VLADIMÍR MAŘÍK
OLGA ŠTĚPÁNKOVÁ
JIŘÍ LAŽANSKÝ
a kolektiv

# Umělá inteligence (5)

ACADEMIA

# Umělá inteligence (5)

*Recenzent*

prof. Ing. Petr Vavřín, DrSc.

**Kniha vznikla a byla vydána s finanční pomocí Akademie věd České republiky a dále těchto grantů a výzkumných projektů:**

**FA8655-06-1-3073** (European Office of Aerospace Research and Development, USA): Autonomous Agents in Air-Traffic Control
**GA AV ČR 1ET100300419**: Inteligentní modely, algoritmy, metody a nástroje pro vytváření sémantického webu
**GA AV ČR 1ET100300517**: Metody inteligentních systémů a jejich aplikace při dobývání znalostí a zpracování přirozeného jazyka
**GA AV ČR 1ET100750408**: Digitální video-senzorický systém záchranného robotu
**GA AV ČR 1ET101120413**: Data a nástroje pro informační systémy
**GA AV ČR AV0Z10300504**: Informatika pro informační společnost: modely, algoritmy, aplikace
**GA AV ČR KJB201210701**: Automatická extrakce znalostí
**GA UK 351/2006/A-INF/MFF**: Virtuální lidé
**GAČR 106/03/0464**: Biomateriál a kontaktní rozhraní – detekce a analýza napěťově deformačních stavů
**GAČR 201/04/0528**: Výpočetní aspekty emergence – teorie a experimenty
**GAČR 201/07/0205**: Dynamické aspekty rozvrhování
**GAČR 408/04/1370**: Projekt výzkumu interaktivních médií
**IST 045 282** (EU STREPS): OLDES – Older People's e-Services at Home
**IST-2003-511598** (EU Network of Excellence): COGAIN – Communication by Gaze Interaction
**LC544** (Centrum základního výzkumu AMU a MU): Výzkum funkcí techniky při vzniku a provozování múzického díla
**MSM 0021620838** (MŠMT ČR): Moderní metody, struktury a systémy informatiky
**MSM 0021620845** (MŠMT ČR): Teoretický výzkum komplexních jevů ve fyzice, biologii a vědách o společnosti
**MSM 0021630529** (MŠMT ČR): Inteligentní systémy v automatizaci
**MSM 1M0021620808** (MŠMT ČR): Molekulárně biologické, genetické a epigenetické aspekty vzniku a rozvoje modelových tumorů dospělého věku
**MSM 1M0567** (MŠMT ČR): Centrum aplikované kybernetiky
**MSM 235200004** (MŠMT ČR): Rozvoj teorie systémů a její aplikace v řízení, inteligentním rozhodování a komunikaci
**MSM 6840770003** (MŠMT ČR): Rozvoj algoritmů počítačových simulací a jejich aplikace v inženýrství
**MSM 6840770012** (MŠMT ČR): Transdisciplinární výzkum v oblasti biomedicínského inženýrství II
**MSM 6840770015** (MŠMT ČR): Výzkum metod a systémů pro měření fyzikálních veličin a zpracování naměřených dat
**MSM 6840770038** (MŠMT ČR): Rozhodování a řízení pro průmyslovou výrobu III
**N00014-06-1-0232** (Office for Naval Research, USA): Meta-reasoning and Adjustable Autonomy in Computational Multi-agent Systems
**NSF ITR 0122466** (National Science Foundation, USA): Multilingual Access to Large Spoken Archives
**UBACYT I056** (Univerzity v Buenos Aires): Technology and Productive Innovation

VLADIMÍR MAŘÍK
OLGA ŠTĚPÁNKOVÁ
JIŘÍ LAŽANSKÝ
a kolektiv

# Umělá inteligence (5)

ACADEMIA
PRAHA 2007

KATALOGIZACE V KNIZE - NÁRODNÍ KNIHOVNA ČR

Mařík, Vladimír
Umělá inteligence. (5) / Vladimír Mařík, Olga Štěpánková, Jiří Lažanský. -- Vyd. 1. -- Praha : Academia, 2007
ISBN 978-80-200-1470-2

004.8 * 001.83
- umělá inteligence -- interdisciplinární aspekty
- sborníky

004.8 - Umělá inteligence [23]

---

Na frontispisu vidíme robot firmy Toyota, který koncertuje na půdě ČVUT v Betlémské kapli.

© Vladimír Mařík, Olga Štěpánková, Jiří Lažanský a kol., 2007

ISBN 978-80-200-1470-2
ISBN 80-200-0502-1 (soubor)
ISBN 80-200-0496-3 (1. díl)
ISBN 80-200-0504-8 (2. díl)
ISBN 80-200-0472-6 (3. díl)
ISBN 80-200-1044-0 (4. díl)

## AUTORSKÝ KOLEKTIV KNIHY


**doc. RNDr. Roman Barták, Ph.D.**
roman.bartak@mff.cuni.cz
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta
Malostranské náměstí 2/25
118 00 Praha 1

**Dr. Diego Bendersky**
dbenders@dc.uba.ar
Universidad de Buenos Aires
Facultad de Ciencias Exactas y Naturales
Ciudad Universitaria
1428 Buenos Aires, Argentina

**Mgr. Cyril Brom**
brom@ksvi.mff.cuni.cz
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta
Malostranské náměstí 2/25
118 00 Praha 1

**RNDr. Jan Cuřín, Ph.D.**
jan_curin@cz.ibm.com
IBM Česká republika
V Parku 2294/4
The Park
148 00 Praha 4

**prof. MUDr. Ivan Čapov, CSc.**
ivan.capov@fnusa.cz
Masarykova univerzita, Lékařská fakulta
I. chirurgická klinika FN u sv. Anny
Pekařská 53
656 91 Brno

**MUDr. Jan Doležel**
jan.dolezel@fnusa.cz
Masarykova univerzita, Lékařská fakulta
I. chirurgická klinika FN u sv. Anny
Pekařská 53
656 91 Brno

**Ing. Marcela Fejtová**
fejtovam@k333.felk.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**doc. RNDr. Jan Hajič, Dr.**
Jan.Hajic@mff.cuni.cz
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta
Malostranské náměstí 2/25
118 00 Praha 1

**prof. PhDr. Eva Hajičová, DrSc.**
hajicova@ufal.mff.cuni.cz
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta
Malostranské nám. 2/25
110 00 Praha 1

**doc. Ing. Ivan M. Havel, CSc., Ph.D.**
havel@cts.cuni.cz
Centrum pro teoretická studia
sdružené pracoviště Univerzity Karlovy
a Akademie věd ČR
Jilská 1
110 00 Praha 1

**Mgr. Matěj Hoffmann**
hoffmann@ifi.uzh.ch
Universität Zürich,
Wirtschaftswissenschaftliche Fakultät
Andreasstrasse 15
CH-8050 Zürich, Švýcarsko

**Mgr. Jana Horáková, Ph.D.**
horakova@phil.muni.cz
Masarykova univerzita
Filosofická fakulta, Ústav hudební vědy
A. Nováka 1,
602 00 Brno

**Ing. Jakub Hrabec**
hrabec@feec.vutbr.cz
VUT v Brně, Fakulta elektrotechniky
a komunikačních technologií
Kolejní 4
612 00 Brno

**prof. Dr. Pablo Miguel Jacovkis**
jacovkis@dc.uba.ar
Universidad de Buenos Aires
Facultad de Ciencias Exactas y Naturales
Ciudad Universitaria
1428 Buenos Aires, Argentina

**prof. RNDr. Jozef Kelemen, DrSc.**
kelemen@fpf.slu.cz
Slezská univerzita v Opavě
Filosoficko-přírodovědecká fakulta
Bezručovo náměstí 13
746 01 Opava

**Ing. Pavel Kordík, Ph.D.**
kordikp@fel.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**Ing. Tomáš Krajník**
krajnt1@fel.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**Mgr. Ondřej Kučera**
kucera@ufal.mff.cuni.cz
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta
Malostranské nám. 2/25
110 00 Praha 1

**RNDr. Miroslav Kulich, Ph.D.**
kulich@labe.felk.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**doc. Ing. Jiří Lažanský, CSc.**
lazan@labe.felk.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**Dr. José Sellés-Martínez**
pepe@gl.fcen.uba.ar
Universidad de Buenos Aires
Facultad de Ciencias Exactas y Naturales
Ciudad Universitaria
1428 Buenos Aires, Argentina

**prof. Ing. Vladimír Mařík, DrSc.**
marik@labe.felk.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**Ing. Roman Mázl**
mazl@labe.felk.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**doc. Ing. Pavel Nahodil, CSc.**
nahodil@k333.felk.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**Ing. Tomáš Neužil**
neuzil@feec.vutbr.cz
VUT v Brně, Fakulta elektrotechniky
a komunikačních technologií
Kolejní 4
612 00 Brno

**Ing. Petr Novák**
novakpe@labe.felk.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**Ing. Tomáš Pajdla, Ph.D.**
pajdla@cmp.felk.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**Ing. Jiří Pavlíček**
pavlicek@labe.felk.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**doc. Dr. Ing. Michal Pěchouček, MSc.**
pechouc@labe.felk.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**Ing. František Pikeš**
frantisek.pikes@hospimed.cz
HOSPIMED, spol. s r. o.
Malešická 51
Praha 3

**Ing. Libor Přeučil, CSc.**
preucil@labe.felk.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**prof. Ing. Josef Psutka, CSc.**
psutka@kky.zcu.cz
Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Univerzitní 8
306 14 Plzeň

**Ing. Milan Rollo**
rollom@fel.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**Ing. Jan Romportl**
romportl@philosophia.cz
Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Univerzitní 8
306 14 Plzeň

**Dr. Juan Miguel Santos**
jmsantos@dc.uba.ar
Universidad de Buenos Aires
Facultad de Ciencias Exactas y Naturales
Ciudad Universitaria
1428 Buenos Aires, Argentina

**Mgr. Vít Šisler**
vsisler@gmail.com
Univerzita Karlova v Praze
Filosofická fakulta
U Kříže 8
158 00 Praha 5

**Ing. David Šišlák**
sislakd@fel.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**doc. Ing. Miroslav Šnorek, CSc.**
snorek@fel.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**prof. Ing. František Šolc, CSc.**
solc@feec.vutbr.cz

VUT v Brně, Fakulta elektrotechniky
a komunikačních technologií
Kolejní 4
612 00 Brno

**prof. RNDr. Olga Štěpánková, CSc.**
step@labe.felk.cvut.cz

ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**prof. Ing. Michael Valášek, DrSc.**
michael.valasek@fs.cvut.cz

ČVUT v Praze, Fakulta strojní
Technická 4
166 07 Praha 6

**MUDr. Lenka Veverková, Ph.D.**
lenka.veverkova@fnusa.cz

Masarykova univerzita, Lékařská fakulta
I. chirurgická klinika FN u sv. Anny
Pekařská 53
656 91 Brno

**Mgr. Barbora Vidová Hladká, Ph.D.**
hladka@ufal.mff.cuni.cz

Univerzita Karlova v Praze
Matematicko-fyzikální fakulta
Malostranské náměstí 2/25
118 00 Praha 1

**MUDr. Petr Vlček, Ph.D.**
petr.vlcek@fnusa.cz

Masarykova univerzita, Lékařská fakulta
I. chirurgická klinika FN u sv. Anny
Pekařská 53
656 91 Brno

**doc. Ing. Jaromír Volf, DrSc.**
jaromir.volf@fs.cvut.cz

ČVUT v Praze, Fakulta strojní
Technická 4
166 07 Praha 6

**Ing. Pavel Vrba, Ph.D.**
pvrba@ra.rockwell.com

Rockwell Automation s.r.o.
Pekařská 10a/695
155 00 Praha 5

**prof. Dr. Kevin Warwick**
k.warwick@reading.ac.uk

University of Reading,
School of Systems Engineering
Whiteknights
Reading, RG6 6AY, Velká Británie
a
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

**prof. RNDr. Jiří Wiedermann, DrSc.**
Jiri.Wiedermann@cs.cas.cz
Akademie věd České republiky
Ústav informatiky
Pod vodárenskou věží 2
182 07 Praha 8

**doc. Ing. Zdeněk Zdráhal, CSc.**
z.zdrahal@open.ac.uk
The Open University
Knowledge Media Institute
Walton Hall
Milton Keynes, MK7 6AA, Velká Británie

**doc. Ing. Luděk Žalud, Ph.D.**
zalud@feec.vutbr.cz
VUT v Brně, Fakulta elektrotechniky
a komunikačních technologií
Kolejní 4
612 00 Brno

**Ing. Filip Železný, Ph.D.**
zelezny@labe.felk.cvut.cz
ČVUT v Praze, Fakulta elektrotechnická
Technická 2
166 27 Praha 6

# Obsah

**Předmluva** ..... 25

**1 Po nás podoba** ..... 29
*I. M. Havel*

1.1 Mrknutí okem ..... 29
1.2 Hraní a zahrávání ..... 30
1.3 Chceme vyrobit umělého člověka ..... 30
1.4 Skeptici, optimisti, pesimisti ..... 32
1.5 Vzhled a podoba ..... 33
1.6 Umění umělého ..... 35
1.7 Obhlížený, ohmatávaný – ale i prožívaný svět ..... 36
1.8 Společenství robotů ..... 38
1.9 Co nás, lidi na tom všem vlastně láká? ..... 39
1.10 Co když je všechno jinak? ..... 41
Literatura ..... 42

**2 Robot – stroj a metafora 20. století** ..... 43
*J. Horáková, J. Kelemen*

2.1 Úvod ..... 43
2.2 Zlaté otrokyně, parní stroje a revoluční proletariát ..... 44
2.3 Nástup robotů ..... 46
2.4 Jak se robot měnil v stroj ..... 48
2.5 Vstup robotů do science fiction ..... 49
2.6 Robot – simplifikovaný dělník vs. komplikovaný stroj ..... 50
2.7 Turingovy hypotézy o stroji a o člověku ..... 53
2.8 Od Alquista k von Neumannovi ..... 57
2.9 Kyborg ..... 60
2.10 Kouzlo interakce ..... 63
2.11 Decentralizovanost ..... 65
2.12 Síla interakce ..... 68
2.13 Závěr ..... 71
Literatura ..... 72

**3 Výpočetní meze kognitivních a inteligentních systémů** ..... 75
*J. Wiedermann*

3.1 Úvod ..... 75
3.2 Jsou živé organismy spíše konečnými automaty anebo Turingovými stroji? ..... 78
3.3 Relativistické kognitivní systémy ..... 81
3.4 Interaktivní evoluční kognitivní systémy ..... 85
3.5 Závěr ..... 88
Literatura ..... 89

**4 Umělé myšlení a kauzální paradox emergentních systémů** ..... 91
*J. Romportl*

4.1 Úvod ..... 91
4.2 Emergence a myšlení ..... 91
4.3 Kauzální domény ..... 94
4.4 Dynamická teorie množství ..... 96
4.4.1 Pohled, obzor a přirozené nekonečno ..... 96
4.4.2 Množinový popis přirozeného nekonečna ..... 98
4.5 Mezidoménová interakce a emergentní efekt ..... 101
4.5.1 Model kauzální domény ..... 101
4.5.2 Model mezidoménové interakce ..... 103
4.5.3 Interpretace kauzálního paradoxu ..... 105
4.5.4 Vztah kauzálního paradoxu a emergujícího myšlení ..... 108
Literatura ..... 112

**5 Virtuální lidé** ..... 113
*C. Brom, V. Šisler, M. Hoffmann*

5.1 Virtuální člověk ..... 113
5.1.1 Trip a Grace: virtuální lidé v akci ..... 114
5.2 Řízení virtuálních lidí ..... 114
5.2.1 Architektura virtuálního člověka ..... 114
5.2.2 Rozhodovací pravidla ..... 117
5.2.3 Reaktivní plánování ..... 118
5.2.4 Rozhodovací stromy ..... 119
5.2.5 Konečné automaty ..... 120
5.2.6 Fuzzy a pravděpodobností přístup ..... 122
5.2.7 Navigace a hledání cesty ..... 122
5.2.8 Alternativní přístupy: plánování a neuronové sítě ..... 124
5.2.9 Nástroje a vývojová prostředí ..... 125
5.3 Aplikace využívající virtuálních lidí ..... 125
5.3.1 Tactical Iraqi ..... 125
5.3.2 Další aplikace ..... 127
5.4 Projekty Enti, IVE a Pogamut ..... 129

5.4.1 Projekt Enti ... 129
5.4.2 Projekt IVE ... 131
5.4.3 Pogamut ... 134
5.5 Virtuální lidé z různých pohledů ... 134
5.5.1 Virtuální člověk z pohledu softwarového inženýrství ... 134
5.5.2 Virtuální lidé, agenti a animati ... 135
5.5.3 Virtuální lidé z pohledu koncového uživatele ... 137
5.5.4 Virtuální lidé z pohledu etiky a práva ... 138
5.6 Výzvy pro umělou inteligenci ... 140
Literatura ... 141

**6 Omezující podmínky: od sudoku po vesmírné aplikace** ... 146
*R. Barták*

6.1 Úvod ... 146
6.2 Problém splňování podmínek ... 147
6.3 Filtrační algoritmy ... 149
6.4 Prohledávací algoritmy ... 155
6.5 Vývojová prostředí ... 161
6.6 Modelování problémů ... 162
6.7 Aplikační oblasti ... 168
6.8 Další zdroje ... 169
6.9 Závěr ... 170
Literatura ... 171

**7 Lze popsat krásu matematickým výrazem?** ... 173
*J. S. Martínez*

7.1 Úvod ... 173
7.2 Proporce a zlatý řez: jak „krásně" rozdělit celek na části ... 174
7.3 Vzory v ploše a prostoru: jednotvárnost a rytmus ... 175
7.4 Symetrie v ploše a prostoru: od zrcadel ke krystalům ... 177
7.5 Perspektiva a projekce: zachycení třetího rozměru ... 179
7.6 Geometrické a fraktální umění ... 182
7.7 Závěrečné poznámky ... 184
Literatura ... 184

**8 Počítač jako laboratoř** ... 185
*P. M. Jacovkis*

8.1 Úvod ... 185
8.2 První počítačové experimenty ... 187
8.3 Laboratoř meteorologa ... 188
8.4 Další vědy ... 189

8.5 Modelování v sociálních vědách ..... 192
8.6 Počítač a klasičtí matematikové ..... 193
8.7 Závěr ..... 194
Literatura ..... 195

**9 Úloha znalostí v inženýrském designu** ..... 197
*Z. Zdráhal*

9.1 Úvod ..... 197
9.2 Dobře a špatně strukturované problémy ..... 198
9.3 Design jako proces ..... 202
9.3.1 Analýza problému ..... 203
9.3.2 Konceptuální design ..... 203
9.3.3 Rozpracování konceptů ..... 204
9.3.4 Detailní návrh ..... 204
9.4 Podpora rozhodování ve špatně strukturovaných úlohách ..... 205
9.5 Explicitní a imlicitní znalosti ..... 207
9.6 Příklady vytváření znalostí ..... 208
9.7 Stručné shrnutí a požadavky na podporu inženýrského designu ..... 213
9.8 Podpora inženýrského designu: projekt Clockwork ..... 214
9.9 Závěrečné poznámky ..... 218
Literatura ..... 219

**10 Mechatronika – navrhování inteligentních strojů** ..... 220
*M. Valášek*

10.1 Úvod ..... 220
10.2 Co je mechatronika ..... 220
10.3 Vznik a historie mechatroniky ..... 221
10.4 Současná mechatronika ..... 222
10.5 Inteligence v mechatronice ..... 225
10.5.1 Komunikační inteligence ..... 226
10.5.2 Funkční inteligence ..... 226
10.5.3 Umělá inteligence ..... 230
10.5.4 Inteligence při navrhování ..... 231
10.6 Závěr ..... 232
Literatura ..... 232

**11 Přírodou inspirované metody optimalizující modely pro vytěžování znalostí** ..... 233
*M. Šnorek, P. Kordík*

11.1 Úvod ..... 233
11.2 Deduktivní a induktivní modelování ..... 234

11.3 Model konstruovaný algoritmem COMBI GMDH (a jeho slabiny) ..... 235
11.4 Model konstruovaný algoritmem MIA GMDH (a jeho slabiny) ..... 236
11.5 Další zlepšování algoritmů pro tvorbu modelu, algoritmus GAME ..... 237
11.6 Automatizace generování modelů v GAME, využití evolučních výpočetních technik ..... 238
11.7 Zachování různorodosti jednotek v GAME ..... 239
11.8 Optimalizace koeficientů přenosových funkcí jednotek v GAME ..... 240
11.9 Gradientní metody optimalizace koeficientů jednotek sítě ..... 242
11.9.1 Kvazinewtonova metoda (Quasi-Newton method QN) ..... 242
11.9.2 Metoda sdružených směrů (Conjugate Gradients) ..... 242
11.10 Přírodou inspirované metody optimalizace koeficientů – evoluce ..... 242
11.10.1 Diferenciální evoluce (DE) ..... 242
11.10.2 Zjednodušená atavistická diferenciální evoluce (Simplified Atavistic Differential Evolution – SADE) ..... 242
11.11 Přírodou inspirované metody optimalizace koeficientů – hejna ..... 243
11.11.1 Optimalizace pomocí hejna částic (Particle Swarm Optimization) ..... 243
11.11.2 Mravenčí kolonie (Ant Colony Optimization) ..... 243
11.11.3 Spojitá optimalizace pomocí mravenčí kolonie (Continuous Ant Colony Optimization) ..... 243
11.12 Hybridní a ostatní metody optimalizace koeficientů ..... 244
11.12.1 Hybrid GA a PSO (HGAPSO) ..... 244
11.12.2 Ortogonální prohledávání (Orthogonal Search OS) a stochastické ortogonální prohledávání (Stochastic Orthogonal Search SOS) ..... 244
11.13 Porovnání výsledků optimalizačních metod ..... 244
11.13.1 Rychlost jednotlivých optimalizačních metod ..... 246
11.14 Závěr ..... 246
Literatura ..... 248

**12 O empirickém výzkumu algoritmického objevování** ..... 250
*F. Železný*

12.1 Fázové přechody ..... 251
12.2 Podivná rozložení ..... 255
12.3 Ockhamova břitva ..... 257
12.4 Metaučení ..... 260
12.5 Závěr ..... 261
Literatura ..... 261

**13 Vliv prostředí v posilovaném učení** ..... 263
*D. Bendersky, J. M. Santos*

13.1 Úvod ..... 263
13.2 Hypotézy a předběžný důkaz ..... 265
13.2.1 Vliv prostředí ..... 265
13.2.2 Automatická konstrukce posilovací funkce ..... 267

13.3 Motivace a diskuse ..... 269
13.4 Závěr ..... 270
Literatura ..... 271

**14 Jazyk a umělá inteligence: kudy a kam?** ..... 272
*E. Hajičová, J. Cuřín, J. Hajič, O. Kučera a B. Vidová-Hladká*

14.1 Úvod ..... 272
14.2 Vytváření rozsáhlých jazykových zdrojů ..... 273
14.3 Využití počítačových jazykových zdrojů v aplikacích ..... 276
14.3.1 Statistický strojový překlad ..... 276
14.3.2 Styx: elektronická cvičebnice češtiny ..... 278
14.4 Příklad integrovaného projektu: MALACH ..... 280
14.5 Závěr ..... 282
Literatura ..... 283

**15 Hlasový dialog s počítačem** ..... 284
*J. Psutka*

15.1 Obecná architektura hlasového dialogového systému ..... 284
15.2 Modul rozpoznávání řeči ..... 285
15.2.1 Analýza akustického signálu ..... 287
15.2.2 Akustické modelování ..... 290
15.2.3 Jazykové modelování ..... 295
15.2.4 Dekódování ..... 296
15.3 Porozumění mluvenému jazyku ..... 297
15.4 Modul syntézy řeči ..... 299
15.4.1 Zpracování přirozeného jazyka ..... 301
15.4.2 Syntéza řeči ..... 305
15.5 Dialogový manažer ..... 310
15.5.1 Vedení dialogu ..... 311
15.5.1.1 Modelování zdrojů znalostí pro vedení dialogu ..... 312
15.5.1.2 Ověřování správnosti interpretace promluvy uživatele ..... 314
15.5.2 Strategie řízení dialogu ..... 315
15.5.2.1 Dialog s iniciativou systému ..... 316
15.5.2.2 Dialog se smíšenou iniciativou ..... 316
15.5.2.3 Dialog s iniciativou uživatele ..... 317
15.5.3 Typy dialogových systémů ..... 317
15.5.3.1 Dialogový systém s konečným počtem stavů ..... 318
15.3.3.2 Dialogový systém využívající strukturu rámců ..... 320
15.3.3.3 Dialogový systém založený na agentech ..... 323
15.6 Závěrečné poznámky ..... 324
Literatura ..... 325

**16 Geometrie v počítačovém vidění – principy a aplikace rekonstrukce prostorové scény z obrazů** ..... 328
*T. Pajdla*

16.1 Geometrie v počítačovém vidění ..... 328
16.2 Konstrukce panoramatického obrazu – klasický problém počítačového vidění ..... 330
16.3 Z obrazu do prostoru ..... 332
16.3.1 Perspektiva a model promítání dírkovou kamerou ..... 332
16.3.2 Konstrukce smysluplných trojdimenzionálních souřadnic z korespondencí v obrazech ..... 335
16.3.3 ImageModeler – interaktivní prostorový modelář ..... 336
16.3.4 Automatická rekonstrukce pohybu kamery v prostoru ze sekvence obrazů ..... 338
16.3.5 Automatické nalezení korespondencí v obrazech ..... 340
16.3.6 Boujou – nástroj pro rekonstrukci pohybu kamery z videa ..... 345
16.4 Závěr ..... 345
Literatura ..... 345

**17 Taktilní informace v systémech s umělou inteligencí** ..... 348
*J. Volf*

17.1 Úvod ..... 348
17.2 Obecné pojmy ..... 348
17.3 Trocha anatomie ..... 349
17.4 Zpracování taktilní informace ..... 351
17.4.1 Průběh zpracování taktilní informace ..... 351
17.4.2 Metody tvorby taktilních obrazů ..... 352
17.4.2.1 Metoda A. A. Charkeviče ..... 352
17.4.2.2 Iterativní metoda určení vrcholů ..... 353
17.4.2.3 Metoda tolerančního pásma ..... 354
17.4.2.4 Metody řetězových kódů ..... 354
17.4.3 Metody klasifikace taktilních otisků ..... 355
17.4.3.1 Metoda Varishold ..... 355
17.4.3.2 Metody využívající řešení Helmholtzovy rovnice ..... 357
17.4.3.3 Metody využívající diskriminační analýzy ..... 358
17.4.3.4 Použití lingvistické metody s využitím Freemanova kódu ..... 359
17.4.3.5 Lingvistická metoda použitá při klasifikaci uchopeného předmětu antropomorfní hlavicí ..... 361
17.4.3.6 Metoda VORO ..... 361
17.4.3.7 Aktivní ohmatávání ..... 363
17.5 Využití taktilní informace ..... 363
17.5.1 Využití v robotice ..... 363
17.5.2 Využití v biomechanice a biomedicíně ..... 366
17.5.3 Taktilní informace jako dodatečný informační zdroj ..... 368
17.6 Závěr ..... 368
Literatura ..... 368

**18 Od jednoduchých robotů k anticipujícím stvořením** ..... 374
*P. Nahodil*

Úvod ..... 374
18.1 Vývoj robotů ..... 374
18.1.1 Vyvíjení robotů – začátky robotizace ..... 374
18.1.2 Mezigenerační posun v možnostech a myšlení ..... 375
18.1.3 Chování inteligentních strojů a jejich vývoj ..... 376
18.2 Umělé formy života ..... 381
18.2.1 Technické předpoklady realizace ..... 381
18.2.2 Historické pozadí vývoje umělého života ..... 383
18.2.3 Základní principy a cíle umělého života ..... 384
18.2.4 Agentové technologie a umělý život ..... 386
18.3 Anticipace a anticipační systémy ..... 387
18.3.1 Srovnání anticipačního chování s reaktivním ..... 387
18.3.2 Anticipace – základní pojmy a definice ..... 388
18.3.3 Rozdělení anticipačního chování ..... 390
18.3.3.1 Implicitní anticipace ..... 391
18.3.3.2 Anticipace odměny ..... 391
18.3.3.3 Anticipace senzorické informace ..... 392
18.3.3.4 Anticipace stavů ..... 392
18.3.4 Možné aplikace anticipačních přístupů v praxi ..... 392
18.3.5 Učící se klasifikační systémy LCS ..... 393
18.3.6 Anticipační třídicí systémy ACS ..... 394
18.3.6.1 Model prostředí vytvářený ACS ..... 395
18.3.7 Příklady anticipačního učení ALP ..... 397
18.4 Budoucnost koexistence člověk–stroj, bezpečnostní rizika ..... 398
Literatura ..... 401

**19 Servisní robotika** ..... 405
*F. Šolc, K. Žalud, T. Neužil, J. Hrabec, L. Veverková, F. Pikeš, I. Čapov, P. Vlček, J. Doležel*

19.1 Úvod ..... 405
19.2 Mikromechanický lékařský operační počítačový systém da Vinci ..... 407
19.2.1 Robotická chirurgie ..... 407
19.2.2 Popis systému ..... 407
19.2.3 Princip robotické operace ..... 410
19.3 Mobilní robotický systém Orpheus ..... 411
19.3.1 Popis robotu ..... 412
19.3.2 Operátorské stanoviště ..... 413
19.3.3 ARGOS – systém pro teleprezenční ovládání ..... 415
19.3.4 Robocup 2003 ..... 417
19.4 Roboty pro armádu ..... 417
19.4.1 Pozemní bezosádkové prostředky ..... 418

19.4.1.1 TALON™ ..... 419
19.4.1.2 EXPERT – PIAP ..... 420
19.4.1.3 MoonBuggy ..... 420
19.4.1.4 MDARS – Mobile Detection Assesment and Response System ..... 421
19.4.1.5 ECA – COBRA M.R. ..... 422
19.4.1.6 ECA – PRM ..... 423
19.4.1.7 RobuROC6 ..... 423
19.4.1.8 BigDog ..... 424
19.5 Robotický fotbal ..... 425
19.5.1 Kategorie robotického fotbalu ..... 425
19.5.2 Systém robotického fotbalu FIRA MIROSOT ..... 426
19.5.3 Počítačové vidění ..... 427
19.5.4 Strategické rozhodování ..... 428
19.5.5 Řízení robotů ..... 428
19.5.6 Robotický fotbalista ..... 428
19.6 Závěr ..... 429
Literatura ..... 430

**20 Od osamocených robotů ke kolaborativní robotice** ..... 431
*V. Mařík, L. Přeučil, M. Pěchouček, M. Kulich, M. Rollo, R. Mázl, P. Vrba, T. Krajník, D. Šišlák*

20.1 Úvod ..... 431
20.2 Robot – základní stavební prvek kooperujících skupin ..... 433
20.2.1 Co je to inteligentní entita, robot? ..... 433
20.2.2 Motivace pro sdružování entit ..... 436
20.2.3 Výhody, nevýhody a rizika sdružování ..... 436
20.2.4 Nové úlohy spojené se sdružováním ..... 437
20.3 Sdružování shodných a odlišných druhů entit ..... 438
20.3.1 Entity shodného druhu (roboty s roboty) ..... 438
20.3.2 Sdružování rozdílných systémů (roboty a lidé) ..... 439
20.4 Agentový přístup k multirobotickým systémům ..... 441
20.4.1 Vztah mezi multiagentovými a multirobotickými systémy ..... 442
20.4.2 Základní vlastnosti multirobotických systémů ..... 444
20.4.3 Výměna informací mezi roboty ..... 446
20.4.3.1 Implicitní komunikace ..... 446
20.4.3.2 Komunikace prostřednictvím detekce stavu (stavová komunikace) ..... 447
20.4.3.3 Explicitní komunikace ..... 448
20.4.4 Zvláštnosti návrhu multirobotických systémů ..... 448
20.4.4.1 Řešení problematiky komunikační nedostupnosti ..... 449
20.4.4.2 Organizační struktury multirobotických systémů ..... 451
20.4.5 Správa znalostí v multirobotickém sytému ..... 453
20.5 Schopnosti robotu pro řešení společných úloh ..... 454

20.5.1 Lokalizace robotu v prostředí ..... 454
20.5.2 Mapování prostředí ..... 456
20.5.3 Plánování pohybu ..... 461
20.5.4 Plánování a koordinování akcí ..... 463
20.6 Simulační nástroje pro vývoj ..... 465
20.6.1 Player, Stage, Gazebo ..... 466
20.6.2 Multiagentní platforma A-globe ..... 468
20.6.3 Simulační nástroj MAST ..... 469
20.7 Příklady multirobotických experimentů ..... 470
20.7.1 Prohledávání neznámého prostředí skupinou robotů ..... 470
20.7.2 Hybridní telematický systém PeLoTe ..... 474
20.7.2.1 Definice úlohy záchranné mise v hybridním systému ..... 474
20.7.2.2 Řešené podúlohy a jejich vzájemné vazby ..... 475
20.7.2.3 Sdílení informací ve smíšeném týmu ..... 476
20.7.2.4 Experimentální výsledky ..... 478
20.7.3 Aplikace systému MAST pro řízení multirobotické linky ..... 479
20.7.4 Distribuovaný průzkum podmořského dna ..... 482
20.7.5 Řízení autonomních bezpilotních letadel ..... 486
20.7.5.1 Systém Air Traffic Control (ATC) ..... 486
20.7.5.2 Algoritmy řešení kolizí ..... 488
20.7.6 Další zajímavé multirobotické projekty ve světě ..... 489
20.8 Závěr – vize dalšího vývoje ..... 491
Literatura ..... 492

**21 Asistivní technologie** ..... 497
*O. Štěpánková, M. Fejtová, P. Novák*

21.1 Co jsou asistivní technologie? ..... 498
21.1.1 Technologie a kognitivní poruchy ..... 499
21.1.2 Kompenzační pomůcky ..... 499
21.1.3 Může pomoci umělá inteligence? ..... 500
21.2 Technologie v roli osobního asistenta nebo společníka ..... 501
21.3 Podpora pro komunikaci člověka s počítačem ..... 502
21.3.1 Počítač řízený myšlenkou ..... 503
21.3.2 Počítač řízený pohybem oka ..... 503
21.3.2.1 Princip systému I4Control® ..... 504
21.3.2.2 Užitečné SW nástavby pro systém I4Control® ..... 507
21.3.2.3 I4Control® a ovládání prostředí a jeho předmětů ..... 509
21.3.2.4 Problémy s ovládáním invalidního vozíku pohledem ..... 510
21.4 Závěr ..... 510
Literatura ..... 511

22 **Evoluce: Poslední hra pro lidstvo** ..... 513
*K. Warwick*

22.1 Úvod ..... 513
22.2 Vytváření rozhraní k nervovému systému ..... 515
22.3 Pokus s implantátem v roce 1998 ..... 515
22.4 Pokus s implantátem v roce 2002 ..... 516
22.5 Pokusná fáze ..... 518
22.6 Některé praktické postřehy ..... 519
22.7 Vylepšení ve stylu Kyborga ..... 520
22.8 Závěr ..... 521
Literatura ..... 521

Summary ..... 523

Česko-anglický slovník ..... 525

Anglicko-český slovník ..... 533

Rejstřík ..... 541

# Předmluva

... a o tom celá tato řada knih o umělé inteligenci vlastně je ...

Pátý svazek má na rozdíl od předchozích dílů jiný, mnohem interdisciplinárnější charakter. Jeho vydání bylo silně motivováno jednáním 2. Česko-argentinského bienále nazvaného „E-Golems", které se konalo 3.–5. července 2005 v rámci 16. Světového kongresu IFAC v Praze. Bienále se vracelo k historickým kořenům spolupráce člověka a stroje počínaje středověkou golemovskou vizí až po moderní interpretaci Golema jako robotického systému využívajícího výsledků umělé inteligence, kybernetiky, virtuální reality, umělého života, nanotechnologií a služeb Internetu. Internetem a gridem inspirované vize vedly dokonce k představě elektronické verze Golema současnosti, tzv. e-Golema, jenž bienále propůjčil své jméno. Více než 120 účastníků ze 13 zemí dostalo unikátní možnost si v rámci interdisciplinárních diskusí uvědomit, jak rychle se mění

naše představy o spolupráci a interakcích člověka a stroje, jak nenápadně, ale intenzivně jsme ovlivňováni praktickými aplikacemi umělé inteligence v robotice, komunikačních systémech i v rozhraních člověk–stroj. Proto má i naše kniha, zejména její první část, poněkud filosofičtější a vizionářštější obsah často konfrontovaný s cestou, kterou umělá inteligence prošla a jíž byla formována. Další části knihy jsou věnovány spíše pokrokům ve vývoji nových výpočetních technologií, algoritmů a systémů, dále pak inteligentním rozhraním a komplexnějším aplikacím technik umělé inteligence v robotice. Čtyři kapitoly byly připraveny zahraničními autory, účastníky bienále z Argentiny a Velké Británie – i tento fakt dává knize nový, vskutku mezinárodní rozměr.

Kniha, kterou držíte v ruce, sdružuje v sobě nejnovější poznatky z období, v němž čím dál tím častěji vystupují do popředí nejen filosofické otázky spojené s hledáním rozumných proporcí mezi využíváním přirozené a strojové (umělé) inteligence v reálných aplikacích, ale dokonce i otázky související s hledáním etické a morální únosnosti využívání umělé inteligence. Takovéto úvahy se vynořují například v souvislosti s pokroky v projektu mapování lidského genomu, s rozvojem oblasti nazývané umělý život i s pokroky v oblasti lékařských aplikací atd.

Jedno je však zcela nesporné. S rostoucím počtem aplikací umělé inteligence v průmyslu, zdravotnictví i běžném životě se bude uvolňovat duševní potenciál lidstva směrem k efektivnějšímu využívání vlastních tvůrčích schopností. Přirozená inteligence čím dál tím většího počtu odborníků bude moci být věnována na výzkum, vývoj a realizaci nových počítačových systémů, které budou nesporně jevit známky umělé inteligence. Tento proces bude samozřejmě podporován rostoucí řadou nástrojů opírajících se o aplikovatelné výsledky umělé inteligence jako vědní disciplíny (půjde například o nástroje typu inteligentní databáze, expertní systémy, softwarová prostředí pro realizace genetických algoritmů, multiagentních systémů apod.).

O nutnosti vydat další díly jsme nepochybovali již v okamžicích, kdy jsme předávali do tisku rukopisy předchozích knih. První díl byl učebnicí základů umělé inteligence. Druhý díl se zaměřoval spíše na doplňkové partie určené pro pokročilejší čtenáře a doktorandy. Třetí díl se věnoval především různým aspektům spolupráce menší či větší skupiny samostatných agentů vybavených různým stupněm schopnosti uvažovat o okolním světě a rozhodovat. Čtvrtý díl se snažil vytvářet předpoklady pro tvorbu metod a nástrojů, které umožní efektivně pracovat s informacemi, tj. organizovat je, třídit, aktualizovat, vybírat z nich to podstatné a zejména je účelně využívat. Pátý díl, který právě držíte v ruce, má oslovit širší veřejnost potenciálních uživatelů technologií, které využívají nejnovějších výsledků výzkumů z oblasti umělé inteligence. Kniha chce především upozornit na obrovskou škálu míst, v nichž se technologie umělé inteligence již dnes úspěšně uplatňují ku prospěchu nás všech. Současně poukazuje i na pozoruhodné směry současného výzkumu, které mohou zásadně ovlivnit to, jaká bude tvář Země i osud lidí v nedaleké budoucnosti.

Do autorského kolektivu se nám tentokrát podařilo získat i zahraniční odborníky, kteří napsali zcela původní, teoreticky fundované, avšak spíše přehledově psané kapitoly, které českému čtenáři umožní jednoduše a rychle se seznámit s podstatou pojednávaného tématu. Mnohdy jde o relativně krátké úvahy, které chtějí upozornit na některé zajímavé jevy, jejichž význam zatím možná ani nedokážeme plně posoudit.

Celý svazek se člení do několika bloků, z nichž každý má potenciál oslovit vlastní skupinu čtenářů. Nemusí jít zdaleka jen o lidi technicky zaměřené. Doufáme však, že

tyto čtenáře téma umělé inteligence natolik zaujme, že zbytek knížky alespoň důkladně prolistují a že při tom naleznou inspiraci pro návrh nových přístupů k řešení úloh, s nimiž se běžně setkávají třeba ve svém profesním životě. Jsme přesvědčeni, že vhodné použití nástrojů umělé inteligence může výrazně přispět jak ke zkvalitnění života pro člověka jako individuum, tak pro celou společnost. Je však třeba správně rozeznat ty příležitosti, kde může právě vhodné nasazení nástrojů umělé inteligence pomoci. A k tomu může přispět především tvůrčí dialog s odborníky z různých oblastí včetně těch tak netechnických, jakými jsou třeba společenské vědy, jež mohou úspěšně těžit ze zkušeností, které se již podařilo získat v rámci budování různých systémů obdařených jistým stupněm umělé inteligence. I když řada uměleckých děl či sci-fi filmů chápe roboty a systémy blízké umělé inteligenci jako jednu z vážných hrozeb pro lidstvo, naším cílem je nabídnout pohled ze strany odborníků na bližší či vzdálenější budoucnost zmiňovaných technologií i na potenciální meze jejich schopností. Do jaké míry se nám tento záměr podařilo splnit (a do jaké míry bude mít pravdu hrdinka z Jiránkovy kresby) však nejlépe posoudí čtenáři sami.

Blok prvních pěti kapitol je věnován filosofickým problémům spojeným s možností vytvářet stále složitější roboty, které díky své stále rostoucí míře samostatnosti více a více připomínají svými vlastnostmi člověka. Jsou zde pokládány zdánlivě jednoduché otázky, které vyžadují hlubší zamyšlení a sahají mnohdy až ke kořenům našeho vlastního bytí a myšlení. Najdeme zde úvahy o paradigmatech umělé inteligence, o robotu jako metafoře 20. století i o výpočetních mezích kognitivních systémů. Z mnoha pohledů jsou zkoumány aspekty atraktivního pojmu „virtuální lidé".

Na matematiku v roli významného inspiračního zdroje, který se uplatňuje jak při vývoji efektivních algoritmů, tak jako nástroj pro modelování složitých kognitivních aktivit, se soustřeďuje druhý blok (6.–8. kapitola). Zde se setkáme i s pozoruhodnou úvahou na téma, zda je možné krásu popsat matematicky.

Třetí blok (9.–13. kapitola) studuje různé fáze práce se znalostmi. Všímá si způsobů zacházení s nimi při inženýrském designu či při návrhu inteligentních strojů. Zvláštní pozornost je věnována postupům, které umožňují znalosti automaticky získávat prostřednictvím přírodou inspirovaných metod i pomocí počítačem prováděných experimentů. Stranou nezůstávají ani moderní metody strojového učení.

Čtvrtý blok (14.–17. kapitola) shrnuje současné možnosti robotického vnímání, tedy aktuální možnosti v oblasti používání přirozeného jazyka a zpracování vizuální či dotykové informace.

Poslední pátý blok (18.–22. kapitola) podává přehled o tom, co dokáže robot nebo kolektiv robotů dnes, i tom, jak a v jakých podobách a formách se jim daří člověku sloužit. Velmi pozoruhodná je úvaha o perspektivách vytvoření sofistikovaných systémů, které spojují člověka a stroj a vytvářejí tak novou entitu, kyborga.

Věříme, že kniha, jejíž strukturu a obsah jsme koncipovali s ohledem na potřeby rozvoje vědeckovýzkumné práce na českých akademických pracovištích i jako studijní materiál, který má být využit při zkvalitňování výchovy doktorandů, přispěje i k dalšímu ujednocení české odborné terminologie. Doufáme, že svou referativností, přehledností, čtivostí a kompaktností bude srozumitelná i kolegům z příbuzných i vzdálenějších oborů a podpoří tak rozvoj interdisciplinárního výzkumu v ČR. Vzájemnou interakci s dalšími obory totiž umělá inteligence jako vědní disciplína bytostně potřebuje, vždy se jí nesmírně obohacovala a nacházela zde inspiraci i problémy k řešení.

Bylo nám ctí spolupracovat s celým širokým autorským, vskutku interkontinentálním kolektivem. Rádi bychom poděkovali všem, kteří přímo či nepřímo přispěli ke zrození tohoto díla. Bez podnětných dotazů, připomínek a námětů našich doktorandů a studentů by kniha jistě neměla svou dnešní podobu. Prvními kritickými čtenáři byli právě doktorandi Gerstnerovy laboratoře ČVUT FEL. Jejich připomínky výrazně ovlivnily řadu partií i jednotlivých formulací. Naše zvláštní poděkování patří doktorandům, kteří zabezpečili překlad kapitol připravených argentinskými kolegy, a to Ing. Miroslavu Buršovi (kap. 8, 13, 22), Ing. Václavu Chudáčkovi (kap. 7), Ing. Filipu Karlovi (kap. 8) a Ing. Tomáši Krajníkovi (kap. 22). Velice si ceníme komentářů a připomínek recenzenta prof. Ing. Petra Vavřína, DrSc., za něž mu upřímně děkujeme. Práci na dokončení knihy nám již tradičně usnadnili dva naši milí spolupracovníci, dlouholetí členové realizačního týmu, a to p. RNDr. Václav Havlíček, CSc., svojí erudicí a vstřícností, s nimiž převáděl rukopisy jednotlivých kapitol do tiskových předloh, a p. Mgr. Aleš Baďura, redaktor nakladatelství Academia, svojí trpělivostí, pečlivostí a entusiasmem při překonávání všech nástrah, které mu naše mnohačetné nedokonalosti dovedly připravit. V neposlední řadě děkujeme i Fondu AV ČR za finanční příspěvek na vydání knihy. Díky za tuto vskutku týmovou spolupráci!

Vladimír Mařík
Olga Štěpánková
Jiří Lažanský

# 1 Po nás podoba

(esej o umělých lidech)[1]

*Čtvrtý robot:* Všechno jsme prohledali, pane. Není lidí.
K. Čapek, *R.U.R.*, dějství III

## 1.1 Mrknutí okem

Pamatuji se na jeden americký film, jehož zápletka byla postavena na vztazích, řekněme citových, mezi lidmi a humanoidními roboty. Co mě nejvíc zaujalo, nebyla ta zápletka, ostatně nepříliš duchaplná, nýbrž pozoruhodný rozpor mezi vzhledem oněch robotů a jejich chováním. Na jedné straně se podobali svým vzhledem – tvarem těla – lidem jen málo: měli sice něco jako hlavu a paže, na člověka byli však příliš kovoví, električtí a komputeroví. Na druhé straně však chování, zvlášť jednoho z nich (jmenoval se „Číslo 5") – jeho pohybová motorika, mimika a gestikulace, včetně otvírání „očí" a zvedání „obočí" bylo natolik lidsky přirozené, expresivní a emocionálně patřičné k situaci, že jsem sám brzy zapomněl na tu jeho kovovost, električnost a komputerovost a skoro jsem ho považoval za odrůdu člověka. V jedné z dramaticky napjatých scén jiného filmu („Já, robot", natočeno volně podle románu Isaaka Asimova) titulní robot výmluvně mrkne okem, aby nenápadně naznačil, že řečené nemusí být bráno vážně. Všichni rozumíme, není ani třeba předchozí domluvy. (Mimochodem, robot neměl toto mimické gesto naprogramováno, sám je odkoukal od lidí a vůbec poprvé je použil právě v dotyčné scéně.)

Víme, co dovedou filmové nápady, střihy a digitálně editované triky. Dovolují mísit realističnost s imaginací v libovolném poměru a vytvářet tak bytosti, jejichž inteligenci, intuici a emoce nelze než obdivovat. Ale i dnešní „reální" roboti z plechu, drátů a čipů, vykračující si po chodbách světových univerzit a výzkumných ústavů, jsou každým dnem v tomto směru sofistikovanější a čím dále tím méně se napohled liší od humanoidních robotů vědeckofantastických filmů. O některých se letmo zmíním v oddílu 1.3.

O fyzickou reálnost či realizovatelnost nám zde však nepůjde. Budeme se ptát na něco jiného: co vůbec nás, *lidi*, nutká k tomu, abychom se pokoušeli – ať už ve fantazii nebo ve skutečnosti – obklopovat se *umělými*, nám však *podobnými* tvory? Nezahráváme si tu s něčím obscénním?[2]

---

[1] Rozšířená a upravená verze textu pro katalog výstavy „Konec světa?" v Národní galerii (Knížák, Demšáková, 2000) s použitím eseje (Havel, 2004).

[2] Srov. (Weizenbaum, 2002).

## 1.2 Hraní a zahrávání

Člověk sám sebe pokládá za bytost hravou. Hravost je sklon k chování, které, ač má (v různé míře) svůj vnitřní smysl, důmysl a řád, není zaměřeno k nějakému zjevnému či výslovně míněnému účelu mimo hru samotnou. Pokud snad nějaký užitek má, není tím hlavním, co zpravidla máme na mysli, říkáme-li o někom, že si čistě jen hraje. „Čisté hraní" je to, čím bych chtěl začít.

Vím, že evoluční psychologové snadno a ochotně vysvětlí hravost jako produkt přírodního výběru těch vlastností, které zvyšují zdatnost druhu: hra přece umožňuje člověku (stejně jako zvířecím mláďatům) nacvičovat si rozmanité užitečné dovednosti pro život a modelovat si situace, v nichž se jednou může nečekaně ocitnout. Možná ano, avšak při čistém hraní se na nějakou užitečnost asi nemyslí (proto píši „čisté").

Vedle sklonu k hraní je tu ještě něco jiného, sklon k *zahrávání si* – člověk není jen bytost hravá, ale i zahrávavá. A opět, zahrávání může mít svou „čistou" podobu, kterou nelze mít ani za hloupou nerozvážnost ani za účelově zdůvodnitelné riziko. Člověk si zahrává rád a nemyslí příliš na to, proč. Člověk – tím zde rozumím některé z nás, ale také celé lidské plémě.

Vydáváme se do dálek, výšek a hloubek, kde ještě nikdo nebyl, oddáváme se rychlostem a zrychlením, které v přírodě neznáme, láká nás vidět, jak mizí hmota a vzniká energie, vylepšujeme přírodu v domnění, že jí rozumíme, a vynalézáme hračky, z nichž jde strach. To vše nám nestačí. Chceme vyrobit umělého člověka.

## 1.3 Chceme vyrobit umělého člověka

Hleďme, tvrzení, které generuje řadu otázek. Opravdu chceme? Proč chceme? Co je to „umělý"? Jaký „člověk" to má vlastně být? Začněme od konce.

Řekneme-li „umělý člověk", nebo techničtěji „humanoidní robot" či „android",[3] naznačujeme tím vlastně dvojí: jednak, že se to člověku nějak (pokud možno co nejvíc) podobá, a jednak, že to člověk vlastně není. Je tu tedy nárok na stejnost i nárok na odlišnost; je ovšem nutné si vždy uvědomit, k jakému aspektu jeden či druhý nárok vztáhneme. Může nás zajímat *vzhled* (jaké to má tělo – má to nohy, ruce, hlavu?), *látka* (z čeho je to vyrobeno – z kovu, hadrů, živé hmoty?), *pohyb a mimika* (jaké to má motorické vlastnosti – kráčí to, mává to rukama, mračí se to, mrká to?), *řeč* (mluví to s námi?), *rozum* (chová se to inteligentně, důmyslně, cílevědomě, racionálně, tvořivě?), *psychika* (má to vědomí, vůli, emoce?). K tomu přistupují dva, svým způsobem hlavní aspekty, které ovšem není navenek vidět: *původ* (jak to vzniklo – narodilo se to v továrně, zkumavce, porodnici?) a *individuální historie* (vyvíjí se to, učí se to, sbírá to zkušenosti?).

Připomeňme si namátkou několik ukázkových příkladů, vymyšlených i skutečných, jak je známe z mytologie, magie, beletrie, divadla, filmu, stejně jako z dílen řemeslníků,

[3] Populární slovo robot je dnes již užíváno snad pro jakékoliv technické zařízení, které je aspoň v něčem automatické. S člověkem to nemá co dělat.

vědeckých laboratoří, a odborné literatury.[4] Čtenář nechť se zamyslí, v kterých aspektech by tito tvorové měli mít lidské vlastnosti a v kterých by se naopak směli od člověka lišit, aby v nás vzbudili onen zvláštní pocit, že tu máme opravdového partnera – nebo alespoň, abychom nějaký zvláštní pocit vůbec měli.

Pygmalionova socha[5] byla ze slonoviny, dokonalá to podoba člověka (dokonce dívky), ale pohyb a duši musela získat od bohů. Rovněž hliněného Golema musel oživit šém, tajemné jméno Boží.[6] Paracelsus uvádí prostý recept na vytvoření homunkula, opravdového živoucího lidského mláděte se všemi údy – jen daleko menšího. Stačí zkumavku s lidským spermatem vložit do koňského břicha a udržovat čtyřicet dní při konstantní teplotě. Recept je však založen na hlubokém tajemství, které má zůstat střeženo až do soudného dne.[7]

Zajímavou literární kreací je robotka – tehdy se říkalo „androida" – jménem Hadali ve Villiersově novele z r. 1886 (Villiers, 1920). Kouzelník století, Thomas Alva Edison vytvořil z fonografů, magnetů, induktorů, izolátorů, kondenzátorů, drátů a jiných technických vymožeností té doby jakýsi polotovar, z něhož pak vyrobil lordu Ewaldovi věrnou kopii jeho krásné milenky včetně jejího líbezného hlasu, ladných pohybů, ba i vůně, hlavně však s jediným rozdílem: nesnesitelně primitivní intelekt vzoru je u kopie nahrazen vynikající oduševnělostí.

Čapkovi proslulí Roboti firmy R.U.R. nebyli z plechu, koleček a drátů, jak si je lidé dnes představují, ale z organické látky, která se od látky, z níž je vybudována živá buňka, liší především tím, že umožňuje daleko rychlejší evoluci. (Sám Čapek později důrazně odmítl odpovědnost za myšlenku, že by mašiny „z plechu, koleček a drátů" mohly nahradit lidi (Čapek, 1935). A to ještě netušil, jak se slovo „robot" ujme nejen v žánru sci-fi, ale i v průmyslu.) Roboti R.U.R. byli obdařeni rozumem, řečí a v závěru hry se dva z nich do sebe i zamilovali (happy-end pro robotí populaci).

V současné vědeckofantastické literatuře a (zejména) filmu jsou roboti již takřka neodmyslitelní. Mají mnoho podob, jednou jsou to zjevné artefakty, které až v průběhu děje projeví lidskou povahu (jako naše „Číslo 5" nebo celá generace robotů ve filmu „Já, robot"), jindy jsou to naopak zcela nerozlišitelní „lidé", o nichž se až v závěru filmu prozradí, že mají vnitřnosti z drátů, čipů a všelijakých součástek. V akčních filmech bývají spíše nepřátelští, o to však obtížněji zničitelní. Bylo by jistě poučné analyzovat rozmanité varianty vymyšlených robotů a humanoidů, dovedených k dokonalosti pomocí dovedných triků a digitálního editování, nicméně jejich libovolnost, přemnožení a převažující nerealističnost by nás odvedla od tématu. Pro tuto chvíli stačí, když si uvědomíme, jak lákavé a vzrušující jsou to témata jak pro autory, tak pro diváky.

Nerad bych opominul humanoidní objekty fyzicky realizované. Vraťme se nejprve trochu do minulosti a připomeňme, že rozličné mechanické loutky a automaty s lidskou podobou byly velice populární, zejména v období rokoka. Figuríny, které hrály na flétnu či píšťalu, písař pohybující rukama i prsty, pianistka, která sledovala očima noty a po

[4] Srov. též (Havel, 1980), kap. I.1. Řadu dalších příkladů čtenář nalezne v následující kapitole od J. Horákové a J. Kelemena.

[5] Ovidius, *Proměny*, kniha X.

[6] O Golemovi jako mýtu a symbolu viz (Neubauer, 2002).

[7] Paracelsus, *De generatione rerum naturalium*. srov. (Gebelein, 1998). Motiv homunkula najdeme též v Goethově Faustovi.

skončení skladby se ukláněla, Turek hrající šachy – to byly slavné atrakce, které budily obdiv k dovednosti svých tvůrců, pochybuji však, že jejich trhané pohyby by mohly vzbudit pocity, které mám na mysli. Vlastně právě tyto trhané pohyby nám z nich zbyly jako charakteristický znak umělých (spíše neumělých) bytostí (později se přidal úsečný a monotónní „mechanický" hlas).

Nová kapitola se ovšem otevřela s příchodem počítačů a metod umělé inteligence. Koncem šedesátých let se objevili první kognitivní roboti.[8] V odborné literatuře proslavený robot „Shakey" Stanfordova výzkumného ústavu byla třaslavá mašinka na kolečkách, tak tak schopná postrkovat krabicí (zato si sama promýšlela jak a kudy). Robot „WABOT" univerzity Waseda v Tokiu, řízený minipočítačem, byl daleko více humanoidní. Dovedl kráčet, měnit směr chůze, nalézt předmět, uchopit jej jednou rukou a dát do druhé, někam položit – to vše na ústní pokyny v japonštině. (Chodící roboti jsou dodnes japonskou specialitou, ti nejpokročilejší současní – „Asimo" a „P3" firmy Honda – dovedou chodit vzhůru a dolů po schodech a když do nich strčíte, udrží rovnováhu.[9])

Zmínil jsem se o hravosti a proto nemohu vynechat roboty, kteří jsou k hraní přímo uděláni. Vypadají spíše jako roztomilá zvířátka než lidi, někteří si hrají se svými chovateli (pejsek „AIBO" firmy SONY[10]), jiní hrají v týmu třeba fotbal s jinými roboty a mají na to i své soutěže v různých kategoriích (Bůcha, 1999).

Historie jde dál, přichází další generace robotů, které nazývám *expresivní*. Příkladem je robot-hlava „Kismet" vyvinutý na MIT,[11] který pomocí výrazů tváře, pozic těla, gest, hlasu a směru pohledu reaguje na tu či onu situaci klidně nebo rozčileně, s odporem, údivem či zaujetím, vesele nebo smutně. Další příklady expresivních robotů jsou „Wendy", „Wamoeba" a „Hadaly" v laboratoři Sugano zmíněné univerzity Waseda.[12]

Záměrně se nezmiňuji o množství užitečných robotů (v průmyslu, lékařství, kosmickém výzkumu atd.), protože u nich se z dobrých důvodů na lidskou podobu nehledí, u jejich konstruktérů se čistá hravost uplatňuje podstatně méně než u výzkumných experimentátorů a užitečnost je tou pravou a neproblematickou motivací k jejich vývoji.

## 1.4 Skeptici, optimisti, pesimisti

Lidé rádi čtou – a (proto) odborníci (i neodborníci) rádi píší – rozmanité úvahy o blízké i vzdálené budoucnosti reálné robotiky. Jde většinou o lepší či horší extrapolace ze současného stavu (kvalitu extrapolace hodnotím podle odvahy opustit zaběhané způsoby technického myšlení avšak při zachování logické možnosti). Většinou se opatrně vyhýbají fantazijním představám vědeckofantastické literatury a filmu a jsou (nedivme se) orientovány hlavně na praktická využití, čímž se vzdalují od našeho tématu – zahrávání si s lidskou podobou.

---

[8] Více o nich a některých dalších viz (Havel, 1980), kap. I.4.

[9] Viz www.honda-robots.com, srov. též (Moravec, 1999a), (Moravec, 1999b).

[10] www.aibo-europe.com, viz též (Bůcha, 1999).

[11] „Sociable Machines Project" v Laboratoři pro umělou inteligenci MIT (www.ai.mit.edu/projects/humanoid-robotics-group/kismet).

[12] Laboratoř Sugano, Waseda University, Tokio (www.sugano.mech.waseda.ac.jp ).

Jeden ze známých reprezentantů současné robotiky, Hans Moravec z Carnegie Mellon University, považuje za rozhodující element pro schopnosti budoucích robotů výkonnost (operační rychlost) počítačů, těch, které mají být součástí jejich „těla". Co dekáda, to nová generace univerzálních robotů (ve slově „univerzální" je v podstatě obsažena podobnost člověku co do vnímání, kognitivních funkcí a fyzické aktivity, vše v reálném čase). Takže rok 2010: všestranné vnímání, manipulace, mobilita („úroveň ještěrky"); rok 2020: schopnost učit se ze zkušenosti („úroveň myši"); rok 2030: vnitřní model světa, na němž si bude robot moci rychleji než v reálném čase ověřovat své alternativní záměry („úroveň opice"); rok 2040: pojmové myšlení, rozvažování o vlastnostech reálného světa. Teprve tuto poslední generaci Moravec označuje jako úroveň člověka, poznamenává dokonce, že roboti budou o hodně lepší myslitelé než člověk (jen v logickém odvozování budou aspoň milionkrát rychlejší). Posléze se roboti vzdají své prostoročasové lokalizace a začnou se měnit v jakési pletivo prorůstající mnoha měřítky velikostí jako fraktály a budou si sami libovolně urychlovat svůj vývoj (Moravec, 1999a), (Moravec, 1999b), kapitoly 4 a 5.

Existují tři extrémní postoje při prognózách budoucnosti; představme si je jako vrcholy trojúhelníka, uvnitř něhož si pak můžeme představovat rozmanité názory podle toho, jak mají blízko k oněm extrémům. Jeden vrchol obsadí Skeptik se svým přesvědčením, že veškeré snahy vyrobit opravdu inteligentního robota jsou nejen nereálné, ale i nesmyslné. Druhý vrchol náleží Optimistovi, který soudí, že tyto snahy nejenže mají smysl, ale úspěšně povedou k cíli, z čehož lidstvu kyne překrásná budoucnost. Třetí vrchol ponechejme Pesimistovi, který spolu s Optimistou věří v úspěch oněch snah, ten však bude tragédií pro lidstvo.

Mám příklady pro všechny tři postoje. Vyhlášeným Skeptikem vůči celému projektu umělé inteligence byl od jejího počátku americký filosof Hubert Dreyfus (Dreyfus, 1979), jehož přirovnání UI k někomu, kdo si myslí, že vylezl-li na strom, přiblížil se k Měsíci, je často zmiňováno. Méně znám je zástupce Optimistů, Bill Hibbard (Hibbard, 2002), který věří, že nás čeká nebe na zemi v družbě se superinteligentním strojem, který nás bude milovat, utěšovat a hýčkat. Zástupce Pesimistů je Kevin Warwick, který je i u nás dobře znám (Warwick, 1999). Jeho dedukce jsou hrozivé, ale (naštěstí) logicky nedomyšlené. Myslí si (nevím proč), že čím větší inteligence, tím větší schopnost a touha ovládat druhé. A uvažuje: proč by inteligentnější roboti měli čekat, až je budeme – ze strachu z role druhořadých občanů – chtít zničit? Raději zaútočí jako první a (jsouce inteligentnější) snadno bez milosti zničí nás. Aspoň pokud se včas nezdokonalíme sami.[13]

## 1.5 Vzhled a podoba

Vraťme se raději k naší otázce: co vlastně dělá robota člověkem? Který aspekt je nejdůležitější?

Avšak nejprve musím poukázat na jistou dvojznačnost ve vyjadřování, které sám podléhám. Ptám-li se: „Co vlastně dělá robota člověkem?", užívám singulár pro robota i pro člověka a může se zdát, že si představuji vedle sebe dvě konkrétní bytosti, robota

[13] Viz Warwickův příspěvek v tomto svazku a rovněž mou kritickou poznámku (Havel, 1999).

a člověka. Do jisté míry ano, ale oba pak zastupují jakýsi (neurčitý a jen myšlený) typový pojem: *typ* robot a *typ* člověk. Podobně slova „vzhled", „chování" apod. je třeba chápat jako *typický* vzhled, *typické* chování (vzorce chování, návyky) atd. a odhlížet od partikulárních, zpravidla nahodilých vlastností jednotlivců. V určitých případech nám však nemusí jít o typické vlastnosti, ale o *možné* vlastnosti (existuje robot, který je má), jindy zase o *nutné* vlastnosti (každý robot je má). Zvlášť důležité je pak odlišovat „vývoj" robota či člověka (ontogeneze) od „evoluce" robotů či lidí (zde už raději plurál) a vědět, kdy slovo „robot" označuje populaci robotů a slovo „člověk" znamená lidstvo.[14]

K naší otázce: zdá se mi, že pokud jde o *podobu s člověkem*, mají z těch aspektů, které jsem prve vyjmenoval (existují jistě i další), asi největší váhu tyto: vnější *vzhled*, repertoár *pohybů* a schopnost komunikovat *řečí*. K tomu samozřejmě přistupuje něco, co není přímo vidět, ale co se v řeči a chování robota jaksi musí odrazit: *rozum*, což je, pravda, věc dosti neurčitá; někdo by raději mluvil o inteligenci, jiný o mysli, třetí o racionalitě, čtvrtý o tvořivém sebezdokonalování. Jde spíše o překročení toho, co už i dnešní umělá inteligence (obor) dovede v primitivní a fragmentární podobě programovat pro počítače, nebo aspoň specifikovat jako zadání pro programátora.

O *duši* robotická věda příliš nemluví, protože o ní nic neví. Nicméně úvahy o emocích, vědomí, vůli počítačů a robotů již nejsou výhradním tématem vědecko-fantastické literatury a přestěhovaly se i na stránky učených, převážně filosofických knih a časopisů, srov. (Havel, 2001). Ostatně záleží spíše na nás, kudy vedeme pomyslnou dělicí čáru mezi tím, když napodobujeme jen vnější chování, nechávajíce si svobodu ve volbě typu vnitřních realizačních procedur, a když chceme napodobit i vnitřní zdroje tohoto chování. Lze dokonce identifikovat velké množství meziúrovní, které by mohly být jednou na té, jindy na oné straně dotyčné dělicí čáry. Mluvit o nápodobě vnitřních procedur, to ještě dovedeme, ale jak je to s podobným *prožíváním*, když si to nedovedeme porovnat ani mezi sebou? K tomu se vrátím v oddílu 1.7.

Tolik prozatím o podobnostech. Jak je to s *odlišností od člověka*? Zde ovšem hraje hlavní roli to, že se robot člověkem nenarodil, protože byl někým vyroben. Tedy jeho původ a individuální historie. S tím souvisí i látka, hmotná stránka jeho ustrojení. Neumíme vyrábět živou, biologickou látku (jako v laboratořích firmy R.U.R.), ale zdá se, že pro náš vztah ke skutečným nebo potenciálním robotům to ani není rozhodující.

Mluvím-li zde takto obeznale o podobnostech a odlišnostech, měl bych si sám nejprve uvědomit, *za koho to vlastně mluvím*. Chvíli jako bych mluvil za jakéhosi Absolutního Rozhodčího, který ví, jak je lidské tělo utkáno a kde se v něm skrývá lidská duše. Robota si dobře proklepe, tu souhlasně přikývne, ano, to je ono, tu zavrtí hlavou, zde už to s člověkem nemá co dělat. Jako by šlo o jakousi „objektivní" podobnost či odlišnost, která buď je, nebo není, a kdo ji nevidí, ať se Ho zeptá.

Avšak jindy jako bych mluvil za obyčejného člověka, za kohokoliv z nás, třeba i za sebe. A hle, buď se mi nějaká podobnost ukáže, pak prostě je, nebo neukáže, pak není. Cit pro podobné a odlišné předchází racionální analýzu (dokonce je její podmínkou). A tak, připomíná-li mi robot člověka, je to nejspíše pro celkový dojem, na který má evidentně vliv vnější vzhled robota, přirozenost jeho pohybů i jeho konverzační schopnosti, přičemž kvalita jednoho může i vyvážit nedostatek jiného. V onom prvním

[14] V technických oborech je slovo „vývoj" víceznačné, podle úrovně typové obecnosti toho, co je vyvíjeno.

celkovém dojmu mi je jedno, čím je co docíleno, protože nevím, čím je co docíleno u mých bližních, vlastně ani u mne samého.

Neřeknu, kdy za koho zde mluvím, jen si to, milý čtenáři, hlídej.

## 1.6 Umění umělého

Klademe-li si otázku po umělém člověku, nelze se nezmínit, co se vlastně míní, řekne-li se o něčem, že je to *umělé*. Zdálo by se, že opakem umělého je přirozené, ale je mnoho nepřirozených věcí, které nejsou umělé, a mnoho neumělých věcí, které nejsou přirozené.

O obecných předpokladech vhodnosti užití přívlastku „umělý" jsem již psal ve 3. svazku této série (Havel, 2001). Uvedl jsem tři hlavní charakteristiky umělého: za prvé je to něco, k čemu známe nějaký *přirozený* vzor (dokonce jaksi přednostně), který připouští možnost napodobení, za druhé musí existovat něčí předběžný *záměr* – úmysl či projekt – onoho napodobení lidským přičiněním dosáhnout (byť z jiného materiálu či jiným postupem) a za třetí, pokud došlo k provedení onoho záměru, čili pokud došlo k náležitému tvůrčímu procesu, pak tento proces musel být *i ve svém průběhu intencionální*, tedy uvědoměný, člověkem vědomě řízený.

Druhé dvě podmínky se opírají o intencionalitu, jednu před akcí a druhou během ní, a intencionalita předpokládá vědomou mysl. (Není tedy nutné se odvolávat na člověka se vším všudy: budeme-li považovat bobry za vědomé intencionální bytosti, můžeme jejich vodní díla nazývat umělá jezera. Avšak neodbíhejme od tématu.)

Umělý člověk. Podle toho, co bylo řečeno, jsou roboti umělí lidé, jakkoli se nám to nezdá. Háček je v tom, že mezi podstatnými vlastnostmi člověka, které bychom rádi viděli také jako součást záměru, je i mysl a vědomí (vnitřně prožívaná subjektivita). Předpokládáme to u tvůrce, proč to nežádat i u výtvoru? Může však vědomí být součástí záměru, když je nelze *objektivně* specifikovat, zadat jako úkol (známe je každý jen u sebe sama)? Vdechovat duše prostě nějak neumíme.

Maximálně lze učinit malý úkrok stranou. Dovolíme konstruktérovi robotů, aby svá díla nedokončoval, ale aby místo toho do nich vložil schopnost se dále vyvíjet, učit se, hromadit zkušenosti z fyzického světa, svobodně komunikovat s jinými roboty (nejen s lidmi) a kdovíjak se ještě vylepšovat. Lidé aspoň takové vlastnosti mají. Co když roboti, možná podobně jako lidé, teprve postupně, sami od sebe naberou rozum i s vědomím? Nebude to umělý rozum, ale nelze mít všechno umělé. Můžeme tomu říkat *emergentní* rozum. Čapkovi Roboti, jak známo, se naučili ještě cosi navíc.

On to ostatně úkrok stranou ani není, je to běžný způsob, jak autoři vědecké fantastiky zajišťují dokonalost a dramatickou nepředvídatelnost robotích populací. Ale i v realizující robotice a umělé inteligenci je učení, vývoj, simulovaná evoluce, emergence a mnohé další věci předmětem rostoucího zájmu posledních dvaceti let. Zájem je to oprávněný a otevírá otázku, do jaké míry je to krok od umělého k přirozenému. Parafrázuji Zdeňka Neubauera: „Vždyť co jiného činí každého živáčka tím, čím je, než způsob, jakým povstal, jeho vznik – vzcházení, růst, vývin! Přirozeně jsoucí jest samo svým vznikem a ten je jeho přirozeností, vlastní povahou."[15]

[15] Volně citováno podle (Neubauer, 2004).

Vztah umělého a přirozeného bychom mohli ovšem relativizovat: nejsou-li vodní díla bobrů umělá (viz předchozí text), lze je zařadit mezi jsoucna přirozená jen za podmínky, že je budeme považovat za součást „rozšířeného" těla bobra. Konec konců i robot by mohl být s dávkou smělosti považován za součást těla svého konstruktéra – označili bychom však takovýto konglomerát jako něco přirozeného?

V našem úkroku stranou jsme dali konstruktérovi možnost své dílo nedokončit a uvolnit je k vlastnímu vývoji. Všimněte si dvou specifických důsledků, které z toho vyplývají: jednak, že roboti by mohli či měli mít šanci si *sami pro sebe* obhlížet a ohmatávat okolní svět (což souvisí s jejich tělesností), a jednak, že by mohli či měli mít možnost se *sociálně sdružovat* (což souvisí se schopností vzájemně komunikovat). Tím by si mohli budovat svou vlastní, „robotí" verzi reálného světa neovlivněnou lidskými předsudky, iluzemi a jistotami, které jsme jinak náchylní jim trpělivě předžvýkávat. K oběma důsledkům bych rád něco poznamenal.

## 1.7 Obhlížený, ohmatávaný – ale i prožívaný svět

Robota lze vsadit (či vysadit) do reálného světa a naprogramovat nebo nacvičit jej, aby se choval tak, jak by se v dané situaci choval člověk, či přesněji, jak si experimentátor myslí, že by se choval člověk. Robota lze naučit nejen správně tahat figurkami po šachovnici, ale i vyhrávat. Roboti dokážou kráčet po schodech, hrát fotbal, montovat automobily, vozit se po Marsu, tvářit se nešťastně, mluvit. Něco však chybí. Jaksi nedovedeme – či ještě hůř: nevíme, zda nedovedeme – je přimět, aby to vše prožívali *vnitřně* tak jako my, či aspoň podobně jako my: nejen kráčet, ale i *chtít* kráčet, něco opravdu *zakoušet* ve chvíli, kdy dají mat, dostanou gól nebo smontují automobil. Aby je toulky po Marsu opravdu *bavily*. Aby se nešťastně nejen tvářili, ale aby přitom *nešťastní byli*. Říkají-li něco, aby to *mínili*. A vůbec, aby věděli, *že jsou*.

Máme být spokojeni s tím, co Josef Čapek kdysi popsal takto: „[vytvářet] na lidskou podobu pouhé stroje, které po některých funkcích člověka jen napodobují, aniž by se podjímaly též úkolu prožívati pocity libosti a nelibosti, jimiž definováno jest veškeré člověkovo konání" (Čapek J., 1924)?

Pravda, nikdo z nás nemá přístup ani k tomu, jak popsané situace v hloubi svého nitra prožívá kterýkoliv jiný člověk, nemáme přístup k fenomenálním dojmům druhého – k onomu „jaké-to-je" rozmanitých prožitků (když má druhý například potěchu z jarní přírody, nebo když je uchvácen hudební skladbou, abych uvedl impresivní příklady, ale platí to i pro prožitky spojené s vnímáním běžných předmětů). Nicméně již to, že mezi sebou sdílíme nejen vzorce chování, ale i jazyk, jímž lze naše vnitřní stavy (někdy) pojmenovávat, nám umožňuje, abychom nejen o okolním světě, ale i o svých prožitcích a zkušenostech přemýšleli a mluvili. Tak se (někdy) dají struktury „našeho" světa zobecnit a objektivizovat, na čemž je i postavena celá přírodověda.

Stvořitelé robotů vypouštějí své výtvory do světa nás, lidí, světa, který si zjednáváme naší lidskou myslí, který vnímáme našimi lidskými smysly, o němž mluvíme a rozmýšlíme pomocí našich, lidských pojmů. Všeobecně se soudí, že robot je tím lepší, čím více přijme tento *náš* svět za svůj a kdoví, snad i včetně *našich* pocitů při jeho prožívání. Nezdá se, že by se někdo trápil ostrým konfliktem mezi tímto naším, *lidským* světem (do

něhož mimochodem náleží i fyzické tělo robota) a světem nutně *ne-lidským* (pokud vůbec nějakým), který by si zjednával – a snad i prožíval – robot sám.[16] Obvyklý přístup odborníků je ovlivněn novověkou vědou: snažme se naše pojmy a výpovědi formulovat pokud možno „objektivně", nezávisle na subjektu, dokonce nezávisle na *typu* subjektu (zda je to člověk, robot, mimozemšťan či chobotnice). Jak jsem poznamenal v oddílu 1.5, i já se zde občas uchyluji k objektivistickému (meta)jazyku (jako bych mluvil ústy Absolutního Rozhodčího). Jinak bych ani nemohl pohodlně mluvit o nějakém konfliktu mezi lidským a ne-lidským světem.

Ve studii (Havel, 2007) jsem rozlišil některé stránky či dimenze lidského situačního prožívání – nazval jsem je *modality epizodických situací* – alespoň ty, které si prožívající subjekt[17] může v reflexi všímat a vzájemně odlišovat, přičemž lze buď klást důraz na jejich subjektivní stránku, anebo se naopak snažit o jejich objektivizaci „uzávorkováním" prožívajícího subjektu. Takovými modalitami jsou například: *čas* (subjekt jej prožívá jako plynutí a rozlišuje v něm minulost, přítomnost a budoucnost na rozdíl od objektivního fyzikálního času), *prostor* (v němž subjekt žije a v němž jedná na rozdíl od geometrického prostoru), *tělesnost* (subjektivně prožívaná jako vlastní tělo na rozdíl od různých pojetí organismu), *aktérství* (pro něž a při němž se subjekt svobodně rozhoduje na rozdíl od kauzálně interpretovaného chování), *vědomí sebe* (vlastní uvědoměné myšlení na rozdíl od předpokládaných neurofyziologických procesů v mozku), *smysl pro druhého* (co pro subjekt znamená druhá osoba na rozdíl od jiných objektů), *socialita* (diferenciace „my" na rozdíl od „oni") a další. Snadno si uvědomíme, že jednotlivé modality jsou navzájem složitě provázány a jejich odlišení je proto spíše teoretické (nicméně možné).

Při naznačeném rozlišení mezi subjektivní a objektivní stránkou uvedených modalit se objeví charakteristická dualita pohledů na ten či onen aspekt dané situace – dualita vnitřního oproti vnějšímu. Tak například čas, který vnitřně prožíváme v asymetrické triádě minulost–přítomnost–budoucnost, v níž „čistá přítomnost je neuchopitelným postupem minulosti hlodající budoucnost" (Bergson, 2003), považujeme vnějšně za něco objektivního, absolutního, homogenního a měřitelného. Stejně vnějšně lze pohlížet i na prostor, přičemž vnitřně prožíváme týž prostor spíše jako individuální pole otevřených možností. Podobná dualita mezi subjektivním a objektivním uchopením se dá předvést i pro mnohé jiné situační modality, jako je například tělesnost, jednání, smysl situace a vztah k druhým.

Tato dualita vede ke čtyřem alternativním postojům k ideálu dokonalého robota jako něčeho, co má být člověku co nejpodobnější.

*První postoj*. Nejde nám o nic víc než o vnější podobu (vzhled) a obdobné chování v obdobných situacích. Vnitřní, psychická stránka je buď irelevantní, nebo se o ní nemá, protože se nedá, mluvit. Tento postoj spoléhá na běžný způsob, jak člověk obcuje s druhými, aniž by se trápil tím, že nemá přístup do jejich nitra[18]; ve filosofické reflexi je to behaviorismus kombinovaný s metafyzickým realismem (díky němuž sám termín „situace" má absolutní, nikoliv pouze intersubjektivní význam, který nezávisí ani na subjektu, ani na jeho typu – zda jde o robota nebo člověka).

[16] K tomuto konfliktu viz též můj úvodník ve Vesmíru 2/2002, s. 63.
[17] Termín „subjekt" užívám spíše jen jako zkratku za „ten, který danou situaci prožívá jako svou".
[18] Srov. závěr oddílu 1.5.

*Druhý postoj.* Robot by měl být člověku[19] co nejpodobnější včetně toho, jak vnitřně prožívá svět a jaký je svět, který prožívá. Měl by ho prožívat stejně jako my, což – nehledě na obtíže s ověřováním – může narazit na zmíněný konflikt mezi ne-lidskou zkušeností robota a lidskou zkušeností člověka.

*Třetí postoj.* Robot by měl být člověku podobný v tom, že by měl (tak jako člověk) vnitřní prožitky, ty by však nemohly mít podobu lidských prožitků, protože by byly (tak jako je tomu u člověka) v souladu s jeho specifickou kovovou tělesností, s jeho původem a individuální historií. Myslím, že jedině při tomto postoji má logicky smysl mluvit o robotu, který by byl obecně „dokonalejší" než člověk (nemusel by čistě kvůli podobnosti být snížen na jeho úroveň).

*Čtvrtý postoj.* Zde robot má (jako člověk) vnitřní prožitky, avšak nejen situace (jako u prvního postoje), ale i prožívání všech modalit situací jsou objektivizovatelné, a to absolutně, nikoliv jen intersubjektivně (analogicky s představou absolutního času a prostoru). Jsou pak popsatelné přesně v té míře, v jaké jsou popsatelné u člověka. A navíc, protože na veškeré prožitky lze v tomto postoji nazírat objektivně, jako na nezávislé na prožívajícím, ba ani na tom, jaký typ entity je tento prožívající, lze klást rovnítko mezi svět robota a svět člověka, obojí včetně světa vnitřního. Pokud jsou rozdíly, jsou jen na úrovni konkrétních individuí (podobně jako jsou rozdíly mezi námi, lidmi). Místo lidské, resp. ne-lidské zkušenosti se dá mluvit jen o jediném, mimo-lidském popisu světa.

Tento čtvrtý postoj je problematický a zdá se, že i vnitřně sporný. Nelze se totiž zbavit skrytého předpokladu onoho mimosvětského Absolutního Rozhodčího, který svým „božím okem" nejen vše vidí, ale má i přístup do nitra všech našich duší.

## 1.8 Společenství robotů

Nyní k dalšímu aspektu, který je třeba vzít do úvahy. Budou se humanoidní roboty sdružovat do kolektivů, společenství, populací? Člověk je bytost sociabilní a kulturní a to, jak pohlíží na svět a sebe v něm, je podmíněno dílem evolučně a dílem kulturně (stupeň této podmíněnosti je předmětem diskusí). Netřeba připomínat, že jednou z hlavních rolí při shlukování jednotlivců do seskupení toho či onoho typu má jejich schopnost komunikovat, tedy především jazyk.

Roboty učíme rozumět lidské nebo počítačové řeči. Není obtížné si představit, že ponecháme-li jich větší množství pohromadě (mám na mysli trochu schopnější roboty blízké budoucnosti), nepochybně užijí tuto naučenou řeč ke vzájemné komunikaci a vzájemná komunikace je tak či onak, ale nutně, stmelí. I tehdy, když je k tomu nebudeme navádět. Literatura a filmy toho dávno využily k dramatickým scénám. Ve zmíněném filmu „Já, robot" dokonce jedna armáda robotů se pustí do křížku s jinou armádou robotů (v čemž jim známé Asimovovy zákony nijak nebrání – pokud se vztahují jen na individua).

Varianty postojů k otázce robota jakožto podoby člověka lze opakovat pro robotickou komunitu jako emergentní nebo programované individuum vyššího řádu. Má se kolektiv robotů chovat obdobně jako kolektiv lidí? Nebo nějakým svým vlastním

[19] Připomeňme si poznámku o singulárech v oddílu 1.5

emergentním způsobem? Nebo mají být natolik podobni lidem na úrovni jednotlivců, že se budou přirozeně vytvářet komunity smíšené?

Nemám v úmyslu se pouštět do fantazií, spíše mi jde o odhalení méně nápadných souvislostí. Pokud vnutíme robotům lidskou řeč (dřív než si oni vytvoří svou), vnutíme jim i velkou část lidského způsobu myšlení, dokonce i lidského světa. Mnohé filosoficko-antropologické názorové směry kladou velký důraz na to, že jazyk a kultura na jazyku postavená je primárním zdrojem našeho prožívání a zjednávání okolního světa (což úzce souvisí s druhým postojem z předchozí kapitoly).

Zdá se, že ten trend v robotice, který usiluje o humanoidní, člověku podobné roboty, jako by možná podvědomě byl motivován snahou přijmout budoucí populaci robotů jako součást populace lidské, nikoliv jako nějaké ne-lidské plémě zrůdných bytostí. Praktické důvody (například průmyslové aplikace) ani takovou amalgamizaci nevylučují. Není vyloučeno, že i to nějak souvisí s lidskou řečí. Má-li tato řeč (třeba jen navenek) nabývat významu opravdové komunikace (výměna názorů, pomoc při rozhodování v lidských záležitostech, snad i terapeutický rozhovor), bylo by celkem přirozené, aby se nám naši partneři v dialogu co nejvíce podobali i vzhledem. I dítě si docela rádo povídá s panenkou.

## 1.9 Co nás, lidi na tom všem vlastně láká?

Zvláštní přitažlivost, až posedlost tématem umělého člověka je evidentní a je společná vědcům, konstruktérům, spisovatelům, futurologům a filosofům. Stejně tak čtenářům románů, filmovým divákům a lidem vůbec. Několik výletů namátkou do minulosti v oddílu 1.3 ukázalo, že tomu tak asi bylo vždy, snad jen s tím rozdílem, že se to dříve rodilo spíše v hlavách výjimečných jedinců a nikoliv rozsáhlých pracovních týmů a že se to týkalo spíše jednotlivých výtvorů než populací androidů.

Co nás, lidi na tomto věčném tématu vlastně tolik láká?

Zprávy konstruktérů nemusí být zcela spolehlivé co do argumentů o jejich vlastních motivacích (je třeba vždy přesvědčit veřejnost, nadřízené a sponzory o smyslu a užitečnosti projektu). Nahlédněme například na webové stránky laboratoří vyvíjejících expresívní roboty (viz oddíl 1.3). Snadno si všimneme důrazu na přímé kontakty robota s člověkem: „Nové aplikační oblasti (domácnost, zábava, zdravotní péče) vybízejí ke konstrukci robotů, které by mohly interagovat a spolupracovat s lidmi a účastnit se jejich každodenního života." Roboti by měli nejen „vnímat a přiměřeně reagovat na přirozené lidské náznaky", ale též umožňovat „nezacvičeným lidským subjektům číst a interpretovat stejné náznaky u robota" [20] (vzpomeňme na mrknutí okem filmového robota). Roboti budou „pracovat spolu s lidskými partnery v našem žitém prostředí a budou sdílet týž pracovní prostor a prožívat tytéž vzorce myšlení a chování jako lidská bytost."[21]

To vše je jistě v pořádku, je však možné (a u nákladných výzkumných projektů časté), že veřejné argumenty o užitečnosti nekopírují skutečné původní motivace konstruktérů. Zkusme proto odhlédnout od těchto argumentů a podívat se na věc i jinak.

---

[20] MIT, cit. webová adresa.

[21] Univerzita Waseda, komentář ve starší webové stránce.

Nelze například vyloučit, že se v robotice spojuje a navzájem posiluje vícero lidských zvláštností, které se dávno odtrhly od svých v praxi užitečných konkrétních podob, pokud je kdy vůbec měly. O první takové zvláštnosti jsem se již zmínil v oddílu 1.2: o *čisté hravosti*. K té totiž patří i tendence zoufale vyhledávat co nejobtížnější úlohy, jak to lze demonstrovat na luštitelích hlavolamů, šachistech a čistých matematicích. Nedá mi, abych se nepřiznal, že čistou hravost považuji za vlastnost veskrze krásnou a mrzí mě, že hraví lidé jsou tak často nuceni předstírat rozmanité užitečné aplikace. Bezpochyby, tu a tam se z hravosti nečekaně zrodí něco takzvaně užitečného, v přírodě tomu ostatně bývá také tak.

Není proto divu, že čistá hravost, ve své tvořivé podobě, by si ráda vybrala úkol nejnáročnější – opakovat druhý největší Stvořitelův čin. Nejde ani o to, že by nás člověk (jakožto cíl projektu) vzrušoval tím, že je z tvorstva nejvznešenější a nejdokonalejší, stačí, že je to, slovy Ladislava Klímy, „nejkurióznější nestvůra zemská". A tu nastupuje druhá pozoruhodná zvláštnost člověka: neochabující *zvědavost*.[22] „Všichni lidé od přirozenosti touží po vědění," zní první věta Aristotelovy Metafyziky. Snad po jakémkoliv vědění – i vědění o člověku samotném. Zdali pak i my jsme něčím, co se dá poznat, čemu se dá porozumět? Konstrukce je dobrým testem porozumění sobě samému: buď se ukáže, že na tom nic není, čili že jsme v jádru věcí dosti prostoduchou a tedy srozumitelnou. Anebo to bude nesmírně náročné a obtížné, pak budeme opravdu pašáci, když se nám to nakonec povede.

Ona první, jednoduchá varianta měla vždy své zastánce. Namátkou cituji z jedné populární předválečné knihy: „Člověk je fyzikálně chemickou soustavou a musí být vysvětlen slovy fyziky a chemie. Je udělán z molekul a elektrických výbojů a má-li být vůbec jeho chování vysvětleno, tedy nechť je vysvětleno v tomto oboru." (Furnas 1939, s. 99). Máme důvod oslavovat: vše je jednoduché, srozumitelné, pochopitelné – a tedy i snadno ovladatelné.

Druhá alternativa, že je to náročné a obtížné, je choulostivá z jiného důvodu: nemá totiž daleko ke zpupnosti, která sama sebe posiluje. Z paradoxu člověka jako zadavatele, řešitele a cíle projektu v jedné a téže osobě vychází řešitel jako vítěz, hrdý na svou dovednost: vždyť dovede vytvořit i něco tak nekonečně složitého, jako je člověk sám. Zdeněk Neubauer ve své golemovské stati na jednom místě poznamenává: „Napětí tohoto vnitřně rozporuplného sebepochopení současného člověka, jako neomezeného vládce na jedné straně a neomezeného tvůrce na straně druhé, se projevuje též na rozpačité podvojnosti našeho vztahu k výtvorům kybernetiky; samočinné počítače jsou jednak dokonalé, poslušné nástroje, výraz lidského důmyslu, avšak zároveň – v robotice – tentýž důmysl přenáší nárok neomezených schopností i na své výtvory." (Neubauer, 2002).

K hravosti a zvědavosti přistupuje, jak již víme, další zvláštnost člověka, totiž že si rád *zahrává*. Nejsem si jist, koho všeho si nyní znepřátelím, ale nedá mi, abych nevyslovil dvě kontroverzní teze. Teze prvá: zahrávání je dobré, pokud přináší dobrým lidem radost. Teze druhá: všechny katastrofické scénáře o lidstvu, vybitém svými vlastními roboty, nejsou v té míře, v jaké jsou míněny jako varování, správně směřovány.

[22] Obecně nerad rozlišuji mezi zvědavostí a zvídavostí, protože to svádí považovat prvé za cosi neslušné. Přijatelnější by bylo chápat zvědavost jako touhu po jedinečných a zvídavost po obecných poznatcích. Pokud potřebujeme ten rozdíl.

Kdo si rád zahrává, dobře ví, že si zahrává, nemusí se mu to v jinotajích ani otevřeně připomínat.

Tím nechci říct, že hrající a zahrávající mají být neteční vůči následkům, které již nejsou v jejich moci. Chopí se jich druzí, kteří si už nehrají. Pravda, tito druzí se třeba ještě nenarodili a leckdo si proto může myslet, že nás se to ještě netýká – a po nás potopa!

## 1.10 Co když je všechno jinak?

Hravost, zvědavost a zahrávání lze přisuzovat lidským jedincům a stejně tak, chceme-li, lidské populaci jako celku – jakožto jedinci vyššího typu. Mnohé dobré, špatné i neutrální vlastnosti jedinců se dají rozpoznat u větších celků, u lidských komunit, ba i u celého lidstva, na které lze hledět, jako by to byla jedna velká superbytost (nekomplikujme si zde úvahy kulturní diverzitou lidské populace). Často čteme o touhách, obavách, náladách, rozumnosti, ztřeštěnosti, pýše, neskromnosti, touze po moci, respektu k minulosti, budoucnosti, přírodě… – to vše ve vztahu k celé lidské populaci, nikoliv k jednotlivým smrtelníkům. Vesměs to nejsou vlastnosti, ke kterým se lze dobrat nějakým průměrováním nebo sčítáním hlasů. Snad původně byly míněny jako metafory, ale co by to bylo za metafory, kdyby na nich přece jen něco nebylo.

Napsal jsem, že se ony vlastnosti celku se dají rozpoznat. Z toho ovšem neplyne, že je můžeme zakusit, vnitřně prožít, rozumět jim. Nejsem lidstvo, abych věděl, jaké to opravdu je, takové vlastnosti mít, stejně jako nevím, jaké je to být vystrašeným netopýrem anebo sebevědomým politikem. Jsem molekulou oné lidské populace, ale nevím, zda a jaký vliv mohu já mít (i kdybych se domluvil s ostatními molekulami) na to, jaká populace je. Mnozí si to myslí a možná, že i nějaký vliv mají, ale ten je třeba i opačný, než si myslí. Nedivil bych se, kdyby to dokonce mnohdy šlo opačným směrem, kdyby to byly vlastnosti lidstva, které se na nás, molekulách, podepisují.

Touhu lidstva (mám na mysli „západní" civilizaci) rozvíjet průmysl a zavádět technické inovace lze docela snadno redukovat na touhu jednotlivců přežít v náročném světě. Tak by snad bylo i možno vykládat snahu konstruovat průmyslové a jiné užitečné roboty. Ale jak je to se snahou vytvořit umělého člověka?

Spekulujme. Možná, že si lidská populace uvědomuje svou konečnost. Má k tomu množství důvodů, jsou toho i plné noviny (a vždy byly). Třeba se nenápadně, aniž bychom k tomu vědomě přispívali, rodí v lidstvu zoufalá pudová touha po nějakém novém, radikálním způsobu reprodukce, po nějaké nové populaci, které by se lépe přežívalo ve světě, který je *pro ni* reálný, než nám ve světě našem. Samozřejmě bychom rádi (teď mluvím za lidstvo i za lidi), aby ta nová populace měla co nejvíc *naši podobu* a aby se, čerstvá a vnímavá, učila *našim* moudrostem, *našemu* chování a *našim* zvykům.

Proč nezačít od jednotlivců – molekul nové populace? Budou-li podobni nám, lidem, může být i jejich populace a jejich svět podoben naší populaci a našemu světu. Člověk se vytratí, jeho svět tu zůstane.

## Literatura


Bergson H.: *Hmota a paměť*. OIKOYMENH, Praha 2003, s. 112.

Bůcha J.: „Robotí kopaná" Vesmír, č. 12, 1999, s. 687.

Čapek J.: „Umělý člověk", In: *Ledacos. Umělý člověk*. Dauphin, (pův. Lidové noviny, 1924), Praha 1997, s. 171–229.

Čapek K.: Autor robotů se brání, *Lidové noviny*, 1935 (přetištěno v F. Buriánek: *Karel Čapek*, Melantrich, Praha 1978).

Dreyfus H.: *What Computers Can't Do*. (2. vydání), Harper & Row, New York 1979.

Furnas C. C.: *Příštích 100 let pokroku ve vědě*. Kvasnička a Hampl, Praha 1939, s. 99.

Gebelein H.: *Alchymie – magie hmoty*, Volvox Globator, Praha 1998, s. 52.

Havel I. M.: *Robotika – Úvod do teorie kognitivních robotů*. SNTL, Praha 1980, kap. I. 1, I. 4.

Havel I. M.: „Budoucnost jako akční thriller", Vesmír, č. 3, 1999, s. 123.

Havel I. M.: „Přirozené a umělé myšlení jako filosofický problém." In: *Umělá inteligence III*. (V. Mařík a kol., editoři), Academia, Praha 2001, s. 17–75.

Havel I. M.: „O proměňování a proměnách. Pokus o studii proměnlivě heterofenomenologickou." In: *Charisteria Sidonio Neubauero Sexagenario* (ed. J. Fiala) Pragae 2002, s. 106–140.

Havel I. M.: „Skryté dilema robotiky", Vesmír, č. 12, 2004, s. 663.

Havel I. M: „Prožívání epizodických situací." In: *Myseľ, inteligencia a život* (V. Kvasnička, P. Trebatický, J. Pospíchal a J. Kelemen, eds). Vydavateľstvo STU, Bratislava 2007.

Hibbard B.: *Super-Intelligent Machines*. Kluwer Academic Publishers, 2002, s. 59.

Knížák M., Demšáková T. (ed.): *Bude konec světa?* Národní galerie, Praha 2000 (angl. mutace *The End of The World?*).

Moravec H.: *Robot: Mere Machine to Transcendent Mind*. Oxford University Press, 1999a.

Moravec H.: *Scientific American*, Dec. 1999b.

Neubauer Z.: *Golem a další příběhy o kabale, symbolech a podivuhodných setkáních*, Malvern, Praha 2002.

Neubauer Z.: „Svatojánský výlet II", díl 3, In: *Bytí je vlnobití*. Středokluky 2004, s. 223.

Villiers de l'Isle-Adam P. A. M.: *L'Eve future*. Praha 1920.

Warwick K.: *Úsvit robotů, soumrak lidstva*. Nakl. Vesmír, Praha 1999.

Weizenbaum J.: *Mýtus počítače* (ed. J. Fiala), Praha 2002, s. 110.

# 2 Robot – stroj a metafora 20. století[1]

## 2.1 Úvod

V této kapitole načrtneme některé – dle našeho mínění a z hlediska tematiky této knihy zajímavé – posuny v chápání významu pojmu *stroj*. Budeme se věnovat fiktivním strojům, s jakými se setkáváme v bájích, v mytologii, ve vyprávěních lidstva. Stejně tak však zaměříme pozornost i k strojům, které lidé dovedli nebo dnes již dovedou technicky zkonstruovat, některé matematicky rigorózně definovat a vlastnosti takových strojů i exaktně, s matematickou přesností zkoumat. Soustředíme se zejména na proměny obsahu pojmu *stroj*, ke kterým došlo v průběhu 20. století v oblasti umělecké kultury, v oblasti (výpočetní) techniky, teoretické a experimentální informatiky a v oblasti pokročilé robotiky[2]. Tyto transformace stroje přiblížíme prostřednictvím relevantních děl osobností, jakými byli v kultuře a vědě 20. století výtvarně a literárně nadaní Karel a Josef Čapkovi, britský matematik Alan Turing, američtí vědci původem z Maďarska John von Neumann a Aristid Lindenmayer nebo Američané Herbert A. Simon a Allen Newell a mnozí jiní.

Čtenáře jistě nepřekvapíme konstatováním, že představa stroje na začátku 20. století byla podstatně odlišná od představy stroje o století později. Co snad překvapí trochu více, je odpověď, kterou si dal na vlastní otázku *Jsem stroj, nebo jsem člověk?* v roce 1988 na přednášce na festivalu Ars Electronica v rakouském Linzi francouzský postmoderní filosof Jean Baudrillard: *Reálně a subjektivně jsem člověk, virtuálně a prakticky jsem stroj. To je charakteristika stavu antropologické nejistoty...* (Baudrillard, 1989). Začátkem 20. století by byl takovýto názor prezentován jako futuristická provokace nebo společenská satira a představa *člověka-stroje* by u širší veřejnosti vyvolávala nejspíše negativní asociace.

Dnes už je zmíněná otázka něčím jiným než provokací. Je jakýmsi řečnickým úvodem k analýze stavu antropologické nejistoty, o které Baudrillard mluví. Z této nejistoty pramení naše potřeba zkoumat hranici mezi koncepty člověka a stroje a současně i obecná tendence zpochybňovat tuto hranici jako něco, co je jasně dané a neměnné.

---

[1] Tato kapitola byla napsána na základě textu přednášky autorů proslovené 12. prosince 2003 v Budapešti na *Mezinárodní konferenci ke 100. výročí narození Johna (Jánose) von Neumanna* a jejich tutoriální přednášky *Robot – A Word and a Machine in the World* proslovené 12. září 2004 v rámci programu *9. mezinárodní konference o simulaci a syntezi živých systémů* (*ALIFE IX*) v Bostonu. Je zpracována využitím textů Jany Horákové a Jazefa Kelemena z let 2003, 2004, 2005.

[2] Oblast nazval takto, novotvarem *robotics*, Isaac Asimov v roce 1942 v povídce *Runaround*. Ta se stala později součástí (druhou kapitolou) jeho proslulé sbírky povídek *I, Robot* (Asimov, 1950).

V čem hledat příčiny tohoto rozechvění konceptů člověka a stroje? Již na první pohled je zřejmé, že nebudou tkvět asi jenom v tom, že se technicky podařilo překonat parní stroj, ani v tom, že jsme dokázali „mechanizovat" početní úkony. Jde spíše o kulturní posun. Odkud, proč a kam? To jsou otázky, o kterých se pokusíme uvažovat.

## 2.2 Zlaté otrokyně, parní stroje a revoluční proletariát

Již před tisíciletími jedna z tehdy vznikajících kultur kdesi na geografickém rozhraní Afriky, Asie a Evropy přemýšlela o tom, jak asi mohl vzniknou první člověk. V kontextu této otázky si uvědomila dvě skutečnosti dalekosáhlého dopadu – uvědomila si, *z čeho* tvořila předměty své denní potřeby, a současně i to, *jak* je tvořila. Tyto řemeselné zkušenosti promítla do příběhu stvoření prvního člověka a tak můžeme v *Genesis*, v první Mojžíšově knize *Starého zákona* (kapitola 2, verš 7), číst:

*I učinil Hospodin Bůh člověka z prachu země...*

Technologická vynalézavost lidstva se utěšeně rozvíjela a tak o něco později mohl Homér v eposu *Ílias* (VXIII, 416–420) budovat svou představu o Hefaistově božské i kovářské dovednosti již na bázi technologie zpracování ušlechtilých kovů:

*Oblékl šat, vzal pádnou hůl, pak ze dveří vyšel,*
*kulhaje, šly s ním služky a byly mu podporou v chůzi,*
*ze zlata – živým dívkám však úplně podobny byly:*
*v prsou rozum jim jest, jest hlas v nich, v nich je síla,*
*znají i umělé práce, jimž věční je učinili bozi.*[3]

Rovněž Aristoteles ve svém spisu *Politika* (kniha 2, kap. 4, 33–39)[4] uvažoval o strojích a nástrojích, které by jednou nahradily člověka v práci: „... *kdyby každý nástroj na rozkaz nebo již předem dovedl vykonati své dílo,... kdyby člunky samy od sebe tkaly a paličky hrály na kitharu, nepotřebovali by stavitelé pomocníků ani páni otroků.*"

Aristoteles uvažoval o strojích nejen v souvislosti s představou umělého člověka (jako v uvedeném příkladu Homér), nýbrž chápal vývoj techniky, strojů, jako prostředek uskutečnění společenských utopií. Snad jako první poukázal na obrovský dopad toho, co bychom dnes mohli nazvat vědeckým a technickým pokrokem, na strukturu a vývoj společnosti. Reálné i konceptuální propojení stroje, člověka a společnosti dosáhlo nebývalé intenzity a s rozvojem informačních technologií i nové dimenze.

Evropa druhé poloviny 19. století již rozhodně nebyla otrokářskou společností a překonala i středověk. Obcházelo jí však, zejména v jejích vyspělejších průmyslových částech, dle první věty úvodu *Komunistického manifestu* vydaného Karlem Marxem v Londýně v roce 1848, zcela konkrétní a ne málo nebezpečné „strašidlo komunismu". Strašidlo vyvolané snad trochu paradoxně právě střety oněch aristotelovských „dokonalých nástrojů", třeba parních strojů 19. století, a těch, kterým měla jejich existence pomoci, tedy zvláštní podoby otroků té doby – proletariátu. Zejména po úspěšné

[3] Překlad O. Vaňorného.
[4] V námi použitém překladu (Aristoteles, 1998) na s. 43.

proletářské revoluci, která proběhla v Rusku (1917) a která v politickém i fyzickém smyslu skoncovala s carismem, se zájem evropských intelektuálů zaměřil zčásti na velebení takovýchto počinů a zčásti na přemýšlení o situacích, ke kterým mohou podobné počiny společnost přivést.

Jednou z krátkých, ale poučných úvah o možnostech a perspektivách takových experimentů s uskutečněním utopických společenských řádů je kratická povídka Josefa a Karla Čapkových *Systém* z roku 1908 (později zařazena do sbírky *Krakonošova zahrada*, shrnující prózy obou bratrů z let 1908–1912). Děj pojednává o tom, jak plantážník a továrník Ripraton zavedl ve svém továrním komplexu Hubertstownu tzv. „kulturní reformu" spočívající v redukci kulturních potřeb dělníků a unifikaci jejich životů. Autoři ilustrují i kritizují dobové reformní vize výrobního procesu, o kterých se později zmínil například i Josef Šafařík ve své studii *Člověk ve věku stroje*, a to takto: „*Henry Ford vysvětloval, že se mu podařilo přivést své továrny k dokonalosti a optimální výkonnosti tím, že – jak se vyjádřil – vypudil z nich všecko umění, čímž mínil, že nahradil živou bytost automatem. V témže smyslu hlásal Taylor, že ideálem průmyslové ekonomie je stroj, k jehož obsluze by stačila vycvičená opice nebo i člověk duševně méněcenný.*" (Šafařík, 1969). Přesně v duchu těchto myšlenek přikyvují posluchači a současně „spolutonoucí" panu Ripratonovi, který jim předkládá svou kulturní reformu dělníka: „*A přece existuje ideál dělníka. Je to jacquard, je to setrvačník, selfaktor, rotačka, je to lokomotiva. Jacquard, pane, nechce soudit a vládnout, nespolčuje se a neřeční; jeho jediná idea, ale silná, pane, veliká a vůdčí idea je: nitě, co nejvíce nití!*" (Čapek J., Čapek K., 2000).

Řád Ripratonovy továrny založený na „totální rovnosti" jejích zaměstnanců (rozuměj unifikaci skrze minimalizaci jakýchkoliv individuálních potřeb) se rozpadne následkem nepatrné organizační nepozornosti: Vzplanutím plamínku svíčky ozářivší tvář ženy, která byla jednomu z dělníků, jak bylo zvykem, dodána za účelem pracovní výkon zvyšujícího a ekonomicky tedy prospěšného pohlavního styku. Ten zhlédne v dokonale uniformním prostředí individuální jinakost a co se nestane: „*Tím v něm procitl smysl pro krásu a vyšší určení člověka, probudily se v něm jemnější city, a již druhého dne přes pokárání dozorců počal prozpěvovati, všelicos kresliti, usmívati se a sníti, mluviti a významně se tvářiti, a vůbec dávati najevo své city humánnějším způsobem.*" (Čapek, 2000).

V Ripratonových dělnících se vzedme vlna odporu k zavedenému systému. Rozklad tohoto umělého řádu vede přes různé inovace humanizující život pracujících a přes různé společenské reformy (zavedení tisku, osvětových programů apod.), aby nakonec kulminoval vyvražděním Ripratonovy rodiny včetně dětí. Otevřený konec povídky ukazuje vůdce povstání barbarského Gibbona, jako nového Adama, který byl na „scestí" svobodného života sveden onou ženou ve světle svíčky. Už jenom krůček, ale krůček velmi významný, zbývá učinit Karlovi k divadelní hře *R. U. R.*, která do slovníků vyspělejších jazyků celé naší planety vložila Josefovo slovo *robot*.[5]

[5] Více k interpretaci *R.U.R.* jako hry o utopické společnosti a řádu, kterého je dosaženo prostřednictvím vyspělých strojů – robotů, viz (Horáková, 2004a).

## 2.3 Nástup robotů

V případě Ripratonových dělníků, formovaných kulturní převýchovou v „*konstrukci typu Operarius utilis Ripratoni*" (Čapek J., Čapek K., 2000), podobně jako v případě pozdějších umělých dělníků-robotů, třeba těch vyráběných továrnou R. U. R., jde o obraz člověka *redukovaného* na pracovní výkon, na fyzickou zdatnost a především na výkonnost. Čapkové ironicky parafrázují dobový ideál dělníka a ekonomické teorie určované měřítkem stroje-mechanismu: „*Rozlétlo se zde světem osvobozující heslo, že člověk se má na své vzestupné dráze více přiblížiti a připodobniti Strojům, jež jsou dnes, za povšechně skvělého pokroku techniky a věd, ze všeho lidského stvoření a tvoření tím nejdokonalejším.*" (Čapek J., 1985).

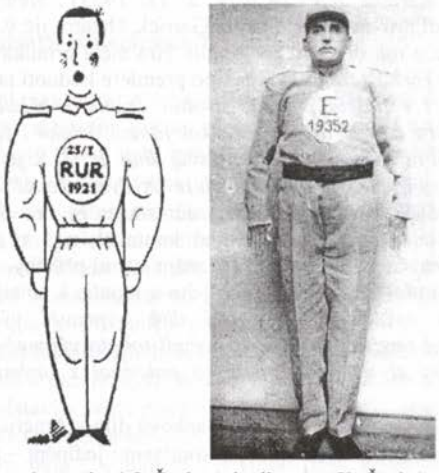
Hru *R. U. R.* psal Karel Čapek v létě 1920 během dovolenkového pobytu u rodičů v Trenčianských Teplicích. Dle jeho vzpomínek mu tam jeho bratr Josef nabídl, aby nahradil svou původně zvažovanou, ale dle jeho vlastního hodnocení poněkud „*papírově znějící*" pojmenování umělého dělníka *labor* (slovo latinského původu avšak s poměrně prvoplánovými politickými konotacemi), pojmenováním *robot*, které má slovanské kořeny, ovšem s velmi hluboko do historie zasahujícími konotacemi, slovem, které zpřítomňuje představu všech otroků, sluhů, nevolníků, dělníků, zkrátka všech těch, kteří se nechávají najímat na práci.

Metoda převýchovy („kulturní reformy"), kterou v povídce *Systém* aplikoval továrník Ripraton, aby formoval své dělníky směrem k ideálu – ke stroji – byla v dramatu *R.U.R.* dovedena k absurdní dokonalosti. Čapek prezentuje tovární systém, ve kterém jsou dělníci nahrazeni svou umělou, a ve své účelné jednoduchosti dokonalou, náhražkou – roboty. Jeho ideální továrna je továrna bez dělníků! Autor takto ironizuje všechna dobová radikální řešení vzrůstajícího napětí mezi továrníky a dělníky. Aby odhalil nereálnost těchto vizí, používá jejich vlastní zbraně a předkládá natolik radikální řešení, že ve své nekompromisnosti překonává i ty nejodvážnější reformátory.

I tito umělí dělníci-stroje-roboti se však v určité fázi své emancipace svým tvůrcům a pánům vzepřou svou vší sílou a „ne-lidskostí", které jim byly (i když nechtěně) dány do vínku a skoncují s lidstvem, jež paralelně s vývojem robotů spěje svou degenerací k zániku. Vzpoura robotů zánik lidstva nezpůsobila, pouze urychlila. Poslední žijící člověk ve hře *R. U. R.* Alquist je však ještě svědkem toho, jak vzplane láska – emoce spojovaná s lidským rodem a s lidstvím – mezi robotem Primem a robotkou Helenou. To mu dává naději, že civilizace Robotů bude dál rozvíjet dědictví svých otců, tedy civilizace lidské.

Místo a datování premiéry hry *R. U. R.* je trochu problematické. V pražském Národním divadle byla hra poprvé představena 25. ledna 1921. Hru však inscenovala již o tři týdny dříve Jednota divadelních ochotníků Klicpera na prknech Městského Klicperova divadla v Hradci Králové, a to 2. ledna 1921. Tuto zvláštní premiéru před oficiální premiérou režíroval inspektor Státních drah Bedřich Stein. První pár robotů hráli F. Paclt (Prima) a J. Skorkovská (Helenu). Obrazová dokumentace se z této premiéry bohužel nedochovala. V pražském Národním divadle se ujal režie Vojta Novák, scénu navrhl Bedřich Feuerstein a kostýmy sám autor slova *robot* – Josef Čapek, pro kterého to bylo první setkání s divadlem v roli výtvarníka. Role robotů Prima a Heleny hráli Eduard Kohout a Eva Vrchlická, která hrála i Helenu Gloryovou.

Josef Čapek zvolil pro roboty-dělníky velmi střízlivé a pouze mírně stylizované kostýmy v podobě modrých pracovních kombinéz, které byly jednoduše odvozeny od podoby pracovního oděvu továrního dělníka. Postava robotky písařky Sully měla obdobně neutrální ženský kostým služky. Ne-lidská podstata robotů byla vyjádřena pouze hrou herců – strnulými výrazy tváří a sekanými pohyby. Jak je uvedeno ve scénických poznámkách k dramatu: „*Roboti v předehře oblečeni jako lidé. Jsou úsečni v pohybech i výslovnosti, bezvýrazných tváří, upřeného pohledu. Ve vlastní hře mají plátěné blůzy v pasu stažené řemenem a na prsou mosazné číslo*" (Čapek K., 1992). Roboti jsou v této své podobě ztělesněním obrazu dělníka, jak jej evokují Čapkové již v *Předmluvě* sbírky jejich juvenilií v odstavci nadepsaném *Velkoprůmysl*: „*Bezpočetně proudí modré blůzy páchnoucí strojem: jutou, olejem, kovem, škrobem a prachem tkaniva. Jsou cítit jedinou člověčinou, a to je pot.*" (Čapek J., Čapek K., 1982) *Proud modrých blůz* evokuje odlidštěnost a zaměnitelnost – charakteristicky spojené s časovým typem dělníka – a odráží v sobě aktuální situaci v oblasti vztahů zaměstnanec–zaměstnavatel i dobové ekonomické a politické ideologie (taylorismus, fordismus, socialismus, komunismus a některé další politicko-společenské ismy).




Obr. 2.1 Návrh kostýmu robotů J. Čapka (s karikaturou K. Čapka) a realizace návrhu z pražské premiéry.

Jestliže tedy Karel Čapek prezentuje v textu své hry roboty jako továrním systémem sériově produkované výrobky, jako komoditu, jde vlastně o metaforické pojmenování motivu dělníka jako zaměnitelné jednotky v soukolí systému továrny.[6] Postava robota však v sobě obsahuje také koncept ideálního dělníka moderní doby, který má podobu *živého a inteligentního pracovního stroje* (Čapek, 1992) plnícího příkazy bez výhrad a bez zaváhání, s rychlostí a s přesností automatu. Proto se také stroj, automat, stává ideálem, ke kterému je dělník (a nejenom dělník, ale obecně pracující člověk průmyslové civilizace) přirovnáván.

[6] Viz povídka *Jednotka* z roku 1910, např. v (Čapek J., Čapek K.,1982).

## 2.4 Jak se robot měnil v stroj

Hra *R. U. R.* – i když je často považována za dílo, které nepatří mezi autorovy nejzdařilejší výtvory – měla úspěch nejen doma, ale i v zahraničí a záhy se dostala na scény divadel po celém světě (v New Yorku byla premiéra v roce 1922, v Londýně v 1923, ve Vídni v 1923, v Paříži v 1924, v Tokiu v roce 1924, ...). Důvod světového úspěchu vidí Otokar Fischer právě v Čapkově „amerikanismu", který v *R. U. R.* dospívá „*...svého vyvrcholení; a spolu svého překonání. (...) Řešení však a východisko této lidské tragédie ukazuje kamsi docela jinam než směrem k určitému středisku lidské civilizace: ukazuje do oblastí všelidských, všecitových a tím i nadhvězdných.*"[7] Tak, jak hra putovala světem a vstupovala do různých kulturních kontextů, měnila se pozvolna i její interpretace. Všelidské a univerzální hodnoty, které se v ní objevují (v podobě motivů tajemství života, stvoření člověka a dalších), jí dodávaly univerzální kosmopolitní charakter, který třeba mnohonárodnostní americké publikum jistě ocenilo.[8]

Hra byla americké veřejnosti premiérově představena na scéně divadla specializujícího se na moderní drama, *které předvádí pouze hodnotné umělecké hry pro úzký kruh vybraného obecenstva* (*Americké dělnické listy*, 2. 11. 1922)[9]. Newyorský divadelní soubor *Theatre Guild* uvedl hru na scéně divadla Garrick Theatre již 9. 10. 1922,[10] byla zde tedy uvedena dokonce o rok dříve než v Anglii. Hra měla pronikavý úspěch. Jeden z recenzentů ji v *The New York Evening Sun* den po premiéře hodnotí takto: „*... podobně jako dříve H. G. Wells, i v našem případě uvolnil dramatik křídla své imaginace a umožnil jí volně létat, aby dopřál divákovi rozkoš jdoucí daleko i za nejsmělejší sny Julesa Verna. Guild splatil návštěvníkům divadla svůj dluh. R. U. R. je supermelodrama plné akce a myšlenek v poměru, který je na našich jevištích zřídkavý.*" Robert Wechsler, který referoval o osudu Čapkova díla v USA, zdůrazňuje, že hra *R. U. R.* vstoupila v Americe do odlišného politického a kulturního kontextu, než ve kterém vznikala v tehdejším Československu. Čapek nebyl v USA znám svými příběhy, ani jako novinář, neznali jeho rané hry, jeho překlady poezie nebo jeho sympatie k americké pragmatické filosofii. *Pro Američany byl Čapek zázračné dítě, vypravěč výmyslů a tvůrce podívaných.*(...) *Čapek byl navždy chápán na pozadí tohoto výjimečného debutu. Byl autorem podívaných, který si náhodou mimo to pohrával s jinými, nižšími žánry.* (Wechsler, 1992).

Změněná perspektiva, ze které bylo vnímáno Čapkovo dílo v Americe, má své kořeny již v prvním – a na dlouhou dobu na americkém kontinentu jediném – překladu *R.U. R.* do angličtiny, jehož autorem je Paul Selver. Již v tomto překladu je výrazně zredukována alegorická rovina nesená typizovanými postavami hry a jejich „mluvícími jmény". Americký čtenář (a zřejmě i divák) tak přicházel o jednu velmi významnou rovinu hry, která se váže na autorův původní záměr – prezentovat jakýsi mozaikovitý obraz lidstva skrze skupinu tvořenou různými lidskými typy.[11] Nepochopení role postav v celkové kompozici hry potvrzuje i skutečnost, že inscenátorům se zdálo nezbytné *udělat několik*

[7] Viz zn. OT.F. (Fischer, O.) v *Národních listech* z 27. 1. 1921 nebo v (Čapek K., 1966).
[8] K osudu hry *R. U. R.* a *Ze života hmyzu* v USA. viz článek (Ambros, 1996).
[9] Citováno podle (Čapek, 1966).
[10] Režie: Philip Moeller, scéna a kostýmy: Lee Simonson, překlad: Paul Selver a Nigel Playfair.
[11] První kompletní a správný překlad hry vytvořila poměrně nedávno Klaudia Novack-Jones na zakázku Divadelního oddělení Brandeisovy University ve Waldhamu, Mass. Tam bylo *R. U. R.* uvedeno Michaelem Murreyem v dubnu 1993.

*úprav (...). Hlavní z nich bylo odstranění dvou jmen ze seznamu postav, Fabryho a Helmana* (Harkins, 1962). Tyto úpravy, které byly zcela v duchu realistické tendence typické pro tehdejší americkou literární produkci, nutně vedly ke zvýraznění dějové linie a naopak k potlačení funkcí jednotlivých typů postav v celkové kompozici hry. V tomto duchu přečetl hru také William Harkins, autor první anglické monografie o díle Karla Čapka: *„R. U. R. jako směs realistického a symbolického stylu je selháním (at fault). Zpětně můžeme usoudit, že románový hektický melodramatický styl nebyl kongeniální s Čapkovým skutečným talentem"* (Harkins, 1962). Harkins mluví v souvislosti s *R. U. R.* o *melodramatickém stylu* (Harkins, 1962) a José Ortega y Gasset konstatuje, že *figurína*, kterou se postava robota zbavená své minulosti bezesporu stává, *„...je melodrama ve své nejčistší podobě"* (Ortega y Gasset, 1994) V jedné z amerických recenzí je Karel Čapek nazván *českým Frankensteinem* (Corbin, 1922) a roboti tak nahrazují ve 20. století známé přízraky 19. století. A protože gotický román, ve kterém se tyto přízraky objevují, byl významným zdrojem repertoáru bulvárních scén první poloviny 19. století, vracíme se opět k melodramatu.

## 2.5 Vstup robotů do science fiction

Paralela mezi Čapkovou hrou o robotech a *Frankensteinem* Mary Shelleyové nás vede ke konstatování, že zbavením postav robotů jejich typové minulosti, ke kterému došlo v kulturním prostředí USA, byla zdůrazněna jejich funkce fiktivního vynálezu nebo vědeckého experimentu, který rámuje dobrodružný děj hry *R. U. R.* Právě s tímto principem výstavby fabulační linie, vycházejícím z předpokladu fantastického vědeckého vynálezu, proniká do gotického románu Mary Shelleyové i utopické hry Karla Čapka poetika žánru *science fiction*. Obě díla jsou navíc uváděna v knihách mapujících historii science fiction jako významné mezníky: *Frankensteina* řada teoretiků považuje za vůbec první science fiction a 19. století počítají jako *„první po Frankensteinovi"* (neboť fantastika gotického románu zde byla poprvé spojena s vědeckou hypotézou). *R. U. R.* je pro změnu uváděno v publikacích věnovaných historii science fiction proto, neboť v této hře se poprvé představily světu postavy robotů, které brzy opustily Čapkovu hru a asimilovaly do americké (a nejen americké) populární kultury, jejíž součástí americká produkce science fiction byla.[12] Zatímco v Evropě byla vědecká fantastika ve dvacátých a třicátých letech dvacátého století spojená s modernistickými uměleckými experimenty,[13] američtí autoři produkovali přibližně ve stejné době nezměrné množství výpravných dobrodružných příběhů ozvláštněných nejrůznějšími vědeckými vynálezy. Na rozdíl od respektovaných autorů evropské produkce vědecké fantasticky měly tyto texty podobu populárního lehkého čtení, orientovaného především na mládež, a uznání

[12] Viz tvrzení Roberta Wechslera: *„Čapek byl chápán jako významný, dokonce téměř ústřední, představitel historie sci-fi, když ne z žádného jiného důvodu, tak proto, že vymyslel jednu ze základních postav sci-fi, robota"* (Wechsler, 1992). Darko Suvin věnuje ve své analyze vývoje sci-fi (Suvin, 1979) Karlu Čapkovi celou samostatnou kapitolu.

[13] Karel Čapek, Jevgenij Zamjatin a Herbert George Wells jsou považováni za autory vědeckofantastické literatury, která je protipólem americké sci-fi produkce té doby. Evropská science fiction (zejména ruský a sovětský futurismus) je od počátku spojena s modernistickými uměleckými experimenty. V USA se oproti tomu produkce science fiction jen velmi pozvolna dopracovala uznání literární kritikou.

literární kritiky si museli teprve později těžce vydobývat.[14] Čapek byl americkou odbornou veřejností nejdříve přijat jako „středoevropský Wells" využívající vědecko-fantastickou hypotézu ve službách satirické hyperboly poukazující k dobovým společenským i obecně filosofickým problémům. S tímto pohledem souvisí zřejmě i výběr scény prvního uvedení *R. U. R.* na americkém kontinentu, kterou bylo jeviště malého experimentálního divadla zaměřeného na repertoár pro „elitní" publikum. Hra byla přijata nejen s velkým ohlasem u odborné kritiky i široké veřejnosti, ale dokonce se stala natolik populární, že se „zlidovělá" podoba fabule *R. U. R.* stala součástí americké populární kultury.

Význam slova *robot* se vzhledem k původnímu smyslu (v jakém je použil Karel Čapek v *R. U. R.*) v kontextu americké vědecké fantastiky a v obecném kulturním povědomí tedy postupně výrazně posouval nebo – možná lépe řečeno – vulgarizoval: V termínech vědecko fantastické literatury označuje slovo *robot* dnes téměř výlučně umělou bytost elektro-mechanického původu – *převody a kov, elektronika a pozitrony* píše o nich Asimov (Asimov, 1981). Zatímco Čapkovi roboti (ale také Frankensteinovo monstrum ze známé povídky Mary Shelleyové, ke kterému byli roboti v americkém kulturním prostředí přirovnáváni) byli stvořeni z organické látky,[15] což dovolovalo autorovi pracovat s motivy vážícími se k nejasným a o to více zneklidňujícím rozdílům a podobnostem mezi postavami lidí a umělými bytostmi – roboty, androidy, golemy atd.

## 2.6 Robot – simplifikovaný dělník vs. komplikovaný stroj

Mechanistickou podobu získali Čapkovi roboti zřejmě již na premiéře *R. U. R.* na scéně newyorského Garrick Theatre (1922), tehdy však ještě pouze v rovině jejich interpretace (viz dříve citované recenze). Na premiéře *R. U. R.* v pařížské Comédie des Champs Élysées (26. 3. 1924) se však objevuje poprvé robot nikoliv hraný hercem se strnulým výrazem ve tváři a s trhanými pohyby, ale jako kovový-plechový panák, tedy skutečně objekt, věc či stroj. V této souvislosti je zajímavé srovnat ohlas *R. U. R.* v odlišných kulturních kontextech New Yorku a Paříže – zatímco pařížské recenze zdůrazňují revoluční, politické aspekty hry, v kontextu americké kultury se objevují spíše vazby na fantastickou literaturu a na vize budoucnosti lidtsva podmíněné vědecko-technickým vývojem.

Současně však můžeme spekulovat o vyznění dramatiké postavy robota ve scénickém prostoru. Postavu robota chápanou ve smyslu *mechanického* humanoidního stroje lze na základě dobových recenzí přiřadit spíše k americké inscenaci a jejímu kulturnímu kontextu. V souvislosti s naší interpretací postav robotů lze chápat tento interpretační posun od původního záměru autora (nejen) jako důsledek směřování amerických insce-

[14] Scholes a Rabkin ve své knize uvádějí, že Amerika dvacátých a třicátých let *„byla stále izolována od ostatního světa a neznalá politického vývoje za hranicemi"* (Scholes, Rabkin, 1977). *„Evropská science fiction byla psána uznávanými autory, kdežto jejich američtí kolegové se stále ještě pokoušeli zvládnout techniku realismu"* ( Scholes, Rabkin, 1977).

[15] Čapek o původu robotů v deníku starého Rossuma píše: „Příroda našla jen jeden způsob jak organizovat živou hmotu. Je však jiný způsob, jednodušší, tvárnější a rychlejší, na nějž příroda vůbec nenarazila. Tuto druhou cestu, po které se mohl brát vývoj života, jsem dnešního dne objevil." (Čapek, 1992)

nátorů ke schematizaci konfliktu mezi postavami lidí a robotů zdůrazněním „černobílého" rozestavení dvou skupin postav hry. Doslovnost realizované metafory robota jevištně ztvárněného do podoby androidního *stroje*, který spojuje v sobě koncept zmechanizovaného člověka s jeho jevištním obrazem v podobě mechanické figuríny (viz pařížská inscenace), balancuje na hraně parodické a komické zkratky doslovného překladu z módu jazyka do módu jevištní vizualizace, pouhého *vyobrazení* pojmu. Současně však tato postava svým účinkem přesahuje až k působivosti tragikomického symbolu samotné doby „věku stroje", ve které Čapkova hra vznikla a kterou v sobě zrcadlí.[16] *„Nuže, tedy chutě vpřed podle hesla Přelez, podlez, jenom nepřelom!, zrak pevně upřený k ideálnímu cíli, jímž má nám býti Zmechaničtění. V běhu po cestě vzestupné nechť je nám ostruhou rozhodující pobídka onoho manifestu*[17]*, věta vůdčí a vrcholná: Soyon donc des machines!*" napsal Josef Čapek (Čapek J., 1985). V každém případě, právě tento vzhled robota jako jakéhosi mechanického panáka se brzy stal nedílnou součástí populární kultury a jedním z atributů právě se konstituujícího žánru *science fiction*.

S touto populární podobou robotů i námětu hry, která se objevuje nejen v povídkách žánru *science fiction*, ale brzy i na plátnech kin, však Čapek nechtěl být nikdy spojován a bránil se tomu v řadě svých článků. *Dost dlouho mlčel a myslel si své, když se o Robotech ustálila představa, že mají údy z plechu a vnitřnosti z koleček a drátů nebo z čeho; (...) ale když se nedávno dočetl, že v Moskvě natočili velký film, ve kterém rozdupávají svět mechanističtí roboti řízení elektromagnetickými vlnami, dostal chuť protestovat proti tomuto aspoň jménem svých vlastních Robotů*, napsal do *Lidových novin* 9. června 1935, patnáct let po prvním vydání své hry, jako svou reakci na uvedení filmu *Konec senzace*, který se měl původně jmenovat *Rur* (Nikolskij, 1978).

Severní Amerika prožívala v časech uvedení *R. U. R.* období masivního technického rozvoje, vznik pásové velkovýroby, první automobilový boom (všechno to, nač tak citlivě reagoval v některých svých filmech třeba Charlie Chaplin a trochu z jiného pohledu ve svých textech například Stuart Chase, který se ve své – k technice a tehdejšímu rozvoji kapitalismu velice kritické – knize *Člověk a stroj* z roku 1929 věnuje i robotizaci).

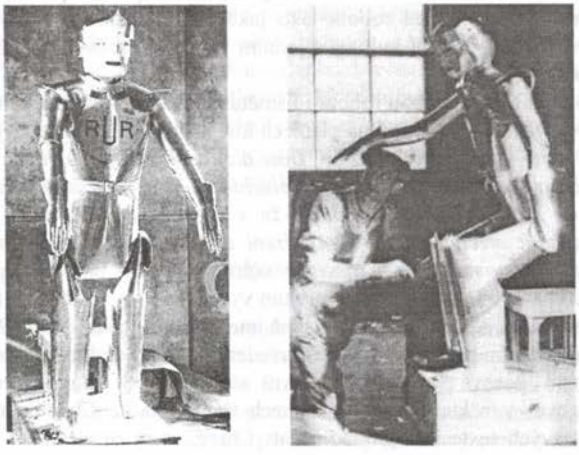
Pro Ameriku nebyl robot *simplifikovaným dělníkem* bez vlastního subjektu, vyráběným v potřebném počtu kopií pro účely tovární produkce, nýbrž to byl *komplikovaný stroj*. Byl to stroj, jaký si tehdy bylo možné představit a alespoň v určité podobě i zkonstruovat. Na začátku minulého století mohl být sestaven jenom z plechu a z koleček s jednoduchými senzory a s prvky rodící se elektro(tech)niky. Mohl to být třeba stroj, jaký viděl Chase: *„... hranatá postava, podobající se vidění kubistovu, zdvíhala ruku, když na ni zahvízdali telefonem. A jak se její paže zdvihla, rozehrálo se mechanické piano, větrák počal vířiti a prací stroj bublati; v místech pak, kde bývá žaludek, se rozsvěcovala a zhasínala elektrická světélka. Byl to sám Mr. Televox, výrobek Westinghousových laboratoří, ztělesňující poslední pokrok v automatizačním procesu. Jím bude nahrazen cvičený dohlížeč na kontrolní přístroje v silárnách...*".[18]

[16] Více k interpretaci robota jako dramatické postavy v kontextu hry *R.U.R.* (Horáková, 2004b).

[17] J. Čapek má na mysli *Manifest futurismu*, který na stránkách deníku *Le Figaro* publikoval Filippo T. Marinetti 20. února 1909.

[18] Zmíněný *Mr. Televox* byl poprvé představen v Pittsburghu v roce 1927.

Čapek odmítal, aby byl považován za autora dobrodružných vědecko fantastických příběhů, ale i za jakéhosi *ludditu* předkládajícího ve své hře varování před hrozbou, kterou představují stroje pro člověka: „*Především se přiznávám, že se cítím poněkud odpovědný za otázku, zda je možno, že stroje konečně nabudou vlády nad člověkem tím, že ho ochudí o schopnost tvořit. Ta otázka mě tíží proto, (...) že jsem v jedné slabé chvíli vynalezl Roboty. Ta okolnost sama způsobila, že si mne někteří lidé představují jako bych s vytesaným perem v ruce hodlal hájit třesoucí se lidstvo proti útočné přesile řvoucích a lidožroutských strojů.*“ Námět *R. U. R.*, převedený do formy otázky by však zněl podle autora jinak: „*Není náš obdiv ke strojům, to jest k mechanické civilizaci, takový, že zatlačuje naši pozornost k skutečným tvořivým schopnostem člověka?*“ (Čapek, 1966).




Obr. 2.2 „Plechový panák“ a scéna z pařížské premiéry *R. U. R.*

Čapkův odpor proti populární podobě jeho díla byl však marný. Po 2. světové válce americký publicista zaznamenává oživený zájmem o jeho tvorbu, Čapek je však v této době již nahlížen skrze proměněnou – mechanistickou – podobu robotů i námětu hry pohlceného populární kulturou. Jeho tvorba se postupně dostává do kontextu paperbackových vydání vědeckofantastických povídek a pozornost budí zejména jeho schopnost předpovídat vynálezy budoucnosti (např. atomovou bombu v románu *Krakatit*, moderní lékařskou etiku a dokonce AIDS v dramatu *Bílá nemoc*, a samozřejmě roboty) jako dříve třeba u Verna označovaného někdy za „básníka hardwaru“ (Scholes, 1977).[19] Robert Wechsler označuje tuto novou vlnu zájmu o Čapka jako „*smrt za živa*“ (Wechsler, 1992).

[19] Jako první vychází v edici science fiction *Krakatit* (pod angl. názvem *An Atomic Phantasy*, 1951), potom jako malé paperbackové vydání *Válka s mloky* (*War with the Newts*, 1955), která byla vydána několikrát znovu (1967, 1975, 1985). *Továrna na absolutno* (*The Absolute at Large*) vyšla v edici The Classics od Science Fiction v roce 1974 (Wechsler, 1992).

Percepci Čapkovy tvorby v USA po druhé světové válce ilustruje také nekrolog publikovaný v časopise *Newsweek* (2. ledna 1939), ve kterém je z Čapkovy tvorby jako jediná zmíněna právě hra *R. U. R.*, o které se zde příznačně říká: *„Ačkoliv věřil, že je to nejméně zajímavé z jeho tvorby, přineslo mu největší slávu a uvedlo slovo 'robot' do anglického jazyka"* (Wechsler, 1992). Z Čapkových robotů jako metafory člověka ve věku stroje se postupně stala metafora stroje s nadlidskými schopnostmi.

Nové chápání robotů se poměrně rychle udomácnilo i v zemi jejich původu. Dokumentuje to třeba televizní seriál, který vznikal začátkem 60. let minulého století v Československé televizi – seriál scenáristy a režiséra Emana Kaněry *Robot Emil*. Velmi pravděpodobně šlo o první dětský televizní seriál vůbec, ve kterém vystupoval robot.

Obr. 2.3 Robot Emil ze seriálu Československé televize ze začátku 60. let 20. století.

## 2.7 Turingovy hypotézy o stroji a o člověku

Kulturnímu vlivu „věku stroje" neodolal člověk, neodolalo mu umění ani obecně vzatá kultura a neodolala mu ani matematika. O mechanizaci základních početních úkonů pokoušeli (a to zcela úspěšně) i vynalézaví matematici a mechanici dřívějších století, např. Blaise Pascal nebo ještě před ním Keplerem z Prahy od roku 1617 úkolovaný Wilhelm Schickart v Tübingenu[20]. O možnostech mechanizace logického uvažování, tedy o možnosti proměnit promýšlení na výpočet snil v Paříži již na rozhraní 13. a 14. století třeba Reymund Lullus, na jehož myšlenky navázal později třeba G. W. Leibniz.[21] Jakmile se objevily roboty, stalo se myšlení jejich atributem[22] a jakmile se objavily počítače, záhy se objevily i spekulace o různých „elektronických mozcích".

[20] První Schickartův návrh mechanického počítacího stroje je popsán v jeho dopisu Johannesu Keplerovi z 20. září 1623.

[21] Trochu víc informací o zmíněných historických faktech obsahuje například publikace (Kelemen, 1989).

[22] V tom vidíme určité pokračování mnohem starší golemovské tradice, podrobněji o ní pojednává (Kelemen, 2001).

Asi nejzřetelnější teoretický posun k cílu pochopit povahu počítání a výpočtů však začal někdy na jaře roku 1935. V té době začal student King's College Univerzity v Cambridge Alan M. Turing navštěvovat přednášky M. H. A. Newmana o základech matematiky. Cyklus přednášek končil tehdy novými a nadmíru provokativními výsledky Kurta Gödela o *logické neúplnosti*, tedy – velmi zhruba řečeno – o nemožnosti rozhodnout o platnosti nebo neplatnosti některých matematických tvrzení pomocí formální logiky. Newman v závěru svých přednášek načrtnul problém: Může existovat nějaký *mechanický* proces, který by umožnil získávat odpovědi alespoň na některé problémy formulované v jazyce matematiky?[23]

Jedna z variant Newmanovy otázky může být formulována třeba ve spojitosti s výpočtem funkcí: Máme-li definovanou funkci tradičně, tedy jako podmnožinu kartezián-ského součinu jejího definičního oboru a oboru jejích hodnot, existuje *mechanický proces*, který by k daným hodnotám proměnných vypočítal jim ve smyslu dané funkce příslušné funkční hodnoty? Když v létě 1935 přemýšlel Turing o této otázce, jako úplně první intuitivní model *mechanismu* pro vykonávání výpočetních procesů mu posloužil mechanický psací stroj (Hodges, 1985).

Výsledkem Turingova přemýšlení byl článek (Turing, 1936) obsahující matematickou definici zařízení, kolem jehož výzkumu se pak točila celá teoretická informatika 20. století. Krátká citace z něho bude snad k užitku: *Můžeme srovnat člověka vyčíslujícího* (v originálu *computing*) *nějaké reálné číslo se strojem, který zvládá pouze konečný počet podmínek* $q_1, q_2, \ldots, q_R$, *které budou nazývány „m-konfiguracemi". Stroj je vybaven „páskou" (analogickou papírové) která jím prochází a která je rozdělena na části (nazývané „čtverečky"), v každé z nich může být zapsán „symbol". V každém okamžiku je „ve stroji" jediný čtvereček, řekněme r-tý, ve kterém je zapsán symbol* $S(r)$. *Tento čtvereček můžeme nazvat „snímaným čtverečkem". Symbol v snímaném čtverečku můžeme nazvat „snímaným symbolem". Snímaný symbol je jediný, na který je stroj – řekněme to tak – „přímo soustředěn". Změnou své m-konfigurace si však může stroj efektivně zapamatovat symboly, které dříve „viděl" (snímal). Možné chování stroje v určitém momentu je determinováno m-konfigurací* $q_n$ *a snímaným symbolem* $S(r)$. *Dvojici* $q_n$, $S(r)$ *budeme nazývat „konfigurací". Konfigurace tedy určuje možné chování stroje. V některých konfiguracích, ve kterých je snímaný čtvereček prázdný (které neobsahují symbol), napíše stroj do snímaného čtverečku nějaký symbol. V jiných konfiguracích odstraní snímaný symbol. Stroj může taky změnit právě snímaný čtvereček, avšak pouze přesunem o jedno místo vpravo nebo vlevo. U všech těchto operací může dojít i ke změně m-konfigurace.*[24]

Krok za krokem, využívajíc svou metaforu, se pak autor dostává k definici zařízení, které nazývá *a-stroj* (původně *a-machine* jako zkratka pro *automatic machine*), které pak Alonzo Church ve své recenzi jeho článku pro *Journal of Symbolic Logic* nazval *Turingovým strojem*, a ke klasifikaci funkcí na rekurzivně vypočitatelné, částečně vypočitatelné a nevypočitatelné. Je třeba přiznat, že Turing navrhl i jiné podoby abstraktních počítačů – tzv. *c-machine* (*c* bylo zkratkou pro *choice*) umožňující využívat při výpočtech interakce s prostředím (k tomuto nápadu se ještě v jednom z následujících oddílů vrátíme) a *u-machine* (*u* jako zkratku pro *univerzal*) inspirovaný strukturou

[23] Více informací možno najít ve skvělém Turingově životopisu (Hodges, 1985).
[24] Viz (Turing, 1936).

nervové soustavy. Nepodařilo se mu však formalizoval tyto zařízení tak, jako koncept *a-stroje* (Wegner, Goldin, 2003).

Emil Post nabídl časopisu *Journal of Symbolic Logic* k publikaci svůj článek s názvem *Finite Combinatorial Processe – Formulation 1*. Redakce ho obdržela 7. září 1936, tedy přibližně v čase, kdy dělal Turing pro *Proceedings of London Mathematical Society* korektury svého článku *On Computable Numbers...* Post si představoval výpočetní proces takto: „*Náš prostor sestává z oboustranně nekonečné posloupnosti míst nebo škatulek. Z hlediska uspořádání se podobá řadě celých čísel ...–3, –2, –1, 0, 1, 2, 3, ... Řešitel problému, nebo dělník, se může v tomto prostoru symbolů pohybovat a pracovat tak, že se v každém momentu nachází a pracuje v jedné ze škatulek. Bez ohledu na přítomnost dělníka může být každá ze škatulek v jednom ze dvou stavů – může být prázdná (neoznačená) nebo označená nějakým znakem, třeba vertikální čárou... Předpokládejme, že dělník může vykonávat následující primitivní operace: (1) označit škatulku, (2) setřít ze škatulky označení, (3) přemístit se do škatulky vpravo, (4) přemístit se do škatulky vlevo a (5) zjistit, zda je nebo není škatulka, ve které se právě nachází, označená...*“ Pak samozřejmě následuje návrh, jak by měl v obecném tvaru pro dělníka vypadat program jeho práce pro konkrétní funkci, kterou vypočítává. Podrobnosti jsou například v práci (Uspenskij, 1979).

V obou uvedených případech návrhů abstraktních zařízení definujících s matematickou přesností pro matematiku tak významný (a tak neuvěřitelně dlouho opomíjený) pojem, jakým je *algoritmus*, vycházejí navrhovatelé z intuitivní inspirace v monotónní, přesné a „mechanické“ (tedy intelektuálně nikterak nenáročné), od okolních vnějších rozptylujících nebo inspirujících vlivů dokonale izolované, činnosti člověka – počtáře a dělníka. Počtář ani dělník se nemýlí a nic, co by ho vedlo třeba jen k droboučké změně předepsaného postupu konání, ho nenapadne. Počtář toho v Turingově stroji moc nenapočítá (on jenom posouvá pásku, zapisuje, propisuje a gumuje) a Postův dělník jenom čile přeskakuje ze škatulky do škatulky... Intelektuálně však taky nemusí dělat nic! Že by skutečně bylo takovéto prosté počínání ekvivalentní i tomu nejsložitějšímu počítání? *Churchova a Turingova hypotéza* tvrdí, že tomu tak je[25]. Vykonání abstraktních výpočtů – jsou-li teoreticky vůbec vykonatelné – je prostinká činnost. Není k ní třeba skoro nic: píle, pečlivost, čas, tužka a dostatečně dlouhý pás papíru...

Stačí takovéto počítání k tomu, aby se stroj mohl stát inteligentním? Turing – zejména po intelektuální konfrontaci se skutečnými počítači – uvažoval o platnosti své (a Churchovy) hypotézy a dospěl k jiné hypotéze, kterou v určité podobě vyjádřil pomocí toho, co v (Turing, 1950) nazval *imitační hrou* a co je dnes známější jako *Turingův test*: Je inteligence projevem výpočtů soustavy funkcí vypočitatelných na Turingově (nebo na libovolném s ním ekvivalentním) abstraktním stroji? Je tedy inteligence v zásadě matematicky definovatelnou a turingovsky vypočitatelnou funkcí? Je inteligence skutečně proces, do kterého *na začátku* vstoupí data (coby hodnoty proměnných) a *na konci* je produkován výstupní údaj (jako hodnota vypočítaná danou funkcí – přesněji Turingovým strojem, který jí odpovídá – z příslušných vstupů)?

[25] Nebo alespoň donedávna bylo. Přehled některých zpochybňujících úvah (jejich počet v posledním desetiletí dost narůstá a některé i my připomeneme v další části této kapitoly) je možné nalézt například v (Eberbach, Wegner, 2003).

Dlouho byl vývoj *umělé inteligence* veden přesvědčením, že tomu tak je! Byl vypracován použitelný arzenál prostředků symbolové *reprezentace* lidských *poznatků*. Byly navrženy a naprogramovány soustavy důvtipných procedur – tzv. *inferenčních strojů* – umožňující počítačům na základě reprezentovaných poznatků řešit problémy, jejichž řešení tradičně vyžadovalo lidskou znalost. Udělalo se, co bylo třeba, i ve prospěch počítačového *zpracování neurčitosti*, kterou jsou lidské poznatky nevyhnutně poznamenány. *Znalostní systémy* tvoří dnes organickou součást mnoha v praxi používaných softwarových produktů pro mnoho aplikačních oblastí.

V současnosti se o platnosti této *druhé Turingově hypotézy* – kterou pro umělou inteligenci jednoznačně zformulovali další její zakladatelé Allen Newell a Herbert A. Simon v podobě tzv. *hypotézy o systému symbolů*; viz např. (Newell, Simon, 1976), (Kelemen a kol., 1992) – dost intenzivně pochybuje. Podívejme se na práci *Turingova stroje*. Na papíře je program, co a jak dělat s páskou, na které je kódována vstupní informace.

Jaký však bude výsledek tohoto počínání, když někdo „zvenčí" (*z prostředí*) popisuje, ve kterém Turingův stroj pracuje, a jehož vliv je v definici Turingova stroje docela ignorován? Přitom právě v umělé inteligenci – v souvislosti s konstrukcí pokročilých softwarových nebo hardwarových *autonomních agentů*, tzv. *softbotů* a *robotů* – se s tímto problémem často a neignorovatelně potýkáme! Co mohou vypočítat počítače, vložíme-li je do dynamického prostředí, které bude ovlivňovat jejich činnost nejenom (dobře definovanými) vstupy na začátku, ale i (méně dobře definovanými) vlivy po dobu „běhu" výpočtu? Jaká bude síla Turingova stroje, bude-li v průběhu výpočtu podléhat vlivům z prostředí, které budou generovány nějakým neznámým Turingovým strojem, nebo několika takovýmito stroji? Jinými slovy: Jaká je síla interakce a emergence? Co se stane, bude-li výpočetní zařízení vystavěno účinkům prostředí, jehož chování není turingovsky vypočitatelné (existují přece i takové funkce!)? Je lidská inteligence funkcí „vypočítající" (nebo pro případ počítačů vypočítající) hodnoty z rigidní sady vstupních dat, nebo je charakteristikou kvality vztahu mezi agentem (člověkem, zvířetem, počítačem, robotem...) a prostředím, které jeho chování nepřetržitě ovlivňuje a kterému se musí v zájmu zachování svého fungování v tomto prostředí přizpůsobovat, které musí měnit, na které musí reagovat vlastními vnitřními změnami...?

Uvažujíc o vlastních vnitřních změnách systémů ponořených do jejich proměnlivých prostředí se dostáváme k druhému okruhu pochybností. Ten vzniká v souvislosti s umělou evolucí. Na začátku zapíšeme pro Turingův stroj na jeho pásku vstupní informaci a on nám vypočte nějaký výstup. My však s výstupem nebudeme spokojeni (prostředí „zatlačí" na Turingův stroj a bude ho nutit k „mutacím" – k drobným úpravám programu právě zapsaného na jeho pásce). Na základě téhož vstupu nám pak může vypočítat v čase $t + n$ cosi jiného než vypočetl v čase $t$ a takto to může probíhat dlouho, protože nalezení globálně optimálního programu může trvat velmi (až nekonečně!) dlouho (dokonce bez ohledu na to, je-li prostor vstupních a výstupních dat konečný nebo nekonečný). Tento celý evoluční proces teď zapíšeme jako program a zapíšeme na pásku jiného Turingova stroje. Může se to podařit? Je-li odpověď kladná, tak co za funkci bude počítat tento druhý Turingův stroj?[26]

[26] Podrobněji jsou tyto rozpaky (spolu s několika dalšími) zpracovány přehledně např. v (Eberbach, Wegner, 2003).

## 2.8 Od Alquista k von Neumannovi

Ve třetím dějství hry *R. U. R.* naléhá na Alquista, posledního z lidí žijících na ostrově s továrnou R. U. R., pět zástupců robotů. Neumějí se množit a tak žádají od něho Rosumův předpis na výrobu robotů: „*Pane, stroje nemohou pracovat. Nemůžeme rozmnožovat Roboty...*" Robot Damon si představuje: „*Budeme rodit strojem. Postavíme tisíc parních matek. Vychrlíme z nich řeku života. Samý život! Samé Roboty! Samé Roboty!*" Alquist však pochyboval o robotech jako jiné formě života: „*Roboti nejsou život. Roboti jsou stroje.*" Kdo se jako první pokusil dát strojům možnost množit se, byl John von Neumann. Pod tímto jménem je znám v informatice jako tvůrce tzv. *von Neumannovy architektury* počítačů a autor prvního návrhu automatu s črtami (umělého) života. Fyzikům se však jeho jméno zřejmě spojí s matematizací kvantové fyziky, matematikům a ekonomům s teorií her, kterou inicioval ve spolupráci s Oscarem Morgensternem.

John von Neumann se narodil v Budapešti 28. prosince 1903 v zámožné bankéřské rodině jako Margittai Neumann János Lajos. S počítači se seznámil po své emigraci před rasovým pronásledováním a válkou a následném trvalém usazení v USA v průběhu 2. světové války jako spolupracovník na projektu Manhattan. V rámci tohoto setkání dal počítačům novou architekturu. Byl však i geniálním matematikem a tak se zamýšlel i nad jejich matematickým opisem a tedy samozřejmě i nad tím, jak pracuje Turingův stroj a jak by měl pracovat, aby se mohl množit: *Turingovy automaty jsou ryze počítače*, píše v (Neumann, 1951). „*Produktem jejich práce je papírová páska, na které jsou napsány symboly 0 a 1. K takové konstrukci, o které mluvím, je třeba automatu, jehož produktem práce bude jiný automat. Nevidím nijakou principiální těžkost, která by nám bránila pojednat o této obecnější myšlence a odvodit s Turingovým výsledkem ekvivalentní výsledek.*"

Konstrukci reprodukujícího se automatu si von Neumann (Neumann, 1951) představoval v několika krocích, které popisují postupně konstrukci a kombinaci různých automatů, ze kterých sestává výsledný reprodukující se automat. Tyto kroky teď načrtneme:

Automat $A$ má vlastnost, že když mu poskytneme vhodný opis libovolného jiného automatu, $A$ tento automat zkonstruuje. Opis nebude v našem případě napsán na pásce papíru, jak to předpokládá Turing... Snadno však můžeme popsat kombinaci konstrukčních prvků tak, aby měl vlastnosti opisu na papírové pásce. Takovýto opis budeme nazývat instrukcí a značit písmenem $I$.

Automat $B$ dokáže vytvořit kopii implementované instrukce $I$.

Zkombinujme navzájem automaty $A$ a $B$ a spojme je s řídicím mechanismem $C$. Automat $C$ nejprve nechá automatu $A$ zkonstruovat automat, který opisuje implementovaná instrukce $I$. Pak automat $C$ nechá automatem $B$ zhotovit kopii této instrukce $I$ a implementuje ji do toho automatu, který právě zkonstruoval automat $A$. Nakonec $C$ oddělí tuto konstrukci od systému $A + B + C$ a vypustí ho jako nový nezávislý automat.

Označme $A + B + C$ jako $D$.

Aby systém $D = A + B + C$ mohl fungovat, musíme ho vybavit opsanou instrukcí. Jak jsme to již naznačili, tuto instrukci je nutné implementovat do $A$. Vytvořme tedy instrukci $I_D$ opisující automat $D$ a implementujme instrukci $I_D$ uvnitř automatu $D$ do automatu $A$. Automat, který takto obdržíme, označme $E$.

„*Je zřejmé*, píše von Neumann, *že E bude reprodukujícím se automatem. Rozhodující krok se uskutečňuje v E tehdy, když konstruujeme instrukce $I_D$ opisující automat D a implementujeme ho do D. Když je třeba konstruovat (kopírovat) $I_D$, automat D již existuje a konstrukce $I_D$ ho nijak nemodifikuje. Tato instrukce se jednoduše přidá, abychom vytvořili E. Existuje tedy jedna časová i logická posloupnost, podle které je třeba vytvářet D a $I_D$. Celý tento proces je logicky korektní a správný.*"

Pak von Neumann ještě upozorňuje na zřejmou analogii mezi rolí genu v biologických systémech a instrukcí $I_D$ v případě reprodukujících se automatů a na to, že kopírovací mechanismus obsažený v *B* hraje roli duplikace genetického materiálu. Připomíná možnost drobných změn automatu *B* i instrukce $I_D$ a přirovnává tyto možnosti k možnosti mutace genetického materiálu u živých organismů. Takže reprodukující se automaty s vlastním genetickým kódem, s evolucí... Jako živé!

Tématem reprodukujících se automatů se von Neumann přestal zabývat poté, co byl po skončení 2. světové války jmenován Kongresem USA za člena Americké agentury pro atomovou energii. Když se posléze zhoršil i jeho zdravotní stav, dal svolení Johnovi Kemenyovi – pozdějšímu autorovi programovacího jazyka BASIC – aby o reprodukujících se celulárních automatech napsal článek on[27]. Komplikovaný von Neumannův návrh do podrobností sepsal a zjednodušil E. F. Codd například tím, že von Neumannem předpokládaný počet interních stavů jednotlivých automatů snížil z 29 na 8 při zachování turingovské výpočetní univerzálnosti modelu. Tento model byl systematicky studován a posléze představen v (Codd, 1968). Další zjednodušení modelu navrhl – inspirován Coddovým návrhem – Christopher G. Langton. Jeho úmyslem nebylo zachovat výpočetní univerzálnost modelu, nýbrž najít nejjednodušší implementovatelnou podobu reprodukujících se celulárních automatů – proměnit možnost, o které psali například von Neumann, Kemeny a Codd, na skutečný (i když „jenom" virtuální) reprodukující se (v paměti počítače) systém umělých buněk. To se mu povedlo.[28]

Jinou motivaci pro informatický – a tedy ve své podstatě „strojový" – pohled na některé charakteristické projevy živých systémů měl Aristid Lindenmayer[29], další významná osobnost, která posunula propojení mezi živými a neživými systémy. Citujeme některé úryvky z rozhovoru s ním, publikovaného v (Kelemenová, Kelemen, 1986):

„*Na přednáškách z biologie a chemie*[30] *jsem narazil na problém, jakým způsobem popsat struktury, které jsou v prostoru i čase částečně diskrétní a částečně spojité, a které pozorujeme například u koloidních suspenzí a gelů. Žádalo se mít formulace v abstraktním jazyce odlišném od spojité matematiky a já jsem se domníval, že je šlo vytvořit v jazyce matematické logiky. J. H. Woodger ... asi od roku 1930 ... zastával názor, že k vyjádření řady biologických fenomenů se relačně-teoretické pojmy hodí lépe než analytické...*

[27] Kemenyův článek vyšel v dubnovém čísle *Scientific American* v roce 1955 (s. 58–68).

[28] Pravděpodobně nejkompletnější opis Langtovova reprodukujícího se automatu je v (Langton, 1984).

[29] Narodil se s vyrůstal také v Budapešti a za zmínku snad stojí, že maturitu získal na Evanjelickém gymnáziu na Alejové ulici (Fasor utca) jako von Neumann. Kromě nich tam maturovali například E. Teller, P. Erdős a G. Pólya.

[30] Lindenmayer navštěvoval zmíněné přednášky na budapešťské univerzitě (tehdy Univerzita Pétera Pázmánye, dnes Univerzita Loránda Eotvöse). Konaly se ve stejné budově na Muzejní okružní, ve které poslouchal jako student téže univerzity přednášky von Neumann.

*Po takovýchto zkušenostech nadešla chvíle mého setkání s teorií automatů. Stalo se tak v létech 1962–63, kdy jsem pracoval v biomatematické skupině univerzity státu Severní Karolina a kdy jsem měl možnost často se setkávat s embryologem J. R. Greggem z Dukovy univerzity. Předtím i on spolupracoval s Woodgerem a napsal článek o množinově-teoretických základech taxonomie a embryologie ...*

*Vznik L-systémů se váže k jednomu problému z Ashbyho knihy*[31]*. V kapitole o konečných automatech se ptá: Jaké chování možno očekávat od dlouhého řetězce obousměrně spojených konečných automatů v případě, dostávají-li vstupy od sousedících automatů? U Kleeneho*[32] *najdeme odpověď pro případ konečněrozměrné sítě konečných automatů. ... Mne však znepokojovala otázka chování rozrůstajícího se pole konečných automatů. Takovéto pole může totiž sloužit jako reálná simulace růstu buněčných vláken, s jakými se setkáváme u řas, mechů nebo hub. Rozuzlení tohoto problému se však ukázalo náročnějším, než jsem předpokládal. To však nic nemění na faktu, že v roce 1968 došlo k definici L-systémů jako interagujících rozrůstajících se lineárních polí konečných automatů*" (Lindenmayer, 1968).

Základní biologickou inspirací pro výzkum *Lindenmayerových* (nebo *L-*) *systémů* byla otázka, jaká míra interakce buněk, majících stejný genetický kód, je nevyhnutná k tomu, aby docházelo k morfogenezi – aby třeba rostliny nabývaly své obvyklé tvary, aby se na schránkách měkkýšů objevovaly charakteristické ornamenty apod. Šlo o to, objasnit globální vývoj (jednoduchých, vláknitých) mnohobuněčných organismů na základě studia lokálních transakcí mezi buňkami. Tyto transakce byly formalizovány do tvaru transformačních pravidel uplatňovaných striktně paralelně na celou populaci buněk tvořících studovaný organismus. Druhým mlčky přijímaným předpokladem bylo, že lokální pravidla jsou stejná pro každou buňku organismu[33].

Tyto počáteční snahy o pochopení fenoménu života našly pokračování v mnoha směrech. O některých podstatných informují například (Csontó, Palko, 2002) nebo – v popularizační podobě – (Levy, 1992). V průběhu 80. let se vynořily i jiné přístupy, blízké problematice *umělého života* (a umělé inteligence), které však nebyly vedeny bezprostředně z pozic výpočetní nebo informatické paradigmy. Skvělým příkladem takovýchto prístupů je (Kováč, 1986).[34]

Podíváme-li se tedy na chování živých bytostí a neživých předmětů, neunikne nám, že zatímco živé bytosti se chovají tak, aby svými zásahy do prostředí udržovaly a prodlužovaly určitý druh rovnováhy mezi sebou a prostředím[35], neživé předměty takovouto tendenci neprojevují. Pokročilá lidská civilizace je charakteristická mimo jiné (a možná především) tím, že (1) sní o možnosti strukturovat neživou hmotu postupně tak, aby se

---

[31] Ashby, W. R.: *An Introduction to Cybernetics*. Chapmann and Hall, London, 1961.

[32] Kleene, S. C.: Representation of events in nerve nets and finite automata. In: *Automata Studies* (C. E. Shannon, J. McCarthy, eds). Princeton University Press, Princeton, N. J., 1956, s. 1–41.

[33] Matematickou teorii L-systémů představuje nejkompletněji (Rozenberg, Salomaa, 1980).

[34] Ladislav Kováč se v citované práci zabývá i problematikou, která se ve stejné době vynořila i v zcela jiných souvislostech – v oblasti pokročilé robotiky – otázkou vzniku a povahy nejnižší úrovně kognice. K problematice tzv. nové umělé inteligence iniciované v robotice Rodneyem A. Brooksem; k tomu podrobně viz (Brooks, 1999), informativně např. (Kelemen, 1994).

[35] Dělají-li to tak, že k udržování rovnováhy s prostředím využívají i určitou „vnitřní" reprezentaci svého prostředí a také reprezentaci svých vlastních možností ovlivňovat svá vnější prostředí, pak bychom jim bez rozpaků měli přiznat i určitou (jejich dovednostem přiměřenou) míru inteligence ve smyslu, v jakém chápe inteligenci například Jean Piaget (Piaget, 1970).

svým chováním přibližovala živým organismům a kulturně reflektuje možné důsledky úspěchu této snahy a (2) vynakládá technické a vědecké úsilí pochopit podstatu takovéto proměny a snaží se proměnit svůj sen ve skutečnost.

Živé bytosti jsou samozřejmě z fyzikálního pohledu stroje – znehodnocují ušlechtilejší formy energie v teplo, mají své vnitřní stavy a ty cyklicky střídají atd. Dělají však něco navíc. Co přesně, to zůstává tajemstvím – život je skutečně asi zázrak. Nemůžeme si však nevšimnout skutečnosti, že se postupně daří uměle konstruovat taková zařízení, která svým chováním překračují tradiční chápání stroje.[36] Nazvat je živými by nebylo korektní, protože, jak jsem viděli, nevíme přesně, co bychom o nich vlastně tvrdili. Na druhé straně nechat bez povšimnutí velice zajímavé vlastnosti jejich chování a ztotožňovat třeba parní stroj s autonomními roboty, které vysíláme na Mars a ony tam autonomně „přežívají", vyhýbají se v rámci svých možností všem různým nástrahám terénu (o kterém jejich konstruktéři mají pouze neúplnou a zprostředkovanou představu) a plní své mise, neodráží zhola nic z odlišnosti těchto zařízení. Tyto robůtky (a mnoho jiných, jim více nebo méně podobných zařízení) nejsou živé, jistě jsou to stroje. Nebylo by však intelektuálně produktivní vymezit pro ně zvláštní kategorii ne sice tak tajemnou, jakou je kategorie živých bytostí, ale zato takovou, která by je vyčlenila (tak jako vyčleňuje tajemství života živé bytosti) z obrovské masy systémů, které vyhovují naší definici stroje? Rozvoj umělé inteligence[37] k takovéto kategorii strojů přivedl – jsou to *agenti*. Tato nová kategorie by nás mohla v našich budoucích úvahách uchránit od uvažování podle tradičního a mnohdy zavazejícího dualistického schématu živé–neživé. Agent je stroj ve smyslu, v jakém je strojem i každá živá bytost. A agent není živá bytost, protože nejsme s to dostatečně „technizovat" pojem život. Agent je stroj s některými atributy živosti, je „technizovatelnou" výsečí z nedozírně tajemného pojmu *živé*. Je tím zkoumatelným a konstruovatelným mezi strojem a živou bytostí.

## 2.9 Kyborg

Naznačili jsme, jak mohla v evropském kulturním a společenském kontextu, zejména v situaci, jaká vládla na začátku 20. století, vzniknout – díky dílu bratří Čapků – představa robota. Připomněli jsme, jakou podobu tato představa měla a jak se její původní podoba postupně měnila v závislosti na proměnách kulturních kontextů, ve kterých se ocitala. Naznačili jsem také, jak se postupně vynořila potřeba vymezit pro novou podobu konstruktů, ve které se postupně díky aktivitám v oblastech, jakými jsou umělá inteligence, kognitivní věda a oblast umělého života, objevil nový koncept – koncept agenta. Jak se tyto proměny odrážejí v chápání robota a člověka? Obecná kultura vždy citlivě reagovala na vynořující se význam a možné proměny pojmu stroj a pokoušela se předvídat možné dopady. Jsme již dnes svědky podobné reakce na vynořující se pojem agenta ve významu, který jsme výše načrtli?

[36] Jsou to pokročilé roboty, při konstrukci kterých se široce uplatňují poznatky získané v oblasti umělé inteligence; více o tom viz např. (Murphy, 2000).

[37] K podobnému přesvědčení se dospělo i v teoretické biologii, jak je to zřejmé např. z (Kauffman, 2004) a jak jsme se o tom zmínili i v (Markoš, Kelemen, 2004).

Felix Guattari se k transformacím pojmu stroj vyjádřil takto: „*Ano, dali jsem stroji to největší rozšíření... Takže někdy mluvíme o technických strojích, v běžném smyslu slova, někdy o sociálních strojích, někdy o strojích touhy (desiring machines). Protože pro nás stroj v žádném případě není v rozporu s člověkem nebo přírodou... Na druhé straně, stroj není v žádném případě redukovaný na mechaniku. Mechanika odkazuje k protokolu nějakých technických strojů; nebo jinak, ke konkrétní organizaci organismu. Ale strojovost (machinism) je něco zcela jiného: označuje každý systém, který ... překračuje mechaniku technologií stejně jako organizaci organismu.*" (Deleuze, Guattari, 1987).

Doposud jsme se zabývali spíše tou pomyslnou oblastí, které dominují objekty ryze „mechanické" a ryze „organické". Co se však ukrývá v „šeru", v prostoru vymezeném Guattariho pojmem *strojovost* (*machinismus*), ze kterého nám Karel Čapek přinesl své roboty? Ukrývají se tam například další, Čapkovými roboty inspirované kreatury, *kyborgové* (nabo snad *kyborgy*?), které známe nejdůvěrněji z různých vědecko-fantastických filmů typu *Terminátor* a snad i z různých počítačových her... Jeden z techno-thrillerů tohoto typu připomeneme, neboť v něm shledáváme i atributy, které se ve standardních dílech zmíněného zaměření obvykle nevyskytují. Jde o novelu napsanou známým autorem textů žánru sci-fi Harry Harrisonem spolu s jedním z průkopníků umělé intelligence, Marvinem Minskym, *The Turing Option* (1992).

*The Turing Option* má dva rozměry, které jsou v našem kontextu zajímavé: Na pozadí poměrně nenáročné detektivní zápletky autoři srozumitelně vysvětlují Minskyho tzv. societní teorii mysli (Minsky, 1986), která v oblasti umělé inteligence a kognitivní vědy ještě zdaleka nevyčerpala svůj myšlenkový potenciál a zůstává provokativní ideou čekající možná na nějaký podstatnější zvrat v oblasti názorů na počítání a nebo také na podstatnější posun v technologii počítání[38]. Nápad Harrisona a Minskyho skrze vědecko-fantastickou detektivku popularizovat nejnovější otázky a problémy exaktní vědy navazuje na tradici renesančních myslitelů. V souvislosti s Aristotelem předestřenými předpoklady zrušení otroctví jsme již mluvili o utopii. Také záměrem anglického politika, katolického mučedníka a zakladatele utopického socialismu Thomase Mora (1478–1535) bylo – svou *Zlatou knihou, stejně tak užitečnou, jako i zábavnou, o nejlepším uspořádání státu a o novém ostrově Utopie* vydanou v roce 1516 – šířit své vize o uspořádání ideální společnosti. To pouze v dnešní době úzké specializace se pokus o propojení vědeckých poznatků a hypotéz s literárním, fabulačním talentem, propojení technického a humanitního vědění ve vědomí kultivovaného člověka, jeví jako nenávratně ztracený fenomén.

Harrisonovu a Minskyho novelu však můžeme číst i jako jiné poselství. Ukazuje nám, jak mohou být (jednou v budoucnu) použity znalosti, ke kterým nás přivádějí dnes spíše za dehumanizační považované výzkumy v oblasti umělé inteligence a pokročilé robotiky, k poznatkům využitelným i pro bezprostřední pomoc lidským bytostem. Na základě takovýchto poznatků a postupů se čtenář seznámí nejen s konstrukcí inteligentního robota, nýbrž i s využitím principů pro rekonstrukci poraněného mozku a nalezení ztracené lidské mysli. Tento fabulační prvek jakoby záměrně zvýrazňoval a dělal zřetelnějším skrytý „filosofičtější" rozměr novely, o kterém jsme se zmínili v předchozím textu.

[38] K tomuto tématu podrobněji viz například (Singh, 2003).

Opět, v podobě ovlivněné desetiletými zkušenosti se stále dokonalejšími stroji, se objevuje i pocit podobný tomu, který na začátku století zaregistrovali futuristé. V prvním *Manifestu futurismu* hlásal Filippo Marinetti potřebu *odstranění statického, nezajímavého umění minulosti* a naléhavost zdůraznění nových technologií (uvádí konkrétně auta, letadla, lokomotivy). Složitost a síla těchto strojů je podle jeho mínění to, co by mělo diktovat estetiku a inspirovat uměleckou tvorbu budoucnosti. „*Teplo kusu železa nebo dřeva je od nynějška rozvášňují více než ženské slzy a úsměvy! Cítíme mechanicky a cítíme se být uděláni z ocele!*", opakuje, parafrázuje a ironizuje výkřiky futuristů Josef Čapek v *Homo Artefactus*.

Jaká byla situace na konci století (a jaká je dnes)? „*Nemá již dále smysl vidět tělo jako sídlo naší psychické a společenské existence. Je spíše strukturou, kterou je třeba monitorovat a modifikovat – nechápejme tělo jako subjekt, nýbrž jako objekt, a nechápejme ho jako objekt, po kterém toužíme, nýbrž jako objekt, který vytváříme*", čteme na stránkách www.stelarc.va.com.au světově proslulého australského performera vystupujícího pod jménem Stelarc. Na rozdíl od umění futuristů, kteří byli fascinováni mechanickým a byli připraveni hledat k němu cestu z domnělé beznaděje lidsky organického, umění dneška jako kdyby hledalo pro vztah organického a mechanického (rozuměj strojového zejména ve smyslu, jak jej formulovali Deleuze a Guattari, viz citát v předchozím textu) místo kdesi uprostřed metaforického šera. „*V konfrontaci s obrazem své sešlosti je tělo traumatizováno a snaží se odštěpit od říše subjektivity zvažujíce nevyhnutelnost svého přehodnocení a možnost nového navržení své vlastní struktury*", hlásá australský multimediální performer Stelarc. A souhlasil by s ním zřejmě i vědec: „*Možná nadešel čas zvažovat přetvoření samých sebe do forem, které poskytují větší možnost individuálního růstu. Mohli bychom uvažovat o možnosti převést se do podoby strojů*", čteme v (Minsky, 1996) a zcela konkrétní pokusy takovéhoto zvláštního sebezdokonalování skrze propojení se se strojem nám čas od času nabízí specialista v oboru kybernetiky a robotiky, Kevin Warwick z univerzity v britském Readingu[39].

Co na to filosofie? Asi musíme přijmout tezi, že člověk není tím, oč evoluci jde. Není cílem vývoje jsoucna, spíše je článkem v jeho vývoji. Egon Bondy se například domnívá, že v tomto vývoji sehraje člověk roli určitého „překročitele biotu". „*Tuto roli splňuje v budoucnosti (ne příliš daleké!) patrně tím*" – psal Bondy začátkem 70. let minulého století – „*že vytváří tzv. (z našeho hlediska) artificiální formu existence...To je nepochybně v silách člověka a jde jen o to, jak brzy si člověk uvědomí, že tu nejde o výrobu 'jeho služebníka', ale o jeho vlastní sebepřekročení jako biologicky vázané bytosti, která se v rámci biologična nemůže už kvalitativně výše rozvinout, tj. nemůže všechno to, na co intencionálně poukazuje svou lidskou existencí*" (Bondy, 1993). Touto tendencí transgrese člověka a pokusem o formulování etických, kognitivních a estetických dimenzí přizpůsobených novým způsobům bytí „nového člověka" se dnes zabývají umělci, filosofové i vědci. V jejich názorech se znovu objevují dva odlišné přístupy, v něčem připomínající odlišnost v pochopení Čapkových robotů zrcadlící se třeba

[39] O jeho názorech na robotiku možno číst česky v (Warwick, 1999).

v odlišnosti kostýmů robotů z pražské a newyorské premiéry *R. U. R.* – přístup charakterizovatelný jako „posthumanistický", zřejmý například v (Petrů, 2005), (Hayles, 1999) a u dalších, a přístup „nehumanistický"[40], s jakým se setkáváme například v (Moravec, 1990) nebo (Minsky, 1996). Dějiny stroje 20. století končí takovýmito knihami – prvními stránkami v hrubém dějepisu o člověku 21. století.

## 2.10 Kouzlo interakce

Robot – to není v naší kultuře pouze metaforická pomůcka pro literární, divadelní nebo kinematografickou tematizaci určitých společenských jevů, problémů, názorů a estetických koncepcí. Rovněž se nestal pouhým neúnavným rychlým a pozorným strojem zastupujícím dokonalého sluhu, pouhým vykonavatelem programů, které do něho vkládáme. Nezůstal ani pasivním předmětem výzkumů rozmanitých disciplín a prostředkem konkrétních aplikací výsledků těchto disciplín. Robot, ve smyslu inteligentního agenta, se stal aktivní součástí našeho světa. Není pochyb o tom, že se stroji nám je a ještě velice dlouho bude souzeno žít a tudíž s nimi kooperovat a komunikovat. Robot a počítač se staly *médiem*. Jejich všudypřítomnost pak nemohla nechat bez odezvy uměleckou citlivost. Tak vstoupily do oblasti umění, zejména do oblasti *umění nových, digitálních* nebo *interaktivních médií*.

Obecně platí, že informace může nabýt smyslu jenom tehdy, když existuje médium, které je schopné ji interpretovat. Interpretace pak může proběhnout jenom v případě, když dojde k interakci mezi zdrojem a adresátem informace. Přitom je nepodstatné, je-li adresátem, popř. zdrojem, informace uměle vytvořený nebo přirozený systém, člověk, jiná živá bytost nebo stroj. Z tohoto obecného faktu plyne, že interakce adresáta s uměleckým dílem je i v oblasti umělecké komunikace asi stejně starým jevem jako umění samo. Bez toho, aby artefakt oslovil, nestane se uměleckým dílem.

Na druhé straně si jenom stěží – jestli vůbec! – dovedeme představit artefakt oslovující důsledně pasivního adresáta. Na obraz se musíme dívat, knihu musíme číst, kolem sochy se musíme projít. Musíme být aktivní, chceme-li, aby nás tyto artefakty nějak oslovovaly. Definice *interaktivního umění* jako *části umění, která nějakým způsobem zahrnuje pozorovatele...* (kterou poskytuje např. internetová encyklopedie www.wikipedia.org) se tedy může zdát málo výstižná. Vždyť součástí porozumění *každému* uměleckému dílu je jeho adresát[41]. Definice se možná stane výstižnější, přidáme-li k požadavku *zahrnování* pozorovatele uměleckým dílem požadavek *přímé* (*a nevratné*) *intervence* vnímatele do uměleckého díla. Socha se tím, že jí obcházím, pro jiného, který jí obchází spolu se mnou, nestává jinou. Interaktivní umělecké dílo by se tím, že zahrne adresáta, jiným stát mělo i pro jiného adresáta, a to právě díky intervenci do něj.

Tak se dostáváme k hledání vymezení toho, co pokládat za *interaktivní médium*. Je to takové médium, které je přiměřené vnějším intervencím. V protikladu k tradiční

[40] Tento pojem tu nepoužíváme jako ekvivalent negativně pojímaného nelidského!

[41] Toto je velice obecný sémiotický princip, který se uplatňuje se i daleko za hranicemi umění. V souvislosti s funkcí genetického kódu pro utváření se organismu a počítačového softwaru pro výkon jím řízeného robota o něm píšeme například v (Markoš, Kelemen, 2004).

představě o uměleckém díle jako jedinečném a neopakovatelném artefaktu a o umělci jako geniálním tvůrci staví umění využívající interaktivní média – *interaktivní umění* – zcela protikladné principy: umělecké dílo vytvořené v prostředí interaktivních médií má potenciálně nekonečnou řadu podob, které nabývá skrze interaktivitu. Proti jedinému autorovi staví kolektivní, kolaborativní autorství. Tyto principy, které jsou analogické anti-uměleckému hnutí 60. let, nejsou pouhou deklarací určitých preferencí umělců, nýbrž vycházejí ze samotných principů tvořících základ interaktivního média, jak je analyzuje americký umělec a teoretik *nových médií* Lev Manovich (2001).

V prostředí interaktivních médií můžeme vysledovat od 2. světové války paralelní tendence s vývojem moderního umění. Tento paralelismus změnil vztah mezi uměním a technologií. V posledních několika desetiletích 20. století technologie interaktivních médií realizovaly některé klíčové vize a projekty moderního umění přibližně ve stejné době, kdy byly tyto vize formulovány. Výsledkem je, že nové technologie – interaktivní média – se samy staly nejúžasnějším uměleckým dílem dneška. Nejrozšířenějším dílem interaktivního umění je z tohoto hlediska interface člověk–počítač.

Paralelismus vývoje umění a interaktivních médií zahrnuje nejen myšlenky, ale i formy, v kterých jsou tyto myšlenky sdělovány. Ve 20. století umělci charakteristicky prezentovali své myšlenky buď psaním manifestů, nebo vytvářením aktuálních uměleckých děl. V případě počítačových expertů máme k dispozici buď teoretické články, které rozvíjejí jak návrhy konkrétních softwarů, tak hardwarů (což jsou jakési analogie k manifestům), nebo více popisné články o již vytvořených prototypech nebo již fungujících systémech (což můžeme chápat jako pandan k aktuální umělecké tvorbě).

Jednoduchým, současně však výjimečně podmanivým příkladem použití interaktivního média byl projekt *Kinoautomat* Radúze Činčery, žnoucí zasloužený úspěch na Expo '67 v kanadském Montrealu v roce 1967. Šlo o filmový příběh s nádechem černého humoru postavený na sérii flashbacků muže, který se činí zodpovědným za požár svého bytu. Po každé ucelenější sekvenci filmu se před plátnem objevila osoba a požádala diváky, aby hlasovali o tom, jakým ze dvou nabízených možností má příběh pokračovat. Hlasovalo se tlačítky spojenými se zelenými a červenými světly obklopujícími promítací plátno. Výsledek hlasování byl takto viditelný a většina rozhodla o pokračování děje, který uskutečňoval kinooperatér obsluhující dvě synchronizované promítací aparatury. Někteří teoretici umění interaktivních médií, např. Michael Naimark (Naimark, 1998), pokládají *Kinoautomat* za jedno z doposud nejzdařilejších děl v této oblasti.

Součástí *Kinoautomatu* však nebyl počítač. Ten se stal poprvé součástí interaktivního uměleckého projevu pravděpodobně v díle Američana Myrona Kruegera, který začal v roce 1969 se svými spolupracovníky na Wisconsinské univerzitě vytvářet interaktivní dílo *Glowflow*. Bylo to prostředí místnosti, ve které snímaly pohyb návštěvníků senzory vbudované do podlahy a citlivé na tlak. V rozích místnosti byly umístěny reproduktory a po stěnách trubice naplněné suspenzemi různých barev. Pohyb návštěvníků byl vyhodnocován počítačem a výsledky vyhodnocení rozhodovaly o tom, jaké zvuky budou vydávat reproduktory a které trubice se rozsvítí[42].

Pokoušíme-li se postihnout, co dává interaktivním uměleckým projektům nebo dílům jejich zvláštní působivost, čím nás oslovují, co jim dává jejich zvláštní kouzlo, možná mnozí budeme souhlasit s tím, že je to právě jejich *připravenost k interakcím* s námi. Je

[42] Více o zkušenostech M. Kruegera s projektem *Glowflow* najdeme v (Krueger, 2003).

to jejich specifická „neukončenost", nebo „neuzavřenost", která je neustálým příslibem, že se v rámci interakce adresáta s dílem může něco nepředvídatelně „přihodit", že může každého interagujícího adresáta stihnout individuální a možná už nikdy jindy se nevynořící zážitek, který svou ojedinělostí může v konkrétním pozorovateli vyvolat *zcela* individuální estetické, etické, emotivní, nebo kognitivní prožitky.

Jako příklad uvedeme „starý ale dobrý" interaktivní videoprojekt Američana Bena Pipera, se kterým jsem se setkali na retrospektivní výběrové výstavě videoumění *Origins* v galerii Art Interactive v massachusettském Cambridge ve dnech 26. 4.–6. 7. 2003 v rámci bostonského festivalu kyberumění: Kamera snímá na bílém pozadí se pohybující postavy pozorující obrazovku, na kterou se snímané sekvence promítají. Náhodně vybírané sekvence jsou pak viditelné spolu s právě aktuálními. Čím delší dobu je sekvence uchovávána v paměti (stroje), tím je neostřejší, až zcela zmizí její kontury v mlze a stane se pro nás „neviditelnou". Celá instalace funguje tedy jako jakási „paměť", která pomalu ale jistě zapomíná, ale současně jako jakýsi prostor imaginace, podobný prostoru naší mysli, odpoutaný od zákonitostí reality, ve kterém se mohou propojit záznamy prožitků, obrazy osob, které by se v kauzální příčinnosti časoprostoru, ve které žijeme, nikdy nemohli setkat...

Jiným dobrým příkladem interaktivní instalace je projekt *A-Volve* Christy Sommererové a Laurenta Mignonneaua s vodní hladinou, pod kterou jsou instalovány obrazovka počítače s živým rybám podobnými virtuálními kreaturami a technické zařízení na zjištění poloh předmětů ponořených do vody. Kreatury mají individuální chování, v jehož rámci reagují na adresátovy pohyby rukou ponořenou do vody nad obrazovkou. Vzniká dokonalá iluze interakce kreatur s pozorovatelem. Systém umožňuje si kreatury i vytvářet dle vlastních figurativních a behaviorálních představ a interagovat pak i s vlastními výtvory[43]. Interakce s *A-Volve* jsme měli možnost vychutnat v září 2000 v rámci specializovaného tutoriálu Christy Sommererové na konferenci ALIFE VII v oregonském Portlandu na tamní Reed College.

## 2.11 Decentralizovanost

Fakt, že se ze stroje stalo (multi)médium interagující s člověkem, představuje výzvu nejenom pro ty, kdo stroje resp. možnosti jejich interakce s lidmi využívají jako prostředek pro zkvalitnění některých lidských činností, schopností nebo prožitků, nýbrž i pro ty, kteří zkoumají a uskutečňují možnosti a způsoby zdokonalování strojů. Podstatným však pro nás teď bude třetí pohled – pohled těch, kteří hledají způsob jak zkoumat interakce stroje a člověka tak, abychom se pokud možno nejméně soustřeďovali na odlišnosti mezi člověkem a strojem a co nejvíce zdůrazňovali to, co je v rámci jejich interakcí spojuje nebo může spojovat. Takto se člověk, stroj a jejich interakce stávají předmětem *kognitivní vědy*.

Zřetelně se kognitivní věda začala z výzkumů umělé inteligence vyčleňovat ve druhé polovině minulého století. V té době koexistovaly v umělé inteligenci dva směry, které se zdály být nesmiřitelné a které nejlépe charakterizují aktivity v oblasti automatického

[43] Umělecko-kritický rozbor a hodnocení projektu *A-Volve* nabízí (Whitelaw, 2004).

dokazování logických teorémů na jedné straně a aktivity v oblasti napodobování psychických procesů předpokládaných při řešení problémů u člověka.

Klasickým příkladem úspěšného projektu „psychologického" směru byly aktivity kolektivu proměnlivého složení seskupeného kolem dvou vůdčích osobností – H. A. Simona a A. Newella – zaměřené na výpočetně uskutečnitelné (programovatelné) postupy řešení problémů. Představa o tom, co jsou vlastně problémy a „mechanismus" nacházení jejich řešení, kterou postupně výpočetně realizovali, je ve stručnosti následující:

Problém se nám objevuje, je-li stav $C$, který si přejeme, odlišný od stavu $E$, který právě existuje. Tuto odlišnost si uvědomujeme tak, že si uvědomíme konečný počet, třeba $n$, různých odlišností $D(C, E) = \{d_1(C, E), d_2(C, E), \ldots, d_n(C, E)\}$ mezi stavy $C$ a $E$. (Neexistují-li mezi dvěma stavy žádné rozdíly, pak jsou z hlediska našeho problému nerozlišitelné.) Intuitivně cítíme, že ne všechny rozdíly mají stejnou důležitost. Proto budeme předpokládat, že z množiny $D(C, E)$ můžeme vždy jednoznačně vybrat tu nejdůležitější odlišnost, kterou (pro stavy $C$ a $E$) označíme $d_{\text{nej}}(C, E)$. K ničemu by nám byla jakkoliv precizní znalost problémů a odlišností mezi dvojicemi stavů, kterými jsou problémy určeny, kdybychom neměli možnost se určitými operacemi dostávat postupně do stavů, které budou stále méně odlišné od cílového stavu. Budeme předpokládat, že máme k dispozici konečnou množinu takovýchto operací $F = \{f_1, f_2, \ldots, f_m\}$, kde libovolná operace $f_i$ z $F$ je z matematického hlediska funkcí (přiřazuje tedy danému stavu, na který je aplikována, jediný nový stav, který vznikne její aplikací). Řešení problému je pak konečná posloupnost operací z $F$, které krok za krokem transformují počáteční stav $E$ na cílový stav $C$.

Jak takovou posloupnost najít? Uvědomíme-li si, že konečných posloupností sestavených z jediného prvku je nekonečně mnoho, přirozeně se nabízí otázka, jak z tohoto množství nějak „mechanicky" (a v poměrně krátkém čase) vybrat právě tu posloupnost, která je řešením našeho problému. Jak asi postupujeme my, lidé? V našich myslích jsou operace, které známe a umíme vykonat, spjaté s příslušnými odlišnostmi. Poznáme-li odlišnost $d_{\text{nej}}(X, Y)$ stavů $X$ a $Y$, pak k ní najdeme operaci z $F$, která je schopna ji redukovat.

Popsaný obecný postup řešení problémů je výpočetně realizovatelný a stal se základem k vytvoření systému pro automatické řešení problémů, známého v literatuře o umělé inteligenci jako systém *GPS* (což je zkratkou z angl. *General Problem Solver*). Jeho specifičností je, že pro řešení konkrétního typu problémů se do něj vždy dostane i „kousek" ze zkušeností toho, kdo ho programuje. Je na programátorovi, jak popíše stavy, jak zvolí odlišnosti, které se rozhodne si všímat, a jak je uspořádá, jaké operátory si zvolí a jak bude vypadat tabulka odlišností a operátorů. Proto se *GPS* stalo výpočtově realizovaným systémem, který umožnil psychologům experimentovat s jejich hypotézami, a stal se tak první vlaštovkou v oblasti rozvoje kognitivní vědy; viz např. (Newell, Simon, 1972)[44].

Systém *GPS* ztělesňoval něco z našich představ o tom, jak přemýšlíme. Jedna okolnost, kterou neztělesňoval, byla lidmi při řešení problémů obvykle hojně využívaná možnost – možnost ptát se zkušenějších, možnost *interakce* a *komunikace* s jinými

[44] Monografické zpracování ucelenější teorie kognitivnosti, která má svoje počátky v experimentech se systémem *GPS*, představuje (Newell, 1990).

řešiteli problémů v zájmu vyřešení individuálního problému. Druhou okolností, při konstrukci *GPS* zcela ignorovanou bylo, že problémy většinou úspěšně řeší *kolektivy* lidí, které mohou být tvořeny různými specialisty pro řešení různých typů podproblémů a mohou být v čase proměnlivé. Řešení problému pak vzniká rozkladem původního problému a řešením jednodušších problémů, jeho přesouváním jiným týmům specialistů, tedy využitím *kooperace*. Takovéto představy však samozřejmě nezůstaly rozvojem kognitivní vědy nereflektovány a byly zahrnuty do širšího diskurzu *decentralizace*.

Poprvé se idea decentralizace objevila implicitně u snah napodobit neuronovou architekturu přirozené nervové soustavy v naději, že chování takovéto napodobeniny bude podobné chování originálu. V čase svých studií na Princetonské univerzitě se začal seznamovat s tehdy intelektuálně mimořádně vzrušující interdisciplinární oblastí – s *kybernetikou* – i jeden z pozdějších průkopníků umělé inteligence a kognitivní vědy – Marvin Minsky. V Cambridgi (Mass.) se na seminářích Norberta Wienera často a mnoho diskutovalo o problematice neuronů a neuronových sítí, o možnostech jejich matematicky precizního popisu a technického napodobení[45]. Inspirován částečně i těmito diskusemi sestavil v roce 1951 Minsky první umělou *neuronovou síť* schopnou učit se a nazval ji *SNARC* (z angl. *Stochastic Neural-Analog Reinforcement Computer*). Dnes je potřebné zvlášť zdůraznit, že šlo o hardwarovou realizaci jednotlivých neuronů i celé sítě, ne o počítačovou digitální simulaci nebo emulaci.

V 80. létech minulého století se začala – i díky technologickému pokroku a následnému poklesu cen procesorů – prosazovat snaha inspirovat se při volbě architektury inteligentních strojů předpokládanou architekturou lidské mysli jakožto vysoce decentralizovaného resp. distribuovaného systému. Částečně to znamenalo renezanci výzkumu neuronových sítí, vždyť tento názor je rozvinutím a zobecněním chápání mysli jako neuronové sítě dle představ kybernetiky 50. let. Zdrojem názorového posunu byla však i reakce na nové způsoby využívání výpočetní techniky – na enormní rozvoj počítačových sítí. Psychologové i neurofyziologové navíc dospěli k představě rozložitelnosti mysli (tedy toho, co dělá mozek) na funkčně specializované vzájemně komunikující moduly. Minsky se pokusil tuto představu precizovat a rozvinout v pojmosloví teorie výpočtů. Jeho záměrem bylo dát strukturám používaným při rámcové reprezentaci poznatků mnohem větší autonomnost, než tomu bylo v původní koncepci. Z jednotek rámcové reprezentace se měly stát autonomně se chovající a vzájemně komunikující softwarové celky uzpůsobené vzájemným interakcím a vytvářením složitějších celků svého vlastního typu. Měly tedy nabýt podobu agentů schopných seskupovat se do větších celků – agentur – které by za určitých okolností samy fungovali jako agenti.

První článek o mysli jako společenství specializovaných výpočetních agentů publikoval Minsky v roce 1980 (Minsky, 1980). Uceleně pak předložil svou teorii ve zvláštně koncipované knize (Minsky, 1986) sestávající z 270 jednostránkových, srozumitelných a vzájemně na sebe odkazujících esejí[46].

Se svou teorií pak udělal to, o čem již byla zmínka: spolu s Harrym Harrisonem napsali technothriller (Harrison, Minsky, 1992). Na pozadí detektivní zápletky z roku 2023 odehrávající se převážně ve vývojové laboratoři inteligentních robotů, se objeví

[45] Působivou a obsažnou intelektuální biografii N. Wienera je (Conway, Siegelman, 2005).

[46] S dopady a rozvojem Minskyho societní teorie mysli po vydání *The Society of Mind* seznamuje v přehledné podobě (Singh, 2003).

nutnost (re)konstruovat poškozenou mysl jednoho klíčového specialisty. To otevírá autorům možnost předvést všechny zásadní myšlenky *societní teorie mysli*, mnoho z jiných teorií umělé inteligence, dokonce četné klíčové projekty Laboratoře umělé inteligence a Laboratoře médií MIT a dokonce i něco z organizačního pozadí a někdy dost komplikovaných vztahů mezi dnešními centry výzkumu a předními výzkumníky (jména jsou pochopitelně pozměněna). V knize jsou přítomny i první náznaky problematiky emotivnosti lidí a strojů, které Minsky věnoval monografii *The Emotion Machine*.

Pocit bezvýchodnosti může z výpočetního hlediska vznikat třeba takto: v mysli řešitele (nebo umělého řešiče) problémů se postupně naplňují nějaké pomocné paměti specifikacemi problémů, které vznikají a čekají na řešení v procesu, jenž jsme naznačili v souvislosti se systémem *GPS*. V mysli existuje nějaký specializovaný agent, který má za úkol sledovat stav naplněnosti těchto pomocných pamětí a v případě jejich přeplňování aktivovat jiného agenta, jehož úkolem je zamezit dalšímu generování podúkolů. Třeba právě aktivaci tohoto agenta vnímá možná naše vědomí způsobem, který jsme nazvali bezvýchodnost. V takovémto stavu vědomí se pak mohou aktivovat jiné strategie řešení problémů (třeba rozhodnutí na základě náhody nebo třeba použití jiných operátorů, než jaká byla v procesu původně při hledání řešení problémů uplatňována...). Často vznikají tyto situace jako následek tlaku reálného prostředí na řešitele problému (prozatím méně často na řešič). Například člověk rozhodující se o volbě vhodné restaurace je čím dál hladovější, člověk hledající odpovídající zaměstnání má stále méně peněz na obživu apod. A možná to platí dokonce celé i obráceně. Uvědomujeme si problémy proto, neboť naše interakce s prostředím v nás vyvolává vznik velice prvoplánových (až v biochemii našich organismů ukotvených) problémů (třeba potřebu doplňovat v organismu hladinu vody nebo cukrů, tuků a bílkovin), které jsme nuceni řešit v prostředích určité komplikovanosti.

Interakce s prostředím je tedy možná velmi aktivním prvkem fungování přirozených i umělých myslí, který je třeba brát v potaz, pokoušíme-li se hledat jakési (výpočetně formulovatelné) vysvětlení nastávání emotivních stavů a jejich souvislost s kognitivní složkou myslí. Jaká je však síla interakce z pohledu teoretické informatiky?

## 2.12 Síla interakce

V našem dosavadním pokusu o analýzu stroje jsme jej uviděli v mnoha podobách, ve kterých posloužil rozvoji naší představivosti, našeho materiálního světa, naší (sebe)-reflexi v uměleckých dílech rozmanitého druhu i našemu chápání procesů určitých druhů, především pak výpočetních procesů. Za předpokladu určitého formálně precizního chápání stroje – třeba v podobě tzv. Turingova stroje – existují určité hranice toho, co je takovýmto strojem (tedy v tzv. *turingovském smyslu*) vypočitatelné.

Koncem 20. století se představa univerzálnosti Turignova stroje postupně problematizovala. Jedním z důvodů bylo zjištění, že ne vždy je nejvýstižnější představou o fungování reálných počítačů jejich chápání jako zařízení transformujících vstupní hodnoty proměnných matematicky definovaných funkcí konečným počtem elementárních operací na příslušné výstupní funkční hodnoty. Reálné výpočetní systémy se postupně

stávaly mnohem otevřenějšími vnějším vlivům, různým *interakcím*, což původní představu o počítání zčásti zpochybňovalo; viz např. (Wegner, Goldin, 2003), (Wegner, 1997).

Pochybnosti vznikaly i pokud jde o přiměřenost konceptuálního modelu reálných počítačů v podobě Turignova stroje jejich fyzikálně (technicky) uskutečněným podobám. Objevily se názory, že právě *ztělesněnost* (angl. *embodiment*) reálných počítačů může podstatným způsobem měnit jejich schopnosti ve srovnání s těmi, na které usuzujeme ze schopností Turingova stroje a s ním ekvivalentních formálních modelů; podrobněji viz například (Sloman, 2002).

V průběhu našeho výkladu jsme se soustředili na problematiku interakce stroje a jeho prostředí, popř. na problematiku interakce strojů navzájem. Viděli jsme, jak může důraz na interaktivitu posunout hranice uměleckého sebevyjádření. Viděli jsme i to, jak může představa jednoduchých interagujících strojů (agentů) ovlivnit představy o našich vlastních mentálních schopnostech. Zůstává už jenom položit si otázku, může-li interaktivita ovlivnit výpočetní sílu našich (abstraktních) strojů.

Dnes se postupně objevují takové formální modely výpočtů, které nabízejí rozšířením jejich tradičního pojímání v přibližně podobném smyslu, v jakém se fyzice podařilo „rozšířit" tradiční newtonovskou mechaniku do podoby relativistické a kvantové mechaniky. Takovéto pokusy o „rozšíření" pojmu *vypočitatelnosti* se dnes dějí v různých směrech.[47] V zásadě všechny snahy jsou postaveny na tom, že do modelu výpočtu v nějaké podobě zahrnou i cosi nevypočitatelného, ale teoreticky dobře zkoumatelného. Třeba obecnější představu o počítání s reálnými čísly, než tomu bylo v (Turing, 1936), nebo představu zařízení, které je schopno v některých momentech výpočtu zodpovědět určitý typ otázek – tzv. *orákula* – apod.

Uvedeme příklad, který staví na předpokladu, že výpočetní zařízení sestává z více agentů, kteří mají individuálně každý nanejvýš tradiční výpočetní sílu Turingova stroje a kteří se mohou v průběhu výpočtu seskupovat z různých důvodů – třeba proto, že mají svoje *těla*, které jim brání vstupovat kdykoliv do procesu výpočtu – do různých „týmů" a počítat pak týmově, ve skupinkách s měnícím se počtem členů. Takovýto předpoklad vykonávání výpočtů se nám jeví jako velmi blízký realitě zpracovávání informace v běžném životě. Proto se nám otázka „výpočetní síly" takovýchto multiagentových systémů jeví jako zajímavá.

Formalizovaný model, který naznačenou představu zčásti vyjadřuje, byl předložen v podobě *systému formálních gramatik*[48] pracujících nad společně sdíleným řetězcem symbolů, který se mění nejen následkem propisování podle pravidel těchto gramatik, nýbrž i následkem vlastní dynamiky (vyjádřené L-systémem). Model jsem nazvali *ekogramatický* (zkráceně *EG*) *systém*. Podařilo se v něm vyjádřit některé vlastnosti, se kterými se setkáváme i u živých bytostí, podíváme-li se na ně z informatického hlediska.

V (Csuhaj-Varjú a kol., 1997) jsme definovali *ekogramatický systém* $\Sigma$ jako systém sestávající z abecedy $V$ symbolů, z fixovaného počtu (třeba z $n$) *agentů* (definovaných pomocí $n$ formálních gramatik) a z *prostředí* (popsaného jako řetězec symbolů).

---

[47] Přehled takových přístupů poskytuje například časopis *Theoretical Computer Science* **317** (2004) No. 1–3, s. 1–267 sestavený M. Burginem a A. Klingerem.

[48] Systémy formálních gramatik jsou představeny například v (Csuhaj-Varjú a kol., 1994).

Prostředí daného EG systému se zásluhou přepisovacích aktivit gramatik (agentů) a pravidel pro změny prostředí mění v *krocích*. Každý krok má dvě fáze:

V první fázi kroku změny EG systému agenti přepisují symboly ve sdíleném prostředí tvaru řetězce $w_E$ symbolů z množiny $V$. Dělají to sekvenčně (v jednom kroku přepisuje jediný agent jediný symbol) a každý podle vlastní konečné množiny přepisovacích pravidel, seskupených do zodpovídajících množin $R_1$, $R_2$, ..., $R_n$. Stav prostředí na začátku práce systému agentů označme řetězcem symbolů $w_0$.

Ve druhé fázi kroku změny EG systému se stav prostředí $w_E$ mění podle vlastních zákonitostí prostředí. Tyto zákonitosti jsou vyjádřeny taky konečnou množinou $P_E$ přepisovacích pravidel. Pravidla jsou aplikována paralelně – tak, jako je tomu u tzv. L-systémů[49]. K jejich aplikaci dochází až poté, co žádný z agentů již nemá co v řetězci $w_E$ přepisovat. Paralelní propis symbolů se přitom smí vykonat pouze v případě, je-li možné pomocí pravidel z množiny $P_E$ přepsat *všechny* symboly vyskytující se v řetězci $w_E$, které nebyly v daném kroku přepisování daným EG systémem přepsány agenty.

Takto krok za krokem postupným vykonáváním dvoufázového přepisování vzniká množina řetězců $\{w_0, w_1, w_2, ...\}$, kterou nazýváme *jazykem* generovaným daným EG systémem. Ve zmíněném článku jsme ukázali, kromě jiných vlastností EG systémů, i vcelku očekávatelný fakt, že takto chápané EG systémy jsou schopny generovat právě ty jazyky, které jsou schopné generovat Turingovy stroje, tedy že náš model a Turingův stroj jsou – pokud jde o jejich „výpočetní sílu" – ekvivalentní.

V (Csuhaj-Varjú, Kelemenová, 1998) byla navržena speciální varianta EG systémů, která generuje jazyky skupinami agentů – tzv. *týmů*. Daným kladným celým číslem $k$, přidruženým k danému EG systému, je určen počet agnetů v týmu. Kteří agenti budou členy týmu není podstatné, musí jich však být přesně $k$ a v průběhu vykonání první fáze daného kroku přepisování se nemohou měnit. Pro daný krok přepisování se tím určí množina agentů, kteří budou v první fázi kroku přepisování prostředí participovat. Po vykonání druhé fáze se může tým změnit, musí však mít opět přesně $k$ prvků.

Wätjen (2003) navrhl fixovaný počet agentů z (Csuhaj-Varjú, Kelemenová, 1998) nahradit představou dynamicky se měnícího počtu agentů v týmu. Jako rekonfigurační mechanismus navrhl použít funkci $f$ definovanou na množině všech kladných celých čísel $N$ s hodnotami v množině $\{0, 1, 2, ..., n\}$, kde $n$ je počet agentů daného EG systému $\Sigma$. Pro $i$-tý krok přepisování daného EG systému udává tedy funkce $f$ číslo $f(i) \in \{0, 1, 2, ..., n\}$. Pro vykonání první fáze kroku přepisování pak bude použita libovolná podmnožina množiny všech agentů daného EG systému $\Sigma$ s počtem prvků právě $f(1)$, ve druhém s počtem prvků právě $f(2)$ atd. Ve (Wätjen, 2003) je dokázáno, že existují takové EG systémy s jeho typem týmů, pro které platí: Je-li $f$ v Turingově smyslu nevypočitatelné funkce, pak je jazyk definovaný takovýmto EG systémem Turingovým strojem neakceptovatelný (nerekurzivní), tj. nedá se Turingovým strojem pro libovolné slovo z tohoto jazyka zjistit (vypočítat), patří-li do tohoto jazyka, nebo do něj nepatří.

Síla výpočetně velice jednoduchých strojů, jakými jsou agenti, jejichž formální modely vystupují v EG systémech, tedy z výpočetního hlediska velice významně narůstá třeba díky jejich seskupování a jejich jednoduchým interakcí s jednoduchým prostředím tvořeným řetězcem symbolů.[50]

---

[49] Trojice $(V, P_E, w_E)$ tedy představuje L-systém ve smyslu (Rozenberg, Salomaa, 1980).

[50] O možných důsledcích tohoto nárůstu pojednáváme v (Kelemen, 2005a, b).

## 2.13 Závěr

Když jsme sledovali vývoj představ člověka o tom, co to vlastně stroj je, a v té souvislosti psali o Turingových úvahách o inteligenci, zformulovali jsme hypotézu, tzv. druhou Turingovu hypotézu, podle níž je inteligence redukovatelná na soubor turingovsky vypočitatelných funkcí. Postupně jsme se dostali k odůvodněnému tušení, že výpočetní síla, kterou Alan Turing (ale v první řadě Alonzo Church) stanovil v obecně známější hypotéze – v tzv. Churchově a Turingově hypotéze – za hraniční pro abstraktní univerzální stroj, který objevil, je překonatelná. V kontextu EG systémů se jí podařilo překonat zejména díky tomu, že do formálního modelu výpočtu byla zahrnuta i možnost nepředvídatelného (ale třeba díky náhodě možného!) seskupování výpočetních zařízení do proměnlivých týmů a možnost práce těchto týmů v prostředí, které sdílejí.

Pokusíme-li se promítnout tuto zkušenost do našich hypotéz týkajících se intelektuálních schopností našich strojů, můžeme dospět k názoru, že inteligence může vzniknout na bázi turingovsky vypočitatelných funkcí, na které je redukovatelná schopnost agentů zpracovávat symbolické reprezentace jejich prostředí, a na bázi možnosti seskupování těchto agentů a jejich aktivit ve sdíleném prostředí. Inteligence se takto stává výsostně kolektivním fenoménem vyvstávajícím (emergujícím) ve společenstvích aktivních subjektů, které na ni individuálně nedosahují. Možná právě v tomto tkví důvod, proč se tak stěží daří inteligenci obsahově vymezit, snažíme-li se ji vztahovat na jednotlivce a vyhnout se tomu, abychom při vymezování odkazovali i na jeho vazby na jiné jednotlivce. Možná právě v tomto spočívá i příčina, proč bylo zapotřebí přesunout v oblasti umělé inteligence orientaci výzkumu z hledání způsobů jak programovat jednotlivé izolované systémy, aby prošly Turingovým testem, na oblast zkoumání významu interakcí mezi agenty, na problematiku multiagentových systémů a na oblast kolektivní robotiky.

Pokusili jsme se také načrtnout podobu kulturního vývoje, který byl často v průběhu minulého století pro úvahy o povaze, schopnostech, podobnostech a odlišnostech strojů a lidí inspirativní nejen pro přemýšlení o lidech, nábrž i pro další poznávání a zdokonalování strojů. Nabízí se tedy nyní, na závěr, dost obecná otázka: Kam takovýto rozvoj povede stroje a zejména lidskou společnost?

Francouzský postmoderní filosof Jean-François Lyotard, shrnuje a kritizuje vývoj názorů ve 20. století na společnost jakožto na samoregulující systém, především Parsonsovy názory o společnosti jakožto kybernetickém systému, napsal: „... *opravdovým účelem systému, tím, k čemu sám sebe programuje jako jakýsi inteligentní stroj, je optimalizace celkového vztahu mezi jeho imputy a outputy, to znamená jeho perfromativita*“ (Lyotard, 1993).

Je zde však jeden aspekt „kybernetického“ pohledu na člověka a jeho společnost, pohled, který nabízí rozvinutí kybernetických myšlenek a představ do podoby současné umělé inteligence. V naznačeném kontextu snad nebude tento aspekt nezajímavý. Málokdy a málokde je o něm zmínka. Snad i proto, že v záplavě opačných názorových proudů působí poněkud paradoxně. Explicitně se však o něm zmínila např. Margaret Boden, když napsala, že jí „...*umělá inteligence přitahuje především pro možnosti, které skýtá pro potírání dehumanizačního vlivu přírodních věd*“ (Boden, 1977). My dodáváme, že nejenom vědy, nýbrž i techniky.

Nový význam, kterým – i pod vlivem oblasti umělé inteligence a kognitivní vědy – v současnosti naplňujeme pojem *stroj*, se zčásti překrývá nejen s významy mnoha pojmů ve fyzice, elektrotechnice a v informatice, nýbrž postupně i s významy pojmů z psychologie, sociologie, obecně humanitních věd, kultury a umění. Stroje postupně opouštějí oblast fyzikálního světa a přibližují se ke stále méně zřetelným hranicím světa živých bytostí a lidí, protože i my lidé jim vycházíme vstříc. Prozatím však obecně zdaleka není zcela jasné, bude-li výsledkem tohoto přibližování humanizace strojů, dehumanizace člověka, nebo konstituce nějakého nového humanismu…

„*Jde jen o to* – píše Egon Bondy (1993, s. 112) – *jak brzo si člověk uvědomí, že nejde o výrobu jeho 'služebníka', ale o jeho vlastní sebepřekročení jako biologicky vázané bytosti, která se v rámci biologična nemůže už kvalitativně výše rozvinout, tj. nemůže rozvinout všechno to, na co intencionálně poukazuje svou lidskou existencí.*" Katherine Hayles je ještě přímočařejší: „*Čím dál, tím méně je otázkou, jestli jsme se již stali posthumánními, protože naše posthumánnost je již skutečností. Raději se ptejme, jakým druhem posthumánních bytostí budeme*" (Hayles, 1999).

## Literatura


Ambros V.: *Jak si v Americe osvojili Čapka aneb ze života hmyzu a robotů v Americe*. Divadelní revue, č. 4, 1996, s. 19–23.

Aristoteles: *Politika*. Petr Rezek, Praha 1998.

Asimov I.: *I, Robot*. Doubleday, New York 1950 (česky např. Odeon, Praha 1981).

Baudrillard J.: *Videowelt und fraktalen Subjekt*. In: Philosophie der neuen Technologie. Merve Verlag, Berlin 1989, s. 113–131.

Boden M. A.: *Artificial Intelligence and Natural Man*. The Harvester Press, Brighton 1977.

Bondy E.: *Filosofické eseje II – Juliiny otázky, Doslov*. Dharma Gaia, Praha 1993.

Brooks R. A.: *Cambrian Intelligence*. The MIT Press, Cambridge, Mass., 1999.

Burks A. W. (ed.): *Essays on Cellular Automata*. University of Illinois Press, Chicago, Ill., 1970.

Codd E. F.: *Cellular Automata*. Academic Press, New York 1968

Conway F., Siegelman J.: *Dark Hero of the Information Age – In Search of Norbert Wiener, the Father of Cybernetics*. Basic Books, New York 2005.

Corbin J.: *A Czech-Slovak Frankeinstein*. The New York Times, 8. 10. 1922.

Csontó J., Palko M.: *Umelý život*. Elfa, Košice 2002.

Csuhaj-Varjú E., Dassow J., Kelemen J., Paun Gh.: *Grammar Systems – A Grammatical Approach to Distribution and Cooperation*. Gordon and Breach, Yverdon 1994.

Csuhaj-Varjú E., Kelemen J., Kelemenová A., Paun Gh.: *Eco-grammar systems – a grammatical framework for studying life-like interactions*. Artificial Life **3**, 1997, s. 1–28.

Csuhaj-Varjú E., Kelemenová A.: *Team behaviour in eco-grammar systems*. Theoretical Computer Science **209**, 1998, s. 213–224.

Čapek, J.: *Homo artefactus*. In: Rodné krajiny (V. Binar, sest.). Mladá fronta, Praha 1985, s. 107–157.

Čapek K.: *R. U. R. – Rossum's Universal Robots* (sestavil M. Halík). Orbis, Praha 1966.

Čapek K.: *Loupežník, R. U. R., Bílá nemoc*. Československý spisovatel, Praha, 1983.

Čapek K.: *Dramata (Spisy VII)*. Československý spisovatel, Praha, 1992, s. 96.

Čapek K., Čapek J.: *Ze společné tvorby (Spisy II)*. Československý spisovatel, Praha 1982, s. 10–11; s. 89–91.

Čapek K., Čapek J.: *Krakonošova zahrada*. Levné knihy KMa, Praha 2000 (reprint vydání z roku 1929).

Deleuze G., Guattari F.: *Thousand Plateaus, Capitalism, and Schizophrenia*. University of Minnesota Press, Mineapolis, Minn., 1987.
Eberbach E., Wegner P.: *Beyond Turing machines*. Bulletin of the EATCS **81**, 2003, s. 279–304.
Harkins W. E.: *Karel Čapek*. Columbia University Press, New York 1962, s. 104.
Harrison H., Minsky M.: *The Turing Option*. Werner Books, New York 1992.
Hayles N. K.: *How We Became Posthuman*. The University of Chicago Press, Chicago 1999, s. 249.
Hodges A.: *Alan Turing – The Enigma of Intelligence*. Unwin Paperbacks, Londýn 1985.
Horáková J.: *Rossum's Universal Robots – továrna utopie*. *Disk* **8**, 2004, s. 96–110.
Horáková J.: RUR – komedie o robotech. *Disk* **10**, 2004, s. 71–86.
Horáková J., Kelemen J.: *Čapek, Turing, von Neumann, and the 20th century evolution of the concept of machine*. In: Proc. Intern. Conference in Memoriam John (János) von Neumann. Budapest Polytechnic, Budapešť 2003, s. 121–135.
Horáková J., Kelemen J.: *Stroj ve 20. století*. In: Kognice a umělý život IV (J. Kelemen, V. Kvasnička, sest.). Slezská univerzita, Opava 2004, s. 179–198.
Horáková J., Kelemen J.: *Kouzlo a síla interakce*. In: Kognice a umělý život V (J. Kelemen a kol., sest.). Slezská univerzita, Opava 2005, s. 195–215.
Chase S.: *Men and Machines*. Macmillan, New York 1929 (česky J. Laichter, Praha 1931).
Kac E.: *Beyond the screen – new directions in interactive art*. Blimp – Film Magazine No. 40, 1999, s. 49–54.
Kauffman S.: *Čtvrtý zákon*. Paseka, Praha 2004.
Kelemen J.: *Myslenie, počítač...* Spektrum, Bratislava 1989.
Kelemen J.: *Strojovia a agenty*. Archa, Bratislava 1994.
Kelemen J.: *Kybergolem*. Votobia, Olomouc 2001.
Kelemen J.: *May embodiment cause hyper-computation?* In: Advances in Artificial Life, Proc. *ECAL 05* (M. S. Capcarrére et al., eds.). Springer-Verlag, Berlín 2005a, s. 31–36.
Kelemen J.: *On computational study of embodiment – some remarks and an example*. Computing and Informatics **24**, 2005b, s. 617–627.
Kelemen J. a kol.: *Základy umelej inteligencie*. Alfa, Bratislava 1992.
Kelemenová A., Kelemen J.: *O biológii, matematike a výpočtoch – rozhovor s A. Lindenmayerom*. Pokroky matematiky, fyziky a astronomie **31**, 1986, s. 96–105.
Kováč L.: *Úvod do kognitívnej biológie*. Biologické listy **51**, 1986, s. 172–190; viz taky: Kováč L.: *Komentovaný úvod do kognitívnej biológie*. In: Kognice a umělý život IV (J. Kelemen, V. Kvasnička, sest.). Slezská univerzita, Opava 2004, s. 233–258.
Krueger M. W.: *Responsive environments*. In: The New Media Reader (N. Wardish-Fruin and N. Montfort, eds). The MIT Press, Cambridge, Mass., 2003, s. 379–389.
Langton C. G.: *Self-reproduction in cellular automata*. Physica *D* **10**, 1984, s. 135–144.
Langton C. G.: *Artificial life*. In: Artificial Life (Langton, C. G., ed.) Addison-Wesley, Redwood City, Cal., 1989, s. 1–47.
Levy S.: *Artificial Life*. Pantheon Books, New York 1992.
Lindenmayer A.: *Mathematical models of cellular interactions in development, I. Filaments with one-sided inputs, II. Simple and branching filaments with two-sided inputs*. Journal of Theoretical Biology **18**, 1968, s. 280–299, s. 300–315.
Lyotard J.-F.: *O postmodernismu*. Filosofický ústav AV ČR, Praha 1993.
Markoš A., Kelemen J.: *Berušky, andělé a stroje*. Dokořán, Praha 2004.
Manovich L.: *The Language of New Media*. The MIT Press, Cambridge, Mass., 2001.
Mignonneau L., Sommerer Ch.: *Creating artificial life for interactive art and entertainment*. Leonardo **34**, 2001, s. 303–307.
Minsky M.: *K-lines – a theory of memory*. Cognitive Science **4**, 1980, s. 117–133.
Minsky M.: *The Society of Mind*. Simon and Schuster, New York 1986.

Minsky M.: *Konštrukcia mysle*. Archa, Bratislava 1996.
Minsky M.: *The Emotion Machine*. Simon and Schuster, New York 2006.
Moravec H.: *Mind Children – The Future of Robot and Human Intelligence*. Harvard University Press, Cambridge, Mass., 1990.
Murphy R. R.: *Introduction to AI Robotics*. The MIT Press, Cambridge, Mass., 2000.
Naimark M.: *Interactive art – maybe it's a bad idea*. In: Cyberarts International Compendium Prix Ars Electronica. Springer, Vídeň 1998.
Neumann von J.: *The general and logical theory of automata*. In: Cerebral Mechanisms in Behavior – The Hixton Symposium (L. A. Jeffres, ed.). Wiley, New York 1951, s. 1–31.
Neumann von J.: *The Computer and the Brain*. Yale University Press, New Haven, N. J., 1958.
Newell A.: *Unified Theories of Cognition*. Harvard University Press, Cambridge, Mass., 1990.
Newell A., Simon H. A.: *Human Problem Solving*. Prentice-Hall, Englewood Cliffs, NJ, 1972.
Newell A., Simon H. A.: *Computer science as an empirical inquiry – symbols and searc*. Communication of the ACM **19**, 1976, s. 113–126.
Nikolskij S.: *Fantastika a satira v dile Karla Čapka*. Československý spisovatel, Praha, 1987.
Ortega y Gasset J.: *Eseje o umení*. Archa, Bratislava 1994, s. 26.
Petiška E.: *Golem*. Martin, Praha 1992.
Petrů M.: *Možnosti transgrese – Lze měnit člověka?* Triton, Praha 2005.
Piaget J.: *Psychologie inteligence*. Státní pedagogické nakladatelství, Praha 1970.
Rozenberg G., Salomaa A.: *The Mathematical Theory of L Systems*. Academic Press, New York 1980.
Scholes R., Rabkin E. S.: *Science Fiction – History, Science, Vision*. Oxford University Press, New York 1977, s. 36, 173.
Singh P.: *Examining the society of mind*. Computing and Informatics **22**, 2003, s. 521–543.
Sloman A.: *The irrelevance of Turing machines to artificial intelligence*. In: Computationalism (M. Scheutz, Ed.). The MIT Press, Cambridge, Mass., 2002, s. 87–127.
Suvin D.: *Metamorphosis of Science-Fiction*. London, Yale Univeristy Press, 1979.
Šafařík J.: *Člověk ve věku stroje*. Atlantis, Brno 1991.
Thagard P.: *Úvod do kognitivní vědy*. Portál, Praha 2001.
Turing A. M.: *On computable numbers, with an application to the Entscheidungs-problem*. Proceedings of London Mathematical Society **42**, 1936, s. 230–265; corrections in **43**, 1937, s. 544–546.
Turing A. M.: *Computing machinery and intelligence*. Mind **59**, 1950, s. 433–460.
Uspenskij V. A.: *Mašina Posta*. Nauka, Moskva 1979.
Warwick K.: *Úsvit robotů a soumrak lidstva*. Vesmír, Praha 1999.
Wätjen D.: *Function-dependent teams in eco-grammar systems*. Theoretical Computer Science **306**, 2003, s. 39–53.
Wegner P.: *Why interaction is more powerful than algorithms*. Communications of the ACM **40**, 1997, s. 81–91.
Wegner P., Goldin D.: *Computation beyond Turing machines*. Communication of the ACM **46**, 2003, s. 100–102.
Wechsler R.: *Karel Čapek in America*. In: On Karel Čapek, Michigan Slavist Materials No. 34 (M. Makin, J. Toman, eds). Michigan University Slavic Publications, Michigan 1992, s. 111, 122.
Whitelaw M.: *Metacreation – Art and Artificial Life*. The MIT Press, Cambridge, Mass., 2004.

# 3 Výpočetní meze kognitivních a inteligentních systémů

*„Everything that's worth understanding about a complex system, can be understood in terms of how it processes information."*
*Lloyd's Hypothesis (Llod a kol., 2001)*

V této kapitole budeme hledat výpočetní meze kognitivních a inteligentních systémů, a to jak biologických, tak i umělých, ale i hybridních, které jsou kombinacích obou předchozích druhů. Společnou platformu poskytne komputacionalismus, tj. víra, že kognitivní resp. inteligentní procesy jsou v konečném důsledku výpočetními procesy. Ukážeme, že v principu mohou existovat kognitivní systémy, a dokonce i v praxi existují „zárodky" takových systémů, které předčí svou výpočetní silou výpočetní sílu Turingových strojů. Tyto výsledky naznačují, že tzv. Churchovou–Turingovu tezi, hovořící o centrálním postavení Turingových strojů ve světě výpočtů a algoritmů, je třeba vidět v souvislosti s fyzikálními principy, které kognitivní systém při své činnosti využívá, a se způsobem, kterým systém komunikuje s okolím.

## 3.1 Úvod

Pojmem *kognice* se obyčejně myslí procesy, pomocí kterých živé organismy získávají, zpracovávají, ukládají a využívají informace z okolí. Tyto činnosti zahrnují zejména percepci (tj. vnímání), učení, pamatování a rozhodování (Shettleworth, 1998). Vzhledem k této definici můžeme považovat inteligenci za součást kognice, která se příliš nezabývá percepcí a spíše se soustřeďuje na kvalitu kognitivních procesů. Oba dva pojmy se však vztahují ke zpracování informací. Přesvědčení, že lidská, resp. biologická kognice či inteligence je druh výpočtů, hlásá tzv. komputacionalistický směr v umělé inteligenci, viz např. (Chalmers, 1994). Dle zastánců tohoto směru je výpočetní modelování kognitivních schopností živých organismů alespoň v principu možné a tímto způsobem lze dosáhnout ne-li věrného modelování všech mentálních činností (včetně myšlení a vědomí), tak přinejmenším jejich věrné aproximace a také podání jejich vysvětlení. V dalším výkladu nás bude zajímat efektivita výpočetních modelů použitelných při modelování kognice živých organismů. Tuto efektivitu budeme „měřit" standardními postupy teorie výpočetní složitosti – tj. budeme ji srovnávat s výpočetní efektivitou známých modelů v této oblasti. Budeme hledat výpočetní meze těchto modelů, tj. jak efektivně – alespoň v principu – dovedou zpracovat data, a na druhé straně, co již zcela určite nedovedou. Takže tímto způsobem budeme hovořit o kvalitě zpracování, jinými

slovy, současně budeme hovořit o výpočetních mezích inteligence. Pro jednoduchost však budeme v dalším výkladu používat převážně pojmu kognice, který je ve smyslu předchozí neformální definice určujícím i pro pojem inteligence.

Komputacionalistická víra dostala koncem minulého století nečekanou podporu ze strany teoretické fyziky a informatiky. V r. 1985 publikoval teoretický fyzik D. Deutsch článek (Deutsch, 1985), ve kterém dokázal, že libovolný skutečný (disipativní) konečný fyzikální systém lze simulovat pomocí kvantového počítače. Protože kvantový počítač lze simulovat například Turingovým strojem (i když, jak se zdá, velmi neefektivně), máme zde důkaz pravdivosti komputacionalistického přesvědčení, že například kompletního člověka lze „věrně", alespoň v principu, simulovat pomocí klasického počítače. Další výsledek z oblasti teorie složitosti tvrdí, že pomocí kvantového počítače bude tato simulace efektivní (Bernstain, Vazirani, 1993). Jedná se o polynomické složitosti vzhledem k velikosti simulovaného systému.

V našem příspěvku půjdeme ještě dále: nebudeme se zajímat o kognitivní schopnosti živých organismů per se, ale do našich úvah zahrneme organismy (např. lidské bytosti) vybavené navíc jakýmikoliv přístroji, které umožní „zesílit" jejich kognitivní schopnosti v jakémkoliv směru. Příkladem takového zařízení je například Hubbleův teleskop umístněný na družici obíhající kolem Země. Za součást zařízení budeme považovat i příslušné řídicí a výpočetní střediska na zemi. Je nesporné, že takový systém posílí kognitivní možnosti pozorovatele, který toto zařízení využívá: prostřednictvím systému má pozorovatel přístup k datům, které by vlastními prostředky, bez použití dalších pomůcek, nemohl získat. Pozorovatel může navíc tato data zpracovat způsobem, který by bez použití dalších prostředků také sám nezvládl. Výsledný systém, tj. pozorovatele plus jeho zařízení určené k pozorování a zpracování takto získané informace, budeme nazývat *kognitivním systémem*. Výsledkem bude nová kvalita kognice, nedosažitelná bez příslušných přístrojů. Budeme velkorysí a nebudeme trvat na tom, aby dané zařízení bylo skutečně sestrojeno a bylo pozorovateli k dispozici při jeho kognitivních experimentech. Spokojíme se pouze s myšlenkovým experimentem, tj. se situací, kdy přepokládaná existence předmětného „kognitivního zesilovače" nebude ve sporu se známými přírodními zákony. V jeho skutečném sestrojení nám může bránit například jeho cena, velikost, nedostatek zdrojů energie atd., ale nikoliv známé fyzikální zákony. Připustíme i modely, sestávající z potenciálně neomezeného (ale vždy konečného) počtu kognitivních subjektů. Takže v myšlenkových experimentech dovolíme například využití známých vlastností časoprostoru, relativistických i kvantových efektů, vznik nových vesmírů, sestrojení rostoucí kolonie živých organismů, „vypěstování" lidské společnosti komunikující přes internet apod. Pokud setrváme na půdě komputacionalismu, tak na jakékoliv zařízení takového typu můžeme v konečném důsledku nahlížet jako na systém pro zpracování informací. Při takovém velmi obecném přístupu se vše nakonec zvrtne na otázku, jaké jsou meze „myslitelných" kognitivních systémů (rozuměj výpočetních systémů), založených na jakýchkoliv principech, které nejsou ve sporu se známými přírodními zákony.

V teorii vyčíslitelnosti a teorii výpočetní složitosti patří výzkum mezí výpočetní síly výpočetních zařízení k základním otázkám. Termínem „výpočetní síla" rozumíme množinu výpočetních problémů, které je daná třída zařízení v principu schopna vyřešit. Čím větší (ve smyslu ryzí množinové inkluze) je tato množina, tím větší je výpočetní síla dané

třídy zařízení. Standardní tradiční odpověď nabízí tzv. Churchova–Turingova teze (Balcázar a kol., 1995), (Hopcroft a kol., 2000), (Kramosil, 2003), (Turing, 1937), kterou pro naše účely můžeme zformulovat například takto (teze má mnoho podob, my jsme zvolili tu, která zachycuje pojem výpočetní síly).

**Churchova–Turingova teze.** Výpočetní síla jakékoliv třídy výpočetních zařízení, řídících se konečným algoritmy, nepřesáhne sílu Turingových strojů.

*Turingův stroj* byl popsán a definován například v práci (Hopcroft a kol., 2000) anebo ve 4. dílu této série knih. Churchova–Turingova teze jinými slovy tvrdí, ze neexistuje žádné výpočetní zařízení, které by mělo větší výpočetní sílu nežli Turingův stroj. Tuto tezi nelze dokázat, protože nedefinuje přesně, co se myslí pojmem „jakékoliv výpočetní zařízení", a tudíž nelze ukázat, jak by takové zařízení mohl simulovat (napodobit) Turingův stroj. Pokud ale sestrojíme (či alespoň ukážeme možnost sestrojení) zařízení, které bude schopno řešit problémy, Turingovým strojem neřešitelné, tak tezi vyvrátíme. Problémy, které Turingův stroj z principu nemůže vyřešit, se v teorii vyčíslitelnosti nazývají *nerozhodnutelné problémy*. Pokusů o vyvrácení Churchovy–Turingovy teze bylo mnoho a jejich neúspěšnost vlastně vedla k formulaci teze a k víře v její platnost. Teprve v posledních letech se ukazuje, že teze je poněkud neúplně formulována: chybějí totiž explicitní předpoklady, za kterých teze platí. Tyto předpoklady se musejí týkat fyzikální teorie, v rámci které jsou realizována výpočetní zařízení, o kterých teze mluví (Etesi, Németi, 2002), a způsobu, kterým systém interaguje s okolím, popř., kterým okolí působí na systém (van Leeuwen, Wiedermann, 2001). Co se týká první podmínky, v současné době se ukazuje, že, zhruba řečeno, teze platí, pokud uvažujeme současné počítačové technologie, či přesněji, pokud se pohybujeme v rámci světa, který dobře aproximují zákony klasické, newtonovské fyziky, a pokud uvažujeme pouze finitní výpočty. Pokud uvažujeme „moderní" fyziku, včetně kvantové a relativistické, tak se zdá, že teze nemusí platit (Etesi, Németi, 2002), (Hogarth, 1994). Jiný způsob, jak se dostat za rámec teze, je uvažovat obecnější scénář výpočtů, než jsou konečné výpočty realizované Turingovým strojem. To vede k myšlence tzv. interaktivních evolučních systémů, pro které je charakteristický potenciálně nekonečný výpočet, který interaguje s okolím, a navíc i hardwarové modifikace systému během jeho výpočtu (van Leeuwen, Wiedermann, 2001). Takto lze zachytit chování nejen například živých organismů či jejich společenství přes mnoho generací, ale třeba i internetu a podobných vyvíjejících se globálních výpočetních sítí (van Leeuwen, Wiedermann, 2001).

Zde se soustředíme na přehled možností, které mají kognitivní systémy jistého typu a které jim nabízejí potenciál pro překonání Churchovy–Turingovy teze. Abychom mohli hovořit o odchylkách od klasických turingovských výpočtů (tj. těch, které jsou realizovány Turingovým strojem), resp. od výpočtů v rámci Churchovy–Turingovy teze, nejdříve se v oddílu 3.2. stručně zmíníme o výpočetních systémech, jejichž výpočetní mechanismy se neliší od výpočetních mechanismů Turingova stroje a které tuto tezi nevyvracejí. Tyto systémy, modelující kognitivní individua, budou však tvořit východisko pro konstrukci složitějších systémů, které již tezi atakují. Dále, v oddílu 3.3 popíšeme kognitivní systém, který překonává Turingovy stroje tím způsobem, že z hlediska pozorovatele realizuje nekonečně mnoho výpočetních kroků v konečném čase. V oddílu 3.4 se soustředím na tzv. evoluční interaktivní výpočty, které lze realizovat jak pomocí umělých systémů podobných internetu, tak i pomocí vyvíjejících se společenství

živých organismů. V závěru této části vyslovíme tzv. rozšířenou Churchovu–Turingovu tezi, která zachycuje nejen potenciál současného způsobu využívání výpočetních technologií, ale i zpracování informací v biologických systémech.

Kapitola má přehledový charakter s cílem upozornit zejména komunitu badatelů pracujících v problematice kognice a umělého života na vývoj v oblasti Churchovy–Turingovy teze, který má dopad na pochopení výpočetních možností kognitivních systémů. Z těchto důvodů se kapitola soustřeďuje především na méně formální popis a komentář odpovídajících výsledků; pokud by některého čtenáře hlouběji zajímaly formální definice a důkazy, nechť se obrátí na originální práce, citované v seznamu literatury.

## 3.2 Jsou živé organismy spíše konečnými automaty anebo Turingovými stroji?

Základním pojmem, se kterým budeme dále nějakou dobu pracovat, je pojem konfigurace nějakého konečného artefaktu, organismu, či pevně určeného objemu prostoru. Všem těmto kategoriím budeme říkat *zařízení*. Konfigurace nějakého zařízení je pojem relativní vzhledem k pozorovateli tohoto zařízení. Budeme totiž říkat, že vzhledem k pozorovateli je zařízení ve stejné konfiguraci ve dvou po sobě jdoucích časových okamžicích, pokud se zařízení v těchto dvou okamžicích jeví pozorovateli stejné (pozorovatel může být pro účel rozlišení konfigurací vybaven zvláštním přístrojem). To znamená, že počet konfigurací, které může zařízení nabýt, je daný rozlišovací schopností pozorovatele. Současná kvantová fyzika určuje teoretickou horní mez pro počet (kvantových) konfigurací, které lze rozlišit u zařízení dané hmotnosti a objemu. S. Lloyd ukázal (Karp, Lipton, 1980), že hmota velikosti 1 kg a objemu 1 litru se může nacházet v nejvýše řádově $10^{31}$ konfiguracích (kvantových stavech). To je obrovské, ale konečné číslo, které můžeme chápat i jako horní mez paměťové kapacity nejefektivnější myslitelné paměti o daných rozměrech. Takovou kapacitu ovšem dosahuje hmota za extrémních podmínek, snad srovnatelných s poměry panujícími v epicentru termonukleárního výbuchu anebo při tzv. velkém třesku. Nicméně, podle předchozích úvah každé konečné zařízení může sloužit jako potenciální paměťové zařízení s kapacitou danou pozorovatelnými měřitelnými fyzikálními parametry tohoto zařízení.

Aby nějaké zařízení „počítalo" podle daného přechodového pravidla (programu), které pro každou konfiguraci a daná vstupní data po každém výpočetním kroku určí následující konfiguraci či konfigurace, vybrané z konečné množiny možných konfigurací v dalším kroku a možná příslušná výstupní data, musí splňovat tři obecné předpoklady:

- musí existovat možnost vstupu a výstupu dat,
- musí být možné nastavit počáteční konfiguraci zařízení,
- musí existovat výpočetní dynamika zařízení, která je zachycena v přechodovém pravidlu, tj. zařízení má vlastnost, že „samo od sebe" způsobuje přechod z jedné konfigurace do druhé (či dalších, v případě paralelního výpočtu) v souladu se svým programem,
- výpočetní dynamika se mění v závislosti na vstupních datech.

Vlastnost, která zaručuje, že zařízení dělá něco „samo od sebe", neznamená nic jiného, než že zařízení má na to mechanismus, který pracuje požadovaným způsobem: zařízení je tak „uděláno". Přechodové pravidlo zařízení nemusí být známo – stačí, když existuje a je konečně popsatelné v nějakém formalismu. Příkladem zařízení, které počítá v uvedeném smyslu, je samozřejmě klasický, ale i kvantový počítač (přechodové pravidlo je známé), dále třeba mozek (pravidlo neznámé, ale komputacionalisté věří, že existuje – viz např. (Chalmers, 1994)) živý organismus, neuronová síť, konečný automat, automat na prodej lístků atd. Příkladem zařízení, které nepočítá v uvedeném smyslu, je např. kámen, obraz, anebo paměťová karta digitálního fotoaparátu. Všimněme si jedné důležité vlastnosti definovaného výpočetního zařízení – totiž skutečnosti, že jsme nestanovili, kdy výpočet skončí ani jeho výsledek. To jsme udělali úmyslně – naše zařízení totiž realizuje potenciálně nikdy nekončící výpočty. Jinak řečeno, zařízení transformuje každý potenciálně nekonečný proud vstupních dat (kterým v případě kognice říkáme stimuly) na potenciálně nekonečný proud výstupních dat (tomu odpovídají v případě živých organismů akce, resp. jejich posloupnost – chování). Přitom je možné a definice to nevylučuje, že některá vstupní data mohou být reakcí okolí na některá výstupní data, takže se jedná o interaktivní výpočty. Samozřejmě, právě popsané zařízení může realizovat i konečné výpočty – jednoduše tak, že uměle omezíme data vstupního proudu například tak, že od jisté pozice budou tvořena pouze posloupností „prázdných" symbolů a budeme se zajímat pouze o výpočty, které skončí.

Dále se budeme zabývat pouze klasickými (tj. diskrétními, ale ne-kvantovými) výpočetními zařízeními. Formálně je každé výpočetní zařízení takového typu ekvivalentní tzv. interaktivnímu konečnému automatu s výstupem, či tzv. interaktivnímu konečnému transduktoru (van Leeuwen, Wiedermann, 2001).

**Definice**. *Interaktivní konečný transduktor* (IKT) je automat, který se skládá z konečného řízení a vstupního a výstupního portu.

- Vstupní port slouží jako zařízení, pomocí kterého automat čte vstupní data, což jsou symboly tzv. vstupní (konečné) abecedy $\Sigma$. Na tomto portu se v každém kroku nachází nějaký vstupní symbol (může to být i tzv. prázdný symbol $\varepsilon \in \Sigma$, který znamená „žádný vstup"); jakmile automat vstupní symbol přečte, na portu se objeví v dalším kroku další symbol.
- Výstupní port je zařízení, kam automat posílá v každém kroku tzv. výstupní symboly – prvky konečné abecedy $\Gamma$.
- Konečné řízení je formálně definováno pomocí konečné množiny stavů $Q$ a pomocí tzv. přechodové relace $f$. Řízení se v každém okamžiku nachází v jednom z konečného počtu stavů. Přechodová relace je daná předpisem (tabulkou), který přiřazuje každému stavu $q \in Q$ a každému vstupnímu symbolu $a \in \Sigma$ konečnou množinu dvojic pozůstávajících z nového stavu $r \in Q$ a výstupního symbolu $b \in \Gamma$.

Každý IKT definuje relaci mezi posloupností stimulů a posloupnostmi chování. V deterministickém případě odpovídá každému vstupnímu proudu dat nejvýše jeden výstupní proud; v takovém případě je přechodová relace ekvivalentní s funkcí $f: Q \times \Sigma \rightarrow Q \times \Gamma$ a hovoříme o překladu proudu vstupních dat na proud výstupních dat.

IKT se tady od „běžného" konečného automatu s výstupem (tzv. Mealyho automatu) liší tím, že jeho vstupní data nejsou předem připravena na pásce a není jich konečný počet; proto i výstupem IKT nemusí být konečný řetězec symbolů.

Některá výpočetní zařízení mohou být obdařena zvláštní schopností (rozuměj: mechanismem) zvětšovat svou paměťovou kapacitu, např. tím způsobem, že za tím účelem využívají jiná zařízení, či se s nimi dokonce „spojují". Tato dodatečná paměťová kapacita jim umožňuje vytvářet a využívat potenciálně neomezené množiny konfigurací. Příkladem takového zařízení je *interaktivní Turingův stroj* (ITS) (van Leeuwen, Wiedermann, 2001), který vypadá jako standardní Turingův stroj, ale data čte a vypisuje podobně jako IKT přes vstupní a výstupní porty. ITS tedy můžeme chápat jako IKT, který pro zvýšení své konečné paměťové kapacity využívá kapacitu potenciálně nekonečné paměťové pásky. Tato páska sama o sobě „nedovede počítat", ale v symbióze s IKT, který je obdařen schopností pohybu podél pásky a schopností čtení a přepisování symbolů na pásce, vede k výpočetně silnějšímu zařízením než je samotný IKT: ITS „počítá" všechno, co počítá i IKT, ale počítá i něco navíc. Z hlediska konstrukce je ITS rozšířením standardního, klasického modelu pro případ nekonečných posloupností dat. Právě schopnost zpracovat tyto nekonečné posloupnosti dává ITS právě zmíněnou možnost počítat „něco navíc". ITS může například zpracovávat nekonečnou množinu konečných datových segmentů. Samozřejmě, že každý takový segment může být zpracován i standardním strojem, ale standardní stroj nemá možnost „přenést" nějakou informaci ze zpracování jednoho segmentu do druhého, protože po skončení výpočtu ho nelze „restartovat" ze stavu, ve kterém výpočet skončil (standardní model Turingova stroje musí zpracování nového vstupu začít z počátečního stavu a s prázdnou páskou). Takže na standardním stroji nemůžeme například realizovat následující překlad: pokud stroj akceptuje nějaký segment (tj. generuje jedničku), tak další segment v každém případě zamítne (generuje nulu). Více o možnostech interaktivních Turingových strojů viz v práci (van Leeuwen, Wiedermann, 2006).

Jiným zařízením, které má potenciál pro vytváření neomezeného množství konfigurací, ale nevyužívá „externí paměťové médium", je potenciálně neomezený systém (tj. tak velký sytém, jak potřebujeme) spolupracujících konečných automatů (příkladem jsou celulární automaty), který má paměťovou kapacitu tvořenou součtem paměťových kapacit jednotlivých konečných prvků systému. Ještě jiným zařízením je společenství ITS, které modeluje například existující internet (van Leeuwen, Wiedermann, 2001).

Věnujme ještě pár slov výběru vhodného modelu pro modelování živého organismu. Pokud nám skutečně jde o modelování za účelem napodobování, tak není chybou, je-li modelující zařízení v nějakém smyslu „silnější" než zařízení modelované. V takovém případě se nám pouze zjednoduší modelování některých vlastností. Například pro modelování činnosti bakterie můžeme využít Turingův stroj (tj. ve smyslu Churchovy–Turingovy teze jakékoliv výpočetní zařízení), ale samozřejmě nemůžeme využívat „na doraz" všechny jeho možnosti (zejména potenciál neomezené paměti). Na druhé straně, chceme-li výpočetní model použít pro odhad skutečné výpočetní síly modelovaného zařízení, výpočetně silnější zařízení zřejmě nelze pro tento účel použít. To znamená, že o zmíněné bakterii nelze tvrdit, že má stejnou výpočetní sílu jako Turingův stroj. To se zdá být samozřejmé, ale například v článcích o umělé inteligenci často můžeme číst, že lidský mozek v principu lze simulovat na Turingově stroji, a tudíž že oba mají stejnou

výpočetní sílu, což není pravda. Z těchto úvah vyplývá, že v případě, kdy nám jde právě o zachycení výpočetní síly kognitivních systémů, musíme tedy dávat pozor na to, aby modelující zařízení nemělo některé vlastnosti, které modelované zařízení nemá a ani v principu nemůže kombinací jiných vlastností nahradit. Pokud bychom se vrátili k případu mozek vs. Turingův stroj, tak mozek nemůže nijak kompenzovat libovolně velkou paměťovou kapacitu pásky Turingova stroje (pokud ovšem nepředpokládáme, že mozek může dle potřeby „narůst").

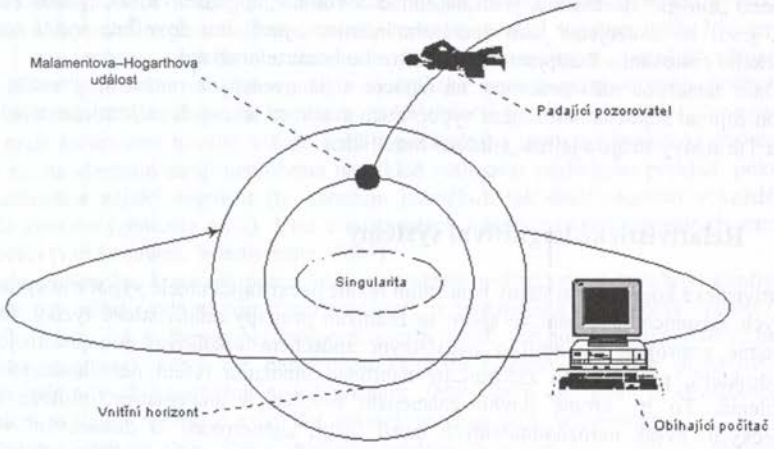
Teď již se můžeme zabývat otázkou z nadpisu tohoto oddílu. Podle předchozího výkladu, organismu samotnému, viděnému bez kontextu svého okolí, odpovídá v našem modelování IKT (protože modelujeme konečný počet konfigurací). Pokud organismu (např. člověku) dovolíme využívat jeho okolí k rozšíření kapacity jeho (konečné) paměti, jako výpočetní zařízení ho lze modelovat pomocí ITS (přechod na potenciálně neohraničený počet konfigurací). Pokud mu povolíme „spolčování" s jinými lidmi, dostaneme společenství IKT anebo společenství ITS, podle toho, jestli každému členu společenství nedovolíme nebo dovolíme využívat okolí jako potenciálně neohraničenou externí paměť. Hrubou analogií společenství IKT by byl například roj včel, společenství ITS pak lidská společnost vládnoucí písmem. Další variace kognitivního zařízení zvaného „člověk" dostaneme, jestli například dovolíme „upgrade" člověka pomocí evoluce, jestli ho uvažujeme jako operátora internetu, jestli mu dovolíme vyžití relativistického cestování v časoprostoru, obdaříme ho nesmrtelností atd.

Dále zaměříme naši pozornost na některé z již uvedených možností. Protože nás budou zajímat zejména horní meze výpočetních možností takových zařízení, soustředíme se na Turingovy stroje a jejich „silnější" modifikace.

## 3.3 Relativistické kognitivní systémy

Relativistické kognitivní systémy jsou zatím reálně neexistující umělé výpočetní systémy, kterých existence však není ve sporu se známými principy relativistické fyziky. Jejich případné sestrojení by zesílilo podstatným způsobem kognitivní schopnosti jejich konstruktérů, protože tato zařízení by například umožnila řešení nerozhodnutelných problémů. To by kromě jiného znamenalo revoluci v matematice, protože řada užitečných, avšak nerozhodnutelných teorií (např. „generování či dokazování všech matematických pravd", po čemž toužil Hilbert) by se stala rozhodnutelnými. Existence takových zařízení je přímým důsledkem faktu, že za jistých okolností je v principu fyzikálně možná realizace tzv. olbřímích úkolů („supertásků") v konečném čase; přitom *olbřímí úkol* je úkol, jehož realizace vyžaduje „za normálních okolností" (tj. použitím klasických prostředků) vykonat nekonečný počet operací (Earman, Norton, 1994). V případě nekonečných výpočtů je myšlenka realizace odpovídajícího olbřímího úkolu ve své podstatě jednoduchá, ale na první pohled vypadá naivně: stačí na Turingově stroji spustit daný výpočet, o kterém chceme zjistit, jestli skutečně neskončí, a nechat ho běžet. Protože na výsledek výpočtu bude nutné čekat nekonečně dlouho, k ukrácení dlouhé chvíle se odebereme do zvláštní „čekárny", kde se, pouze pro nás, čas nekonečně zpomalí. Turingův stroj je naprogramován tak, aby nám do čekárny vyslal signál v případě, že zastaví. Důležité je, aby se do čekárny takový signál dostal. Nedostaneme-li

takový signál, víme, že náš stroj nikdy neskončí. Takový scénář je na první poslech natolik nereálný, že bez dalšího vysvětlení riskujeme pověst vědců a aspirujeme na post pohádkářů. Nuže, zde je vysvětlení z dílny současné relativistické teoretické fyziky černých děr. Předchozí scénář je možné realizovat v tzv. *Malamentově–Hogarthově časoprostoru.* Charakteristickou vlastností takového časoprostoru je skutečnost (Norton, 2000), že připouští existenci dvou různých trajektorií, $T_1$ a $T_2$, vycházejících z jednoho bodu, přičemž putování podél $T_1$ trvá nekonečně dlouho, zatímco pro cestovatele podél $T_2$ uplyne pouze konečný čas. Takže, když se vrátíme k našemu potenciálně nekonečně dlouho počítajícímu Turingovu stroji, stačí, aby se během výpočtu pohyboval po trajektorii $T_1$ a zmíněná „čekárna" po trajektorii $T_2$. Existence Malamentova–Hogarthova časoprostoru vyplývá z teorie, ale zatím jeho existence nebyla empiricky ověřena. Kandidáty na takový časoprostor jsou rotující černé díry.[1] Trajektorie $T_1$ je oběžná dráha (za horizontem událostí) kolem takové díry, trajektorie $T_2$ je trajektorie padajícího pozorovatele do černé díry. Fyzika černé díry a jejího okolí umožňuje padajícímu pozorovateli přijímat signály vysílané obíhajícím počítačem. Schéma výpočtu je zobrazeno na obr. 3.1.




Obr. 3.1 Schéma relativistického výpočtu s využitím černé díry.

Z „praktického" pohledu má předchozí scénář několik vad: za prvé, není jasné, jestli je možné udržet nějaký počítač v chodu nekonečně dlouho. Vyžadovalo by to neustálý přívod energie, protože počítač by spotřebovával stále více paměti (pokud by výpočet neupadl do věčného cyklu – ale to se dá snadno odhalit) a tudíž by musel „růst". Za druhé, úloha pozorovatele v čekárně je nevděčná. Pokud se smíří s tím, že jeho civilizace mezitím zanikne, pořád zůstává možnost, že poslední pozorovatelova myšlenka před jeho smrtí, v okamžiku, kdy pozorovatel dosáhne vnitřní horizont černé díry a nastane tzv.

[1] „Nejslibnější" černou dírou je superhmotný kompaktní objekt v centru Mléčné dráhy; definitivní odpověď, týkající se existence rotujících černých děr, bude nalezena snad v příštích desetiletích (Karp a Lipton, 1980).

Malementova–Hogharthova událost, kdy pozorovatel dokončí „prohlídku" minulosti Vesmíru, bude platit obíhajícímu Turingovu stroji: „Žádný signál jsem nedostal, tudíž výpočet nekončí! Áááá..." Za třetí, není jasné, jestli je možné realizovat navazující výpočty typu „nekonečný výpočet v konečném čase". První možnost – neomezená kapacita paměti, neomezený přísun energie – je reálná v jisté třídě kosmologických modelů uvažovaných extravagantním matematickým fyzikem F. Tiplerem (tzv. teorie bodu omega) (Deutsch, 1997), (Tipler, 1995). Druhá možnost – destrukce pozorovatele na horizontu událostí rotující černé díry – je vyvrácena v práci (Etesi, Németi, 2002). Třetí možnost – realizace vícefázového nekonečného výpočtu – by snad byla možná „vypučením" nového vesmíru z kolabující černé díry anebo přechodem do něj přes tzv. červí díru (Greene, 1999). To znamená, že pro každý návazný nekonečný výpočet bychom potřebovali nový vesmír.

Opusťme nyní fascinující svět relativistické fyziky a věnujme se otázce výpočetní síly příslušných zařízení. V práci (Tipler, 1995) je studován model tzv. relativistického počítače a je ukázáno, že tento počítač může přijímat komplementy všech rekurzivně vyčíslitelných jazyků (což standardním Turingovým strojem není možné) a některé ještě složitější nerozhodnutelné jazyky. Jiná charakteristika výpočetní síly relativistických výpočtů je podána v práci (Wiedermann, van Leeuwen, 2002), kde je navržen předchozími úvahami inspirovaný model tzv. relativistického Turingova stroje (RTS), který je schopen realizovat libovolný konečný počet nekonečných relativistických výpočtů, každý z nich v konečném čase.

**Definice.** *Relativistický Turingův stroj* (RTS) je zvláštní druh deterministického mnohopáskového Turingova stroje s oddělenou vstupní páskou. Mezi jeho stavy se nachází zvláštní, tzv. relativistické stavy a také tzv. signální stavy. Signální stavy jsou dvojího druhu: pozitivní a negativní. Jakmile stroj dospěje do relativistického stavu, spustí tím relativistickou fázi výpočtu. Nastartuje se tím zvláštní výpočet, jehož cílem je dodat odpověď na dotaz „pokud bude pokračovat výpočet dále, dospěje v konečném čase do koncového stavu?" Jestliže ano, tak RTS vstoupí v dalším kroku do pozitivního signálního stavu; v opačném případě do negativního signálního stavu. V obou případech může výpočet pokračovat dále klasicky, počínaje v situaci, ve které byl při vstupu do relativistické fáze. Výpočet může obsahovat konečný počet střídání klasického a relativistického výpočtu.

Všimněme si, že relativistický výpočet po sobě nezanechá žádné stopy, tj. nemění obsah pásky. Jediné svědectví o tom, jestli skončil, vydávají právě signální stavy. Pokud postulujeme, že předchozí model odpovídá fyzikálnímu modelu popsanému v předešlém textu, tak lze dokázat překvapivý výsledek, dávající do souvislosti nekonečné výpočty s konečnými tzv. neuniformními výpočty. *Neuniformní výpočty* jsou takové výpočty, jejichž „program" závisí na daném vstupu. To znamená, že množina zařízení, realizujících nějaký neuniformní výpočet pro nekonečně mnoho vstupů, nemá konečný popis. Klasickým příkladem neuniformního výpočetního zařízení je *Turingův stroj s orákulem* (TSO), zavedený samotným Turingem (Turing, 1939). Je to Turingův stroj, který je vybaven další tzv. dotazovací páskou, na kterou stroj může napsat slovo (tj. konečnou posloupnost symbolů) $w$ a vstupem do tzv. dotazového stavu může zjistit, jestli slovo $w$ patří nebo nepatří do nějaké předem dané množiny $M$. Odpověď je indikována vstupem do stroje, do speciálního pozitivního nebo negativního signálního stavu. Příslušnost

prvku do množiny $M$ může pro klasický Turingův stroj představovat nerozhodnutelný problém, avšak stroj s orákulem $M$ jej „řeší" v jednom výpočetním kroku. V takovém případě je množina $M$ nekonečná a popis stroje s orákulem $M$ je tedy nekonečný, protože musí obsahovat popis množiny $M$.

Jiným příkladem neuniformního výpočetního zařízení je množina obvodů. Je zřejmé, že každý obvod může zpracovat pouze vstupy jisté velikosti, takže pro zpracování vstupů rostoucí velikosti potřebujeme i obvody rostoucí velikosti. Tyto obvody mohou být v principu sestrojeny „na míru" svým vstupům, tj. nemusí existovat algoritmus, který vygeneruje potřebný obvod pro vstup dané velikosti. Více o neuniformních výpočtech najdeme například v práci (Balcázar a kol., 1995).

Věnujme se opět problematice relativistických výpočtů. Pro ně platí, že nekonečný výpočet realizovaný RTS polynomické prostorové složitosti je ekvivalentní výpočtu tzv. neuniformního Turingova stroje, přičemž neuniformita roste polynomicky s rozměrem vstupu a je (relativisticky) vypočitatelná právě pomocí RTS polynomické složitosti (Wiedermann, van Leeuwen, 2002). Technicky řečeno, *neuniformní Turingův stroj* je Turingův stroj se speciálním orákulem, které umožní stroji během výpočtu získávat specializovanou, obecně nevypočitatelnou informaci „zvenčí". Na rozdíl od „běžných" TSO, o nichž je zmínka v předchozím výkladu a které mohou dostat speciální informaci pro každý vstup, neuniformní Turingův stroj může dostat pouze informaci, která je stejná pro všechny vstupy stejné délky; v tomto ohledu se podobá uvažovaným obvodům. Neuniformní Turingovy stroje jsou známé i pod názvem *Turingovy stroje s rádcem*. Rádce je blíže nespecifikované zařízení, které na požádání dodá stroji onu zmíněnou informaci – radu. Formálně se činnost tohoto mechanismu modeluje zavedením speciálního, tzv. dotazovacího stavu, podobně jako u strojů s orákulem; jakmile stroj do takového stavu vstoupí, na zvláštní pásce se mu „objeví" příslušná rada (Karp, Lipton, 1980). Lze rigorózně matematicky dokázat, že Turingův stroj s rádcem je silnější než klasický Turingův stroj, který rádce nepoužívá. Například v práci (van Leeuwen, Wiedermann, 2001) je navržen Turingův stroj používající rady lineární velikosti pro řešení klasicky nerozhodnutelného *problému zastavení*.

Idea sestrojení takového stroje M je následující. Potřebujeme „vymyslet" radu, pro každou velikost vstupu jinou, která umožní stroji M zjistit o nějakém jiném stroji T s popisem $\langle T \rangle$, jestli výpočet stroje T se vstupem $\langle T \rangle$ skončí anebo neskončí. Nechť $n$ je délka popisu $\langle T \rangle$ stroje T; to znamená, že i délka vstupu pro stroj T je $n$. Strojů s popisem délky $n$ je konečný počet. Vyberme mezi nimi ten stroj, který počítá nejdéle na vstupu, který je shodný s jeho popisem. Nazvěme takový stroj $S_n$ a jeho popis $\langle S_n \rangle$. Stroj M bude fungovat takto: na vstupu $\langle T \rangle$ bude simulovat stroj T a současně i stroj $S_n$, jehož popis $\langle S_n \rangle$ bude právě rada pro velikost vstupu $n$. Jestliže simulace stroje T skončí dříve než simulace stroje $S_n$, tak se zřejmě T se vstupem $\langle T \rangle$ zastaví. Jinak, pokud simulace $S_n$ skončí jako první, tak se stroj T zřejmě nikdy nezastaví vzhledem k tomu, že $S_n$ je stroj s maximální délkou výpočtu. Velikost rady je zřejmě stejná jako délka popisu stroje T, tj. rada je velikosti $n$.

Výsledek z práce (Wiedermann, van Leeuwen, 2002) říká, že stačí (v polynomickém prostoru) relativisticky vypočitatelná rada polynomické velikosti k tomu, abychom nahradili relativistické nekonečné výpočty generované v prostoru polynomické velikosti konečnými výpočty. Tato rada je konstruovaná podobně jako v případě předchozího

stroje M. Pokud někoho zajímá přesná charakterizace výpočetní síly RTM, tak další výsledek práce (Wiedermann, van Leeuwen, 2002) ukazuje, že tyto stroje rozeznávají přesně třídu $\Delta_2$ aritmetické hierarchie.

Bude někdy možné dosáhnout hranice kognice dané relativistickými úvahami? Dle již zmíněného Tiplera (Tipler, 1995) (a nesčetných autorů sci-fi) je možné očekávat nekonečně dlouhý vývoj inteligence, která se nakonec oprostí od své tělesné existence, bude nesmrtelná a bude existovat ve vesmíru či vesmírech, poslušná pouze zákonům fyziky. V takovém případě naše relativistické úvahy snad budou naplněny, ne-li dokonce překonány novými objevy.

Praktický význam relativistických výpočtu je zatím nulový. Příslušné úvahy však již v současné době mají velký význam například z hlediska teorie vyčíslitelnosti – kromě jiného ukazují, že při určování hranic vypočitatelnosti je třeba brát do úvahy fyzikální zákony, v rámci nichž je výpočet realizován. Dále ukazují, že Churchova–Turingova teze byla (či je) poplatná době svého vzniku a předchozí příklad relativistických výpočtů vyvrací její pravdivost. Pokud by se někomu zdálo, že předchozí „vyvrácení" teze pomocí relativistických výpočtů je přece jen poněkud spekulativní, v dalším výkladu si ukážeme jiný způsob jak dosáhnout podobného cíle současnými výpočetními prostředky v současném světě.

## 3.4 Interaktivní evoluční kognitivní systémy

Vraťme se nyní zpět k našim IKT zavedeným v oddílu 3.2. Jejich výpočetní síla není nikterak oslnivá – dovedou „pouze" transformovat potenciálně nekonečnou posloupnost stimulů na výstupní proud různých akcí, které ve svém úhrnu tvoří „chování" IKT. Všimněme si, že množina různých akcí jednoho IKT je konečná, protože existuje pouze konečný počet různých konfigurací IKT. Nicméně, v obecném případě dokáže IKT vygenerovat nekonečně (dokonce nespočetně) mnoho různých (nekonečných posloupností) chování v závislosti na vstupních datech, která se mohou vyskytovat v nespočetných vstupních proudech. „Návod" pro převod vstupní posloupnosti na výstupní je konečný, daný přechodovou funkcí IKT. Zdálo by se, že po tomto konstatování není co zkoumat.

Položme si však otázku, jestli dovedeme pomocí našeho aparátu zachytit evoluci – tj., zhruba řečeno, postupnou proměnu nějakého IKT, který počítá nějakou transformaci, na IKT, který počítá jinou transformaci, atd. Zachytit takto chápanou (časově neomezenou, tj. nekonečnou) evoluci lze pomocí nekonečné posloupnosti (navzájem různých) IKT a mechanismu, který stanoví okamžik „přesměrování" proudu vstupních dat z jednoho IKT na druhý IKT právě tehdy, když se daný IKT „evolučně" změní na jiný IKT. Navíc můžeme chtít, aby „nový" IKT pokračoval ve zpracování vstupních dat ve stejné konfiguraci, ve které jeho předchůdce ukončil svou činnost. To proto, abychom dovedli modelovat „předávání znalostí" z generace na generaci. To vše vede na pojem evoluční posloupnosti IKT s globálními stavy, která byla po prvé definovaná v práci (van Leeuwen, Wiedermann, 2001).

**Definice.** *Evoluční posloupnost interaktivních konečných transducerů s globálními stavy* je posloupnost IKT tvaru $I_1, I_2, I_3, \ldots$, přičemž $i$-tý transducer má množinu stavů $S_i$, z níž

se vyděluje speciální podmnožina tzv. globálních stavů $G_i$ a platí $G_i \subseteq G_{i+1}$. Předpokládejme, že v čase $t$ je proud vstupních dat nasměrován do trandsuceru $I_j$. Pokud poté transducer $I_j$ počítá ve stavech množiny $S_j - G_j$, tak proud dat stále směřuje do něj. Jakmile ale transducer vstoupí do jakéhokoliv globálního stavu $g \in G_j$, proud dat se přesměruje do dalšího automatu $I_{j+1}$, který začne počítat ve stavu $g \in G_{j+1}$. Pokud někdy později $I_{j+1}$ opět vstoupí do globálního stavu, proud dat se přepne na další automat atd.

Ptejme se dále, co „počítá" evoluční posloupnost IKT s globálními stavy. Z definice tohoto zařízení je zřejmé, že to opět bude nějaká transformace proudu vstupních dat na výstupní data. Nyní však je „návod" pro výpočet této transformace nekonečný, daný posloupnosti popisů jednotlivých členů posloupnosti! Samozřejmě, může se stát, že existuje konečný algoritmus, který pro daný index vygeneruje příslušný IKT, ale v obecném případě tomu tak nemusí být. Evoluční posloupnost interaktivních konečných transducerů s globálními stavy je tedy dalším případem neuniformního výpočetního zařízení, o kterých byla zmínka v předchozím textu. Pojem neuniformity tady zachycuje rozdíl mezi výpočty jediného IKT, který je popsán konečným předpisem, a výpočty realizovanými posloupností IKS, které jsou popsány nekonečným předpisem. Tato charakteristika nám zatím mnoho neříká, protože v konfrontaci konečna s nekonečnem intuice selhává. Lepší by samozřejmě bylo srovnání s Turingovým strojem, kde již máme představu, co takový stroj dovede spočítat.

V práci (van Leeuwen, Wiedermann, 2001) autoři dokázali, že výpočetní síla evoluční posloupnosti IKT s globálními stavy je ekvivalentní výpočetní síle interaktivního Turingova stroji s rádcem. ITS „bez rádce" byl popsán v oddílu 3.2; jeho rozšíření o rádce ve stejném duchu, jako jsme to dělali v předchozím textu, je přímočará záležitost. Přitom se velikost rady měří vzhledem k délce již přečteného prefixu potenciálně nekonečného proudu dat, zpracovávaného ITS, v okamžiku vyžádání rady. Poměrně jednoduše lze ukázat, že ITS s rádcem je výpočetně silnější než stroj bez rádce, viz např. (van Leeuwen, Wiedermann, 2001a), (van Leeuwen, Wiedermann, 2001b). Použije se přitom překlad, ve kterém je zakódován problém zastavení. Vstupní proud dat se skládá z nekonečné posloupnosti segmentů a každý segment obsahuje popis nějakého Turingova stroje. Výstupní proud obsahuje na pozici odpovídající danému popisu stroje jedničku (nulu), pokud stroj odpovídající popisu zastaví (nezastaví) na vstupu, který je roven popisu daného stroje. Toto je pro ITS bez rádce nerozhodnutelný problém, ale pokud stroj vybavíme pro každý popis radou, kterou jsme zkonstruovali pro stroj M v předchozí části, tak tento problém bude řešitelný.

Ekvivalence evoluční posloupnosti IKT s globálními stavy s interaktivním Turingovým strojem s rádcem se dokáže takto. ITS dostane ve své radě popis IKT, který právě zpracovává vstup, takže stroj může tento transducer simulovat. Opačně, popis ITS s rádcem je v každém okamžiku konečný, a proto mu odpovídá nějaký IKT.

V rámci původní motivace pro zavedení pojmu evoluční posloupnosti IKT s globálním stavy můžeme předchozí výsledky interpretovat i tak, že výpočetní síla evoluce (přesněji: evolučních interaktivních systémů) je vetší než výpočetní síla „klasických" Turingových strojů, uvažovaných sice v interaktivním kontextu, ale bez možnosti evoluce. Tato skutečnost má řadu zajímavých dopadů v teorii umělého života, které byly zkoumány v práci (Wiedermann, van Leeuwen, 2002). Například se ukazuje, že pomocí modelu ITS s rádcem (anebo evoluční posloupnosti IKT s globálními stavy) lze

modelovat třeba výpočetní sílu rozvíjejících se společenství IKT, tj. kognitivních jedinců, komunikujících prostřednictvím všech známých komunikačních prostředků, jakými je hovořená řeč, dopisy, e-maily, mobilní telefony apod. Také internet, včetně jeho prostřednictvím komunikujících jedinců, spadá do kategorie zařízení, která jsou ekvivalentní, z hlediska jejich výpočetní síly, interaktivnímu Turingovu stroji s rádcem. Protože víme, že Turingův stroj s rádcem je silnější než klasický stroj bez rádce, můžeme předchozí výsledky interpretovat v kognitivní teorii jako důkaz, že kolektivní inteligence vysoce, kvalitativně překračuje inteligenci osamoceného jedince. Mimochodem, zde může být skryto vysvětlení problému, který tak trápil Penrose (Penrose,1994): jak je možné vysvětlit, že matematikové jsou schopni dokazovat pravdivost některých teorémů přes to, že víme (dle Gödelovy věty či Turingova výsledku (Turing, 1937)), že obecně neexistuje algoritmus, který by zaručeně vedl k cíli, pro důkaz či vyvrácení jakéhokoliv teorému. Vysvětlení je, že matematikové najdou „neuniformní" důkaz, tj. postup, který se hodí pro důkaz dané teorémy a pravděpodobně už pro nic jiného. Otázkou je, jak matematikové takový důkaz najdou, tj. jak získají nevypočitatelnou informaci, která „otevře cestu" k řešení daného problému. Odpovědí je, že komunita matematiků představuje interaktivní evoluční systém, který si v nepredikovatelné interakci a evoluci (myšlenek) generuje pro své vlastní účely klasickým způsobem nevypočitatelné informace, jež jsou podrobeny „evolučnímu tlaku" v konkurenci s jinými myšlenkami. Ty myšlenky, které přežijí, matematikové použijí jako „tvořivý" doplněk již známých tvrzení a vyzkoušených postupů. Ve vybraných případech mohou vést k řešení nových problémů. Ne nadarmo existují odborné časopisy, prostřednictvím kterých matematikové (ale i jiní vědci) interagují.

Celkově se ukazuje a formálně lze pomocí modelů příslušných zařízení ilustrovat, že interaktivní Turingův stroj s rádcem představuje nové výpočetní paradigma, které modelově vystihuje podstatu současného využívání výpočetní techniky. Toto rozšířené paradigma lze po vzoru (van Leeuwen, Wiedermann, 2001) formulovat tímto způsobem.

**Rozšířená Churchova–Turingova teze.** Výpočetní síla jakékoliv třídy interaktivních výpočetních zařízení řízených neuniformními algoritmy nepřesáhne výpočetní sílu interaktivního Turingova stroje s rádcem.

Pro úplnost si zopakujme výpočetní scénář klasického Turingova stroje, tak jak ho implicitně předpokládá klasická Churchova–Turingova teze: předem daná vstupní data konečného rozsahu, výpočet, do kterého nelze v jeho průběhu zasahovat (tj. nelze měnit program během výpočtu, ani měnit či přidávat vstupní data), výsledek definovaný pouze v případě, kdy výpočet skončí. A nyní to porovnejme se scénářem výpočtu dle rozšířené Churchovy–Turingovy teze: data do systému vstupují interaktivně, vystupují průběžně, výpočet může být potenciálně nekonečný, program lze měnit během výpočtu. Intuitivně, interaktivita představuje tu vlastnost výpočtů, která jim dává potenciál nikdy nekončících procesů. Nevypočitatelný či neuniformní prvek, modelovaný pomocí rádce, v „praxi" představuje například u internetu jeho nepredikovatelný vývoj, způsobený jeho lokální hardwarovou či softwarovou modifikací jeho uživateli, jeho nepredikovatelný růst přidáváním nových pracovních stanic, a v neposlední řadě nepredikovatelnou charakteristikou doručovacích časů zpráv vyměňovaných prostřednictvím internetu. Podobně lze nahlížet i na vyvíjející se společenství komunikujících kognitivních zařízení. Všechna tato zařízení, která jsou z hlediska jejich výpočetní síly ekvivalentní interaktivnímu

Turingovu stroji s rádcem, resp. posloupnostem IKT s globálními stavy, nazýváme *evolučními interaktivními výpočetními systémy.*

Je dobré si uvědomit, že předchozí formulace teze je skutečně rozšířením klasické teze. Je tomu tak proto, že klasické pojetí algoritmů jako neinteraktivních a uniformních algoritmů je vlastně speciálním případem evolučních interaktivních algoritmů, podobně jako je klasický Turingův stroj speciálním případem interaktivního Turingova stroje s rádcem. Navíc, rozšířená teze pokrývá i případ relativistických výpočtů. Důvod je ten, že výpočty RTS, definovaného v oddílu 3.3, jsou ekvivalentní výpočtům (neinteraktivního) Turingova stroje s rádcem (viz také konec 3.3), což je zřejmě speciální případ interaktivního Turingova stroje s rádcem.

Poslední zajímavá otázka, na kterou je dobré dát jasnou odpověď, je tato: pokud tedy tvrdíme, že evoluční interaktivní výpočetní zařízení mají větší výpočetní sílu než klasický interaktivní Turingův stroj, znamená to, že mohou řešit nerozhodnutelné problémy? Odpověď zní, ano, ale pouze za určitých předpokladů, které v praxi nelze splnit: nevypočitatelná informace, která je nutná pro řešení nerozhodnutelných problémů, by se musela do evolučních systémů dostávat z jejich okolí prostřednictvím interakce či prostřednictvím evoluce. Takže praktická implikace odpovídajících výsledků o výpočetní síle těchto systémů je spíše opačná: tyto výsledky dokazují, že evoluční interaktivní výpočetní systémy nelze modelovat klasickými prostředky. Jsou sice výpočetně silnější než klasické systémy, avšak tuto sílu nelze využít pro řešení zajímavých, námi určených nerozhodnutelných problémů.

## 3.5 Závěr

V předchozích dvou oddílech jsme naznačili dva způsoby, jak se dostat za hranici vypočitatelnosti dané sílou standardního Turingova stroje: buď pomocí relativistické fyziky, anebo změnou výpočetní scénáře na scénář evolučních interaktivních výpočtů. Oba dva způsoby mění pravidla hry, implicitně uvažované Turingem a obsažené v klasické Churchově–Turingově tezi. V prvním případě využíváme na doraz relativistické vlastnosti časoprostoru, které v běžném životě nehrají roli. Ve druhém případě vlastně uvažujeme nekonečně velké výpočetní zařízení, resp. přesněji, výpočetní zařízení s potenciálně nekonečně velkým programem. Vtip je ale v tom, že v každém okamžiku je toto zařízení konečné, ale jeho budoucí evoluci nelze „vypočítat". V prvním případě dostaneme relativistické výpočetní zařízení, které nám umožní řešit konkrétní nerozhodnutelné problémy. Ve druhém případě, evoluční výpočetní systémy řešení konkrétních, námi definovaných nerozhodnutelných problémů neumožňují, protože okolí s nimi správným způsobem „nespolupracuje", nedodává jim potřebné rady. Jinými slovy to znamená, že relativistické výpočetní systémy představují konstruktivní řešení některých doposud neřešitelných problémů, kdežto interaktivní evoluční představují nekonstruktivní řešení: víme, že interaktivní evoluční systémy řešící některé nerozhodnutelné problémy existují, ale je nedovedeme sestrojit. Na druhé straně, posledně zmíněné systémy dovedou řešit jeden problém, o kterém (zatím) nedovedeme dokázat, že je nerozhodnutelný, protože ho nedovedeme formalizovat: je to problém vzniku a evoluce života. Zde naopak je prostředí velmi podstatné, usměrňuje vývoj výpočetního systému

tak, aby zaplnil danou „niku". Naše výsledky naznačují, že bez interakce s prostředím nelze vývoj živých organismů (a tedy ani kognici v plném rozsahu) simulovat pomocí běžných počítačů. Takže závěr je docela zajímavý. Na jedné straně, v případě relativistických výpočtů dovedeme, alespoň v principu, řešit formální nerozhodnutelné problémy dle naší libovůle, ale nedovedeme řešit problém vzniku a vývoje života (v pozemském smyslu), protože (kromě jiného) nedovedeme nasimulovat věrně interakci s okolím. Na straně druhé, tento problém řeší evoluční interaktivní výpočty, které se svému okolí přizpůsobují „automaticky" (pokud jsou správně naprogramovány). V reálném světě se „správné naprogramování" děje prostřednictvím přirozeného výběru darwinovským způsobem. V teorii se tento princip využívá v oblasti genetických algoritmů. Nicméně na základě předchozího výkladu lze konstatovat, že problém mezí kognitivních výpočtů je dán možnostmi výpočetních systémů v hranicích platných fyzikálních teorií a v rámci uvažovaného výpočetního scénáře. Jinými slovy, co lze a co nelze vypočítat není výlučně matematický problém, jak se domnívali matematikové v průběhu 20. století, ale je to i, a především, problém fyzikální (Deutsch, 1997). Co zůstává, je problém sestrojení konkrétních umělých kognitivních systémů, které by využívaly principy evolučních interaktivních algoritmů. Možné kroky v tomto směru, využívající principy evoluční sebereprodukce a sebesestavování, jsou naznačeny v práci (Wiedermann, 2005).

## Literatura


Balcázar J. L., Diáz J., and Gábarró J.: *Structural complexity, Vol. I*, 2nd Edition, Springer-Verlag, Berlín 1995.

Bernstain E., Vazirani U.: *Quantum Complexity Theory*. Proc. of the 25-th Annual Symposium on the Theory of Computing, ACM, New York 1993, s. 11–20.

Deutsch D.: *Quantum Theory, the Church-Turing principle and the universal quantum computer*. Proc. of the Royal Society of London A 400, 1985 s. 97–117.

Deutsch D.: *Fabrics of Reality*. Penguin Books, 1997, s. 390.

Earman J., Norton J.: *Infinite Pains: The Trouble with Supertasks*. In P. Benacerraf: The Philosopher and His Critics, S. Stick ed., Blackwell, New York 1994.

Etesi G., Németi I.: *Non-Turing computations via Malament–Hogarth space-times*, Int. J. Theor. Phys. 41, 2002, s. 341–370, viz také http://xxx.lanl.gov/ps/gr-qc/0104023

Greene B.: *Elegantní vesmír*. Mladá fronta, 1999, s. 398

Hogarth J.: *Non-Turing computers and non-Turing computability*, Proceedings of the 1994 Bienial Meeting of the Philosophy of Science Association PSA 1994, Vol. One, s. 125–138, 1994.

Hopcroft J., Motwani R., and Ullman J. D.: *Introduction to automata theory, languages and computation*, 2nd Edition, Addison-Wesley, Reading, MA, 2000.

Chalmers D. J.: *A Computational Foundations for the Study of Cognition*. Minds and machines, Vol. 4, No. 4, 1994.

Karp R. M., Lipton R. J.: *Some connections between non-uniform and uniform complexity classes*, in Proc. 12th Annual ACM Symposium on the Theory of Computing (STOC'80), 1980, s. 302–309.

Kramosil I.: *Pravděpodobnostní algoritmy*. In: Umělá inteligence (4), V. Mařík, O. Štěpánková, J. Lažanský a kol. (eds), Academia, 2003, 475 s.

Leeuwen J. van, Wiedermann J.: *Theory of Interaction*. Kapitola v knize: Goldin D., Smolka S. and Wegner P. (eds): Interactive Computing: A New Paradigm. Springer, 2006, 487 s.

Leeuwen J. van, Wiedermann J.: *The Turing machine paradigm in contemporary computing*, B. Enquist and W. Schmidt (eds), Mathematics Unlimited – 2001 and Beyond, Springer-Verlag, 2001, s. 1139–1155

Leeuwen J. van, Wiedermann J.: *Beyond the Turing limit: evolving interactive systems*. Pacholski L., Ruzicka P. (eds.), SOFSEM'01: Theory and Practice of Informatics, 28th Conference on Current Trends in Theory and Practice of Informatics, Lecture Notes in Computer Science Vol. 2234, Springer-Verlag, Berlín 2001, s. 90–109.

Lloyd S.: *How Fast, How Small, and How Powerful*: Moor's Law and the Ultimate Laptop. In: Rebooting Civilization, http://www.edge.org/3rd_culture/rebooting/ rebooting.html , 2001.

Melia F.: *The heart of the Milky Way*, American Scientist, s. 346–353, July–August 2000.

Norton J. D.: *What Can We Learn about Ontology of Space and Time from the Theory of Relativity?* http://philsci-archive.pitt.edu/documents/disk0/00/00/02/25/index.html, Sept. 2000.

Penrose R.: *Shadows of the Mind (A Search for the Missing Science of Consciousness)*. Oxford University Press, Oxford 1994, 457 s.

Shettleworth S.: *Cognition, Evolution, and Behavior*, Oxford University Press, 1998.

Tipler F.: *The Physics of Immortality*, Doubleday, 1995.

Turing A. M.: *On computable numbers, with an application to the Entscheidungs-problem*, Proc. London Math. Soc., 42-2 (1936) 230–265; A correction, ibid., 43 (1937) 544–546.

Turing A. M.: *Systems of logic based on ordinals*, Proc. London Math. Soc. Series 2, 45 (1939), s. 161–228.

Wiedermann J., Leeuwen J. van: *The Emergent Computational Potential of Evolving Artificial Living Systems*. Ai Communications, IOS Press, Vol. 15, No. 4, 2002, s. 205–216.

Wiedermann J., Leeuwen J. van: *Relativistic Computers and Non-Uniform Complexity Theory Unconvential Models of Computation* (UMC'2002), LNCS Vol. 2509, Springer, Berlín 2002, s. 287–299.

Wiedermann J.: *Globular Universe and Autopoietic Automata: A Framework for Artificial Life*. Capcarrère M. S., Freitas A. A., Bentley P. J., Johnson C. G., Timmis J. (Eds.): Advances in Artificial Life, 8th European Conference, ECAL 2005, Canterbury, UK, September 5–9, 2005, Proceedings. Lecture Notes in Computer Science Vol. 3630 Springer, 2005, s. 21–30.

# 4 Umělé myšlení a kauzální paradox emergentních systémů

## 4.1 Úvod

Diskuse o možnostech a mezích vytvoření umělého vědomí je zcela samozřejmě předcházena nejen úvahami o podstatě vědomí přirozeného, ale také o vymezení a uchopení toho, co činí přirozené přirozeným a umělé umělým. Kognitivistické (resp. logicko-symbolické) paradigma klasické umělé inteligence zatím nedokázalo vysvětlit podstatu vědomí a vědomého prožívání, byť koncepce mysli jako výpočetních procedur operujících nad souborem mentálních reprezentací dokáže poměrně dobře popsat některé dílčí kognitivní funkce (Thagard, 2001). Teoretické úvahy nad původem a podstatou vědomí se tak stále častěji dostávají do emergentistického rámce uvažujícího vzájemné působení a ovlivňování velkého množství prvků, z jejichž interakce vyvstává jev, který lze jako vědomí chápat.

V této kapitole se nebudeme primárně zabývat otázkami vědomí jako takového – tj. samotným jeho ontologickým zakotvením či možnostmi jeho poznání a vědeckého zkoumání. Záměrně opomíjíme též důležitost rozlišování „vnitřního" (v první osobě) chápání vědomého mentálního života a „vnějšího" (ve třetí osobě), jakož i jeho fenomenální a performační složky.[1] Situaci si zjednodušíme tím, že dále budeme hovořit spíše jen o myšlení (umělém či přirozeném) nebo mentálních stavech a procesech, ať již jejich chápání bude zčásti intuitivní či nikoli. Naším hlavním cílem bude snaha alternativním způsobem ukázat některé zajímavé důsledky, jež se týkají emergence jevů, a tudíž i úvah o přirozeném myšlení či vytváření myšlení umělého. Výklad se přitom hojně opírá o první kapitolu třetího dílu knihy Umělá inteligence (Havel, 2001).

## 4.2 Emergence a myšlení

Termín *emergence* se v našem pojetí týká kolektivních systémů, jež se na jisté úrovni skládají z velkého množství vzájemně interagujících prvků. Na úrovni „vyšší"[2] pak toto

[1] To, co je na daném mentálním procesu objektivně, „zvnějšku" popsatelné, je jeho performační komponenta; to, co přitom subjekt „vnitřně" prožívá, je fenomenální komponenta. (Havel, 2001)

[2] Uvozovky jsou použity, aby podtrhly metaforický význam rozlišování „vyšších" a „nižších" úrovní, neboť – jak bude ukázáno později – se nechceme omezovat pouze na lineárně uspořádanou hierarchii, která by odpovídala stupnici měřítek různých fyzikálních veličin.

bohaté strukturní uspořádání dává vzniknout novým, kvalitativně odlišným jevům, jež nelze redukovat na pouhé chování prvků podkladových či je z něj odvodit.

Význam pojmu emergence definuje I. M. Havel (Havel, 1993), (Havel, 2001) několika způsoby. Jedním z nich je vysvětlení jevů v ontologickém smyslu, kdy se jevy přirozeně vyskytují na určité úrovni, avšak něco brání jejich redukci na odpovídající nižší úroveň. Druhým je vysvětlení jevů v gnoseologickém smyslu – jistá vlastnost specifikovaná teorií $T_1$ je emergentní, jestliže má skutečné výskyty, které jsou doprovázeny výskyty některé jiné vlastnosti definované redukující teorií $T_2$, avšak nelze ji na žádnou vlastnost teorie $T_2$ redukovat.

Ve vztahu emergence k přirozenému myšlení můžeme přijmout tzv. *emergentistickou tezi* (Havel, 2001).

*Mentální stavy a procesy lze pojmout jako emergentní jevy na některé vyšší úrovni dostatečně složitého dynamického systému.*

Záměrně jsme zdůraznili, že jde o mentální stavy a procesy myšlení přirozeného. Myšlení umělé (ve smyslu kritérií „umělosti", tj. zejména existence záměru člověka vytvořit daný artefakt a zároveň realizace intencionálního procesu vedoucího od záměru k vytvoření) může být nejspíše v rámci emergentistické teze také zahrnuto, avšak zde již musíme být mnohem opatrnější: zřejmé jsou epistemologické překážky bránící poznání, zda to, co v umělém systému probíhá, je opravdu myšlení (čili nesdělitelnost fenomenální složky mysli); zároveň je nutné si uvědomit, že lidské myšlení (resp. vědomí) je situováno do specifického tělesného, fyzikálního a socio-kulturního prostředí – a právě takto je jako myšlení identifikujeme.

Co bychom měli nejprve hledat u umělého systému (např. počítače), je přítomnost emergentních jevů – tzn. jevů ve smyslu ontologické či gnoseologické definice emergence, lhostejno však, dají-li se považovat za projevy umělého myšlení jako duplikátu myšlení přirozeného. Takovou přítomnost emergentních jevů nazvěme *emergentní efekt* systému. Vztah mezi emergentním efektem a přirozeným myšlením (alespoň po stránce performační) může dále zkoumat kognitivní věda – o tom více (Pstružina, 1998).

Emergentním efektem systému umělé inteligence tak může být určitá jeho žádaná (či naopak spontánní) vlastnost, jako například různé predikční, rozhodovací a optimalizační schopnosti, či modely některých kognitivních funkcí (obrazové a zvukové rozpoznávání, apod.). Návrh systémů v rámci tzv. „Nové umělé inteligence" (Havel, 2001), (Pstružina, 1998), (Brooks, 1999) může pak být vnímán jako snaha o vytvoření takového umělého „média", na jehož základě díky emergentnímu efektu vyvstanou požadované vlastnosti či schopnosti systému jako celku.

Naskýtají se však důležité otázky: jaká je podstata emergentního efektu? Proč nelze emergentní chování redukovat na chování konstituujících prvků? Proč při dosažení určitého množství prvků a jejich vzájemných vztahů dojde k emergenci? Jaké musí být toto množství? Co dalšího musí emergentní systém splňovat? Co vůbec emergenci *způsobuje*?

Hovoříme-li o emergentních systémech, můžeme tak v zásadě (a velice zjednodušeně) činit ze dvou úhlů pohledu: z *analytického* a z *fenomenologického*. Analytický pohled považuje reálné emergentní systémy za realizace složitých abstraktních nelineárních systémů (tj. abstraktních systémů popsaných netriviálními nelineárními diferenciálními či diferenčními rovnicemi), čili emergence je brána toliko za epifenomén, zdání v člověku vyvolané na základě neznalosti či neschopnosti zjistit nebo zajistit řešení příslušných rovnic a z toho plynoucí nemožnosti činit o systému a jeho chování jasné a ostré před-

povědi a závěry. Emergentní jevy vznikají díky neúplnému popisu jisté části reality – jde tak v zásadě pouze o „jazykový problém".

Jinými slovy: emergence v tomto pojetí je způsobena neznalostí pozorovatele či jeho technickou neschopností nalézt analytické řešení složitých rovnic popisujících chování systému (či nalézt přesný tvar těchto rovnic). Bude-li se pozorovatel dostatečně dlouho snažit, podaří se mu rozšířit obzor svého poznání, čímž rozptýlí zdání daného emergentního efektu. Jediné, co nadále bude způsobovat nepredikovatelnost a „chaotičnost" chování systému, bude šum znemožňující přesné stanovení počátečních podmínek, což díky bifurkacím nelineárních rovnic povede k velkým rozdílům v možných průbězích stavové trajektorie. Ovšem přítomnost tohoto šumu může sama být vysvětlena dvěma způsoby: šum *vyvstává* (neboli emerguje) ze složitosti vnějšího prostředí, do něhož je systém umístěn; popř. šum je způsoben metafyzickým principem čisté náhody. První vysvětlení zakládá počátek nekonečného regresu, který může být přerušen pouze redukováním vysvětlovaného jevu či přijetím druhého řešení – principu čisté náhody. Reduktivní vysvětlení emergentního efektu však rozhodně nepřispěje k pochopení jeho podstaty a přijetím čisté náhody získáme maximálně určitý explanační princip, nikoli vyvratitelnou hypotézu.

Naproti tomu fenomenologický přístup zastává názor, že emergentní efekt je důsledkem *aktů vědomí*. Na půdě vědomí se setkáváme s jevy vnějšího světa a emergence může být jedním z prostředků vědomí, který při uchopování jevů dovoluje zachovat jejich soudržnost: například tvar předmětu emerguje z jeho vnitřní struktury a vědomí při daných rozlišovacích schopnostech uchopuje tento tvar, nikoli samotnou vnitřní strukturu. Díky bezprostřední danosti vědomých mentálních stavů (Havel, 2001) se s tvarem předmětu setkáváme jako s něčím, co *je*, nikoli jako s něčím *zdánlivým* (byť by to de facto zdánlivé i bylo – jako například fata morgana), stejně jako se na půdě vědomí setkáváme s vlastní bolestí (či jinými tělesnými pocity), o níž dozajista nikdy nelze mluvit jako o zdánlivé (tvrzení „zdánlivě mě bolí hlava" je samozřejmě nesmyslné).

Chceme-li se tedy dopátrat vnitřní struktury předmětu, s jehož tvarem se setkáváme, zvolíme nejspíše analytický přístup. Avšak k vysvětlení, proč a jak se setkáváme právě s emergujícím tvarem a nikoli jednoduše rovnou s vnitřní strukturou, nám reduktivní pojetí emergence ve smyslu analytickém nepomůže. Podobně ačkoli *víme*, že červená barva je projevem elektromagnetického pole určitých parametrů, vždy se na půdě vědomí setkáváme s prožitkem červenosti, nikoli s prožitkem elektromagnetického pole a k nereduktivnímu kvalitativnímu popsání tohoto prožitku nám teorie elektromagnetického pole nepomůže.

Fenomenologický přístup je z hlediska exaktně vědeckého poznání taktéž problematický, neboť naráží na překážky subjektivismu vědomého prožívání. Na druhou stranu alespoň nepopírá existenci jevu, jehož podstatu nyní chceme objasnit (tj. jevu emergence). V této kapitole se tedy pokusíme k podstatě emergentního efektu přistoupit způsobem, který se snaží do jisté míry propojit obě zmíněná pojetí, a to pomocí fenomenologicky motivovaného matematického aparátu, který do svého rámce včleňuje i pozorující vědomý subjekt.

## 4.3 Kauzální domény

Kauzální výklad fungování světa je ve vědeckém diskursu velmi důležitým prvkem. *Kauzální vztahy* – tj. vztahy umožňující výklad pozorovaného jevu jako způsobeného jevem jiným – jsou implicitně přítomny i v teleologickém výkladu (tj. nastal-li daný jev proto, *aby* bylo dosaženo jevu jiného, znamená to, že první jev má kauzální potenci, v níž je obsažen i jev druhý).

Jak ukazuje K. R. Popper (Popper, 1997), kauzalita je do daného fragmentu světa vnesena až (vědeckou) teorií (čili jazykem), která tento fragment popisuje (či vysvětluje) – čili není v něm přítomna a priori. Vědecká teorie je z formálního hlediska systém znaků nebo symbolů a pravidel manipulace s nimi. Je-li pak teorie nahlížena v empirickém kontextu, jde vlastně o univerzální tvrzení. Teprve prostřednictvím teorií se nám objekty vnějšího světa a vztahy mezi nimi zjevují.[3]

Kauzální vztahy (tj. vztahy příčinné souvislosti) spojují takové jevy (či události), u nichž je možné podat *kauzální vysvětlení*. „Podat kauzální vysvětlení nějaké události znamená dedukovat tvrzení, které ji popisuje, z jednoho či více *univerzálních zákonů* spolu s jistými singulárními tvrzeními, *počátečními podmínkami*" (Popper, 1997). Ilustrace tohoto pojetí kauzality je zřejmá z jednoduchého Popperova příkladu: můžeme říci, že zavěšení dvoukilogramového závaží na vlákno o nosnosti 1 kg *způsobilo* přetržení vlákna, neboť naše teorie disponuje univerzálním tvrzením (resp. hypotézou), že je-li na vlákno o dané nosnosti zavěšeno závaží o větší hmotnosti, dojde k přetržení vlákna, a máme též dvě singulární tvrzení: „nosnost našeho vlákna je 1 kg" a „hmotnost našeho závaží je 2 kg". Mezi jevy zavěšení dvoukilogramového závaží a přetržení vlákna tudíž existuje kauzální vztah.

Hovoříme-li o světě, můžeme to v obecnějším pojetí činit tak, že je svět fragmentován do množství tzv. *domén diskursu*. Budeme-li se na tyto domény dívat prostřednictvím kauzálních vztahů, jež do nich při našem porozumění vneseme a které nás také budou nejvíce zajímat, můžeme domény diskursu omezit na *kauzální domény*, jež I. M. Havel zavádí takto (Havel, 2000), (Havel, 2001):

*Kauzální doména je jakákoli oblast (výsek, fragment, komponenta) skutečnosti, v jejímž rámci se nám kauzální vztahy jeví jako zjevné, srozumitelné a vzájemně koherentní – přinejmenším jsou zjevnější, srozumitelnější a vzájemně koherentnější než vztahy mezi různými doménami.*

Hranice kauzální domény jsou často neostré a vágně vymezené, tudíž o dvou oblastech skutečnosti můžeme říci, že jsou různými kauzálními doménami až tehdy, jsou-li od sebe v nějakém smyslu dostatečně daleko. Zatímco interakce mezi jevy v rámci jedné domény jsou zřetelné a kauzální, vztahy mezi jevy napříč doménami se tomuto přímému popisu většinou vymykají a obvykle o nich předpokládáme, že mohou být též emergentní povahy. Při popisu světa tak můžeme myšlenkově přecházet z jedné kauzální domény do jiné, avšak jedním pohledem velice obtížně dokážeme pojmout více domén současně. Je nutné se též tázat, zda vztahy mezi doménami vůbec mohou být kauzální povahy, a pokud ano, čím se tyto vztahy liší od vztahů uvnitř domén.

---

[3] „Teorie jsou sítě, házené, aby chytly to, co nazýváme ‚svět'." (Popper, 1997)

Zavedením kauzálních domén dostáváme možnost vyhnout se představě, že k dosažení emergentního efektu je zapotřebí hierarchický řetězec úrovní, u nichž lze vždy říci, která z nich leží „výše" a která „níže". Toto platí obzvláště v případě problému mentálních procesů: lze vůbec tvrdit, že úroveň mentální leží nad úrovní neuronovou? A kde se například nachází úroveň evoluční či sociální? Nad nebo pod těmito dvěma?

Na základě úvah týkajících se kauzálních domén formuluje I. M. Havel *zobecněnou emergentistickou tezi* (Havel, 2001):

*Mentální stavy a procesy lze pojmout jako emergentní jevy nad rozsáhlou množinou vzájemně vázaných kauzálních domén.*

Tuto tezi lze ilustrovat mnoha příklady, z nichž některé I. M. Havel v jednom ze svých textů (Havel, 2000) analyzuje. Holistická provázanost velkého množství kauzálních domén je tam patrná zejména na příkladu motolice, jejíž rafinované (z pohledu člověka) chování se dá nejlépe vysvětlit teleologicky, zatímco kauzální vysvětlení nejčastěji inklinuje ke „snadné" redukci opírající se o evoluční princip (tj. ze všech možných variant motolic se nejvíce dokázala množit ta, která náhodou „způsobovala", že se její hostitel mravenec ve strnulé poloze na stéblu trávy nechal pozřít kolem procházející ovcí či jiným přežvýkavcem).

Chceme-li se však dobrat konkrétnějších důsledků zobecněné emergentistické teze, jež by něco určitého vypovídaly o povaze emergentního efektu jako souboru jevů nedílně provázejících holistický pohled na síť vzájemně interagujících kauzálních domén, musíme přistoupit k formalizaci pojmu kauzálního vztahu a kauzální domény samotné. Nejzjevnějším a formálně nejuchopitelnějším rysem emergentního systému je jeho složitost chápaná jako *množství* vztahů mezi jeho prvky, proto se zaměříme zejména na jevy podložené právě množstvím, které odpovídá souborům *abstraktních objektů*, jež jistým způsobem zastupují objekty přirozeného světa. Tyto abstraktní objekty pak v námi vytvářené formalizaci vstupují do vzájemných vztahů místo skutečných prvků systému, přičemž na rozdíl od těchto prvků nesou pouze vlastnosti vyplývající z množství, nikoli žádné další vlastnosti (kvality), jež u prvků samotných můžeme pozorovat. Jednou z nejvhodnějších formalizací těchto vztahů je vytvoření jejich množinového modelu, přičemž vztahy mezi abstraktními objekty v modelu obsaženými budou považovány za odpovídající vztahům, jež platí v modelované sféře světa přirozeného.

Klasická teorie množin však nenabízí prostředky k modelování kauzálních domén, neboť podstatou kauzální domény je právě její niterná svázanost s pozorujícím subjektem, zatímco klasická teorie množin ze své podstaty vylučuje jakéhokoli pozorovatele a na její objekty musí být nahlíženo zcela ostře, atemporálně a „odnikud". Vytvořením modelu kauzální domény v klasické teorii množin bychom tak přišli o její klíčový fenomenologický rys, který se projevuje i ve zmiňované její neostrosti.

Z těchto důvodů na několika následujících stránkách provedeme velice stručný exkurs do alternativní teorie množin a zjednodušeně představíme některé její základní principy, na nichž dále vystavíme množinový model kauzální domény a mezidoménové interakce, jenž nám pomůže odpovědět na dílčí otázky vyplývající z nastíněných témat týkajících se povahy emergentního efektu.

## 4.4 Dynamická teorie množství

*Dynamická teorie množství* (DTM) je matematickou teorií dále rozpracovávající myšlenky *alternativní teorie množin* P. Vopěnky (Vopěnka, 1989), (Vopěnka, 2001), jíž provází snaha, která by se dala stručně shrnout jako hledání odpovědi na otázku: „Jak je možné, že některé poznatky klasické infinitní matematiky jsou použitelné v přirozeném reálném světě a jiné nejsou použitelné dokonce ani při výkladech světa reálného?“[4] (Vopěnka, 2007).

Důležitou vlastností DTM je snaha o zachycení *neostrosti* z fenomenologického hlediska, tj. snaha matematizovat to, jak se v přirozeném reálném světě přítomná neostrost dává vědomému subjektu, jenž si pro jevy tohoto světa zjednává porozumění. Toto je zároveň základní požadavek, který klademe na formální aparát, v němž bychom měli být schopni zachytit model kauzálních domén.

Jev neostrosti je velice těsně spjat s jevem *nekonečna*, jak ukazuje (Vopěnka, 2000) a (Vopěnka, 2001). Uchopení neostrosti tak jde ruku v ruce s uchopením nekonečna, což se v následujících odstavcích pokusíme nastínit. Náš výklad se opírá zejména o knihu (Vopěnka, 2001), kde se však stále hovoří spíše o alternativní teorii množin (z této knihy při popisu DTM doslovně přebíráme i některé formulace, neboť se domníváme, že nejlépe vystihují podstatu dané věci). Termín „Dynamická teorie množství“ zavádí P. Vopěnka v připravované knize (Vopěnka, 2007), kde je zároveň z matematického hlediska tato teorie formulována důsledněji. Alternativní teorie množin samotná je z ryze množinově-teoretického hlediska plně popsána v (Vopěnka, 1989).

### 4.4.1 Pohled, obzor a přirozené nekonečno

Každý *pohled* (nejedná se jen o pohled vizuální, ale spíše obecně o pohled jako akt lidského vnímání[5]), ať vedený v jakémkoli směru, je omezen. Buď mu stojí v cestě nějaká hranice, která jej ostře přerušuje, nebo je omezen *obzorem*, směrem k němuž se zřetelnost vytrácí a ostrost otupuje. Je-li pozorován nějaký předmět (nejen fyzický, ale obecně jakýkoli, i abstraktní objekt), z perspektivy pozorovatele je rozdělen na dvě části – *osvětlenou* (pohledem zachycenou) a *neosvětlenou* (pohledem nezachycenou). Mezi těmito dvěma částmi prochází hranice, jež je od sebe odděluje, a tak je přesně definována její podstata, nicméně neostře a vágně vymezen její rozsah.

Čili osvětlená a neosvětlená část předmětu dohromady pokrývají celý tento předmět, přičemž se v žádném místě neprotínají. Obzor nepatří ani do jedné z uvedených částí a na rozdíl od nich není ani částí pozorovaného předmětu – je přisouzen pohledu samotnému. Ježto je tedy obzor sám o sobě ostrý a přesně definovaný jev, na němž se z jeho podstaty zobrazuje neostrost, poslouží právě on k zachycení a popsání vágnosti a neostrosti.

Část předmětu, která je nám známa, nazvěme *krajina známosti* (terra cognita) – přesahuje osvětlenou část předmětu, neboť i za obzorem pozorovaný předmět dále

---

[4] *Přirozeným reálným světem* rozumíme v podstatě to prostředí, jehož jevy nazíráme. *Reálným světem* pak jakési vědou předpokládané rozšíření přirozeného reálného světa (Vopěnka, 2007). Hlubší diskusi nad těmito pojmy lze nalézt v knize (Vopěnka, 2001).

[5] „Pohled je hledáním toho, s čím se lze shledat (tj. toho, co lze zahlédnout), a hlídáním, hleděním si a ohledáváním toho, co jsem zahlédl.“ (Vopěnka, 2001)

plynule pokračuje. *Přesahem krajiny známosti* rozumíme to, čím krajina známosti přesahuje osvětlenou část. Dále pak již je *terra incognita* („neznámá krajina").

Při zkoumání předmětu zrak (opět v abstraktním smyslu) postupuje v jednotlivých krocích směrem k obzoru. Takto vykonávanou cestu je možné rozlišit na *cestu na obzor* a *cestu směřující k obzoru*. V obou případech jde o cestu, která postupuje osvětlenou částí předmětu a míří k nějakému bodu na obzoru, jenž vyznačuje její směr. Cesta na obzor vede přímo do tohoto bodu a ani o kousek dále. Zde se ztrácí a její případné pokračování by na ni z tohoto bodu plynule navazovalo a šlo dále do přesahu krajiny známosti. Cesta směřující k obzoru končí někde v osvětlené části předmětu a její případné pokračování by začínalo až někde dále za obzorem (v závislosti na velikosti kroku překračujícího obzor), a to v místě zakrytém oním bodem, jenž udává směr. Je účelné krokům na těchto cestách přiřazovat přirozená čísla a tyto cesty – seskupení kroků – vykládat jako abstraktní třídy přirozených čísel.[6]

Cestu směrem k obzoru je zřejmě možné popsat ostře vymezenou konečnou množinou kroků. Poněvadž ta končí v oblasti před obzorem, je určitě kratší než cesta na obzor. Avšak cestu na obzor není možné popsat konečnou třídou kroků – při cestě na obzor se k němu stále „blížíme", nicméně nebude existovat poslední krok, o němž bychom mohli prohlásit, že vede přesně na obzor. Jako příklad je možné uvést situaci, kdy zmenšujeme kousek papíru odtrháváním jeho částí – v jistém kroku již nejsme schopni zachytit ani velikost papíru, která se propadne do neosvětlené části světa poznávaného tělesnými smysly, zanechávajíc po sobě na obzoru pouze stopu v podobě geometrického bodu. I kdybychom zvyšovali smyslové schopnosti nade všechny meze, nepodaří se nám vytvořit z papíru geometrický bod. Cesta na obzor je tedy nekonečná.

Ačkoli tomu tak je, můžeme stále vykonat cestu, která je jejím prodloužením do přesahu krajiny známosti – za obzor. Třída všech kroků na této cestě je množinou, a to konečnou. Přitom třída těch z nich, které leží pouze na cestě na obzor, je zřejmě její neostře vymezenou podtřídou. Neostře vymezené (nekonečné) podtřídy ostře vymezených konečných množin nazývejme *polomnožinami*.

Polomnožiny ilustruje kupříkladu notoricky známý paradox hromady – polomnožina je třída těch přirozených čísel, pro něž platí, že odebereme-li z dané hromady písku takovýto počet zrníček, stále ještě zbude hromada písku. Odebereme-li totiž kdykoli z hromady písku jedno zrníčko, vždy zůstane hromada, avšak původní hromada jistě obsahovala konečný počet zrníček (zde je zřejmá polomnožinová povaha jazykových pojmů). Jiným příkladem polomnožiny je váha nákladu, kterou unese osel (abychom jmenovali opět tradičně uváděné zvíře): přidáme-li kdykoli oslovi k nějakému nákladu jeden gram, víme, že osel takto zvětšenou zátěž unese. Víme však též, že celou tunu dozajista neunese, proto množství jednogramových přídavků nákladu, které osel unese, odpovídá polomnožině.

Dalším známým příkladem polomnožiny je třída všech těch přirozených čísel, kterou lze v českém jazyce jednoznačně popsat v několika větách neobsahujících dohromady

[6] Zde zatím chápejme pojem „třída" jako něco velmi podobného pojmu „množina". Rozdíl mezi třídou a množinou je vysvětlen o několik odstavců dále. *Abstraktní* třída navíc v tomto kontextu znamená třídu, která obsahuje pouze abstraktní objekty (tj. objekty, které primárně nenesou žádnou jinou vlastnost než „být objektem" a v množinovém modelu zastupují objekty přirozeného světa, popř. objekty, které obsahují pouze jiné abstraktní objekty), nikoli například objekty přirozeného světa.

více než tisíc písmen. Všech nejvýše tisícičlenných posloupností české abecedy je jen konečně mnoho, a tedy všechna čísla zachytitelná uvedeným způsobem jsou menší než nějaké hodně vysoké přirozené číslo, takže uvedená třída je podtřídou nějaké množiny. Kdyby však tato třída sama byla množinou, bylo by nejmenší přirozené číslo, které do ní již nenáleží, jednoznačně popsáno, a sice těmi větami, které jsme právě pronesli. Jelikož však tyto věty mají dohromady méně než tisíc písmen, náleželo by i toto číslo do uvedené množiny, což je spor.

Klasický výklad nekonečna umisťuje nekonečné množství až za všechna množství klasicky vykládaná jako konečná. Toto je natolik zažitý zvyk, že je velice obtížné jej překonat, přitom ze samotného pojmu „nekonečno" není nikde patrné, proč by tak tomu mělo být – nekonečno netvrdí, že musí obsahovat více objektů než konečno; nekonečno znamená, že pouze není jeho konec (Cantor, 1965). To, co pozorujeme na polomnožinách, nazvěme *přirozené nekonečno*. Vztah přirozeného nekonečna k jiným nekonečnům a jeho úlohu v lidském poznávání hlouběji analyzuje (Vopěnka, 2000).

### 4.4.2 Množinový popis přirozeného nekonečna

To, co se jeví na kauzálních doménách, je právě přirozené nekonečno, které pomocí přirozených čísel uchopíme na oboru abstraktních tříd a jehož popsání nám umožní přiblížit se k podstatě emergentního efektu. K tomu však musíme použít určitého formálního vyjadřovacího aparátu.[7] Jelikož potřebujeme nějak jednoduše nazývat dohromady množiny a polomnožiny, začneme užívat termínu „třída" jako souhrnného názvu pro množiny a polomnožiny. Každá množina je tak zároveň třídou (a to ostře a konečně vymezenou), avšak nikoli každá třída je též množinou – může to být i polomnožina.

Nyní stručně uveďme přehled základních operací, které nám umožní přirozené nekonečno kauzálních domén uchopit (vesměs jde o klasické a běžně známé množinové operace). Užijme následujícího značení: $A \times B$ označuje kartézský součin tříd $A$, $B$, tj. třídu všech uspořádaných dvojic $\langle x, y\rangle$ takových, že $(x \in A) \wedge (y \in B)$. Relací (binární) rozumíme takovou třídu, jejíž každý prvek je uspořádanou dvojicí.[8] $A \mid B$ označuje parcializaci třídy $A$ na třídu $B$, tj. třídu všech uspořádaných dvojic $\langle y, x\rangle$, kde $(\langle y, x\rangle \in A) \wedge (x \in B)$. Definiční obor relace $R$ značíme $dom(R)$ a je to třída všech $x$ takových, že $\langle y, x\rangle \in R$, obdobně obor hodnot relace $R$ značíme $rng(R)$ a je to třída všech $y$ takových, že $\langle y, x\rangle \in R$. Je-li $A$ třída, jejíž prvky jsou též třídy, pak $\bigcup A$ označuje sjednocení všech tříd, jež jsou prvky třídy $A$. Analogicky je definován i průnik $\bigcap A$.[9]

Relace $F$ je funkcí, když ke každému $x \in dom(R)$ existuje právě jedno $y$ takové, že $\langle y, x\rangle \in F$; toto $y$ značíme $F(x)$. Relace $R$ má konjugovanou funkci, jestliže pro každé $x \in dom(R)$ je třída $y$ takových, že $\langle y, x\rangle \in R$, množinou. Konjugovanou funkcí té relace je funkce $F$ taková, že $dom(F) = dom(R)$ a pro každé $x \in dom(R)$ je $F(x)$ množina všech $y$ takových, že $\langle y, x\rangle \in R$. Konjugovanou relací funkce $F$ rozumíme třídu všech $\langle y, x\rangle$ takových, že $y \in F(x)$.

---

[7] Zde uvedeme jen pro nás nejnutnější základy. Podrobnosti nechť čtenář hledá v (Vopěnka, 2001) a (Vopěnka, 2007).

[8] Množinové relace modelují vztahy mezi objekty – je-li mezi objekty $x$ a $y$ daný vztah, pak je uspořádaná dvojice $\langle x, y\rangle$ prvkem třídy, která tento vztah reprezentuje. Vztahy mohou být například „být těžší než", „být sourozencem někoho", ale i „být příčinou něčeho" (tj. kauzální vztah mezi dvěma jevy).

[9] Je-li například $A = \{\{1, 2, 3\}, \{1, 3, 4\}, \{1, 3, 5\}\}$, pak $\bigcup A = \{1, 2, 3, 4, 5\}$ a $\bigcap A = \{1, 3\}$.

Třídu všech objektů spadajících do oboru abstraktních tříd označíme $V$. Třídu všech abstraktních objektů náležejících do $V$, jež nejsou množinami, označíme $V_0$. Malými písmeny latinské abecedy budeme označovat pouze prvky třídy $V$. Podtřídy třídy $V$ budeme označovat velkými latinskými písmeny. Třídu přirozených čísel (náležejících do $V$) označíme $N$. Přirozená čísla samotná budeme označovat malými řeckými písmeny.

Řezem na přirozených číslech rozumíme takovou třídu přirozených čísel, která spolu s každým v sobě obsaženým číslem obsahuje i všechna přirozená čísla menší. Je-li $\alpha \in N$, pak třída všech přirozených čísel menších než $\alpha$ je řezem na přirozených číslech, je to množina a značíme ji $[\alpha]$. Tento řez nazýváme řezem hlavním a číslo $\alpha$ je jeho (vnější) hlavou.

Je-li $H$ řez na přirozených číslech, říkáme, že posloupnost nějakých objektů na řezu $H$ je na něm stabilní, jestliže pro každé přirozené číslo $\alpha \in H$ je úsek této posloupnosti až do čísla $\alpha$ (včetně) ostře vymezen.

Klíčovým pojmem k uchopení prve zmiňované podoby přirozeného nekonečna (a s tím i kauzálních domén) je *obzorný řez*. Obzorným řezem rozumíme takový řez $H$ na přirozených číslech, pro který platí:

a) Řez $H$ nemá poslední (tj. největší) prvek a třída $H$ je polomnožinou.

b) Každou posloupnost na řezu $H$, která je na tomto řezu stabilní, lze prodloužit do posloupnosti, která je stabilní na nějakém hlavním řezu na přirozených číslech, jehož podtřídou je třída H.

Abychom mohli obzorný řez formalizovat, zavedeme ještě následující termíny: řekneme, že funkce $F$ je stabilní na řezu $X$, je-li $dom(F) = X$ a pro každé $\gamma \in X$ je $F \mid [\gamma + 1]$ množina náležející do třídy $V$. Řekneme, že funkce $f$ je tvrdým prodloužením funkce $F$ definované na řezu $X$, jestliže existuje $\alpha$ takové, že $dom(f) = [\alpha]$, a $F = f \mid X$.

Na základě toho řekneme, že neprázdná třída $H \subseteq N$ je obzorným řezem, jestliže platí:

a) $H$ je řez, který v uspořádání přirozených čísel nemá největší prvek.

b) Je-li $F$ stabilní funkce na řezu $H$, pak existuje její tvrdé prodloužení $f$.

Řekneme, že třída $A$ je přehledná na obzorném řezu $H$, jestliže existuje stabilní funkce $F$ na řezu $H$ taková, že $A = rng(F)$. Řekneme, že $X$ je $\sigma$-třída (vzhledem k řezu $H$), jestliže existuje přehledná třída množin $A$ (tj. třída $A$, jejíž prvky jsou množiny) taková, že $X = \bigcup A$. Řekneme, že $X$ je $\pi$-třída (vzhledem k řezu $H$), jestliže existuje přehledná třída množin $A$ taková, že $X = \bigcap A$.

V rámci DTM mají mezi všemi třídami obzvláštní význam právě $\sigma$- a $\pi$-třídy, jejichž vzájemné vztahy ukazují na souvislosti, jež klasická teorie množin vytěsnila z oblasti svého studia. Není naším cílem uvádět a odvozovat zde všechna tvrzení týkající se $\sigma$-tříd a $\pi$-tříd – za tímto účelem čtenáře odkážeme na knihy (Vopěnka, 2001), (Vopěnka, 2007), odkud zde vybíráme dvě základní tvrzení, jež mají pro nás v danou chvíli zásadní význam (důkazy těchto tvrzení lze samozřejmě nalézt tamtéž).

*Tvrzení 1.* Buď $X \subseteq w$, kde $w$ je množina. Nechť $X$ je $\sigma$-třída (popř. $\pi$-třída). Potom $w - X$ je $\pi$-třída (popř. $\sigma$-třída).

*Tvrzení 2.* Třída $X$ je současně $\sigma$-třída i $\pi$-třída právě tehdy, když $X$ je množina.

Je důležité povšimnout si rozdílu mezi vnitřní strukturou $\sigma$-třídy a $\pi$-třídy (tento rozdíl později využijeme při modelování interakce kauzálních domén), který je dán použitím odlišného operátoru při jejich konstrukci (operace sjednocení proti vůči ní v jistém smyslu komplementární operaci průniku). Ve výsledné podobě $\sigma$-třídy se totiž mnohem lépe odráží charakter funkce $F$ definované na daném obzorném řezu, jež stojí za vznikem oné $\sigma$-třídy, neboť tato $\sigma$-třída obsahuje všechny prvky, na něž funkce $F$ poukazuje prostřednictvím svých funkčních hodnot (díky jejich sjednocování). Naproti tomu $\pi$-třída – byť vztažená ke stejnému obzornému řezu – tento charakter stále více zakrývá tím, jak se s postupným přibližováním pohledu k obzoru zmenšuje a své prvky „ztrácí" (po operaci průniku vždy zůstanou jen ty objekty, jež byly prvky současně všech funkčních hodnot funkce $F$ až do daného kroku).

Dále zaveďme pojem *prvoevidovatelný jev*. Prvoevidovatelným jevem nazvěme takový jev, který jsme schopni evidovat, tj. vidět ho a vědět, že ho vidíme, jakmile ho evidovat můžeme. Například jev červenosti je prvoevidovatelný, zatímco jev nečervenosti prvoevidovatelný není, neboť k jeho seznání potřebujeme mít již uchopen jev červenosti. Obdobně jev rozlišitelnosti je prvoevidovatelný, jev nerozlišitelnosti prvoevidovatelný není. Obzvláště důležité je uchopení rozdílnosti ve strukturách jevů prvoevidovatelných a jejich doplňcích (negacích).

Nastíněné matematické vztahy je vhodné interpretovat tak, že v oboru abstraktních tříd odpovídají prvoevidovatelným jevům $\sigma$-třídy. Znamená to tedy, že za účelem studia neostrosti prvoevidovatelných jevů je možné tyto jevy podkládat různými $\sigma$-třídami a z vlastností takových $\sigma$-tříd pak zpětně nacházet vlastnosti jimi podložených prvoevidovatelných jevů.

Z uvedeného Tvrzení 1 pak vyplývá, že doplňkem určité $\sigma$-třídy do její nadmnožiny je $\pi$-třída, čili negace prvoevidovatelných jevů mají jinou strukturu než prvoevidovatelné jevy samotné. Negace prvoevidovatelných jevů je tedy možné popsat $\pi$-třídami. Vyjádříme-li se více v duchu analytické filosofie, abychom neužili fenomenologicky laděného termínu „jev", pak můžeme říci, že například extenzí pojmu „červený" je $\sigma$-třída všech objektů, které jsou červené, a extenzí pojmu „nečervený" (jako negace „červeného") je $\pi$-třída všech objektů, které nejsou červené. Oproti klasickému množinovému vyjadřování tak získáváme účinnou formalizaci důležitého kvalitativního rozdílu spočívajícího v tom, že červené věci jsou červenými *jinak* než nečervené věci nečervenými. Ze znalosti toho, jaké to je *být nečervené*, nelze vzhledem k charakteru $\pi$-třídy nijak odvodit, jaké to je *být červené*, což dobře odpovídá naší běžné zkušenosti.

Podle Tvrzení 2 dále můžeme usuzovat, že jev, který je prvoevidovatelný stejně jako jeho negace, je jevem ostře vymezeným, jelikož doplněk $\sigma$-třídy $X$ je též $\sigma$-třídou pouze tehdy, je-li $X$ množinou. Má-li naopak doplněk třídy $X$ být pouze $\pi$-třídou, musí být $X$ pouze $\sigma$-třídou, a tudíž neostře vymezenou.

## 4.5 Mezidoménová interakce a emergentní efekt

### 4.5.1 Model kauzální domény

V tuto chvíli by bylo vhodné poněkud zúžit a precizovat, co v dané situaci míníme, užijeme-li slova „jev". Z hlediska fenomenologického můžeme jako jev chápat všechno, „...co lze pohledem zachytit a odlišit od všeho ostatního, na co lze takto ukázat, [pro co] si mohu zjednat první porozumění spočívající v tom, že to nějak 'je' a že to je 'to'." (Vopěnka, 2001) Nás však spíše zajímají takové jevy, které s sebou nesou určitou kauzální potenci, tzn. mohou něco způsobit. Fenomenologickému vymezení jevu by tak odpovídala i objektová individua – např. nějaká kočka. Nelze však říci, že by kočka sama o sobě jako jsoucno něco způsobovala. Jazykové zkratky sice dovolují říci větu: „Kočka způsobila, že pes začal štěkat.", avšak ve skutečnosti štěkot psa byl způsoben tím jevem, že se kočka objevila na zahradě. V tomto případě výrazy „kočka" a „pes" máme na mysli jednu konkrétní kočku a jednoho konkrétního psa, např. Mindu a Žeryka. Označuje-li výraz jednotlivý předmět, nazvěme jej *vlastní jméno* (Fiala, 2006). Za jevy v zúženém smyslu budeme tedy nadále pro naše potřeby považovat propozice,[10] v nichž figurují pouze vlastní jména.

Omezením se na vlastní jména sice možná ztrácí takto chápaný jev na své obecnosti, avšak naší snahou je vyhnout se co možná nejvíce rozličným logicko-jazykovým či jiným problémům, které by vyvstaly například při zahrnutí propozic, jež by mimo vlastní jména obsahovaly i pojmy. K modelování mezidoménové interakce a emergentního efektu zjednodušeně chápané jevy bohatě postačí – dokonce i v případě mentálních stavů lze učinit dohodu, že každá mysl bude v našem modelu disponovat svými vlastními „exempláři" rozličných mentálních stavů – takže například deprese či migréna jednoho člověka bude jiným individuem než u člověka druhého. Za jevy tedy budeme, zhruba řečeno, považovat jednotlivé události, které se ve světě mohou stát, nikoli univerzální tvrzení (univerzální tvrzení – jak bylo diskutováno dříve – nám umožňují kauzální výklad jevů).

Předpokládejme, že můžeme uvažovat dostatečně rozsáhlé universum jevů přirozeného reálného světa – natolik rozsáhlé, aby v sobě obsahovalo alespoň dvě kauzální domény. Každému objektu z tohoto universa jednoznačně přiřadíme jeden abstraktní objekt (resp. objekt z oboru abstraktních tříd). Takto vzniklou třídu abstraktních objektů označme $U$ (jde vlastně o množinový model universa jevů).

Zabývejme se nyní kauzální doménou, již označíme symbolem $i$. Tato doména nechť vypovídá o dostatečně bohatém fragmentu přirozeného reálného světa (bohatém v tom smyslu, aby například neobsahoval pouze chování nějakého jednoduchého logického automatu), který je zároveň součástí universa $U$. Třídu všech jevů domény $i$ označme $H_i$. Zřejmě $H_i \subset U$.

Kráčíme-li od jádra[11] domény $i$ směrem vytyčeným její hranicí, můžeme se v každém kroku u každého jevu zeptat, zda stále patří do této domény (tj. zda o něm konzistentně a přirozeně můžeme uvažovat ve vztahu k ní). Odpovíme-li si kladně, pak

[10] Propozicí chápeme větu, u níž lze hovořit o její pravdivostní hodnotě.

[11] „Jádrem" domény nazvěme tu podtřídu jejích jevů, u nichž shledáváme nejvyšší „zřejmost" kauzálních vztahů a „přirozenost" nacházet se v této doméně.

zároveň víme, že u bezprostředně následujícího jevu (tj. jevu v epistemickém smyslu „nejbližšímu") odpovíme též kladně. Zároveň však víme, že pokud bychom naráz přeskočili dostatečně velké množství jevů, nalezli bychom takový jev, který už do kauzální domény *i* bezpečně nepatří. V jedné kauzální doméně tak budou například jevy týkající se fungování neurotransmiterů; jsou-li tam tyto jevy, pak tam nejspíše budou i jevy týkající se mitochondrií[12]; jsou-li v doméně zahrnuty mitochondrie a jejich kauzální vztahy, pak tam zřejmě mohou být kauzální vztahy, v nichž vystupují synapse či celé neurony; obsáhne-li tato kauzální doména i neurony, pak může popisovat též dynamiku celých neuronálních struktur a je zřejmě možné, aby v ní byly konzistentně zahrnuty i dílčí kognitivní funkce. Takto můžeme kauzální doménu rozšiřovat o další a další jevy (v daném případě nejspíše s rozvojem příslušného vědního oboru) a vždy, když do ní určitý jev vložíme, nemůžeme do ní nezahrnout i jev jemu nejbližší.

To, co jsme si ukázali, bychom mohli charakterizovat jako projev principu jakési „přirozené indukce", který na universu jevů vytváří projevy přirozeného nekonečna. Z toho budeme usuzovat, že $H_i$ je obzorný řez.

To, jak se věci mezi sebou v kauzální doméně *i* mají, nazíráme prostřednictvím jisté teorie $T_i$ – v nejobecnějším případě jazykem, který nám umožňuje vztahy mezi jevy strukturovat a zachytit. Tato teorie vlastně říká, které kauzální vztahy jsou zjevné, srozumitelné a vzájemně koherentní, tudíž je v epistemologickém smyslu vlastně konstituujícím elementem kauzální domény. Naproti tomu má-li jakákoli teorie o kauzální doméně vůbec vzniknout, musí tato kauzální doména již nějakým způsobem existovat (v ontologickém smyslu). Teorie je tak se svojí kauzální doménou spjata v hermeneutickém kruhu, který úzce souvisí se zjednáváním světa jakoukoli vnímající bytostí (Brooks, 1999).

V oboru abstraktních tříd můžeme vytvořit jednoduchý model kauzálních vztahů v rámci kauzální domény pomocí relace, do níž budou vstupovat právě ty objekty, které jsou ve vzájemném kauzálním vztahu. Nechť tedy existuje relace $R_i$ taková, že $dom(R_i) = H_i$, $rng(R_i) \in H_i$ a dva objekty (jevy) $x$, $y$ jsou v relaci $R_i$ (tj. $\langle y, x \rangle \in R_i$) právě tehdy, když $x$ přímo způsobuje $y$ (resp. $y$ je přímým důsledkem $x$). Například specifická hladina fluoxetinu (jednoho z inhibitorů zpětného vychytávání serotoninu[13]) způsobí změnu zpětného vychytávání serotoninu, což způsobí změnu hladiny serotoninu.

Nechť dále $F_i$ je konjugovaná funkce k relaci $R_i$. Abychom mohli zajistit její existenci, budeme po teorii $T_i$ požadovat, aby třída všech přímých důsledků každého jevu $x$ byla ostře vymezena, a tudíž byla množinou. Teorie ve smyslu souborů formálních tvrzení budou nejspíše tento požadavek splňovat, stejně tak i teorie v obecnějším smyslu teorií vědeckých. Teorie ve smyslu vágně vymezených a formulovaných oblastí lidského poznání, vyjádřená přirozeným jazykem, tento požadavek samozřejmě splňovat vždy nemusí, podobně si lze i představit speciálně vytvořenou formální teorii, která jej splňovat nebude záměrně. Tyto případy z našeho zkoumání vyloučíme z technických důvodů, abychom si usnadnili práci v oblasti operací s abstraktními třídami.

---

[12] Neurotransmitery (nervové přenašeče) jsou objekty, které se významným způsobem podílejí na synaptickém signálovém přenosu. Mitochondrie je částice nacházející se v tzv. presynaptické části chemické synapse neuronových buněk.

[13] Serotonin je jeden z řady neurotransmiterů. Zpětné vychytávání (či znovuvstřebávání, angl. reuptake) je jedna z fází procesu tohoto přenosu, která reguluje počet uvolněných neurotransmiterů. Více např. (Novák a kol., 1998).

Pro každé $x \in H_i$ je $F_i(x)$ množinou všech důsledků jevu $x$ v rámci kauzální domény $i$. Funkce $F_i$ je tak vlastně modelem její vnitřní struktury. Relace $R_i$ je stabilní na řezu $H_i$, proto je na něm stabilní i funkce $F_i$, a tedy třída $A_i = rng(F_i)$ je na obzorném řezu $H_i$ přehledná. Z toho vyplývá, že třída $D_i = \cup A_i = \cup rng(F_i)$ je σ-třída. $D_i$ zjevně obsahuje všechny jevy z domény $i$, jež jsou důsledky jiných jevů ze stejné domény (čili $D_i$ neobsahuje „počáteční podmínky" – tj. jevy nemající v dané doméně žádnou příčinu). Třídu $D_i$ nazvěme *kostra kauzální domény i*.

Interpretace kostry kauzální domény jako σ-třídy odpovídá přirozené představě kauzální domény jako jevu prvoevidovatelného. Člověk obklopený spletí jevů, jíž si zjednává porozumění, jaksi automaticky poznává, co jsou kauzální domény (opírajíc se o jejich kostry v podobě prvoevidovatelných seskupení objektů, z nichž domény sestávají), avšak z této schopnosti či znalosti nikterak neurčí, co kauzální domény nejsou (čili co není v nějaké kauzální doméně, pro to si nezjedná porozumění). Ve vztahu k teorii $T_i$ je kostra kauzální domény $i$ vlastně třídou všech jevů, které jsou v této teorii kauzálně vysvětlovány.

### 4.5.2 Model mezidoménové interakce

Jak již bylo dříve naznačeno, máme-li se dopátrat zajímavějších výsledků, musí být universum $U$ dostatečně bohaté, aby v sobě obsahovalo přinejmenším dvě kauzální domény – jednu výchozí a druhou, z níž budeme na tu výchozí pohlížet. O jaké konkrétní domény půjde, je vcelku lhostejné, avšak v souladu s naším prvotním tématem týkajícím se emergentního efektu v systémech s umělým či přirozeným myšlením můžeme model mezidoménové interakce ukázat na příkladu, v němž výchozí kauzální doménu $i$ nazveme třeba doménou nervových procesů[14] a pokusíme se ukázat, jak se vztahuje k doméně jiné, která složité nervové procesy obvykle doprovází, a to doméně mentálních stavů. Dále v textu budeme tyto domény zkráceně nazývat „nervová" a „mentální".

Nechť tedy všechny jevy domény mentální jsou zastoupeny třídou $C_m$, přičemž platí, že $C_m \subset U$ a zároveň $H_i \cap C_m = \emptyset$ (tj. doména nervová nemá žádné společné jevy s doménou mentální – jinak by to nebyly dvě různé kauzální domény).

Zkoumáme-li mezidoménovou interakci, chceme vlastně vědět, co způsobují jevy z výchozí kauzální domény v kauzální doméně cílové – jinými slovy se můžeme ptát, jaké jevy z domény nervové způsobily pozorovaný jev v doméně mentální. Zajímají nás tedy kauzální vztahy, do nichž vstupují jevy z nervové domény na jedné straně a jevy z domény mentální na straně druhé. Pakliže bychom se z vědeckého hlediska opravdu zajímali o tyto dvě konkrétní domény, chtěli bychom též vytvořit teorii, která by tyto mezidoménové kauzální vztahy shrnovala. Model této teorie bychom v oboru abstraktních tříd mohli zachytit funkcí podobnou funkci $F_i$. V rámci našeho problému nás samozřejmě nezajímá konkrétní podoba této teorie či její modelující funkce, ale spíše epistemologické hranice, jež možnosti jejího nalezení vymezují.

Náš pohled na výchozí (nervovou) doménu $i$ rozšíříme nyní tak, aby dosáhl až k doméně mentální a pokryl alespoň ty jevy z ní, které nás zajímají. Vytvoříme tedy

[14] „Nervové procesy" jak ve smyslu původním (biologickém), tak i ve smyslu přeneseném – „in silicio" (tj. „v počítači").

množinu $w$ takovou, že $D_i \subset w$ a současně $w \cap C_m = \emptyset$. Množina $w$ obsahuje všechny jevy kostry kauzální domény $i$ a zároveň k nim přidává třídu dalších jevů, mezi nimiž jsou ty z mentální domény, jež nás zajímají.

Nyní se pohledem přesuneme z nervové domény směrem k doméně mentální a z této perspektivy se pokusíme nahlédnout, jak jsou jevy ze třídy $X_i = w - D_i$ kauzálně závislé na jevech z $D_i$. Analogicky k funkci $F_i$ se pokusíme najít stabilní funkci $F_{Xi}$, která bude též definována na řezu $H_i$ (tj. stejné třídě příčin jako má $F_i$), avšak obor hodnot se rozšíří o množiny důsledků, jež nejsou v $D_i$, tj. patří do mentální domény (či do „temného prostoru" mezi oběma doménami, který nás však nyní nezajímá).

A zde se dostáváme k velice důležitému bodu: podle Tvrzení 1 je třída $X_i$ π-třídou a tudíž $X_i = \bigcap rng(F_{Xi})$ a jelikož $D_i$ a tedy ani $X_i$ nejsou množiny, tak podle Tvrzení 2 ani neexistuje takové $F'_{Xi}$, aby platilo $X_i = \bigcup rng(F'_{Xi})$. Z toho vyplývá klíčový poznatek, že všechny prvky třídy $X_i$ nutně musejí být rovněž obsaženy ve *všech* prvcích třídy $rng(F_{Xi})$ (které samy jsou množinami).[15] Vezmeme-li v úvahu interpretaci funkce $F_{Xi}$, pak to znamená, že *každý* jev z mentální domény musí být v nově vznikající teorii principiálně zahrnut mezi důsledky *všech* jevů z domény nervové!

Jinými slovy řečeno: pokud se zeptáme z hlediska mentální domény, které jevy nervové domény *způsobují* daný jev v doméně mentální, a budeme trvat na kauzálním vysvětlení, odpovědí bude, že všechny jevy obsažené v nervové doméně jsou jeho příčinou. Znamená to tedy, že při doménové fragmentaci světa můžeme vykládat vztahy mezi jevy napříč doménami kauzálně, avšak v tom případě musíme připustit, že jde o kauzalitu poněkud zvláštního rázu. Může se totiž zdát, že tento výsledek není v souladu se „zdravým rozumem" a základními pozorováními jevů reálného světa: je přece „jasné", že inhibujeme-li v nervové kauzální doméně zpětné vychytávání serotoninu, v mentální doméně člověka trpícího depresí to *způsobí*, že deprese poleví či zmizí. Podle námi prezentovaných výsledků však je vymizení deprese *způsobeno* i například jevem zcela opačným, tj. posílením zpětného vychytávání serotoninu, byť by tento jev nebyl v danou chvíli pozorován. Z těchto důvodů nazvěme náš poznatek o mezidoménové kauzalitě jako *kauzální paradox vícedoménových systémů* (v užším případě též kauzální paradox emergentních systémů).

Zcela záměrně se nebudeme zabývat jakýmkoli klinickým faktem, jako například že někdy inhibice zpětného vychytávání serotoninu nejen k vyléčení deprese v danou chvíli nepomůže, ale depresivní stavy se dále prohloubí[16] (můžeme pak říci, že tato inhibice *způsobila* zhoršení deprese?) – tímto bychom totiž odváděli pozornost od „zásadnějšího" problému (zásadnějšího pouze v rámci našeho textu – pochopitelně nikoli pro člověka trpícího depresí): je vůbec možné tvrdit (ve smyslu hlubším, než je pouhá jazyková konvence), že nějaká změna stavu věcí v jedné kauzální doméně *způsobí* nějaký jev v jiné kauzální doméně? Dle našeho názoru nikoli, neboť pak se nelze vyhnout kauzálnímu paradoxu. Dozajista existuje nějaký vztah mezi stavem serotoninu na synapsích a depresí, avšak tento vztah nelze konzistentně chápat jako kauzální (tj. příčinný).

---

[15] Toto vyplývá z logiky operace průniku; tak například máme-li funkci $G$ definovanou tak, že $G(1) = \{1, 2, 3, 4\}$, $G(2) = \{1, 2, 3\}$, $G(3) = \{2, 3\}$, pak $rng(G) = \{\{1, 2, 3, 4\}, \{1, 2, 3\}, \{2, 3\}$ a $rng(A) = \{2, 3\}$.

[16] K činění podobných tvrzení se samozřejmě ani zdaleka necítíme být kompetentní.

### 4.5.3 Interpretace kauzálního paradoxu

Kauzální paradox a jeho důsledky se zde pokusíme interpretovat ze dvou základních hledisek: z hlediska epistemologického – tj. co nám kauzální paradox vypovídá o možnostech, způsobech a hranicích poznání přirozeného reálného světa, speciálně pak jaké z něj můžeme učinit závěry v rámci logiky vědeckého bádání (ve smyslu „teorie teorií", viz (Popper, 1997)); a z hlediska umělé inteligence, resp. kognitivní vědy – tj. jak se kauzální paradox promítne do prve nastíněné problematiky umělého a přirozeného myšlení.

Z logiky množinové formulace kauzálního paradoxu vidíme, že důležitým aspektem je fakt, že třída $X_i$ je π-třídou vzhledem k obzornému řezu $H_i$, stejně jako je vzhledem k němu třída $D_i$ σ-třídou. Z toho plyne, že výklad příčinné souvislosti v mezidoménové interakci nutně požaduje, aby kauzální vztahy byly mezi všemi jevy výchozí kauzální domény (tj. všemi prvky řezu $H_i$) a všemi zkoumanými jevy mimo tuto doménu. I kdyby se nám však podařilo klást na mezidoménovou „kauzální" funkci $F_{Xi}$ takové požadavky, aby nebyla definována na celém obzorném řezu $H_i$ (tj. na celé kauzální doméně $i$), ale pouze na některé jeho části – čímž by se zúžila třída myslitelných příčin (například kdybychom do celého formálního popisu vnesli časovost a nějak bychom zužovali třídu jevů z domény $i$, jež mohou být myslitelnými příčinami, toliko na ty jevy, které jsou v daném časovém okamžiku pozorované, resp. „aktuální") – pořád bychom se nezbavili těch vlastností třídy $X_i$, které z ní činí π-třídu, a kauzální paradox by se nikterak neredukoval. Stále by totiž všechny jevy třídy $X_i$ byly důsledky všech jevů ze zúžené třídy příčin (tj. z definičního oboru funkce $F_{Xi}$).

Z třídy $X_i$ činí π-třídu právě přirozeně nekonečné narůstání σ-třídy $D_i$, jemuž lze dle Tvrzení 2 „zabránit" pouze ostrým množinovým vymezením. Kauzální paradox tak můžeme obejít jedině tím, že vybereme nějakou ostrou podmnožinu jevů ze třídy $H_i$, s níž budeme nadále pracovat. Tímto „seříznutím" původní kauzální domény se však všechna další tvrzení budou týkat nikoli původní kauzální domény, jak ji empiricky seznáváme, ale jejího více či méně zkresleného ostře vymezeného množinového modelu. K výsledné redukované (či zkomolené[17]) doméně lze připojit podmnožinu jevů z třídy $C_m$ a nad tímto můžeme dále hledat kauzální funkci již bez kauzálního paradoxu. Dostáváme však kauzální doménu novou, zcela odlišnou od obou původních kauzálních domén, jejichž vztah nás nejdříve zajímal. Naše závěry činěné nad touto novou doménou potom však hovoří (byť naprosto konzistentně a kauzálně uspokojivě) o něčem zcela jiném, než o co nám zprvu šlo.[18]

Budeme-li se naopak snažit o nereduktivní výklad daného fragmentu světa, kauzálnímu paradoxu se nevyhneme. Jelikož na základě empirických kritérií nemůžeme jednoduše přijmout představu, v níž nelze každý jev jedné domény kauzálně vysvětlit jinak než jako důsledek všech jevů druhé domény, musíme nalézt jiné východisko. To spočívá v zamítnutí předpokladu kauzální interakce mezi jevy z různých kauzálních domén. Nemůžeme tedy tvrdit (rozumí se jinak, než v rámci prosté jazykové konvence),

[17] Termínu „zkomolené jevy" užívá P. Vopěnka pro ostré „aproximace" jevů neostrých – neostrý přírodní jev se stane ostrým pouze tehdy, vytvoří-li se seznam všech jeho výskytů; takový seznam však původní jev komolí a jeho zkomolenina je přísně vzato jevem novým (Vopěnka, 2007).

[18] Vhodným zamaskováním tohoto aspektu pak dochází k dnes leckdy vídaným sebevědomým prohlášením typu „podstata vědomí vysvětlena" apod.

že jev z jedné kauzální domény (například již zmiňované zpětné vychytávání serotoninu) *způsobuje* jev z jiné kauzální domény (opět například fenomenologicky prožívané psychické stavy, jako deprese apod.). Nemůžeme ani tvrdit, že by tento vztah byl kupříkladu charakteru pravděpodobnostního – do pravděpodobnostních vztahů mohou vstupovat pouze jevy, které jsou prvky pravděpodobnostní algebry, a tudíž jde v důsledku o jevy jiné (zástupné), než jsou původně zkoumané jevy z kauzálních domén. Pravděpodobnostní výklad je opět zkomolením (byť nesmírně účinným) výchozí vícedoménové situace, neboť opět nahrazuje dvě kauzální domény doménou jedinou, v rámci níž jsou pravděpodobnostní jevy pak již kauzálně soudržné a bezesporné.

Podobně lze v rámci snahy o porozumění vědomí charakterizovat Chalmersův tzv. *„snadný problém"* (Chalmers, 1996), cit. podle (Havel, 2001). Problém, který lze takto pojmout, je v jistém (metodologickém) smyslu „snadný", neboť jeho vymezení odpovídá redukci mezidoménové interakce na kauzální vztahy v rámci jedné kauzální domény, vytvořené například prostřednictvím teorie pravděpodobnosti. Příkladem může být snaha odpovídat jazykem neurologie na otázky kladené jazykem (objektivní) psychologie. Naproti tomu *„těžký problém"* formulujeme takto (Havel, 2001):

*Jak vůbec může fyzické (chemické, biologické) dění v mozku vést k mému subjektivnímu prožitku?*

Jinými slovy řečeno: co *způsobuje*, že projevy naší nervové soustavy nevnímáme jako projevy naší nervové soustavy, ale jako subjektivně prožívané mentální stavy, jež disponují svojí fenomenální komponentou? Kupříkladu proč excitaci jistých skupin neuronů nezpracováváme jednoduše jako excitaci jistých skupin neuronů, ale jako pohled na rozkvetlou louku? Ve formulaci „těžkého problému" je z naší perspektivy jasně patrná snaha o nereduktivní zachování mezidoménové interakce.

Jakého charakteru však mohou být vztahy, které působí mezi jevy napříč kauzálními doménami, nejsou-li to vztahy kauzální? Chápeme-li kostry kauzálních domén jako třídy jevů, které mohou být teoriemi kauzálně vysvětleny, pak je naše interpretace kauzálního paradoxu v souladu s představou, že kauzální vysvětlení nemůže překračovat hranice vymezené těmito teoriemi. I v případě, že bychom ve vztahu ke kauzalitě zaujali radikální postoj po vzoru D. Huma – že totiž všechny vztahy mezi jevy, které považujeme za kauzální, nejsou ve skutečnosti ničím jiným než pouhou koincidencí (tj. souvýskytem) daných jevů a jen induktivní přístup spolu s pravidelným opakováním této koincidence nám umožňuje *věřit*, že vztahy jsou svázány kauzálně – nepodaří se nám uspokojivě překonat problematičnost mezidoménových vztahů. Po formální stránce se totiž v našem modelu nic nezmění – pouze relaci $R_i$ budeme vykládat tak, že do ní vstupují dva jevy tehdy, jestliže jejich výskyty spolu pravidelně koincidují. Výsledkem pak bude paradox týkající se pravidelné koincidence jevů, zcela analogický ke kauzálnímu paradoxu.

Činění soudů o povaze vztahů či zákonů mezi doménami je dle našeho názoru záležitostí metafyziky, neboť jakékoli tvrzení o nich, jež by neredukovalo podstatu disjunktnosti obou neostře vymezených domén, by bylo nefalsifikovatelné.[19] Totiž i samotný princip kauzality stojí mimo vědu jako takovou:

[19] Kritérium falsifikovatelnosti navrhuje K. R. Popper jako konvenci k rozlišování mezi vědecko-empirickými a metafyzickými systémy (mezi „metafyzické" systémy by se zde řadila i matematika a logika): empirické systémy lze z principu falsifikovat (tj. ukázat nějakou empirickou evidenci, která je v rozporu s univerzálním tvrzením daného systému), zatímco metafyzické nikoli.

*Víra v kauzalitu je metafyzická. Není to nic než typická metafyzická hypostáze dobře zdůvodněného metodologického pravidla – vědcova rozhodnutí nikdy se nevzdat hledání zákonů* (Popper, 1997).

A právě jedním z projevů tohoto metodologického pravidla je snaha interpretovat mezidoménové vztahy jako vztahy kauzální, a to zmíněným komolením vzájemné interakce dvou či více kauzálních domén do kauzální domény jediné, pokud možno ještě ostře vymezené.

Tolik bychom mohli poznamenat k epistemologickým závěrům plynoucím z kauzálního paradoxu. Emergentní efekt, o němž byla řeč v oddílu 4.2, lze též vykládat jako projev kauzálního paradoxu. Ať je totiž v daném časovém okamžiku vstup emergentního systému jakýkoli, systém se z vnějšího pohledu jaksi „rozhodne", jaký výstup bude indukovat (dalo by se říci, že projevuje určitou „svobodnou vůli"). Jak toto „rozhodování" probíhá, je otázka v zásadě shodná s otázkou po povaze mezidoménových vztahů. Bezespornou, avšak ryze metafyzickou odpovědí může být například tvrzení, že v daném časovém okamžiku výběr aktuálně nastávajících jevů ze všech ekvivalentně příhodných důsledků provede princip čisté náhody, Bůh či nějaký strukturně nadřazený systém.

Emergentní efekt jako projev kauzálního paradoxu je (s přihlédnutím ke vztahu kauzálních domén a teorií) tak vlastně důsledkem nahlížení světa prostřednictvím dvou či více vzájemně nekompatibilních teorií (v širším slova smyslu). To by mohlo na první pohled znamenat, že emergence je opravdu pouhý epifenomén, jak jsme o něm diskutovali z analytického hlediska v předchozím výkladu. Tento zjednodušující názor ale předpokládá, že neznalost[20] pozorovatele, která dle něj zapříčiňuje emergentní efekt, může být poznáváním eliminována (čímž je eliminováno i zdání emergence), přičemž bariéry tohoto poznávání nejsou principiální, ale pouze technické. Za tímto předpokladem se však skrývá představa, jež je v jistém smyslu neoprávněná. Za neznalost je zde považována situace, kdy na jistý úsek reality pohlížíme prostřednictvím dvou či více vzájemně neslučitelných (či velmi těžko slučitelných) teorií, jimž odpovídají různé kauzální domény. „Poznávání" v tomto kontextu pak znamená vytvářet novou teorii, která „spojí" původní teorie neslučitelné. Tím vzniká též nová kauzální doména, jež se rozepne přes původní disjunktní domény, čímž je sloučí a umožní koherentní kauzální výklad všech jevů v nich obsažených, vyhýbajíc se tak kauzálnímu paradoxu a emergentnímu efektu. Jak jsme však ukázali, dojde tím ke zkomolení původního uchopení stavu věcí, čímž nakonec poznáme něco jiného, než jsme nejdříve zamýšleli.

Doménová fragmentace světa je nedílnou součástí lidského vnímání. Emergentní efekt je pak důsledkem této fragmentace. Lidské poznávání v podstatě přeskupuje strukturu rozvrstvení domén, nové domény vytváří či staré opouští, takže emergentní efekt postupně mění svoje nositele, na nichž se ukazuje. Spojení jedné či více teorií[21] nevysvětlí dříve přítomný emergentní efekt, pouze jej opustí a ponechá zcela mimo dosah svého zájmu (spolu s celou řadou jevů, které „epistemicky zaniknou" tímto zkomolením původních kauzálních domén). Navíc tím vznikne nová kauzální doména, díky jejímuž obzoru se objeví nový emergentní efekt. Dalo by se říci, že cílem poznání je tedy

[20] „Neznalost" v tom smyslu, jak byla formulována v diskusi o analytickém pojetí emergence.

[21] „Spojení teorií" ve smyslu vytvoření nové teorie, která vysvětluje jak všechny jevy původních neslučitelných teorií, tak i vztahy mezi jevy napříč těmito teoriemi.

zachycení celého světa v rámci jedné kauzální domény. To by však mělo velmi závažné důsledky v dalších úvahách o emergujícím myšlení.

### 4.5.4 Vztah kauzálního paradoxu a emergujícího myšlení

Vraťme se nyní k zobecněné emergentistické tezi. Dle ní lze mentální stavy a procesy chápat jako emergentní jevy nad třídou kauzálních domén. Již jsme poznali, že kauzální domény (i jejich modely v rámci DTM) jsou úzce spjaty s pohledem pozorujícího subjektu. Emergující myšlení tedy vyvstává nad kauzálními doménami, avšak aby tomu tak mohlo být, musí již nějaký pozorující subjekt předem existovat, aby se k němu mohly tyto domény vztahovat. Pozorující subjekt však sám (jsouc subjektem vědomým) musí vycházet z emergentního efektu. Dostáváme se tímto opět k hermeneutickému kruhu. Můžeme si představit, že jeden z těchto dvou objektů (třída kauzálních domén či pozorující subjekt) byl dán a priori (například Bohem), nebo žádný z těchto objektů nebyl dán a priori, přičemž druhá možnost by mohla být vysvětlena „evolučně" (resp. „vývojově"): vědomá mysl a kauzální domény vznikaly souběžně krok za krokem ve vzájemném zjednávání až do takového komplexního stavu, v jakém o nich mluvíme dnes.

Nicméně i druhé vysvětlení je problematické, neboť aby bylo toto postupné vzájemné zjednávání možné, musí být již ve světě přítomná neostrost, která by umožnila realizaci principu „přirozené indukce"[22] namísto principu klasické neslábnoucí indukce. Není-li totiž nějaký systém v daném časovém okamžiku pozorujícím subjektem, pak jím nebude ani v bezprostředně následujícím časovém okamžiku. Neslábnoucí indukce by tudíž narozdíl od „přirozené indukce" nedovolila pozorujícímu subjektu vzniknout, neboť by nikdy nemohlo dojít k takovému „fázovému přechodu". Neostrost umožňující tuto „přirozenou indukci" je v oboru abstraktních tříd také podložena obzorným řezem, takže dochází k jakési emergenci „druhého řádu", z níž emergence „prvního řádu" emerguje. Hermeneutický kruh je tím rozšířen do „hermeneutické sféry". Můžeme nyní dále analogicky pokračovat a předpokládat emergence stále vyšších řádů (spolu s „hermeneutickými hypersférami"), a to až do nekonečna (zřejmě též přirozeného). Všechny tyto kroky mají společnou jednu základní věc, jíž je přítomnost neostrosti. Má-li tedy mysl (resp. mentální stavy a procesy) být emergentním jevem, musíme přijmout alespoň jeden metafyzický princip, jímž je právě apriorní přítomnost neostrosti ve světě.

Takto chápaná mysl přináší ale zároveň dostatek prostoru pro *svobodnou vůli*, neboť vymaňuje lidské (a možná i umělé) myšlení ze zajetí přímé kauzální determinace dané slepým během procesů v biologických a fyzikálních doménách, z nichž mysl emerguje. Jak jsme již totiž ukázali, emergentní efekt žádnou takovou kauzální determinaci nemá. To, co „rozhoduje" o výsledných projevech emergentních systémů, a tedy i mysli (čili to, co „způsobuje" svobodnou vůli), je skryto v neprostupnosti $\pi$-třídy, z níž na podkladové kauzální domény hledíme, a při snaze o zachycení prostřednictvím jiné kauzální domény (tedy $\sigma$-třídy) to nenávratně mizí (zde bychom mohli poukázat na analogii s Heisenbergovým principem neurčitosti ve fyzice).

Úvahy o možnostech zkonstruování umělého myšlení se také nevyhnou konfrontaci se závěry plynoucími z kauzálního paradoxu. Vytvořit umělé myšlení znamená na

[22] „Přirozená indukce" je zde opět chápána ve smyslu komplementárního principu k přirozenému nekonečnu – viz odstavce o DTM.

základě konstrukčního plánu sestrojit systém, který by duplikoval myšlení přirozené, tedy lidské. V rámci emergentistického paradigmatu by šlo o vytvoření takového prostředí *in silicio*, v němž by díky vzájemné interakci jeho prvků došlo ke vzniku (při pohledu z vnějšku) emergentního efektu, který by odpovídal přirozenému myšlení. Konstrukční plán by tedy měl charakterizovat všechny relevantní kauzální vztahy mezi prvky podkladového prostředí a požadovanými projevy systému jako celku. Jak je však z kauzálního paradoxu zřejmé, přítomnost emergentního efektu vylučuje smysluplné charakterizování takovýchto mezidoménových kauzálních vztahů. V kontextu tohoto nejsilnějšího pojetí umělého myšlení se tudíž domníváme, že umělé myšlení nelze v tomto smyslu zkonstruovat.

Jiným případem však je vytvoření takového umělého emergentního systému, jehož primárním cílem je simulace určitého typu evoluce. Sekundárně se pak v tomto systému mohou objevit emergentní jevy, které by určitému typu myšlení odpovídaly, avšak vztah konstruktéra k těmto jevům je již zcela jiného typu než u prve zmiňovaného případu. Další variantou může být situace, kdy konstruktér začne iterativně vytvářet takovéto emergentní systémy, přičemž bude experimentálně (čili metodou pokus–omyl) měnit jejich parametry (což odpovídá jakési „metaevoluci" systému) do té doby, než se po nesmírném množství iterací v systému objeví nějaký emergentní efekt, jenž by odpovídal přirozenému myšlení. Tehdy by však bylo na pováženou mluvit o systému *umělém*, když většina jeho zajímavých vlastností mu byla dána odjinud než konstruktérem, pro něhož by byla podstata emergentního efektu stejně nepřístupná jako u přirozeného myšlení samotného. Stejná situace by nastala i tehdy, kdyby někdo dokázal buňku po buňce sestavit repliku skutečného člověka. Šlo by pak vůbec o člověka umělého?

Obecně lze říci, že podaří-li se uměle vytvořit emergentní systém, o němž se prohlásí, že explicitně simuluje přirozenou kognitivní funkci či mentální proces a zároveň se svému konstruktérovi zcela nevymkne kontrole, pak přesto dochází k redukci požadovaného jevu (tj. mentálního či kognitivního procesu) na některé jevy podkladové kauzální domény – tj. o požadovaných jevech se řekne, že jsou ekvivalentní s některými jevy domény, v rámci níž je systém zkonstruován. Interpretace výsledku je však zcela v rukou pozorovatele systému – systém sám není interpretantem. Tento fakt však není v souladu s požadavkem přítomným v sémiotické formulaci umělého myšlení, která vychází z pojetí sémiotiky (nauky o symbolech) C. S. Peirce, jenž symboly charakterizuje triadickým popisem, skládajícím se z těchto aspektů: 1) znak, neboli representamen; 2) bezprostřední objekt; 3) interpretant. Základní problém umělé inteligence se dá v tomto kontextu formulovat tak, že je třeba určit, které kroky se mají provést, aby se počítač ocitl v roli interpretanta (Markoš, Kelemen, 2004). Lze to ilustrovat na velice jednoduchém příkladu umělé neuronové sítě, jejímž cílem je rozpoznávat ručně psaná písmena. Rozpoznané písmeno je indikováno aktivací daného neuronu ve výstupní vrstvě sítě, avšak interpretovat, že tento jev je rozpoznáním písmena, rozhodně není v kompetenci sítě, ale jejího uživatele. Stejně tak v potenciálním umělém modelu lidské mysli nemůže přicházet v úvahu, že by aktivace neuronu s názvem „strach" jakkoli odpovídala skutečnému mentálnímu stavu „mít strach". Z těchto skutečností je zřejmé, že sémiotický aspekt hraje v úvahách o přirozeném a umělém myšlení důležitou roli.

Také doménová fragmentace světa je svázána se způsobem, jímž lidé provádějí interpretaci symbolů, které je obklopují. Možný výklad sémiotického přístupu chápe

symbol takovým způsobem, že jeho prostřednictvím interpretant přímo vyděluje bezprostřední objekt z chaosu vjemů okolního světa. V časovém běhu však nelze celou symbolovou triádu chápat staticky – znak, bezprostřední objekt i interpretant jsou v neustálé přeměně, celá triáda se může stát jedním z prvků triády jiného symbolu (tj. celý symbol může z perspektivy jiného symbolu figurovat například jako bezprostřední objekt) a dochází k neustálému a nekonečnému interpretačnímu kroužení. Na základě aktivní snahy, již interpretant projevuje při tomto *zjednávání*, se mu začnou vyjevovat zákonitosti panující mezi jevy okolního světa. Uvnitř těchto zákonitostí vyvstávají z bezprostředních objektů objekty dynamické, které na rozdíl od objektů bezprostředních nejsou ve světě jaksi „pasivně" přítomny, jednoduše svázány poukazem k nějakému symbolu, ale aktivně jsou interpretantem zjednávány v rámci celého sémiotického procesu, aby mu výklad okolního světa „dával smysl". Zákonitosti panující ve světě potom lze v tomto kontextu považovat za *zvyklosti* zavedené dlouhým interpretačním procesem (tradicí).

Na svět tedy Peircova sémiotika nahlíží jako na nikdy nekončící výkladový proces (sémiosis). K sémiotické triádě navíc přistupuje další faktor – zvyk – díky němuž můžeme celou sémiosi *jakoby* pozastavit a v rámci tohoto vymezení času a prostoru definovat významy *přesně*. Proces sémiose však ve skutečnosti běží dál, proto jsou tyto definice přesné pouze v daném zafixovaném výřezu, nikoli ve zjednávaném světě jako takovém. Při jeho poznávání se však chováme, jako by byly definované přesně, a to tak dlouho, dokud trvá zvyk. Poznání vědecké pravdy tak jde ruku v ruce s kulturou, neboť závisí na dlouhodobém úzu celého společenstva.[23]

Aktuální rozložení kauzálních domén tedy zrcadlí průběh sémiose a odpovídá osobním, kulturním i evolučním „zvyklostem". Zachycení tohoto rozložení prostřednictvím polomnožin v rámci DTM v sobě obsahuje i dynamiku sémiotického procesu. Prve zmiňované komolení jevů zastoupených polomnožinami pak odpovídá fixování statických „výřezů" tohoto dynamického procesu. Emergentní efekt je tedy zejména projevem dynamiky nikdy nekončícího výkladového procesu nahlížení světa, jeho různé podoby a přítomnosti se však mění, a to s postupnými změnami zvyklostí daných proměnami společenského úzu, evolučními změnami či osobními postoji.

Netvrdíme, že by z principu nebylo možné sestrojit umělou bytost myšlením stejnou jako člověk. Záměrně se však vyhýbáme nazvat tuto bytost „umělou bytostí s vědomím". Kauzálnímu paradoxu, který by bránil vytvoření „umělé bytosti s vědomím", se totiž „jednoduše" můžeme vyhnout redukcionistickým vyprázdněním mentální domény s tím, že mentální stavy a procesy (strach, láska, bolest, vědomí apod.) nejsou *nic jiného než* jejich koreláty z domény nervové. V důsledku tohoto redukcionismu dojde k postupnému odstranění neostrosti, čímž zanikne zde přítomný emergentní efekt a bude již možné kauzálně popsat vztahy mezi nervovými jevy a redukovanými „mentálními" jevy.

Kupříkladu je-li v našem porozumění chování mravence přítomná neostrost a mezidoménová interakce, je toto chování provázeno emergentním efektem, a mravence tak vnímáme jako intencionální bytost, která projevuje jisté aspekty svobodné vůle (např. „mravenec se *chce* dostat tam a tam", či „mravenec cítí bolest, a proto běží pryč" apod.). Nicméně podaří-li se nám (nyní alespoň v rovině myšlenkového experimentu) redukovat celé mravenčí chování a život do domény jeho nervového fungování, mravenec přestává

[23] Výklad o procesu sémiose zde byl volně převzat z (Markoš, Kelemen, 2004).

být intencionálním tvorem a stává se prostým reaktivním agentem, u něhož nemá příliš smysl hovořit o tom, že „žije".[24]

Podobně většina živých bytostí je „sestavena" z miliard reaktivních agentů a pouze přítomnost neostrosti umožňuje prostřednictvím mezidoménové „paradoxní kauzality" vnímat je jako živé bytosti, nikoli jako množiny miliard reaktivních agentů. Jelikož však toto vnímání vychází z aktuálního rozvržení doménové fragmentace světa, můžeme předpokládat, že se s postupem času bude v rámci nekonečného procesu sémiose měnit.

Z uvedených úvah je patrné, že konstitutivním prvkem vědomí[25] (ať přirozeného, nebo umělého) je „stav věcí", který bychom mohli nazvat *„fundamentální agnosticismus"*, a to nikoli jako pouhý záměrný epistemologický přístup nějaké osoby, ale ústrojnou součást (zjednávaného) vědomě vnímaného světa. Můžeme říci, že vědomé mysli zjednávají sebe samé tím, že „nevidí všechno". Každé vědomí může z principu rozšířit svůj obzor a vysvětlit jevy, které vnímalo předtím, jako něco jiného (tj. prostřednictvím jevů nových, dříve neviděných). Nicméně to, co bylo vnímáno předtím, je pak navždy ztraceno a nemůže být znovu vyvoláno. Pro jednoduchou ilustraci nechť si čtenář představí libovolný dům (tj. ve své mysli vyvolá obraz domu). Má-li již představu domu hotovou, nechť nyní odpoví na otázku, jakou barvu mají okapy tohoto domu. V naprosté většině případů si čtenář bude muset přimyslet k domu ještě okapy, čímž začne pracovat s jinou představou, než původního domu, o němž nebylo možné říci, že by okapy neměl, pouze že jejich (ne)přítomnost byla skrytá za obzorem. Nyní však již čtenář tento experiment zodpovědně zopakovat nemůže, neboť se nelze vrátit k původní představě stejného domu s okapy za obzorem – dokážeme si představit již pouze dům s okapy či dům bez okapů, popř. představu o tom, jaké to je představit si tento stejný dům s okapy za obzorem. Jiným příkladem je představa Boha, která v naší společnosti (s přihlédnutím ke stavu poznání v rámci sémiose, jež v naší společnosti probíhá) již nejspíše nikdy nebude taková jako před tisíci lety. Můžeme se pouze snažit představit si, jak si tehdejší člověk Boha (či bohy) představoval.

Otázkou zůstává, zda takovéto neustálé sémiotické rozšiřování obzoru může být nekonečné, nebo se jednou zastaví (resp. zkolabuje). V případě, že by bylo přirozeně nekonečné (neboť to je právě ten typ nekonečna, který se v přirozeném reálném světě nachází), pak by nejspíše mohlo dojít k jeho kolapsu, a to v situaci, kdy by nastalo úplné vyprázdnění mentální domény a mysl by byla definitivně vysvětlena pouze v pojmech nastíněného redukcionismu. Chtělo by se říci, že tato změna by de facto neměla vliv na naše vědomí, které je nám (jako druhu) jednou dané, bezprostředně je cítíme, a tudíž bychom pouze poznali jeho podstatu, aniž by ji to ovlivnilo. Musíme si však uvědomit, že naše vědomí je emergentním efektem nad nesmírně složitým souborem vzájemně interagujících kauzálních domén, jejichž rozvržení je dáno právě sémiotickým procesem, který sám je zpětně ovlivňován činností našeho vědomí (resp. našich vědomí – v plurálu). Vliv poznávání (ve smyslu reduktivním) vědomí v tomto procesu vzájemného ovlivňování a tvarování vede, přísně vzato, k mizení vědomí samotného, tj. zjednávání světa bez vědomí jednotlivců, jak je známe dnes. Samozřejmě jde o časově velice dlouhodobý proces – nemůže se stát, že by se dotkl života jednotlivců – avšak máme za

[24] Snad s výjimkou vymezení „živého systému" jako systému autopoietického, tj. systému se specifickým případem rovnováhy sítě procesů, která průběžně regeneruje sama sebe (resp. svoji vlastní strukturu) a zároveň konstituuje systém jako konkrétní individuum v prostoru (Maturana, Varela, 1980).

[25] Od tohoto místa již záměrně užíváme termínu „vědomí" místo původního „myšlení", byť tím klademe větší požadavky na jeho intuitivní uchopení.

toho, že již dnes člověk vnímá své vlastní vědomí jinak než například před dvaceti generacemi. Důkazem může být třeba pocit jakési nepatřičnosti, máme-li se v dnešní době vyjadřovat o vědomí v rámci vědeckém diskursu – ve vyostřeném případě dokonce předstíráme, že žádné vědomí ani nemáme.

Společnost (a tudíž i její kulturní úzus) je v procesu sémiose ovlivňována a směrována svými vlastními předchozími interpretačními procesy. Působením otupujícího vlivu zjednodušujícího a zavádějícího způsobu přenosu informací v dnešním světě (zejména prostřednictvím současných sdělovacích prostředků, jejichž styl lze parafrázovat prohlášením typu: „Američtí a japonští vědci *objevili* robota, který dokáže mít rád svého majitele.") začíná již množství lidí žít tak, že kupříkladu láska pro ně není *nic než* „vzbouření hormonů" či navyklé vzorce chování, nebo že za činy jednotlivce nemůže on sám, nýbrž pouze jeho psychika (či dokonce mozek). Všechny tyto nekonečně malé (opět v přirozeném smyslu) kroky se však z vývojového hlediska podílejí na postupných kvalitativních změnách vědomí jako takového.

Tudíž pokud by se nám (jako lidstvu) nakonec přece jen podařilo vytvořit *umělé vědomí*, nebylo by to pro nás již stejně žádné *vědomí*, neboť by už nebylo žádné „my" a ani žádné přirozené vědomí, které by dokázalo rozhodnout, zda to, co jsme vytvořili, je opravdu to správné *Umělé Vědomí*. Nebyli bychom totiž *nic jiného než* reaktivní agenti, neboť poslední kauzální paradox by byl odstraněn a s ním i poslední emergence, přinášející nám náš život, jak jej známe nyní.

## Literatura


Brooks R.: *Cambrian Intelligence: The Early History of the New AI*. MIT Press, Cambridge, MA, 1999.

Cantor G.: Gesammelte Abhandlungen mathematischen und philosophischen Inhalts. Berlín 1932, (duhé vydání G. Olms, Hildesheim 1962).

Fiala J.: *Analytická filosofie – úvod*. OPS, Plzeň – Nymburk 2006.

Havel I. M.: *Artificial thought and emergent mind*. In: Proceedings IJCAI'93, Morgan Kaufman Profesional Book Center, Denver, CO, USA, 1993, s. 758–766.

Havel I. M.: *Causal domain and emergent rationality*. In: B. Smith, B. Broguard (editoři): Rationality and Irrationality. Proceedings of the 23rd International Wittgenstein Symposium, Kirchberg, Rakousko, 2000.

Havel I. M.: *Přirozené a umělé myšlení jako filozofický problém*. In: Mařík V. a kol.: Umělá inteligence (3). Academia, Praha 2001, s. 41, 43, 69.

Chalmers D. J.: *The Conscious Mind: In Search of a Fundamental Theory*. Oxford University Press, Oxford 1996.

Markoš A., Kelemen J.: *Berušky, andělé a stroje*. Dokořán, Praha 2004.

Maturana H. R., Varela F. J.: *Autopoiesis and Cognition*. D. Reidel, Dordrecht, Holandsko, 1980.

Popper K. R.: *Logika vědeckého bádání*. Oikoymenh, Praha 1997, s. 42, 43, 266.

Pstružina K.: *Svět poznávání (K filozofickým základům kognitivní vědy)*. Nakladatelství Olomouc, Olomouc 1998.

Thagard P.: *Úvod do kognitivní vědy (Mysl a myšlení)*. Portál, Praha 2001.

Vopěnka P.: *Úvod do matematiky v alternatívnej teórii množín*. Alfa, Bratislava 1989.

Vopěnka P.: *Úhelný kámen Evropské vzdělanosti a moci*. Práh, Praha 2000.

Vopěnka P.: *Meditace o základech vědy*. Práh, Praha 2001, s. 11.

Vopěnka P.: *Dynamická teorie množství* (připravováno). OPS, Plzeň–Nymburk, vyjde 2007.

# 5 Virtuální lidé

Virtuální lidé jsou zatím zřejmě posledním přírůstkem do rodiny agentů, animatů, holonů a dalších člověkem vytvořených systémů, na které se můžeme dívat jako na entity vykonávající zadané úkoly samostatně a v jistém smyslu inteligentně. Co si představíme, když se řekne virtuální člověk? V jakých typech aplikací je máme hledat? A čím se vlastně liší od svých kolegů agentů?

Cílem této kapitoly je čtenáře s problematikou virtuálních lidí seznámit z pozic umělé inteligence. Začneme předběžným vymezením pojmu *virtuální člověk* a ukázkou konkrétní dvojice virtuálních lidí. Poté načrtneme schéma jejich řízení. Problému řízení se pak budeme věnovat důkladněji. Následně budeme podrobněji studovat aplikace s virtuálními lidmi, přičemž představíme tři ukázkové projekty řešené na MFF UK v Praze. Nakonec se vrátíme k pojmu virtuální člověk, abychom ho prozkoumali zevrubněji. Budeme diskutovat o příbuznosti termínu s pojmem autonomní agent, čímž navážeme na kapitoly z předchozích svazků, kde byly agenti a multiagentní systémy studovány (Štěpánková a kol., 1997), (Mařík a kol., 2001), (Kelemen, 2001), (Mařík a kol., 2004). Zaměříme se také na otázky, které se váží k výzkumu virtuálních lidí, ale mají přesah i mimo oblast umělé inteligence – zejména do práva a etiky.

## 5.1 Virtuální člověk

Předběžně si za pojmem *virtuální člověk* představme počítačový program, který *imituje* lidské chování ve *virtuálním prostředí* a který disponuje *virtuálním tělem*, prostřednictvím nějž vnímá a jedná (tuto předběžnou představu zpřesníme v oddílu 5.5). Virtuálním prostředím se rozumí určitá abstrakce našeho přirozeného reálného světa – většinou půjde o umělý 3D svět, jaký známe například z akčních počítačových her, není to ale nezbytné. V této kapitole budeme z výkladových důvodů často používat i příklady z jednodušších, diskrétních 2D světů. Virtuální svět bývá obvykle řízen samostatným programem nebo komponentou – *simulátorem světa*. Virtuální světy i lidé v nich bývají často zobrazeni graficky.

Klíčové na virtuálním člověku je to, že chování imituje – nejedná se o biologicky nebo psychologicky plausibilní model. O imitaci chování se stará modul, který budeme nazývat *umělá* (nebo též *virtuální*) *mysl*. Základními problémy, jež s programováním umělé mysli souvisí, jsou reprezentování znalostí o tom, co a jak může virtuální postava provádět, a efektivní generování konkrétních akcí na základě těchto reprezentací. Dalšími dílčími problémy jsou například modelování percepce, vyjadřování emocí či komunikace s ostatními virtuálními postavami a lidskými uživateli. Naprogramovat

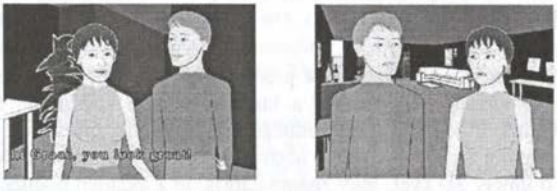
virtuálního člověka je komplexní úkol vyžadující znalosti ze softwarového inženýrství, počítačové grafiky, lingvistiky, umělé inteligence a často i psychologie. V této kapitole nicméně zůstaneme na poli umělé inteligence a budeme se věnovat zejména problému generování akcí.

### 5.1.1 Trip a Grace: virtuální lidé v akci

Podívejme se na obrázek 5.1 (též barev. příl.). Vidíme dvojici virtuálních lidí: Tripa a Grace. Jsou to manželé, kteří „bydlí" ve virtuálním apartmánu nad městem. Jednoho dne pozvou na skleničku svého přítele ze studií. Během večírku ovšem vyjde najevo, že jejich manželství se rozpadá a příteli nezbude, než jim pomoci dát ho znovu dohromady. Přítel je postava, za kterou jedná uživatel.

Aplikace se jmenuje Façade, z větší části byla vyvinutá na Carnegie Mellon University a vydaná v roce 2005 (ProcArts, 2005). Ovšem pozor! Nejedná se o běžnou počítačovou hru. Předně celý večírek trvá asi dvacet minut a rozvíjí se během něj příběh strukturovaný jako jednoaktové drama. Příběhy počítačových her podobu dramatu nemívají. Zadruhé, události při každém spuštění běží trochu jinak. Celý večírek si musíte přehrát pětkrát, šestkrát, než projdete všechny možné větve příběhu. To také nebývá v počítačových hrách časté. Za třetí, s postavami lze komunikovat v angličtině a ony anglicky odpovídají. Mnoha věcem sice nerozumějí a všechny věty jsou předmluveny (až na jméno přítele, které je dosazováno za běhu), v lidech to ale přesto vesměs vzbuzuje silný dojem. A konečně – atmosféra během večírku by se dala krájet – postavy svým výrazem ve tváři, postojem, chováním, intonací... vyjadřují emoce.

Co je Façade zač? Jedná se o *virtuální drama* („virtual drama"), zcela novou kategorii multimediálních aplikací, jejichž možnosti se teprve začínají zkoumat, např. (ICVS, 2001–2005), (IVA, 1999–2006). Virtuální dramata jsou typickými aplikacemi s virtuálními lidmi. Dalšími příklady jsou výukové počítačové hry, některé akční a dobrodružné hry, terapeutické aplikace nebo i film.



Obr. 5.1 Grafické rozhraní nám umožňuje nahlížet do světa Tripa a Grace – manželů z virtuálního dramatu Façade. Na obrázku vlevo jsou oba, Trip i Grace, spokojení. Vpravo už poněkud méně. Převzato ze stránek ProcArts (2005), kde je možné si software zdarma stáhnout.

## 5.2 Řízení virtuálních lidí

### 5.2.1 Architektura virtuálního člověka

Bylo řečeno, že každý virtuální člověk existuje ve virtuálním světě. Z toho plynou určité podstatné důsledky.

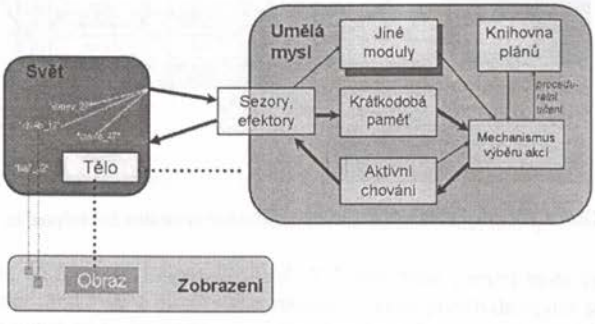
Za prvé, virtuální člověk musí mít *tělo*. Tělo tvoří rozhraní do světa; prostřednictvím něj vnímá a jedná. Z „tělesnosti" virtuálního člověka například plyne, že člověk musí dodržovat pravidla světa, jakési „fyzikální zákony" (neprocházet zdmi apod.).

Za druhé, aby mohl virtuální člověk ve světě jednat, musí simulátor světa rozeznávat určitá *funkční primitiva – atomické akce*. Jejich množina je obvykle předem daná programátorem aplikace a po dobu simulace se nemění. Co přesně je atomickou akcí, závisí na úrovni abstrakce přijaté v dané aplikaci a na typu světa. V oblasti virtuálních lidí jde většinou o děj trvající několik desítek milisekund až jednotek sekund (typickou akcí je „krok" resp. „pohyb o $x$ metrů ve směru $z$"). Umělá mysl virtuálního člověka pak generuje sekvence těchto akcí.

Za třetí, virtuální člověk musí také vnímat. Technicky řečeno mu simulátor světa musí posílat, co vnímá, prostřednictvím *senzorických primitiv*. Senzorická primitiva opět závisí na úrovni abstrakce a na typu světa. Ve 2D světě bude běžným primitivem „vidíš objekt na pozici $x$, $y$", zatímco v 3D světě akční hry se setkáme i s informacemi typu „paprsek v tomto směru narazí za 150 cm na zeď".

Typická architektura virtuálního člověka je zobrazena na obrázku 5.2. Je obdobná klasické architektuře autonomního softwarového agenta či řídicího systému robota, viz též (Kelemen, 2001). Jedná se o architekturu, která vychází z *funkcionální dekompozice*. To znamená, že jednotlivé moduly plní funkce, které přisuzujeme částem živého člověka, typicky končetinám, smyslovým orgánům a mozku. Například *senzory* se starají o vnímání (přijímání a částečné zpracování senzorických primitiv) a *efektory* o jednání (posílání atomických akcí). Pochopitelně jde o metaforu.

Jaké komponenty vyjma efektorů a senzorů vidíme na obrázku? *Knihovna plánů* obsahuje předdefinované znalosti o tom, co a jak má virtuální člověk provádět[1].
V *krátkodobé paměti* si virtuální člověk pamatuje cokoli, co se vztahuje k plnění aktuálních úkolů. *Mechanismus* (nebo *algoritmus*) *výběru akcí* se stará o generování akcí na základě znalostí uložených v knihovně plánů a v krátkodobé paměti. Komponentu




Obr. 5.2 Typická architektura aplikace. Vidíme vzájemnou interakci mezi virtuálním člověkem a jeho prostředím. Vše zobrazuje GUI.

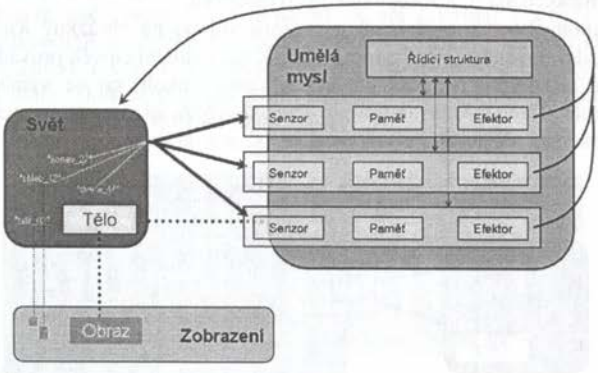
[1] Knihovně plánů se někdy říká *dlouhodobá paměť* („long-term memory") nebo *procedurální paměť* – například v architektuře Soar (Newell, 1990). Z psychologického hlediska ani jedno označení není přesné.

*aktivní chování* zatím přeskočme. Co se rozumí „jiným modulem"? Například u Tripa a Grace půjde o komponenty zajišťující projevování emocí a komunikaci ve zjednodušené variantě angličtiny. Nedílnou součástí aplikace je také grafické rozhraní (GUI).

Typicky pak jedná virtuální člověk ve stále se opakujícím cyklu: 1) přijme podněty od prostředí skrze senzory, 2) zpracuje podněty (například uloží je do paměti) 3) vybere následující akci, 4) skrze efektory „odešle" své rozhodnutí simulátoru světa.

Existují i jiné architektury. Za zmínku stojí především architektura založená na *behaviorální dekompozici*. Při ní dekomponujeme člověka na *chování* („behavioral modules"). Každé chování je abstraktní modul zajišťující určitý okruh chování, včetně vnímání a vykonávání atomických akcí, tedy zahrnuje v sobě vlastní senzory, efektory, paměť atd. Moduly jsou na sobě nezávislé. Výhoda je zřejmá: moduly umožňují implementovat doménově specifické prvky (paměť pro situace, v nichž se Trip a Grace hádají, by mohla pracovat s jinou reprezentací než paměť pro vaření). Architektura navíc umožňuje relativně jednoduše virtuální lidi rozšiřovat – přidáme nový modul a máme nové chování. Ve funkcionální architektuře znamená přidání nového chování nutnost modifikovat všechny moduly. Slabým místem behaviorální dekompozice je fakt, že jednotlivé moduly spolu musí nějak interagovat (Trip nemůže zároveň nalévat víno a telefonovat). Potřebujeme tedy centrální složku, která je nějakým způsobem řídí.[2]

V dalším textu budeme vycházet z funkcionální architektury.




Obr. 5.3 Virtuální člověk sestavený podle behaviorální dekompozice.

[2] Behaviorální architektury vychází z původních prací Brookse (Brooks, 1986), (Brooks, 1991), nověji se jim věnuje například (Bryson, 2001), (Bryson, 2003). V Brooksově architektuře centrální prvek není, což činí integraci více modulů dosti komplikovanou. Bryson chápe behaviorální moduly jako objekty ve smyslu objektově orientovaného programování, přičemž senzorickým primitivům a atomickým akcím odpovídají, zjednodušeně řečeno, metody objektů, které centrální složka volá během rozhodování o tom, co má být provedeno v dalším kole, a při následném spouštění akcí.

### 5.2.2 Rozhodovací pravidla

Obraťme nyní naši pozornost k tomu jak virtuální lidi řídit. Otázka má dvě polohy: jak reprezentovat znalosti v knihovně plánů a jak realizovat mechanismus výběru akcí.

Snad nejjednodušší reprezentace je založena na *rozhodovacích pravidlech* („decision rules"). Jedná se o techniku, která původně pochází z oblasti expertních systémů. Obecně byla v této knize představena v (Zdráhal, 1993). Zjednodušeně řečeno, rozhodovací pravidlo (nebo if-then pravidlo, reaktivní pravidlo, produkční pravidlo) je pravidlo ve tvaru:

```
if podmínka then příkaz
```

V kontextu virtuálních lidí lze dělit příkazy na *vnější* (tedy funkční primitiva – atomické akce) a *vnitřní* (modifikující pouze vnitřní stav člověka – např. něco si zapamatuj). Podmínky nějakým způsobem testují informace přijaté v podobě senzorických primitiv. Zpracování primitiv do podoby, na niž se můžeme v podmínkách dotazovat, provádí senzory. Podmínkou může být například logická formule, pokud senzory vytváří v paměti na základě podnětů symbolickou reprezentaci světa. Může jít ale i o doménově specifický test, dotazy na vnitřní stav virtuálního člověka aj.

V nejjednodušší formě jsou podmínky booleovské a příkaz buď proveden je, nebo není.

Použití rozhodovacích pravidel si ukážeme na příkladu virtuálního houbaře. Co chceme, aby houbař dělal? Chceme, aby ráno vyrazil z chalupy s prázdným košíkem, bloumal po lese a sebral houbu, když na nějakou narazí. Dále chceme, aby se pokud možno vyhýbal stromům a nezakopával o pařezy a aby se vrátil, když má košík plný, ale nejpozději kolem poledne (i když ho plný nemá).

Představme si nyní, že máme jednoduchý 2D simulátor virtuálního světa, do kterého můžeme nahrát virtuální les plný hub a stromů a také našeho virtuálního houbaře. Jak popíšeme jeho chování pomocí rozhodovacích pravidel? Příklad uvádíme na obr. 5.4.

```
1. if vidis-prekazku then zmen-smer
2. if doma and sbiram then jdi_k_lesu
3. if doma and NOT sbiram then poloz-kosik and END
4. if kos-plny-hub and sbiram then set Sbiram = FALSE
5. if vidis-houbu and sbiram then seber-tu-houbu
6. if 11:30 and sbiram then set Sbiram = FALSE
7. if sbiram then pohybuj-se-nahodne
8. if NOT sbiram then pohybuj-se-smerem-domu
```

Obr. 5.4 Schematický zápis toho, jak má houbař sbírat houby.

Toto schéma uložíme do knihovny plánů. V krátkodobé paměti bude zaznamenána hodnota proměnné `Sbiram` a obraz světa rekonstruovaný prostřednictvím senzorických primitiv (nikoli nutně historické záznamy, ale aktuální údaje – množství hub v košíku, seznam viditelných překážek apod.). Fakta ocitající se v krátkodobé paměti na základě vnějších vstupů budou testována prostřednictvím: `vidis-prekazku`, `vidis-houbu`, `doma`, `kos-plny-hub`. Testy jiného typu jsou: `sbiram` a `11:30`. Všechny příkazy vyjma nastavování proměnné `Sbiram` povedou k vygenerování atomické akce nebo sekvence atomických akcí.

Houbařův algoritmus výběru akcí pak bude pracovat tak, že bude neustále ohodnocovat podmínky všech pravidel. Všimněte si, že pravidla jsou postavena tak, aby vždy platilo alespoň jedno. Pokud nějaké pravidlo platí, provedeme jeho akci.

Může se stát, že by platilo více pravidel naráz? Ano, například pravidla 1, 4 a 7. V takovém případě přijde ke slovu *mechanismus pro řešení konfliktu* mezi pravidly („conflict resolution"). Nejjednodušší je pravidla ostře uspořádat a dávat přednost nejvyššímu pravidlu z těch, co v dané chvíli platí. Čísla na obrázku 5.4 tak můžeme chápat jako priority, nižší číslo znamená vyšší prioritu. Víte, co se stane, když houbař uvidí zároveň překážku a houbu, přičemž už bude 11:45?

Proč algoritmus neustále ohodnocuje ty samé podmínky? Je tomu tak proto, že prostředí kolem houbaře se dynamicky mění – v kterémkoli okamžiku může nastat cokoli (nebo téměř cokoli) a my nedokážeme dopředu dobře předvídat co. Houbař může například při chůzi k houbě, tedy vykonávání příkazu `seber-tu-houbu`, narazit na nový pařez (což by mělo vykonávání příkazu `seber-tu-houbu` přerušit).[3]

### 5.2.3 Reaktivní plánování

Způsob řízení popsaný v minulé sekci spadá mezi techniky *reaktivního plánování* („reactive planning"). Obecně jde o techniky, které v každém kroku určují *právě následující akci*, v závislosti na aktuálním kontextu a popisu chování. V podstatě tedy počítají funkci:

$$S \times P \rightarrow A, \tag{5.1}$$

kde $S$ je množina všech možných vnitřních stavů (včetně paměti), $P$ je množina všech možných vjemů a $A$ je množina všech akcí, které připadají v úvahu. V tomto kontextu termín *reaktivní* znamená to, že v každém kroku se generuje právě jedna akce. Termín *plánování* souvisí s tím, že funkce bývá realizována pomocí procházení předem daných schémat (například toho z obr. 5.4), jimž se říká *plány*[4]. Tím se vysvětluje původ pojmu *knihovna plánů* z obr. 5.2. Běžně jsou plány fixní – virtuální bytost se neučí novému chování. Dodejme, že reaktivně plánovací techniky se nepoužívají pouze v oblasti virtuálních lidí, ale i u autonomních agentů a v robotice.

Jeden způsob reprezentace plánů je právě pomocí rozhodovacích pravidel uspořádaných podle priorit. Zaručuje nám ale to, že vždy platí alespoň jedno pravidlo, že se houbař bude chovat rozumě? Jinými slovy má tato reprezentace nějaká omezení?

Zmiňme se jen o těch nejzákladnějších. Zaprvé, pokud houbař zabloudí do nepřehledného terénu, není moc jasné, jestli ho pravidlo 1 opět vyvede. Na hledání cesty

---

[3] V počítačových hrách je frekvence ohodnocování pravidel typicky v řádu jednotek až desítek Hz. Existují implementační triky jak ohodnocování pravidel zrychlit. Jednak si můžeme pamatovat výsledky z minulého kola, jednak lze využít mechanismu „event-driven", kdy budeme ohodnocovat určité pravidlo pouze tehdy, pokud nastane nějaká událost (u níž se očekává, že by mohla platnost pravidla změnit). V prvním případě můžeme při logické reprezentaci faktů v krátkodobé paměti použít například algoritmu Rete známého v oblasti expertních systémů (Forgy, 1982). V druhém případě budeme typicky stavět aplikaci podle architektury využívající návrhového vzoru Observer, např. (Gamma, 1995).

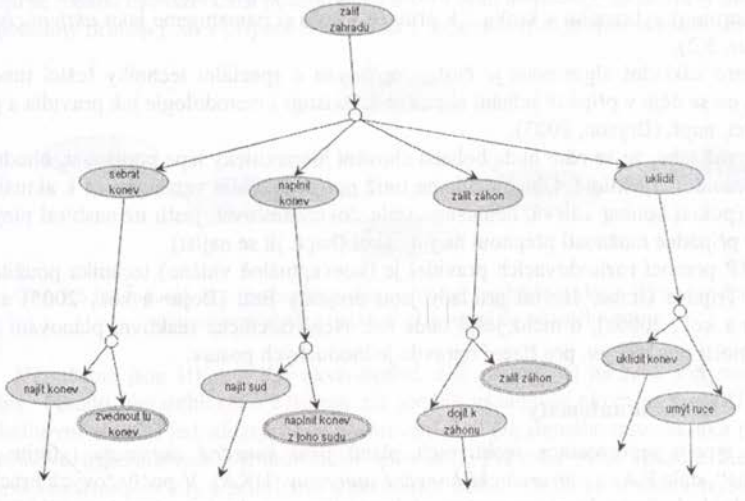
[4] Plánem se zde skutečně rozumí předdefinovaná struktura, nikoli sekvence akcí vygenerovaná na základě popisu domény jako v případě klasického plánování, o kterém se lze dočíst v prvním dílu této knihy (Lažanský, 1993). V anglické literatuře lze k pojmu reaktivní plán někdy nalézt doplňující termín „look-up structure".

a vyhýbání se překážkám se obecně používají speciální techniky – přehledově se jim budeme věnovat v odstavci 5.2.7. Zadruhé, když bude chování houbaře bohatší, bude stále výhodné, aby bylo popsáno řekněme stovkou ostře uspořádaných pravidel? Zřejmě ne – na alternativní reprezentace se podíváme v odstavcích 5.2.4 a 5.2.5. Zatřetí, houbař se ve stejných situacích bude chovat stále stejně – čili jeho chování bude predikovatelné. Tato vlastnost většinou není výhodná – postava působí nevěrohodně. V odstavci 5.2.6 se podíváme, jak to napravit.

### 5.2.4 Rozhodovací stromy

Pokud má virtuální člověk řešit více úkolů, je výhodné popisovat jeho chování hierarchicky – úkoly jako skládající se z podúkolů, ty ještě z menších podúkolů a tak dále. Tím získáme variantu reaktivního plánování nazývanou *hierarchické reaktivní plánování* (HRP).

Reprezentovat plány hierarchicky můžeme opět pomocí rozhodovacích pravidel. Tentokrát ovšem konsekventem pravidla bude moci být nejen příkaz, ale i odkaz na další sadu pravidel. Algoritmus bude pracovat tak, že ohodnotí hlavní pravidla (ta, na která nevede žádný odkaz), z nich vybere jedno, ohodnotí jeho „podpravidla", z nich opět jedno vybere – a tak bude pokračovat dál, dokud nenarazí na pravidlo, jež obsahuje příkaz. Popsaný mechanismus si můžeme představit jako procházení *rozhodovacího stromu* (viz obr. 5.5). Každý uzel stromu je představován sadou pravidel. Strom procházíme shora dolů a při návštěvě každého uzlu ohodnocujeme všechna pravidla uzlu,




Obr. 5.5 Představme si, že náš houbař umí i zalévat. Obrázek ukazuje rozhodovací strom pro tento úkol. Příkazy jsou znázorněny dvojitým rámečkem. Zde všechny příkazy generují atomické akce. Pozor – záhonů může být více a během jejich zalévání se konev může vyprázdnit. S tím si ale rozhodovací pravidla poradí – během vykonávání plánu může jedno pravidlo začít platit několikrát.

přičemž vybíráme vždy právě jedno, které určí větev, po které se spustíme o hladinu níž. V listech budou příkazy.[5]

Formalizaci tohoto konceptu podává například (Bryson, 2001). Zde předkládáme zjednodušenou verzi. *Knihovnou chování* virtuálního člověka nazveme dvojici $\langle f, Zrp \rangle$, kde $Zrp$ je množina základních reaktivních plánů pro jednotlivé hlavní úkoly a $f$ je funkce, která určuje, jaký hlavní úkol má být prováděn v tom kterém okamžiku (vždy může být aktivní pouze jeden). *Základní reaktivní plán* (ZRP) je množina pravidel, přičemž pravidlem rozumíme trojici $\langle p, c, A \rangle$, kde $p$ je priorita, $c$ podmínka a $A$ buď příkaz, nebo podřízený ZRP (tedy další množina trojic). Priority *v rámci* jednoho ZRP jsou ostře uspořádané. Prvky $Zrp$ jsou tedy rozhodovací stromy pro jednotlivé hlavní úkoly.

Algoritmus výběru akcí pak funguje takto:

**Vstup**: knihovna $\langle f, Zrp \rangle$ (5.2)

1) $zrp \leftarrow$ vyber z množiny $Zrp$ pomocí $f$ jeden konkrétní ZRP
2) $A \leftarrow$ příkaz toho pravidla ze $zrp$, jehož podmínka $c$ platí a které má zároveň nejvyšší prioritu ze všech platných pravidel v rámci $zrp$
3) **if** $A$ je příkaz **then** vykonej $A$ a **end**
4) **else**
   4.1) $zrp \leftarrow A$
   4.2) jdi na krok 2

Algoritmus provádíme znovu a znovu v každém kroku simulace. Cestu od kořene ZRP (stromu) vybraného v kroku 1 k příkazu v listu si pamatujeme jako *aktivní chování* (viz obr. 5.2).

Tento základní algoritmus je často doplňován o speciální techniky řešící timeouty úkolů, co se děje v případě selhání akce apod. Existují i metodologie jak pravidla a plány vytvářet, např. (Bryson, 2003).

Kromě toho, že se nám bude bohatší chování hierarchicky lépe popisovat, ohodnocování pravidel zrychlíme. Ohodnocujeme totiž pouze pravidla vztahující se k aktuálnímu úkolu (pokud houbař zalévá, nemusíme stále dokola testovat, jestli už nasbíral plný koš hub) a případné možnosti přepnout na jiný úkol (např. jít se najíst).

HRP pomocí rozhodovacích pravidel je (konceptuálně viděno) technika použitá pro řízení Tripa a Grace. Jinými příklady jsou projekty Enti (Bojar a kol., 2005) a IVE (Brom a kol., 2006a), o nichž ještě bude řeč. Nehierarchické reaktivní plánování se ve skutečnosti používá jen pro řízení opravdu jednoduchých postav.

### 5.2.5 Konečné automaty

Jiným typem reprezentace reaktivních plánů jsou *konečné automaty* („finite state machine", dále KA) a *hierarchické konečné automaty* (HKA). V počítačových hrách jde o standard *de facto*, který se možná už brzy stane standardem skutečným (IGDA-AI, 2002–2006).

[5] Termín *rozhodovací strom* zde používáme v mírně posunutém významu oproti jeho „klasickému" použití, např. (Kubát, 1993). Rozhodovací stromy pro popis chování například bývají vytvářeny návrhářem ručně, nejsou budovány automaticky pomocí mechanismů strojového učení.

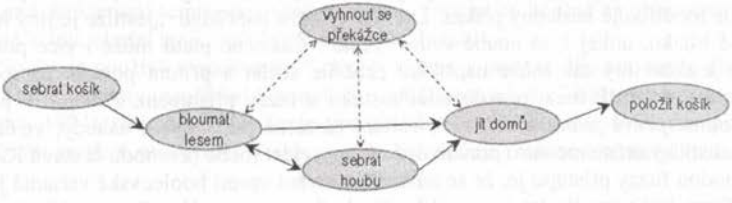
Typický deterministický KA[6] použitý pro řízení virtuálních lidí sestává z množiny *stavů* a *přechodů mezi stavy*. V každém okamžiku je aktivní jeden stav automatu, přičemž řídicí algoritmus testuje, jestli nelze přejít do jiného stavu. V jednom okamžiku může být platných více přechodů (z jednoho stavu), přičemž konflikt mezi nimi lze opět řešit například prioritami. Přechody lze chápat jako rozhodovací pravidla; budou mít následující formu:

```
if jsem-ve-stavu-X a platí-podmínka then aktivuj-nový-stav
```

Máme dvě možnosti, jak pomocí KA generovat akce: buď můžeme příkazy asociovat (1) se stavy, nebo (2) s přechody. Příkazy mohou opět generovat atomické akce nebo měnit vnitřní stav postavy. Zvláště ve variantě (1) často příkazy negenerují jen jednu atomickou resp. vnitřní akci, ale celou sekvenci. V takovém případě se většinou hovoří o tom, že se stavem je asociovaný *skript*, tedy podprogram zapsaný v procedurálním jazyce (který může obsahovat cykly, podmínky apod.).

V případě (1) se opakovaně provádí příkaz aktivního stavu, dokud není aktivován jiný stav. Pokud je aktivován jiný stav během provádění příkazu, prováděný příkaz se přeruší. V případě (2) se příkaz provede jednou při přechodu (lze přecházet i ze stavu do toho samého stavu). Příklad jednoduchého konečného automatu je na obr. 5.6.

Pokud je chování virtuální postavy komplexní, vede to většinou k tomu, že automat má mnoho stavů, nebo že jsou skripty komplikované. V takovém případě bývá výhodné chování hierarchicky dekomponovat a použít HKA. Struktura HKA je tvořena z hlavního konečného automatu, do jehož stavů jsou „vnořeny" podautomaty. Ve stavech podautomatů se mohou nacházet další podautomaty nebo s nimi mohou být asociovány příkazy. Jde o podobný princip jako v případě hierarchicky uspořádaných rozhodovacích pravidel.




Obr. 5.6 Jednoduchý KA pro houbaře. Stav vyhnout se překážce slouží jako přerušení – po vyhnutí se překážce se řízení vždy vrací do minulého stavu.

Výpočetně jsou HKA a KA ekvivalentní, čili mohou být na sebe vzájemně převáděny. Použití hierarchického přístupu ale jednak usnadňuje návrh a pozdější orientaci v knihovně plánů a jednak zrychluje generování akcí (ze stejného důvodu jako při použití stromově uspořádaných rozhodovacích pravidel). Převodu mezi HKA a hierarchicky uspořádanými pravidly s prioritami a porovnání obou technik se věnuje (Bryson, 2004). V krátkosti lze říci, že pravidla vychází ze srovnání lépe tehdy, když máme v automatu přechody mezi (téměř) všemi stavy navzájem.

[6] V kontextu virtuálních lidí se konečné automaty většinou pojímají mírně odlišně od klasických automatů Moorea a Mealyho.

Automaty se používají zejména v oblasti počítačových her. Příklad postav řízených KA, popř. HKA, jsou ve (van Waveren, 2001), (Isla, 2005). Specifickou aplikací virtuálních lidí je filmový průmysl. Virtuální lidé zde mohou být použiti jako virtuální komparsisté, a tak nahradit skutečné komparsisty. Příkladem nástroje, kde lze pro vytváření 3D filmových scén specifikovat chování virtuálních aktorů pomocí HKA, je produkt Softimage/Behavior (AvidTech, 2006).

### 5.2.6 Fuzzy a pravděpodobnostní přístup

Přístupy, v nichž se pracuje s booleovskými rozhodovacími pravidly, lze rozšířit o pravděpodobnostní či fuzzy variantu. Vznikají tak fuzzy rozhodovací stromy, fuzzy konečné automaty, pravděpodobnostní konečné automaty a podobně.

Pravděpodobnostní přístup je dobrý všude tam, kde chceme, aby virtuální člověk nejednal ve stejné situaci pokaždé stejně. Jedná se o dosti rozšířenou techniku. Existuje několik míst, kudy pravděpodobnost do systému zavést. Jednou z nich je mechanismus pro řešení konfliktu mezi pravidly, resp. přechody – není snad obtížné si představit, že mechanismus se nemusí rozhodovat jen na základě priorit, ale též na základě náhody. Pokud by chtěl houbař ráno zalévat i jít na houby, „hození si korunou" může být dobrá volba jak konflikt vyřešit. Jinou možností je zavést pravděpodobnost až na úrovni příkazů – s každým pravidlem resp. stavem můžeme asociovat více příkazů a konkrétní z nich pak může být vybírán náhodně. Například pokud houbař uvidí malou překážku, někdy se jí může vyhnout a jindy ji přeskočit. Další možností je modifikovat na základě pravděpodobnosti platnost podmínek pravidel. Houbař by se mohl například začít vracet domů „plus minus" v 11.30.

Při fuzzy variantně (Zadeh, 1965) jsou podmínky pravidel (resp. přechody) dány fuzzy formulemi a míra jejich pravdivosti (to je číslo z $\langle 0, 1 \rangle$) určuje sílu platnosti pravidla. Tato síla modifikuje následný příkaz. Lze tak vyjádřit například: „jestliže je jiný houbař k houbě blízko, utíkej k té houbě velmi rychle". Částečně platit může i více pravidel, postava z akční hry tak může například „zběsile střílet a přitom pomalu ustupovat". Uvědomme si rozdíl mezi pravděpodobnostním a fuzzy přístupem. Zatímco v prvním vždy volíme *právě jednu* z možných variant částečně pod vlivem náhody, ve druhém deterministicky určujeme míru pravdivosti *více* pravidel (nebo přechodů či stavů KA).

Výhodou fuzzy přístupu je, že se návrháři chování oproti booleovské variantě jednodušeji formulují pravidla (to je typická výhoda fuzzy systémů) a že se snáze dosahuje chování s plynulými přechody mezi akcemi. Nevýhodou fuzzy metod je pomalejší vyhodnocování a složitost implementace plánovače. Snad z tohoto důvodu se zatím tento přístup příliš nepoužívá.

Přehled fuzzy a pravděpodobnostních metod podává Champandard (2003), který ukázal rovněž prototypovou implementaci na postavě z akční hry Quake. Úvod do fuzzy technik lze také nalézt ve čtvrtém dílu této knihy (Hájek, 2004).

### 5.2.7 Navigace a hledání cesty

K tomu, abychom dokázali řídit virtuálního člověka, nám stále zbývá vyřešit mnoho problémů, počínaje reprezentováním znalostí v krátkodobé paměti a konče modelováním sociálních vztahů (mají se Trip a Grace rádi, nebo jsou na sebe naštvaní?). Všem problémům se zde věnovat nemůžeme, nelze ovšem alespoň krátce nezmínit o jednom ze

základních – pohybu prostorem. Uvědomme si, že téměř všechny houbařovy příkazy ze schématu 5.4 souvisí s pohybem z místa na místo. Pro řízení pohybu máme v zásadě dvě skupiny technik: navigaci a plánování cesty.

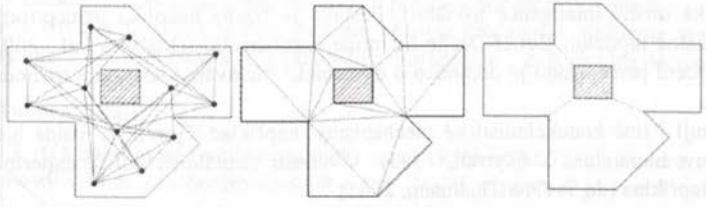
*Navigační pravidla* („steering rules") představují speciální „nízko-úrovňovou" reaktivně plánovací techniku pro řešení některých problémů souvisejících s lokální navigací či směřováním ke vzdálenému cíli v nekomplikovaném (nicméně spojitém) terénu. Na poli virtuálních lidí se vesměs vychází z práce Craiga Reynoldse (Reynolds, 1987)[7]. Jeho technika je založena na tom, že na virtuálního člověka působí několik sil, které ho přitahují nebo odpuzují, a tyto síly svým složením určují výsledný směr pohybu. Síly se dynamicky mění podle toho, jak se mění terén, jímž se postava pohybuje. Pomocí mechanismu je možné jednoduše dosáhnout například:

- navigace postavy směrem k cíli,
- pozvolného zpomalování před cílem,
- vyhýbání se překážkám,
- sledování zdi nebo cesty,
- přibližování se cizím postavám nebo vzdalování se od nich,
- koordinace postav ve skupině, s vedoucím i bez vedoucího („flocking").

Výhoda techniky je, že představuje jednoduchý a výpočetně efektivní mechanismus. Například ve strategických počítačových hrách jsou řízeny navigačními pravidly stovky postav. V případě komplikovaného terénu (například v budovách) nebo při chůzi na větší vzdálenost ovšem bývá nutné zkombinovat tento mechanismus s plánováním cesty.

Při *plánování cesty* máme zadaný počáteční bod, cíl a zjednodušenou reprezentaci terénu pro potřeby hledání cesty. Algoritmus nejprve v dané reprezentaci najde (optimální nebo suboptimální) cestu od počátku k cíli. Virtuální postava se potom drží vybrané cesty, přičemž její pohyb může být lokálně ovlivňován i navigačními pravidly, například překážkami, se kterými se při plánování nepočítalo (pokud by houbař na plánované cestě narazil na kládu, obešel by ji, aniž by přeplánovával cestu).

Nejčastěji se používá algoritmus A* (byl pojednán v prvním dílu této knihy (Mařík, 1993)) a jeho varianty, například hierarchický A* (Botea a kol., 2004) nebo dynamický A* (Stenz, 1994), (Stenz, 1995). Terén pro potřeby plánovaní cesty bývá dán pomocí *navigačního grafu* („way-points") nebo *navigační mřížky* tvořené trojúhelníky či kon-



Obr. 5.7 Srovnání tří reprezentací terénu pro potřeby algoritmu plánujícího cestu. Vidíme shora místnost, uprostřed níž je sloup. Vpravo je podlaha pokryta navigačními body, uprostřed mřížkou s trojúhelníky a vpravo mřížkou s *n*-úhelníky. Převzato z (Gemrot, 2006).

[7] Před Reynoldsem se nicméně problémem navigace zabývala oblast robotiky.

vexními *n*-úhelníky („nav-mesh"), viz obr. 5.7. V nich bývají často zakódovány i dodatečné sémantické informace; vzdálenosti, fakt, že mezi určitými místy lze pouze skákat apod.

O reprezentaci terénu a výhodách a nevýhodách jednotlivých reprezentačních technik se lze dočíst v (Gemrot, 2006). Obecně lze říci, že navigační mřížka se více hodí pro reprezentaci otevřeného prostoru. Její výhodou je, že ji lze snáze automaticky generovat a že *n*-úhelníků je obvykle méně než navigačních bodů, což zrychluje prohledávání. Na druhou stranu lokalizace postavy na mřížce je složitější než při použití navigačních bodů. S navigačními body lze také snáze asociovat doprovodné informace.

### 5.2.8 Alternativní přístupy: plánování a neuronové sítě

Podstatou reaktivního plánování je, že určuje pouze následující akci dle aktuálního kontextu (to jest stavu světa a postavy). Nabízí se otázka, jestli by nešel tento přístup zkombinovat s klasickým plánováním, viz (Lažanský, 1993). Výhodou by bylo, že by se virtuální lidé mohli chovat uvěřitelně i v delším časovém úseku, lépe by se modelovalo dosahování dlouhodobých cílů, řešení problémů a podobně.

Faktem však zůstává, že v současnosti se pro řízení virtuálních lidí klasické plánování téměř nepoužívá. Důvody jsou nasnadě: rozhodování musí probíhat v reálném čase a virtuální prostředí je většinou nepředvídatelné a dynamicky se mění.

V akční hře F.E.A.R. (Orkin, 2005) sice bylo reaktivní plánování zkombinováno s variantou klasického plánování, ovšem zjednodušenou. Experimentálně se zkoumá *anytime plánování* (Nareyek, 2001) a *HTN plánování* (Cavazza a kol., 2002), (Muñoz-Avila a kol., 2006), výsledky ale nejsou jednoznačné.

Také neuronové sítě se pro řízení používají zřídka, a pokud už, pak pouze doplňkově či experimentálně. Důvod je ten, že práce s nimi je výrazně obtížnější než s rozhodovacími pravidly či konečnými automaty. Snad jediné zásadnější výjimky představují hry Creatures (Grand a kol., 1997) a Black&White (Evans, 2002). V prvně jmenované se hráč stará o virtuální zvířátka – norny. Norn žije asi deset hodin, má základní životní potřeby a je schopen se rozmnožovat. Je vybaven simulovanou biochemií a je řízen dopřednou neuronovou sítí se zhruba tisíci umělými neurony. Síť se učí zpětnovazebně. V druhé hře hráč učí postavu, která vystupuje v roli poloboha vůči vesničanům (jež jsou z hlediska umělé inteligence triviální). Postava je řízena několika perceptrony, které konceptuálně modelují Belief-Desire-Intention rozhodování ((Mařík a kol., 2001) odst. 5.3.4). Učení perceptronů je doplněno o dynamické budování klasických rozhodovacích stromů[8].

Existují i jiné konekcionistické mechanismy, například Tyrrellovy volné hierarchie („free-flow hierarchies"), (Tyrrell, 1993). V oblasti virtuálních lidí je experimentálně použili například (de Sevin a Thalmann, 2005).[9]

---

[8] V případě Black & White se jedná o klasické rozhodovací stromy (Kubát, 1993). Mechanismus jejich budování vychází z algoritmu ID3 (Quinlan, 1993).

[9] Konekcionistické mechanismy používané pro řízení virtuálních postav obecně splňují definici (1); pojem reaktivní plánování nicméně někteří autoři chápou spíše v kontextu symbolických metod – pravidel a konečných automatů.

### 5.2.9 Nástroje a vývojová prostředí

Můžeme shrnout, že z konceptuálního hlediska je návrh řízení virtuálního člověka – houbaře, Tripa, Grace – otázkou zápisu požadovaného chování pomocí rozhodovacích pravidel, konečných automatů či jejich variant za kombinace s mechanismy pro navigaci a hledání cesty. Reaktivní plánování má sice určitá omezení (Brom, 2005), v běžných případech s ním ale vystačíme.

Mít koncept ale znamená mít ideu, náčrtek na papíře. Jak proměnit koncept v program? Možnosti jak programovat virtuální lidi jsou v zásadě dvě: buď použijeme nějaký speciální programovací jazyk či modelovací nástroj, nebo budeme koncept emulovat v klasickém programovacím jazyce, kupříkladu C++.

Například Trip a Grace jsou programováni v jazyce Hap/ABL (Mateas, 2002). Jinou volbou je reaktivní plánovač PyPOSH (Kwong, 2003) vycházející z behaviorálně orientovaného mechanismu Brysonové (Bryson, 2001), (Bryson, 2003). Hap/ABL i PyPOSH staví na principu rozhodovacích pravidel.

Lze použít i nějaký volně dostupný BDI jazyk používaný primárně pro řízení agentů, například JACK Agent Language (Jack Toolkit, 2006) nebo JAM (Hubber, 1999). Poměrně robustním a v akademickém prostředí často používaným vývojovým nástrojem je symbolický Soar (Newell, 1990). Zatímco první zmíněné s rozhodovacími pravidly pracují pouze konceptuálně, v Soaru tvoří pravidlo jazykové primitivum.

Dnes jsou již dostupné i komerční middlewary pro vývoj umělé inteligence v hrách, například AI-Implant (EngTech, 2006) nebo X-AI-tment (X-AI-tment, 2006). Middlewary většinou nabízí minimálně knihovny pro implementaci konečných automatů a řešení navigace a plánování cesty. Pro řešení problémů spojených s hledáním cesty se často používá PATHEngine (PathEngine, 2006). Počítačové hry navíc často disponují vlastními skriptovacími jazyky. Pro vývoj řídicí logiky virtuálních aktorů ve filmu se používá již zmíněný produktu Softimage/Behavior pracující s HKA (AvidTech, 2006).

## 5.3 Aplikace využívající virtuálních lidí

Zatím jsme se setkali pouze s jednou konkrétní aplikací s virtuálními lidmi – s dramatem Façade – a zmínili jsme se o uplatnění virtuálních postav v počítačových hrách a ve filmu. Aplikacím s virtuálními lidmi se nyní budeme věnovat důkladněji. Začneme rozborem výukové aplikace Tactical Iraqi a poté přehledově pojednáme o dalších oblastech použití.

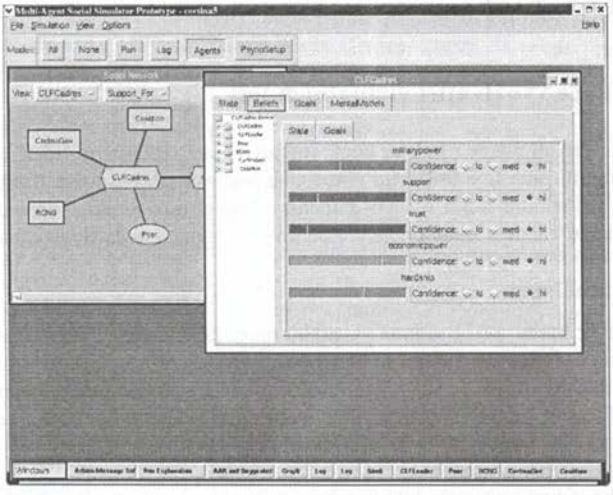
### 5.3.1 Tactical Iraqi

Na obr. 5.8 (též barev. příl.) vidíme skupinku amerických vojáků v rozhovoru s kavárenskými hosty v blíže neurčeném iráckém městě. Voják v popředí je avatar ovládaný uživatelem, všechny ostatní postavy jsou virtuální lidé řízení počítačem. Snímek pochází z programu Tactical Iraqi (TI) vyvinutého v roce 2005 v Center for Advanced Research in Technology for Education na University of Southern California v projektu financovaném americkou armádou.

TI je výuková hra („educational videogame", „serious game"), jejímž cílem je naučit vojáky americké armády základy irácké hovorové arabštiny a seznámit je se zvyklostmi

a společenskými rituály irácképo prostředí. Důležitým prvkem této aplikace jsou právě virtuální lidé a umělá inteligence. Program je postaven na systému vyvinutém původně pro akční hru Unreal Tournament 2003 (Epic Games, 2003), dále využívá software pro rozpoznání hlasu a systém PsychSim pro modelování sociálních interakcí mezi virtuálními postavami (CARTE, 2004).

Obr. 5.8 Tactical Iraqi. Seržant John Smith v čele skupiny amerických vojáků hovoří s hosty irácképo kavárny.



Obr. 5.9 PsychSim. Systém pro modelování sociálních interakcí.

PsychSim je postavený na multiagentním přístupu, kdy je každý vytvářený virtuální člověk nahlížen jako autonomní entita. Nástroj umožňuje navrhovat a řídit sociální vazby ve skupině virtuálních postav, přičemž počítá jak s komunikací postav mezi sebou navzájem, tak s jejich interakcí s uživatelem. Každá postava v systému má své povahové rysy, vlastní cíle a domněnky („beliefs"). Specifičností modelu je, že si postavy navíc utváří proměnlivý mentální model ostatních virtuálních lidí (tedy jejich cíle, povahové rysy aj.), z čehož lze následně odvozovat vztahy k ostatním. To v posledku ovlivňuje chování postav (Marsella a kol., 2004).

Výuka arabštiny v TI se opírá o interaktivní příběh v grafickém 3D prostředí, v němž uživatel hlasem, myší a klávesnicí ovládá avatara seržanta Johna Smithe. Úkolem hráče je v komplikovaném sociálním prostředí malého irácké ho města navázat kontakt s lokálními autoritami a s jejich pomocí přestavět válkou zničenou dívčí školu. Pro úspěch ve hře je nutné osvojit si schopnost komunikovat s virtuálními obyvateli města v arabštině a vhodně reagovat v simulovaných sociálně vypjatých situacích. Použije-li student nevhodné obraty nebo špatně ohodnotí kulturní význam neverbálních gest, dávají virtuální lidé najevo své nepochopení či nelibost, ve vyhrocených případech na uživatelova avatara křičí nebo mu dokonce vyhrožují fyzickým násilím. Série více nevhodných interakcí s virtuálními lidmi způsobí neúspěch mise.

Aplikace se tak snaží co nejvěrněji modelovat prostředí a situace, s nimiž se vojáci pravděpodobně setkají při svém skutečném nasazení v Iráku. Zde se ovšem zároveň odhalují současné limity virtuálních lidí. Množina programem rozpoznávaných řečových aktů a možných reakcí na ně, jakkoli komplexní, je vždy konečná. Ačkoliv je umělá inteligence využitá v programu schopná se učit, tato schopnost se omezuje na přizpůsobení se hlasovému projevu individuálních studentů. Někteří uživatelé si také stěžují, že aplikace zcela vylučuje používání angličtiny, ačkoli v reálné misi podobného typu je často komunikace vedena ve zjednodušené hybridní směsi obou jazyků (Losh, 2005).

Výrazná část mezilidské komunikace se odehrává v neverbální rovině. Systém Unreal Tournament 2003, na kterém je TI postaven, umožňuje do aplikace začlenit gestikulaci, zobrazení obličejové mimiky v něm ovšem možné není. TI obchází tento problém postupem převzatým ze sféry počítačových her. U každé virtuální postavy se uživateli zobrazuje grafický ukazatel, indikující míru důvěry, kterou tato postava má k osobě Johna Smithe. Změny ukazatele pak uživateli poskytují zpětnou vazbu získávanou v reálném životě na základě mnoha různých faktorů.

TI získalo v roce 2005 cenu agentury DARPA za významný technologický přínos. Podrobné informace o aplikaci lze získat na webových stránkách projektu (Tactical Iraqi, 2005), sociologicko-lingvistickému rozboru TI se věnuje Elizabeth Losh (Losh, 2005).

### 5.3.2 Další aplikace

Viděli jsme, že virtuální lidé pronikli do počítačových her, výukových aplikací, filmu a interaktivního dramatu. Mezi jiné praktické aplikace patří terapeutické programy a simulace. Virtuální lidé se používají i jako takzvaní konverzační agenti, například jako virtuální průvodci muzeem. Společně s fenoménem virtuální reality si jich povšimli umělci. Virtuální lidé se dají použít i při verifikaci návrhu přístrojů a pracovních postupů. Podívejme se nyní na některé oblasti důkladněji.

*Výukové aplikace* („educational applications", „serious games"). Virtuální prostředí umožňuje studentům vyzkoušet si postupy, které by jinak pro finanční náročnost zkoušet nemohli, nebo „prožít" nebezpečné či fakticky nerealizovatelné situace. Při tréninku jim

buď asistují *tutoři* („tutors") – virtuální učitelé, nebo je modelovými scénáři, které by se měl student naučit zvládat, provázejí *aktoři* („actors") – virtuální herci. Příkladem tutora je agent Steve (Rickel a Johnson, 1999), který učí vojáky ovládat vysokotlaký kompresor. Příkladem aktorů jsou popsaní vojáci a Iráčané z TI. Existují ale i výukové aplikace s virtuálními lidmi z lékařského prostředí (Games For Health, 2006).

Programovat virtuální aktory a tutory je poměrně náročné. Kromě popisu scénáře a modelu rozhodování často potřebujeme modul na zpracování přirozeného jazyka, někdy i na jeho generování, jak bylo naznačeno v předchozím výkladu.

*Terapeutické aplikace.* Při léčení fobií je možné použít kognitivně-behaviorální terapii, kdy je pacient za asistence terapeuta opakovaně vystavován situacím navozujícím jeho fobii. Cílem terapie je pochopit podstatu problému a následně najít způsob, jak ho řešit. Ukazuje se, že pacienti reagují, i když jsou situace napodobovány ve virtuálním prostředí. Příkladem je Herbelinova práce z roku 2005, která se týká pacientů trpících sociofobií, to znamená strachem z přítomnosti v prostoru, kde se nachází větší množství lidí. Technicky se virtuální lidé pro terapeutické aplikace podobají aktorům z výukových aplikací.

*Komerční počítačové hry* („computer games"). Zejména v akčních („first person shooter", FPS) nebo dobrodružných hrách („adventure", role-playing game", RPG) se často vyskytují virtuální postavy: lidé, zvířata či různé stvůry většinou s antropomorfními rysy. Tyto umělé bytosti představují buď protivníky, nebo postavy, jež dokreslují příběh – jde o tzv. „non-player characters" – doslova postavy neovládané hráčem. Je sice otázka, jestli virtuální vojáky ze hry Wolfenstein 3D (idSoftware, 1992), jež na PC zpopularizovala žánr FPS, již lze považovat za virtuální lidi, nicméně v počítačových hrách posledních let o tomto typu postav asi můžeme jako o virtuálních lidech hovořit. V porovnání s virtuálními lidmi z ostatních aplikací bývají ti z her dobře graficky vyvedeni, avšak nejsou příliš „inteligentní". Co je důvodem?

Zhruba řečeno, většinu procesorového času spotřebuje grafika, umělá fyzika a režie hry. I když hráčská komunita reflektuje, že se postavy často chovají hloupě, vývojářská studia soudí (nebo ještě v nedávné době soudila), že sázka na grafiku se vyplatí více než na umělou inteligenci. Udává se, že na řízení postav v akčních a dobrodružných hrách je spotřebováno okolo 5–15% procesorového času (Woodcock, 2003). I když se situace s příchodem grafických procesorů a vícejádrových procesorů zlepšuje, ještě to zřejmě nějakou dobu potrvá, než se dočkáme výraznějšího pokroku. Dalším důvodem je to, že programovat umělou inteligenci pro hry je opravdu těžké (například ve strategických hrách je stavový prostor hry řádově větší než v šachu) a vývojáři na to většinou nedostanou potřebný čas.

Je nicméně pravda, že i v oblasti počítačových her vznikly, co se umělé inteligence týče, výjimečné počiny. Příkladem jsou již zmíněné hry Black&White (Evans, 2002) nebo Creatures (Grand a kol., 1997). Je také pravda, že existují snahy o razantnější obrat. To se odráží například ve snaze o standardizaci algoritmů pro herní umělou inteligenci, hovoří se i o specializovaném hardwaru (IGDA-AI, 2002–2006).

Je také dobré vědět, že oblast her je zajímavá z hlediska výuky umělé inteligence. Díky tomu, že je kód některých počítačových her částečně otevřený[10], je možné využít existujících herních prostředí na prototypování a demonstraci různých algoritmů (které

[10] Jedná se například o hry Unreal Tournament či Quake, počet takových her však stále vzrůstá. S jádrem hry je většinou možné komunikovat na principu klient–server, nebo lze programovat vlastní virtuální postavy přímo ve skriptovacím jazyku hry.

s virtuálními lidmi třeba vůbec nesouvisí). Virtuální postavy jsou názorné a studentům se líbí. Jeden konkrétní projekt týkající se tohoto tématu popíšeme v odst. 5.4.3.

*Virtuální vypravěčství* („virtual storytelling"). Jde o „přehrání příběhu na počítači", přičemž příběhem se myslí například noveleta nebo divadelní hra. Jedná se o aplikace na pomezí umění a počítačových her. Narozdíl od počítačové hry je kladen silný důraz na narativní strukturu děje a umělecké zpracování simulace. Uživateli bývá dovoleno příběh ovlivňovat. Klasickým příkladem je Façade. Velmi těžkým problémem je zpracování jazyka a popis příběhu – příběh je nelineární a interaktivní.

*Film.* Virtuální lidé mohou nahradit komparsisty při davových scénách. Ovšem stovkami postav nepohybuje animátor obrázek po obrázku, nýbrž má každá postava svůj program a vlastnosti a výsledná interakce vzniká emergentně. Dobrým příkladem je souboj dvou armád. Již jsme se zmínili, že v aplikaci Softimage/Behavior (AvidTech, 2006) jsou postavy řízeny konečnými automaty. Virtuální komparsisty od předchozích typů virtuálních lidí odlišuje to, že jejich chování není nutné počítat v reálném čase.

*Verifikace návrhu přístrojů.* Konstruktéři, kteří navrhnou v nějakém CAD systému model stroje, si kladou otázku, jestli je stroj navržen dobře z hlediska ovládání. Tím se myslí, jestli může stroj ovládat jak malý, tak velký člověk, jestli ovládání není nebezpečné (například jestli se těsně vedle často používané páčky nenachází horká plocha) a podobně. Verifikaci modelu lze částečně provést ve virtuálním prostředí provedením série experimentů s virtuálními lidmi (Badler ad., 2002). Oproti předchozím příkladům je větší důraz kladen na reálný 3D model postavy.

*Modelování biologických organismů* („modelling biologically inspired systems"). Počítač představuje nástroj, pomocí kterého je možné verifikovat biologické teorie. Většinou se ale pracuje s blízkými příbuznými virtuálních lidí – s animaty, o nichž více pojednáme v odst. 5.5.2. S jistým zjednodušením by bylo možné označit daný obor za komputační etologii[11]. Příklady prací jsou modelování chování tlupy opic (Sellers a kol., 2005) či šíření strachu ve stádě ovcí prostřednictvím feromonů (Mata, Aylett, 2005). V Čechách se tomuto tématu systematicky věnuje skupina Pavla Nahodila z ČVUT (Kadleček, Nahodil, 2001), (Nahodil a kol., 2004), o jejímž výzkumu se lze také více dočíst v 18. kapitole tohoto svazku.

## 5.4 Projekty Enti, IVE a Pogamut

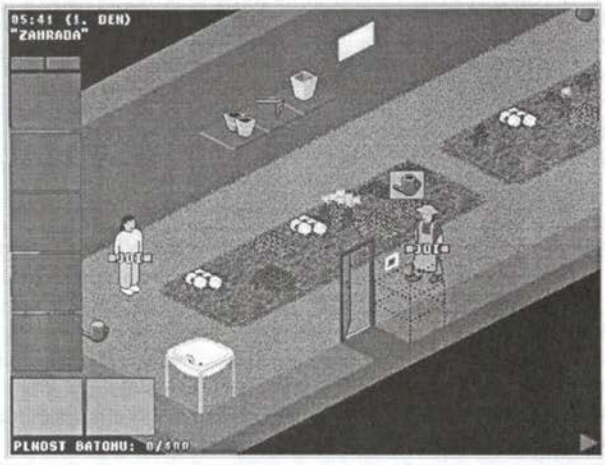
Na Matematicko-fyzikální fakultě Karlovy Univerzity v Praze pracujeme na třech projektech, které se týkají virtuálních lidí.

### 5.4.1 Projekt Enti

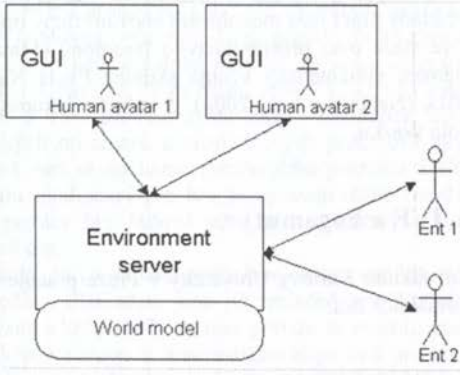
Cílem projektu bylo vytvořit nástroj na prototypování chování virtuálních lidí – *entů* (obr. 5.10, též barev. příl.) – a zkoumání mechanismů jejich řízení. Aplikace byla vyvinuta jako studentský softwarový projekt v letech 2000–2002 (Bojar a kol., 2003), (Bojar a kol., 2005). Hlavním přínosem je to, že projekt zahrnuje jak nástroje na řízení virtuálních lidí, tak simulátor virtuálního světa.

[11] Etologie je disciplína, která studuje chování živočichů, především v jejich přirozeném prostředí.

Aplikace sestává ze tří samostatných linuxových programů: simulátoru světa, GUI a simulátoru individuálních entů (obr. 5.11). Simulátor světa umožňuje pracovat s různými modely světa; standardní balík obsahuje prostředí rodinného domu. Simulátor světa rozeznává zhruba 70 atomických akcí a 60 objektů.




Obr. 5.10 Pohled do světa entů. Zahradník nese naplnit konev, avatar uživatele ho sleduje. Zahradník je řízen reaktivním plánem, jehož struktura schematicky odpovídá obr. 5.5.




Obr. 5.11 Architektura projektu Enti.

Každé grafické rozhraní simuluje jednoho avatara, prostřednictvím nějž může uživatel interagovat s virtuálním světem. Základní idea je taková, že uživatel může ovlivňovat dění stejně jako enti a naopak. Interakce probíhá dvěma způsoby; jednak může avatar vykonávat stejné akce jako běžný ent a jednak může na enty mluvit ve zjednodušené variantě přirozeného jazyka.

Simulace je kroková a prostor je rozdělen na místnosti oddělené dveřmi, přičemž místnosti sestávají ze čtvercových (atomárních) dlaždic. Předměty, enti a avataři jsou popsány různými vlastnostmi; například záhon může být zalitý, ent může být hladový apod. Vzhledem k tomu, že aplikace je určena pro zkoumání řízení chování, nikoli zobrazování, je diskrétní šachovnicový svět dostatečný.

Architektura simulátoru enta odpovídá obr. 5.2. Reaktivní plány jednotlivých entů je možné programovat ve speciálním jazyce E. Jazyk poskytuje primitiva analogická rozhodovacím pravidlům. Krátkodobá paměť entů ukládá data ve tvaru prologovských faktů. Simulátor má navíc lingvistický modul, jenž zpracovává a generuje promluvy.

Projekt Enti se ukázal být vhodný pro živou demonstraci problematiky řízení virtuálních lidí během výuky. Projekt byl zároveň využíván pro výzkum možností a rozšíření reaktivního plánování (Brom, 2005), (Mikula, 2006). Hlavním přínosem bylo rozšíření hierarchického reaktivního plánování popsaného v odst. 5.2.4 o mechanismus přechodů a odkladů, s nimiž se v klasické variantě pracuje špatně. Přechodným chováním („transition behaviour") například je, když ent při přerušení zalévání úkolem „napít se" položí konev a umyje si ruce (kdyby neprovedl přechod, odešel by se napít s konví v ruce). Odklad je, když se ent rozhodne, že jelikož zalévání za chvíli dokončí, přerušovat ho kvůli žízni vůbec nebude.

Další výzkum na této platformě se nyní neplánuje.

### 5.4.2 Projekt IVE

Projekt IVE – inteligentní virtuální prostředí (Brom a kol., 2006a) – vznikl jako odpověď na určité problémy, na něž nestačil projekt Enti. Konkrétně šlo o potíže se simulacemi velikých světů s desítkami postav (aplikace se neúnosně zpomalí už při řízení tří entů), o přidávání nových objektů a nových akcí (enti pracují s fixní množinou) a o centrální koordinaci aktorů při plnění společných úkolů; například při hraní karet či tanci v páru (všechny postavy v entech jsou řízeny autonomně – jde o agenty). V projektu IVE jsou řešení těchto problémů implementovaná.

Projekt vznikal v letech 2004–2005. Jedná se o samostatnou aplikaci napsanou v Javě pod BSD licencí.[12]

Podobně jako v entech jsou postavy v IVE koncepčně řízeny pomocí rozhodovacích pravidel, tentokrát jsou pravidla zapisována ve speciálním jazyce v xml. Prostor je opět diskrétní, čas ovšem běží pseudospojitě. Postavy v IVE nejsou vybaveny jazykovými schopnostmi, narozdíl od projektu Enti však IVE disponuje:

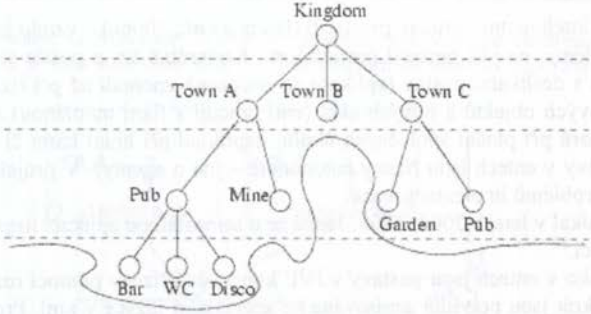
1. Efektivním vyhodnocováním rozhodovacích pravidel; je použita varianta algoritmu Rete (Forgy, 1982).
2. Level-of-detail technikou (LOD), která automaticky zjednodušuje simulaci na místech, jež v daný okamžik nejsou ve středu dění (tedy nesleduje je uživatel nebo se tam nic podstatného neděje). Narozdíl od klasického použití LOD v oblasti počítačové grafiky je technika v IVE použita pro zjednodušování virtuálního prostoru a chování postav (Šerý a kol., 2006). Na abstrakci prostoru z obrázku 5.13b je dění v místech označených značkou „A" simulováno pouze částečně (to znamená, že například postavy v dole nedolují uhlí, ale určitou dobu jen stojí, přičemž poté se náhle objeví uhlí celá

[12] Stránka projektu je: `http://urtax.ms.mff.cuni.cz/ive/public/about.php`

hromada). Na místech, která označena nejsou, simulace probíhá detailněji. LOD technika v IVE je obecná a robustní. Zaprvé umožňuje pracovat s více úrovněmi detailu (viz obr. 5.12), to znamená s různou mírou zjednodušení dění. Zadruhé dokáže simulovat libovolné místo v libovolném detailu. Zatřetí umožňuje úroveň detailu simulace kdykoli efektivně měnit, například podle toho, jaká místa sleduje uživatel (Brom, 2006).

3. Reprezentace virtuálního světa vychází z psychologických teorií afordancí (Gibson, 1979), což je tzv. ekologická percepční teorie, a praktického rozhodování (Bratman, 1987), jehož implementační varianta je známá jako BDI architektura (viz třetí díl této knihy – odst. 5.3.4). Díky této kombinaci lze reaktivní plány ukládat „distribuovaně" do prostředí, nikoli je mít reprezentovány centrálně v knihovnách plánů jednotlivých postav (Brom a kol., 2006a). Tak například, když postava přijde do baru, bar jí poskytne reaktivní plány pro všechny úkoly vztahující se k jednání v baru. Tento mechanismus umožňuje přidávat nové objekty a akce do virtuálního světa (teoreticky i za běhu simulace), čímž řeší problém s rozšiřitelností, jímž trpěli enti.

4. IVE umožňuje řídit virtuální postavy jak autonomně, tak centrálně, i oba přístupy kombinovat (Brom a kol., 2006a), (Brom, Lukavský, 2006). Na obr. 5.13a jsou ve skutečnosti všichni aktoři ve frontě řízeni centrálně, jejich vlastní řídicí mechanismus pouze kontroluje, jestli není čas frontu opustit (například pokud postava potřebuje na toaletu). Tento mechanismus zjednodušuje programování scén, kde spolu interaguje více aktorů.

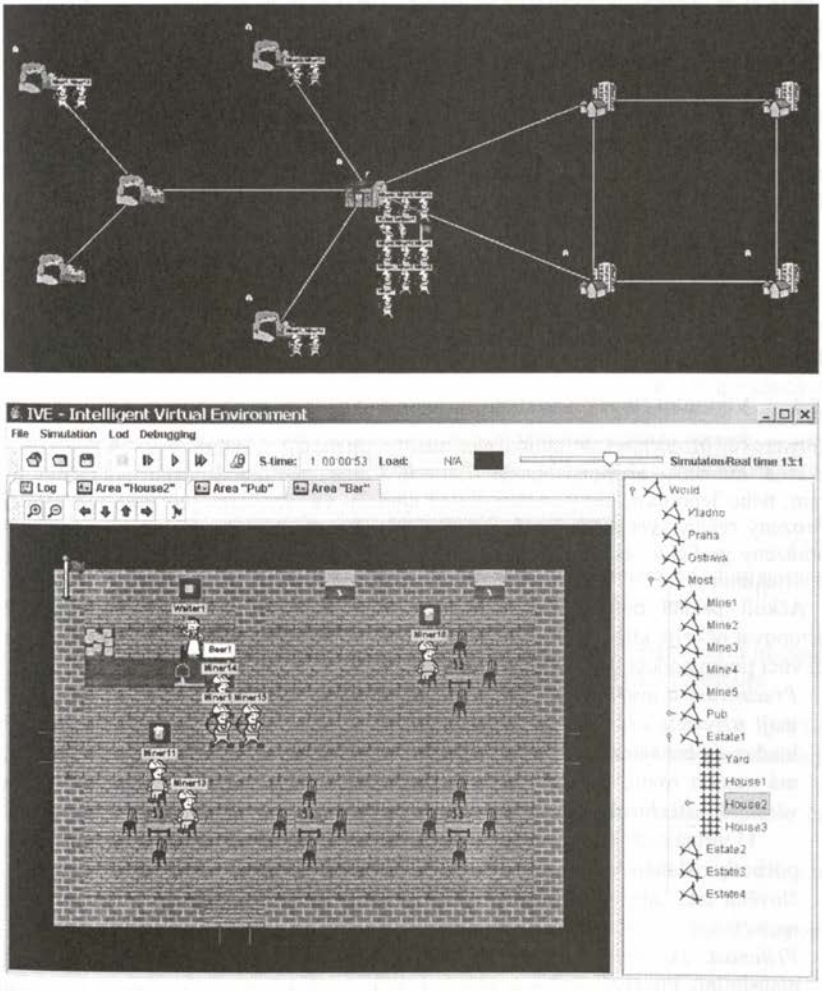



Obr. 5.12 Schematické znázornění level-of-detail v IVE. „Membrána" procházející stromem oblastí určuje míru detailu simulace v tom kterém místě. Každá úroveň ve stromu představuje jednu možnou úroveň detailu simulace.

Především díky třetí vlastnosti je možné na virtuální svět v IVE nahlížet jako na prostředí, které samo naviguje postavy, doslova jim říká, co mají dělat. Postavy vnímají svět jako sadu nabídek vykonat určité úkony vztahující se k jejich cílům – nabídky na ně „volají", že jejich cíle „uspokojí". Podobné reprezentační techniky se často používají i v počítačových hrách. Vesměs vychází z konceptů *chytrých objektů* („smart object") (Kallmann a Thallman, 1998), což jsou objekty, které v jistém smyslu samy dokáží postavy navigovat prostředím. IVE tento koncept rozšiřuje.

Z hlediska hierarchického reaktivního plánování je zase klíčové, že projekt IVE ukázal, jak ho lze využít pro LOD techniku aplikovanou v umělé inteligenci.

V současné době se IVE využívá jak během výuky, tak jako platforma pro výzkum reaktivního plánování a techniky level-of-detail pro umělou inteligenci. Za účelem výzkumu v oblasti interaktivního dramatu rovněž IVE rozšiřujeme o komponentu nazývanou „drama manager", jejímž smyslem je uvést ve virtuálním světě v život interaktivní příběh. Scénáře příběhů jsou reprezentovány pomocí Petriho sítí (Brom a Abonyi, 2006).




Obr. 5.13 Ukázka z projektu IVE. Testovací svět má pět úrovní detailu. Scénář z baru v plném detailu (dole) a pohled na vesnici s pěti doly (nahoře). Místa označená „A" jsou simulována na úrovni 3, ostatní místa detailněji (4 nebo 5).

### 5.4.3 Pogamut

Pogamut (Brom a kol., 2006b) je platforma integrující prostředí akční hry Unreal Tournament (UT) a samostatný reaktivní plánovač PyPOSH pracující na bázi rozhodovacích pravidel (Bryson, 2001), (Kwong, 2003). Pro připojení k UT využívá platforma rozhraní Gamebots (Adobbati a kol., 2001).

Pogamut představuje vývojové prostředí s jednoduchým debuggerem a editorem pro prototypování umělé inteligence v akčních počítačových hrách. Výhodou spojení je, že UT poskytuje kompletní 3D virtuální prostředí, které je dynamické a nedeterministické a svým způsobem podobné přirozenému reálnému světu, a že PyPOSH představuje plnohodnotný reaktivní plánovač pro řízení postav, kterým UT nedisponuje. UT umožňuje nahrávat různé modely virtuálního světa. Poněkud problematicky se zatím jeví rozhraní Gamebots, které neposkytuje korektně všechny informace předávané z UT plánovači PyPOSH.

Pogamut je využíván pro vlastní výzkum umělé inteligence ve zmíněném typu her a plánuje se jeho rozšíření za účelem dalšího využití při výuce.

## 5.5 Virtuální lidé z různých pohledů

### 5.5.1 Virtuální člověk z pohledu softwarového inženýrství

Softwarové inženýrství a aplikovaná umělá inteligence chápe *virtuálního člověka* („virtual human", „artificial human", „life-like character" apod.) jako počítačový program, nebo jeho část, který *imituje* lidské chování ve virtuálním prostředí modelujícím přirozený reálný svět a který má *virtuální tělo*. Prostředí a virtuální tělo jsou obvykle zobrazeny graficky. *Avatarem* („avatar") se pak rozumí grafická reprezentace uživatele ve virtuálním světě.

Ačkoli přesná definice virtuálního člověka zřejmě neexistuje, v literatuře lze vystopovat několik klíčových vlastností, které se obecně považují za vymezující virtuální lidi vůči jiným podobným programům.

- *Předstírání a uvěřitelnost.* Virtuální lidé pouze *předstírají*, že se chovají jako lidé; mají uživatele *oklamat*, to znamená chovat se *uvěřitelně* („believable"). Důraz je kladen na behaviorální stránku, virtuální lidé se snaží uspět v Turingově testu[13]. To má mnoho rovin, počínaje věrohodností pohybu a vyjadřováním emocí a konče věrohodností chování, jíž jsme se v této kapitole zabývali především.

  U virtuálních lidí se většinou nemodelují vnitřní – fyziologické nebo mentální – pochody. Mohou však být imitovány, můžeme například chtít vytvořit virtuálního člověka tak, „aby si uživatel myslel, že virtuální člověk má hlad a chce se tudíž najíst".
- *Vtělenost.* Jak bylo řečeno, virtuální lidé mají tělo, byť ne nutně fyziologicky plausibilní. Používá se často přímo termín *grafické vtělení* („graphical embodiment"), čímž se myslí, že těla jsou zobrazena graficky.

[13] O Turingově testu blíže viz (Havel, 2001). Klasický Turingův test je definován poněkud úžeji, než jak ho zde pojímáme.

- *Dynamika.* Virtuální lidé jednají v prostředí, jež je *dynamické* (mění se i bez přičinění virtuálního člověka), *nedeterministické* (výsledek akce nelze vždy přesně předpovědět) a pouze *částečně pozorovatelné* (virtuální člověk vnímá vždy jen část prostředí, nikoli celé prostředí naráz)[14].
- *Komplexnost.* Virtuální lidé jsou komplexní ve dvou aspektech. Za prvé v tom smyslu, že imitují určitou ucelenou škálu lidských činností. Obvykle se požaduje, aby entita, pokud o ní máme mluvit jako o virtuálním člověku, dokázala provádět více úkolů, z nichž alespoň dva jsou konfliktní. Druhý aspekt komplexnosti spočívá v celistvosti těla. Mluvící virtuální hlava je toliko hlava – nikoli virtuální člověk.
- *Interaktivita.* Aplikace s virtuálními lidmi jsou většinou interaktivní. Tím se rozumí, že uživatel může ovlivňovat běh simulace; buď prostřednictvím svého avatara, nebo jinak. Výjimku tvoří filmoví aktoři.

Dodáváme, že v počítačových hrách se často můžeme setkat s virtuálními zvířaty s antropomorfními rysy, jež lze snad také považovat za svého druhu virtuální „lidi" – vzpomeňme například na Creatures (Grand a kol., 1997).

### 5.5.2 Virtuální lidé, agenti a animati

Obecně řečeno se *agentem* myslí určitá podtřída pojmu *systém*, ovšem v interpretaci různých autorů pokaždé podtřída jiná. Pro nás bude klíčové to, jak pojem chápe současná aplikovaná umělá inteligence. Agenti v tomto smyslu již byli v této knize studováni v (Štěpánková a kol., 1997), (Mařík a kol., 2001), (Kelemen, 2001), (Mařík a kol., 2004). Připomeňme snad nejrozšířenější definici (Wooldridge, 2002), jedná se o upravenou definici z (Wooldridge a Jennings, 1995):

*Agent*: (5.3)

„ *Agent* je počítačový systém, který je *situovaný* v nějakém prostředí a který je schopen *autonomně jednat* v tomto prostředí tak, aby plnil cíle, pro něž byl navržen."[15]

Pojem je natolik obecný, že zahrnuje v podstatě i termostat (i když přísně vzato jde spíše o „kontrolní systém" než „počítačový systém"). Autor proto termín dále zpřesňuje.

*Inteligentní softwarový agent*: (5.4)

Inteligentní softwarový agent je agent (v původním smyslu), který splňuje následující vlastnosti:

- je *reaktivní* („reactive"): „...agent je schopen vnímat své prostředí a v pravý čas reagovat na jeho změny tak, aby plnil cíle, pro něž byl navržen."
- je *proaktivní* („proactive"): „...agent je schopen chování zaměřeného na cíl; sám *převezme iniciativu* tak, aby plnil cíle, pro něž byl navržen."
- je *sociální* („social ability"): „...agent je schopen interakce s ostatními agenty (a popř. lidmi) tak, aby plnil cíle, pro něž byl navržen."

Vzniká otázka, jestli virtuální lidé nejsou určitou podtřídou inteligentních agentů. Toto pojetí naráží na dva problémy. Virtuální lidé jsou snad vždy reaktivní a většinou se dají

[14] Vlastnosti virtuálních prostředí jsou podle (Russell a Norvig, 1995).
[15] Kurzíva v originálu, autorem tohoto a dalších překladů je Matěj Hoffmann.

chápat i jako proaktivní, ovšem se sociální vlastností může být potíž. Někteří virtuální lidé jsou ve svém prostředí sami (i když většinou interagují alespoň s uživatelem). Druhý, zásadnější problém je s autonomií. Virtuální lidé jsou někdy řízeni jedním kontrolním programem – jakousi „sdílenou myslí". Ve strategických hrách se na postavy často nahlíží jako na pouhé figurky na šachovnici. I Trip a Grace z Façade jsou koordinováni společně. Přestože pojem autonomie není přesně definován a různí autoři ho chápou různě, asi nikdo si pod ním nepředstaví centrální řízení.

Zatímco typičtí autonomní agenti se rozhodují nezávisle, u virtuálních lidí chceme pouze to, aby tak působili navenek. To je hlavní důvod, proč nelze paušálně virtuální lidi za agenty označit. Někteří konkrétní virtuální lidé ovšem agenty podle (5.4) být mohou – například aktoři z aplikace Tactical Iraqi. Takoví agenti se někdy označují jako *inteligentní* a *virtuální* („intelligent virtual agent" někdy též „human-like agent".

Z teorie víme, že agenty lze přibližně zařazovat na jakousi pomyslnou stupnici od čistě reaktivních po čistě plánovací (či deliberativní). Pokud bychom se snažili na tuto škálu umístit i virtuální lidi, stáli by poblíž reaktivní hranice. Nebývají ovšem čistě reaktivní – jednak mívají paměť (či zjednodušený model virtuálního světa) a jednak jejich reaktivní plány bývají explicitně strukturovány podle cílů (například houbař pracuje na cíli sbírat houby). Mezi jednotlivými virtuálními lidmi samozřejmě existují rozdíly – například postavy z akčních her budou typicky více reaktivní než aktoři z Tactical Iraqi.

Termínem z jiné oblasti je *animat*. Pojem označuje doslova *umělé zvíře* („artificial animal") a pochází z pera Stewarta Wilsona (Wilson, 1985), (Wilson, 1991). Stojí v názvu konference SAB – plným jménem „From animals to animats: The International Conference on Simulation of Adaptive Behavior" – kterou Wilson spoluzakládal. Zde vyjdeme z definice upravené podle Champandarda (Champandard, 2003).

*Animat:* (5.5)

„*Animat* je umělá bytost žijící ve virtuálním prostředí, která je vtělená („embodied"), tj. interaguje se svým prostředím výhradně prostřednictvím svého těla, které podléhá omezením prostředí."

Pojem animat se od pojmu virtuální člověk ve dvou ohledech liší. Zaprvé je obecnější v tom smyslu, že může jít o model jakéhokoli živočicha. Za druhé, a to je to hlavní, jsou animati stavěni především za účelem pochopení fungování živých organismů, nikoli za účelem prosté imitace. Vesměs se jedná se o biologicky (neurobiologicky, fyziologicky) plausibilní modely. V článku (Wilson, 1991) se přímo píše:

„...základní *hypotézou* přístupu pracujícího s animaty ('animat approach') je, že díky simulování a pochopení celých systémů inspirovaných zvířaty se můžeme postupně přibližovat člověku."

„Doufáme, že se nám podaří dosáhnout lidské inteligence 'zdola'."

„ je nezbytné ... maximalizovat 'fyzičnost' senzorických signálů tak, abychom se vyhnuli předem definovaným symbolickým vstupům."

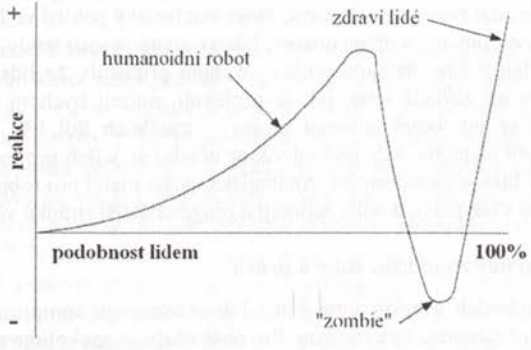
Virtuální lidé jsou většinou budovaní shora-dolů („top-down"), animati zdola-nahoru („bottom-up"). Virtuální lidé jsou věrohodní „zvenku", animati „zevnitř" i „zvenku". Animati například těžko mohou být řízeni centrálně. Někteří autoři ovšem pojem animat rozšiřují i na postavy z počítačových her, například právě (Champandard, 2003).

Existují i další obdobné termíny, například *bot* (počítačem řízený protivník z akční hry). Přesná taxonomie neexistuje, klíčové je orientovat se podle účelu, za jakým umělá bytost vznikla, a pochopit výchozí abstrakci toho kterého modelu. Problematikou „umělých bytostí" se úzce zabývá také oddíl 18.2.

### 5.5.3 Virtuální lidé z pohledu koncového uživatele

Na otázku, jak reagujeme na člověku více či méně podobné bytosti, se pokusil odpovědět již v 70. letech průkopník robotiky Dr. Masahiro Mori. Grafické zachycení tohoto vztahu má nestandardní průběh, způsobený přítomností „údolí odpornosti"[16] (obr 5.14). Na ose *x* máme podobnost nějaké bytosti člověku (danou většinou jejím vzhledem či pohybem), na ose *y* pak reakci lidí na onu bytost – zda na ně působí pozitivně či negativně. Bytostí zde můžeme myslet v podstatě cokoli. Primárně šlo o roboty, ale vztah se týká zrovna tak dobře i virtuálních lidí, zvířat, kyborgů či lidí s postižením.

Průběh křivky je následující: Začneme-li od bytostí velmi odlišných od člověka, pak zvyšování podobnosti způsobuje, že naše reakce je pozitivnější. Tedy například přidáním člověku podobných vlastností nějaké virtuální bytosti, jež ani není antropomorfní, zvýší naše sympatie k ní. Stále jsme si ale bezpečně vědomi, že nemáme tu čest s člověkem. Od určité míry podobnosti však dochází k obratu: více lidských atributů začíná naše sympatie k bytosti snižovat. Na dně tohoto „údolí" pak dlí člověk-zrůda, například zombie či Frankenstein. Dokážeme-li se člověku dále přibližovat, začínají sympatie opět růst, až k pomyslnému vrcholu v normálním zdravém člověku.




Obr. 5.14 Údolí odpornosti. Lidská reakce na bytost či podobnost bytosti člověku.

Nutno dodat, že tvar křivky je hypotetický a představuje spíše intuitivní náhled na věc (stačí si uvědomit, kdy původní článek vznikal). Faktem nicméně je, že na této intuitivní bázi s křivkou pracují například tvůrci prostetik, filmů nebo výtvarného umění. U virtuálních lidí si můžeme vyzkoušet působení údolí pohledem na nedokonalé 3D avatary. Pokud jsou postavy znázorněny schematicky (např. obr. 5.12), reagujeme neutrálně nebo

[16] V originálu „uncanny valley", do češtiny se někdy překládá jako údolí záhad, tajemné údolí nebo tajemný val.

vcelku pozitivně. Pokud jsou znázorněny v 3D prostředí, což se obvykle chápe jako posun směrem k vyšší reálnosti, často spokojeni nejsme (mnohdy aniž bychom dokázali říci proč). Důvodem je to, že postavy připomínají člověka natolik, že již podvědomě začínáme očekávat lidská gesta, pohyby očí, emoce ve tváři a podobně – jenomže tyto projevy často chybí. Toto také představuje motivaci pro podstatnou část výzkumu v oblasti virtuálních lidí. Je nasnadě, že hlavní roli zde hraje psychologie. Je například možné sledovat míru angažovanosti diváka, výukové aplikace či počítačové hry pomocí měření frekvence tepu a kožního odporu. K odhadování těchto reakcí je možné opět použít metody umělé inteligence, například bayesovské učení (McQuiggan a kol., 2006).

Požadavek na „lidskost" chování přerůstá zejména v počítačových hrách v požadavek na zvyšování zábavnosti či navozování pocitu hrdinství. Zábavnost a lidskost mohou být dvě různé věci. V akčních hrách například potřebujeme, aby počítačem řízené postavy, přestože teoreticky znají přesnou pozici protivníka, nestřílely přesně. Potřebujeme, aby střílely zhruba tak nepřesně jako člověk, přičemž se ovšem mohou zlepšovat (jako člověk). K tomu se dají nacvičit umělé neuronové sítě (Champandard, 2003). Zde požadavek na zábavnost a lidskost vede k použití téže techniky, až na jednu výjimku: pokud virtuální protivník spatří postavu hráče, zatímco hráč o protivníkovi neví, střílí protivník obvykle vedle, i když by správně měl zasáhnout (i se započtením chyby). To jednak zvyšuje napětí hry a jednak zbytečně v hráči nevyvolává pocit, že virtuální protivník podvádí (i když by fakticky nepodváděl).

S tím, jak se bude nadále věrohodnost virtuálních lidí zvyšovat, začneme se pravděpodobně setkávat s otázkami typu: „jsou virtuální lidé živí?", „je možné je vypnout?" či „mají mít vlastní práva?". Přestože se z pohledu umělé inteligence a softwarového inženýrství mohou zdát tyto otázky bizarní, funkcionalistický pohled ve filosofii, který na nejjednodušší úrovni pracuje s přisuzováním lidství určité bytosti tehdy, pokud se lidsky projevuje, je vcelku dobře obhajuje. Pokud bychom připustili, že lidství živým lidem přiznáváme pouze na základě toho, jak se projevují, museli bychom minimálně začít zvažovat, v čem se od našich projevů projevy virtuálních lidí liší, abychom mohli potenciálně zachytit okamžik, kdy (pokud vůbec někdy) se jejich projevy přiblíží našim natolik, že rozdíly budou zanedbatelné. Analogická úvaha platí i pro robotiku.[17]

V tuto chvíli se však právo a etika zajímají o pragmatičtější stránku věci.

### 5.5.4 Virtuální lidé z pohledu etiky a práva

Detailní rozbor právních aspektů virtuálních lidí představuje komplexní problematiku a přesahuje rozsah i zaměření tohoto textu. Pro naše účely se spokojíme s konstatováním, že z hlediska práva představuje virtuální člověk souhrn více jevů, z nichž každý je samostatně definován. Nadto pak může právní klasifikace virtuálního člověka v určitých případech záviset na kontextu, ve kterém je použit.

Základ každé virtuální bytosti je *počítačový program*, který je jako právní kategorie sama o sobě uveden v autorském zákoně (č. 121/2000 Sb., o právu autorském). Většina virtuálních bytostí se uživateli představuje v nějakém grafickém zobrazení, které je autorským zákonem chápáno jako *výtvarné dílo*. V neposlední řadě umělá inteligence pracuje při vyhodnocování svých kroků s obsáhlým a strukturovaným souborem dat,

[17] Více o funkcionalismu v kontextu umělé inteligence se lze dočíst v (Havel, 2001).

který spadá pod právní pojem *databáze*. Všechna tato díla jsou pak chráněna autorským právem, pokud jsou jedinečným výsledkem tvůrčí činnosti autora a jsou vyjádřeny v objektivně vnímatelné podobě.[18] Jak vidíme, vše uvedené se vztahuje na většinu softwarových aplikací a virtuální lidé nezaujímají v právním systému žádné zvláštní místo. V těchto uvedených kategoriích se také virtuální lidé stávají předmětem majetkových a právních vztahů. V populární hře The Sims On-line (ElArts, 2002), která umožňuje hráčům rozvíjet osobnost a kariéru virtuálního člověka v simulovaném světě, již dnes vzkvétá obchod s virtuálními bytostmi a jejich nejrůznějšími doplňky (modely obličejů, šatstva nebo celých bytů a budov).

Nicméně některá využití virtuálních lidí v konkrétních aplikacích přinášejí nové, dosud plně nereflektované eticko-právní otázky. Jednou z nich je problematika sociálních a rasových stereotypů v počítačových hrách. Virtuální postavy ve hrách vycházejí již ze své podstaty ze zjednodušeného modelu, který ovšem v některých případech může vést k vytváření negativních stereotypů. D. Chan analyzuje ve své práci vazbu mezi rasovou a sociální identitou virtuálních lidí v počítačových hrách (Chan, 2005). Argumentuje, že některé etnické skupiny (zvl. afroameričané, hispánci, arabové) jsou ve hrách reprezentovány v několika málo převážně negativních sociálních rolích (gangster, terorista). Dlužno dodat, že tento problém se zdaleka neomezuje jenom na západní hry, stejné otázky si můžeme klást u virtuálních lidí zobrazujících Izraelce v akční propagandistické hře hnutí Hizballáh Special Force (Solution, 2002) nebo v sérii syrských propalestinských her Under Siege (Afkar Media, 2005). Více o vzájemné stereotypizaci virtuálních „nepřátel" v dynamicky se rozvíjejícím průmyslu počítačových her pojednává práce (Šisler, 2006).

Dalším, pro mnohé závažnějším tématem, je využívání virtuálních lidí ve filmovém průmyslu. Virtuální herci záhy pronikli i do produkce pornografických filmů. Zvláště citlivou otázkou je v těchto případech použití počítačových modelů nezletilých a dětí („virtual child pornography"). Technika nahrazení živých herců virtuálními je poměrně často využívána zejména proto, že je v některých právních řádech zcela legální. V českém trestním právu je výklad příslušného ustanovení nejednoznačný a vyskytují se protichůdné názory. Klíčové je zde rozdílné chápání zákonodárcem užitého pojmu *zobrazení dítěte*.[19] Většina autorů se kloní spíše k tomu, že uvedený pojem zahrnuje i virtuální bytosti a takové zobrazení je tedy v pornografickém snímku trestné. Na poli mezinárodního práva se již delší dobu projevuje zřetelná snaha o změnu národních právních řádů směrem k zákazu využívání virtuálních modelů dětí a nezletilých k těmto účelům (Šisler, 2004). Tato citlivá problematika je jen jedním z příkladů zmiňované závislosti právní klasifikace virtuálního člověka na kontextu jeho užití, podobných situací by bylo možné najít více (např. specifické otázky využívání virtuálních bytostí v reklamním průmyslu atd.).

[18] § 2 odst. 1, 2 zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů, ve znění pozdějších předpisů.
[19] § 205 odst. 1, písm. a) zákona č. 140/1961 Sb., trestní zákon, ve znění pozdějších předpisů.

## 5.6 Výzvy pro umělou inteligenci

Asi je už jasné, že v oblasti virtuálních lidí nejde o modely racionálního rozhodování, optimální řešení problémů a podobně. Jde o to co nejlépe v reálném čase imitovat lidské chování, tedy svým způsobem „podvádět" uživatele tak, aby imitaci neprohlédl. Pokud si odmyslíme, že bychom potřebovali přesně zjistit, co to znamená imitovat, respektive kdy skutečný člověk imitaci prohlédne a kdy ne, což je otázka zejména psychologie, máme před sebou několik problémů, jež se v doméně virtuálních lidí zkoumají nebo začínají zkoumat.

První oblast souvisí s modelováním pozornosti a vidění, jinými slovy se zpracováním senzorických primitiv a vytvářením reprezentace okolního světa. Vzhledem k omezeným výpočetním zdrojům virtuální lidé často nemohou „snímkovat" okolí, naopak musí zaměřovat svou pozornost pouze na relevantní objekty a děje. Modelování percepce tak hraje důležitou roli především ve hrách a výukových aplikacích. Příklady prací jsou (Kim a kol., 2005), (Leonard, 2003). Některé poznatky jsou do modelů přenášeny z percepční psychologie.

Další velikou oblastí je adaptace. Bylo by možné pomocí mechanismů strojového učení automaticky nastavovat určité vlastnosti modelu? Mohl by se virtuální člověk naučit sám něco vykonávat, například imitováním avatara ovládaného uživatelem? Výzkum v této oblasti probíhá poměrně intenzivně, zkoumají se například neuronové sítě či evoluční algoritmy, např. (Blumberg, 1996), (Champandard, 2003), (Cornelius a kol., 2006), do praxe se však tyto techniky zatím příliš neprosadily (výjimky tvoří například již zmíněné hry Creatures a Black&White). S adaptací souvisí také problematika anticipace – předjímání toho, co by se mohlo stát. Tomuto tématu se podrobně věnuje oddíl 18.3.

Jinou intenzivně zkoumanou problematikou je řízení příběhu ve virtuálních dramatech a výukových aplikacích. Problém s příběhem je následující: na jednu stranu potřebujeme, aby probíhal tak, jak je dáno, na druhou stranu je žádoucí, aby mohl uživatel svou interakcí příběh ovlivňovat, čímž ho ovšem v důsledku může „zkazit". Tomuto problému se často říká „interactive – narrative tension" nebo „boundary problem" (Magerko, 2006). Jakým způsobem příběh popisovat? Je možné příběh automaticky upravovat podle toho, jak jedná uživatel? Pro popis osnovy příběhu je možné použít například konečné automaty nebo Petriho sítě (Brom a Abonyi, 2006), používá se také HTN formalismus, např. (Aylett a kol., 2005).

Poměrně neprobádaná oblast související s počítačovými hrami vzniká s tím, jak se začínají prosazovat víceprocesorové počítače (jak na PC, tak na herní konzole). Jak paralelizovat již existující algoritmy? Jak úlohy automaticky rozvrhovat na různé procesory tak, aby došlo k maximálnímu vytížení? Bylo by například možné využít několik procesorů jen na řízení virtuálních postav? Zřejmě dojde k využití některých mechanismů z oblasti plánování a rozvrhování.

Existují i další dílčí problémy. Namátkou jmenujme automatické řízení kamery v 3D prostředí, level-of-detail pro umělou inteligenci, čili automatické zjednodušování chování postav mimo místa ve středu dění, a znovu generování emocí, any-time plánování a zpracování (generování) přirozeného jazyka či alespoň předdefinovaných dialogů. Virtuální lidé přináší pro umělou inteligenci mnoho výzev. Oblast se rychle vyvíjí a očekává doménově specifická řešení, která půjdou co nejdříve aplikovat.

Aktuální výsledky bývají obvykle prezentovány na konferencích týkajících se počítačové grafiky, existují ale již i oborové konference o umělé inteligenci v počítačových hrách (Game-On, 2003–2005), (AIIDE, 2005–2006), virtuálním dramatu, např. (ICVS, 2003–2005), a o virtuálních lidech obecně (IVA, 1999–2006). O vytváření animatů se lze dočíst například ve sbornících konferencí SAB (1990–2006) či MNAS (2005). Další informace lze také hledat v přehledovém sborníku (Prendinger a Ishizuka, 2003), knize (Champandard, 2003) a sérii knih AI Game Programming Wisdom, jejichž editorem je S. Rabin, z let 2002, 2004, 2006.

## Literatura


AIIDE. *Artificial Intelligence and Interactive Digital Entertainment conference*. 2005–2006. Stránka konference: http://www.aiide.org/.

Adobbati R., Marshall A. N., Scholer A., Tejada S.: *Gamebots: A 3d virtual world test-bed for multi-agent research*. In: Proceedings of the 2nd Int. Workshop on Infrastructure for Agents, MAS, and Scalable MAS, Montreal, Kanada, 2001.

Afkar Media. *Tahta al-Hisār* (Under Siege). Afkar Media, Damašek, 2005. Stránka hry: http://underash.net/.

AvidTech. *Softimage/Behavior*. Softimage Co. Avid Technology, 2001–2006. Stránka produktu: http://www.softimage.com/Products/Behavior/v2/.

Aylett R. S., Louchart S., Dias J., Paiva A., Vala M.: *FearNot! – An Experiment in Emergent Narrative*. In: Proceedings of 5th Intelligent Virtual Agents (Panayitopoulos ad., eds), LNAI Volume 3661, Springer, Německo, 2005, s. 305–316.

Badler N., Erignac C., Liu Y.: *Virtual humans for validating maintenance procedures*. In: Communications of the ACM, Vol. 45, No. 7, ACM Press, 2002, s. 56–63.

Blumberg B. M.: *Old Tricks, New Dogs: Ethology and Interactive Creatures*. PhD thesis, MIT, Media Laboratory, Learning and Common Sense Section, 1996.

Bojar O., Brom C., Hladík M., Vejlupek M., Toman V., Voňka D.: *ENTs – A Simulator of Human-like Natural Environment*. In: Sborník příspěvků z MIS 2003, Matfyzpress, Česká republika, 2003, s. 3–14.

Bojar O., Brom C., Hladík M., Toman V.: *The Project ENTs: Towards Modeling Human-like Artificial Agents*. In: 31st SOFSEM Communications – student research forum (Vojtáš P., Bieliková M., Bernadette C., Sýkora O., eds), Slovensko, s. 111–122, 2005.

Botea A., Müller M., Schaeffer J.: *Near Optimal Hierarchical Path-Finding*. In: *Journal of Game Development*, Vol. 1, No. 1, 2004, s. 7–28.

Bratman M. E.: *Intention, plans, and practical reason*. Cambridge, Mass: Harvard University Press, 1987.

Brom C.: *Hierarchical Reactive Planning: Where is its limit?* In: Modelling Natural Action Selection: Proceedings of an International Workshop (Bryson J. J., Prescott T. J., Seth A. K., eds), Edinburgh, Skotsko, 2005, s. 235–242.

Brom C., Lukavský J., Šerý O., Poch T., Šafrata P.: *Affordances and level-of-detail AI for virtual humans*. In: Proceedings of Game Set and Match 2, The Netherlands, Delft, 2006a.

Brom C., Gemrot J., Bída M., Burkert O.,Partington S. J., Bryson J. J.: *POSH Tools for Game Agent Development by Students and Non-Programmers*. In: Proceedings of CGAMES 06, Dublin, Irsko, 2006b, s. 126–133.

Brom C.: *Level-of-detail pro umělou inteligenci: Je tato technika přínosná?*. In: Sborník příspěvků ITAT 2006, Slovenská republika, 2006, s. 153–160.

Brom C., Abonyi A.: *Petri-Nets for Game Plot*. In: Proceedings of AISB Artificial Intelligence and Simulation Behaviour Convention, Bristol, Vol. 3, 2006, s. 6–13.

Brom, C., Lukavský, J.: *Proč je hranice mezi virtuální bytostí a jejím světem neostrá*. In: Sborník příspěvků z konference Kognice a umělý život, Třešť 2006, s. 97–106.

Brooks R. A.: *Robust Layered Control System for a Mobile Robot*. In: IEEE Journal of Robotics and Automation, Vol. 2, No. 1, 1986, s. 14–23

Brooks R. A.: *Intelligence without representation*. In: Artificial Intelligence 47, 1991, s. 139–159.

Bryson J. J.: *Intelligence by Design: Principles of Modularity and Coordination for Engineering Complex Adaptive Agents*. PhD thesis, Massachusetts Institute of Technology, 2001.

Bryson J.: *The Behaviour-Oriented Design of Modular Agent Intelligence*. In: Proceedings of Agent Technologies, Infrastructures, Tools, and Applications for E-Services, Springer LNCS 2592, Německo, 2003, s. 61–79.

Bryson J.: *Action Selection and Individuation in Agent Based Modelling*. In: The Proceedings of Agent 2003: Challenges of Social Simulation (Sallach D. L., Macal C., eds), 2003; finální verze: http://www.cs.bath.ac.uk/~jjb/web/publications.html, 2004.

CARTE. *PsychSim*. University of Southern California, 2004. http://www.isi.edu/isd/carte/proj_psychsim.

Cavazza M., Charles F., Mead S. J.: *Planning Characters' Behaviour in Interactive Storytelling*. In: The Journal of Visualization and Computer Animation 2002, Vol. 13, s. 121–131.

Cornelius R., Stanley K. O., Miikkulainen R.: *Constructing Adaptive AI Using Knowledge-Based Neuroevolution*. In: AI Game Programming Wisdom III (Rabin S., ed.), Charles River Media, Inc., Hingham, Massachusetts, 2006, s. 693–708.

ElArts. *The Sims*. Počítačová hra. Electronic Arts. 1999–2005. Stránka hry: http://thesims.ea.com/

EngTech. *AI.implant*. AI middleware for games, animation and simulation. Engenuity Technologies, Inc. 2000–2006. Stránka produktu: http://www.biographictech.com/products.html.

Epic Games. *Unreal Tournament*. Počítačová hra. Epic Games, 2003, Inc. Stránka hry: http://unrealtournament.com.

Evans R.: *Varieties of Learning*. In: AI Game Programming Wisdom I (Rabin S., ed.), Charles River Media, Inc., Hingham, Massachusetts, 2002, s. 567–579

Forgy C. L. Rete: *A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem*. In: Artificial Intelligence, Vol. 19, 1982, s. 17–37.

Gamma E., Helm R., Johnson R., Vlissides J.: *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional, 1995.

Game-ON research conference. EUROSIS. 2000–2006. Stránka konference: http:// www.ibr.cs.tu-bs.de/news/ibr/game-on-2006/index.html.

Games for Health. Annual Conference. Games for Health/The Serious Games Initiative. Stránka konference: http://www.gamesforhealth.org/index3.html.

Gemrot, J.: *Mapování, zpracování a reprezentace map pro Unreal Tournament boty*. Bakalářská práce. Matematicko-fyzikální fakulta, Karlova Univerzita, Praha, Česká republika, 2006.

Gibson J. J.: *The Ecological Approach to Visual Perception*. Houghton Muffin, Boston 1979.

Grand S., Cliff D., Malhotra A.: *Creatures: Artificial life autonomous software-agents for home entertainment*. In: Proceedings of the First International Conference on Autonomous Agents (Johnson W. L., ed.). ACM press, 1997, s. 22–29.

Havel I.: *Přirozené a umělé myšlení jako filozofický problém*. In: Umělá inteligence 3 (Mařík V., Štěpánková O., Lažanský J., eds), Academia, Praha 2004, s. 17–75.

Hájek P.: *Deduktivní systémy fuzzy logiky*. In: Umělá inteligence 4 (Mařík V., Štěpánková O., Lažanský J., eds), Academia, Praha 2004, s. 71–92.

Huber M. J.: *JAM: A BDI-theoretic mobile agent architecture*. In: Proceedings of the 3rd International Conference on Autonomous Agents (Agents'99), Seatle 1999, s. 236–243.

Champandard A. J.: *AI Game Development: Synthetic Creatures with learning and Reactive Behaviors*. New Riders, USA, 2003.

Chan D.: *Playing with Race: The Ethics of Racialized Representations in E-Games*. In: The Ethics of E-Games, International Review of Information Ethics, Vol. 4. 2005, s. 24–30. http://www.i-r-i-e.net/inhalt/004/chan.pdf.

ICVS. International Conference on Virtual Storytelling. 2001–2005. Stránka konference: http://www.virtualstorytelling.com/indexVS.htm

idSoftware. *Wolfenstein 3D*, počítačová hra, 1992. Stránka hry: http:// www.idsoftware.com/games/wolfenstein/wolf3d/.

IGDA-AI, 2002–2006

IGDA International Game Developers Association. Special Interest Group on Artificial Intelligence. Stránka skupiny: http://www.igda.org/ai/.

Intelligent Virtual Agents, a working conference. 1999–2006. Stránka konference: http://iva05.unipi.gr.

Isla D.: *Handling complexity in Halo 2*. In: Gamasutra Online, November 3, 2005.

IVA, 1999–2006

JACK toolkit. Agent Oriented Software Group. Projektová stránka: http://www.agent-software.com/shared/home/.

Kadleček, D., Nahodil, P.: *New Hybrid Architecture in Artificial Life Simulation*. In: Advances in Artificial Life, LNAI 2159, Springer-Verlag, Berlín, 2001, s. 143–146.

Kallmann M., Thalmann D.: *Modeling Objects for Interaction Tasks*. In: Proceedings of EGCAS 98, Lisabon, Portugalsko, 1998, s. 73–86.

Kelemen J.: *Reaktivní agenti*. In: Umělá inteligence 3 (Mařík V., Štěpánková O., Lažanský J., eds), Academia, Praha 2001, s. 161–188.

Kim Y., Hill R. W. Jr., Traum D. R.: *A Computational Model of Dynamic Perceptual Attention for Virtual Humans*. In: Conference on Behavior Representation in Modeling and Simulation (BRIMS), Universal City, CA, 2005.

Kubát M.: *Strojové učení*. In: Umělá inteligence 1 (Mařík V., Štěpánková O., Lažanský J., eds), Academia, Praha 2001, s. 168–183.

Kwong A.: *A Framework for Reactive Inteligence through Agile Component-Based Behaviors*. Master's thesis (Bryson J., dir.), Department of Computer Science, University of Bath, 2003.

Lažanský J.: *Plánování*. In: Umělá inteligence 1 (Mařík V., Štěpánková O., Lažanský J., eds), Academia, Praha 2001, s. 184–216.

Leonard T.: *Building an AI Sensory System: Examining The Design of Thief: The Dark Project*. In: Gamasutra Online, March 7, 2003.

Losh E.: *In Country with Tactical Iraqi: Trust, Identity, and Language Learning in a Military Video Game*. In: Digital Experience: Design, Aesthetics, Practice, University of Copenhagen, Kodaň 2005, s. 69–78.

Magerko, B.: *Intelligent Story Direction in the Interactive Drama Architecture*. In: AI Game Programming Wisdom III (Rabin S., ed.), Charles River Media, Inc., Hingham, Massachusetts, 2006, s. 583–595.

Marsella, S., Pynadath, D., Read, S.: *PsychSim: Agent-based modeling of social interactions and influence*. In: Proceedings of the International Conference on Cognitive Modeling, 2004, s. 243–248.

Mařík V., Pěchouček M., Štěpánková O.: *Multiagentní systémy: Principy komunikace a základní formální architektury*. In: Umělá inteligence 3 (Mařík V., Štěpánková O., Lažanský J., eds), Academia, Praha 2001, s. 189–236.

Mařík V., Vrba P., Pěchouček M.: *Od holonů k virtuálním organizacím*. In: Umělá inteligence 4 (Mařík V., Štěpánková O., Lažanský J., eds), Academia, Praha 2004, s. 407–450.

Masahiro M.: *Bukimi no tani the uncanny valley*. In: Energy, Vol. 7, No. 4, 1970, s. 33–35 (japonsky).

Mata C. D., Aylett R.: *Having it both ways – the impact of fear on eating and fleeing in virtual flocking animals*. In: Modelling Natural Action Selection: Proceedings of an International Workshop (Bryson J. J., Prescott Tj. J., Seth A. K., eds), Edinburgh, Skotsko, 2005, s. 152–157.

Mateas M.: *Interactive Drama, Art and Artificial Intelligence*. PhD thesis. Department of Computer Science, Carnegie Mellon University, 2002.

McQuiggan S., Lee S., Lester J.: *Predicting User Physiological Response for Interactive Environments: An Inductive Approach*. In: Proceedings of the Second Conference on Artificial Intelligence and Interactive Entertainment (AIIDE), Marina del Rey, California, 2006, s. 60–65.

Mikula, T.: *Hierarchické reaktívne plánovanie s prechodmi*. Bakalářská práce. Matematicko-fyzikální fakulta, Karlova Univerzita, Praha, Česká republika, 2006.

MNAS. Modeling natural action selection: an interdisciplinary workshop. 2005. http://www.bath.ac.uk/comp-sci/ai/MNAS-2005/.

Muñoz-Avila H., Hoang H.: *Coordinating Teams of Bots with Hierarchical Task Network Planning*. In: AI Game Programming Wisdom III (Rabin S., ed.), Charles River Media, Inc., Hingham, Massachusetts, 2006.

Nahodil P., Slavík P., Řehoř D., Kadleček D.: *Dynamic Analysis of Agents Behaviour – Combining ALife, Visualization and AI*. In: Engineering Societies in the Agents World: 4th International Workshop, ESAW 2003 (Omicini A., Petta P., Pitt J., eds), London, Springer-Verlag, LNCS Volume 3071, 2004, s. 346–359.

Nareyek A.: *Constraint-Based Agents – An Architecture for Constraint-Based Modeling and Local-Search-Based Reasoning for Planning and Scheduling in Open and Dynamic Worlds*. PhD thesis. Reading, Springer LNAI 2062, 2001.

Newell A. *Unified Theories of Cognition*. Harward University Press, Cambridge, Masachusetts, 1990.

Orkin J.: *Agent Architecture Considerations for Real-Time Planning in Games*. In: Artificial Intelligence and Interactive Digital Entertainment Conference, Menlo Park, California, AAAI Press, 2005, s. 105–110.

PathEngine. *PATHEngine*. Intelligent agent movement. PathEngine, France, stránka produktu: http://www.pathengine.com/.

Prendinger H., Ishizuka M.: *Life-like Characters: Tools, Affective Functions and Applications*, Springer, Německo, 2003.

ProcArts. *Façade: a one-act interactive drama*. Procedural Arts, 2005–2006. Stránka aplikace: http://www.interactivestory.net/.

Quinlan, J. R.: *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.

Rabin S., ed.: *AI Game Programming Wisdom I*. Charles River Media, Inc., Hingham, Massachusetts, 2002.

Rabin S., ed.: *AI Game Programming Wisdom II*. Charles River Media, Inc., Hingham, Massachusetts 2004.

Rabin S., ed.: *AI Game Programming Wisdom III*. Charles River Media, Inc., Hingham, Massachusetts 2006.

Reynolds C. W.: *Flocks, Herds, and Schools: A Distributed Behavioral Model*. In: Computer Graphics, Vol. 21, No. 4. (SIGGRAPH '87 Conference Proceedings), 1987, s. 25–34.

Rickel J., Johnson W. L.: *Animated Agents for Procedural Training in Virtual Reality: Perception, Cognition, and Motor Control*. In: Applied Artificial Intelligence, Vol. 13, 1999, s. 343–382.

Russell S. J., Norvig P.: *Artificial Intelligence, A Modern Approach*. Prentice Hall, Englewood Cliffs, New Jersey 1995.

SAB. From Animals to Animats: The International Conference on Simulation of Adaptive Behavior. International Society for Adaptive Behavior. 1990–2006. Stránka konference: http://www.isab.org/sab06/.

Sellers W. I., Hill R., Logan B.: *Biorealistic simulation of baboon foraging using agent-based modelling*. In: Modelling Natural Action Selection: Proceedings of an International Workshop (Bryson J. J., Prescott Tj. J., Seth A. K., eds), Edinburgh, Skotsko, 2005, s. 127–134.

de Sevin E., Thalmann D.: *A motivational Model of Action Selection for Virtual Humans*. In: Computer Graphics International (CGI), IEEE Computer Society Press, New York 2005.

Solution. *Al-Quwwa al-Chāsa* (Special Force). SunLight, Beirut, 2002. http://www.specialforce.net.

Stenz A.: *Optimal and Efficient Path Planning for Partially-Known Environments*. In: Proceedings of International Conference on Robotics and Automation, 1994.

Stenz A.: *The Focussed D* Algorithm for Real-Time Replanning*. In: Proceedings of IJCAI, Montreal, Quebec 1995, s. 1652–1659.

Šerý O., Poch T., Šafrata P., Brom C.: *Level-Of-Detail in Behaviour of Virtual Humans*. In: Proceedings of SOFSEM 2006: Theory and Practice of Computer Science, LNCS 3831, Springer-Verlag, Česká republika, 2006, s. 565–574.

Šisler V.: *Convention on CyberCrime*. In: Parlamentní zpravodaj, Vol. 10, No. 3, 2004, s. 25–27.

Šisler V.: *Representation and Self-Representation: Arabs and Muslims in Digital Games*. In: Gaming Realities: A Challenge for Digital Culture, Proceedings of the International MediaTerra Conference, Fournos, Atény, 2006, s. 85 – 92.

Štěpánková O., Mařík V., Lhotská L.: *Distribuovaná umělá inteligence*. In: Umělá inteligence 2 (Mařík V., Štěpánková O., Lažanský J., eds), Academia, Praha 1997, s. 142–177.

Tactical Iraqi. CARTE, University of Southern California, 2005. http://www.tacticallanguage.com/tacticaliraqi/.

Tyrrell T.: *Computational Mechanisms for Action Selection*. PhD thesis. Centre for Cognitive Science, University of Edinburgh, 1993.

van Waveren J. M. P.: *The Quake III Arena Bot*. Master thesis. Faculty ITS, University of Technology Delft, 2001.

Wilson S. W.: *Knowledge growth in an artificial animal*. In: Proceedings of the First International Conference on Genetic Algorithms and Their Applications (Grefenstette J. J., ed.), Lawrence Erlbaum Associates, Hillsdale, New Yersey 1985, s. 16–23.

Wilson S. W.: *The animat path to AI. In: From Animals to Animats*. (Meyer J. A., Wilson S. W., eds), The MIT Press, Cambridge 1991, s. 15–21.

Woodcock S.: *AI Roundtables Moderators' Report*. Game Developers Conference, San Jose, California, 2003. Přístupno z: http://www.gameai.com/cgdc03notes.html.

Wooldridge M.: *An-Introduction to MultiAgent Systems*. John Wiley & Sons, 2002.

Wooldridge M., Jennings N. R.: *Intelligent agents: theory and practice*. In: The Knowledge Engineering Review, Vol. 10, No. 2. 1995, s. 115–152.

X-AI-tment. Exalted Artificial Intelligence Technologies for Entertainment, 2006. Stránka produktu: http://www.xaitment.com/.

Zadeh L. A.: *Fuzzy Sets*. Information and Control, 8(3), 1965, s. 338–356.

Zdráhal Z.: *Reprezentace znalostí*. In: Umělá inteligence 1 (Mařík V., Štěpánková O., Lažanský J., eds), Academia, Praha 1993, s. 99–122.

# 6 Omezující podmínky: od sudoku po vesmírné aplikace

## 6.1 Úvod

Každý se ve svém životě setkáváme s mnoha omezeními, která musíme různým způsobem vzít v úvahu při řešení každodenních osobních či pracovních problémů. Vezměme si takové naplánování schůzky či pracovního oběda. Každý z účastníků má jistá omezení na čas konání, ke kterým se mohou přidat omezení na dostupnost jednací síně, stolu v restauraci apod. Pokud tento problém trochu zobecníme a budeme hovořit o naplánování více schůzek, resp. akcí najednou, dostaneme klasický rozvrhovací problém, ať už například v podobě tvorby rozvrhu pro personál nemocnice, školního rozvrhu či přiřazení posádek jednotlivým letům podle letového řádu. A nebo zcela jiný problém – zabalení kufru na cestu. Je tady zpravidla omezení na velikost kufru a často také na maximální hmotnost zabalených věcí. Proti těmto omezením jdou požadavky na to, co všechno je potřeba do kufru dostat. Opět můžeme celý problém balení povýšit a hovořit o problému naložení kamionu zbožím nebo třeba o umístění vědeckého zařízení do nákladového prostoru raketoplánu. Jako poslední příklad si vezměme sestavení nějakého právě zakoupeného domácího spotřebiče. Jednotlivé kroky často můžeme dělat v různém pořadí, na posloupnost kroků jsou ale přece jen kladena jistá omezení – například „nepřipojujte do elektrické sítě, pokud není vložena pojistka". Opět můžeme problém „trochu" zvětšit a hovořit například o naplánování jednotlivých kroků stavby nového mostu nebo třeba o nalezení sekvence oživování palubní elektroniky kosmické lodi Apollo 13.

Na první pohled se jedná o zcela různé problémy, při trochu abstraktnějším pohledu je ale vidět jistá podobnost. Cílem je vždy najít řešení problému, které vyhovuje zadaným omezením, nebo alespoň porušuje jen minimum z nich. Jednotlivé problémy se vlastně liší jen v tom, jaká omezení se v nich vyskytují. Řadu z popsaných problémů, jako je naplánování společného oběda nebo sestavení kuchyňského robota, každý jistě umí běžně „vyřešit". Komplexnější varianty, jako je postavení mostu nebo rozložení nákladu v nákladovém prostoru raketoplánu, ovšem vyžadují zkušené odborníky. Někteří mají řešení podobných problémů natolik rádi, že se mu věnují i ve volném čase. Vždyť vyluštění křížovky, vyřešení algebrogramu nebo dnes populárního sudoku není nic jiného, než hledání řešení splňujícího jisté přesně dané podmínky.

V následující kapitole si ukážeme, že existuje obecná technologie, kterou je možné využít při řešení zmíněných problémů. Jedná se o splňování omezujících podmínek.

## 6.2 Problém splňování podmínek

V úvodu jsme se zmínili o řadě problémů, které sice na první pohled vypadají dost rozdílně, ale mají společný charakter. Jedná se o problémy kombinatorické optimalizace, tj. o problémy, kde je potřeba prozkoumat různé kombinace možných řešení a mezi nimi vybrat ty kombinace, které splňují daná omezení a případně jsou nějakým způsobem optimální. Pro automatické řešení takových problémů existuje řada různých přístupů. Pokud jsou například všechna omezení v podobě lineárních nerovností, je možné využít technik matematického programování. Pokud jsou omezení spíše logického charakteru a navíc jistým způsobem homogenní, můžeme se obrátit na diskrétní matematiku. Nás zde ale bude zajímat jiný, zcela obecný popis problému, který má původ v oblasti umělé inteligence. Řeč je o *problému splňování omezujících podmínek*, ve zkratce CSP (z anglického Constraint Satisfaction Problem).

Formální definice CSP je velmi jednoduchá – problém je zadán množinou proměnných, množinou možných hodnot těchto proměnných (tzv. doménou) a množinou podmínek omezujících kombinace hodnot, které lze proměnným přiřadit. Zpravidla předpokládáme, že všechny tyto množiny jsou neprázdné a konečné (a v této kapitole se budeme zabývat pouze konečnými množinami). Někdy bývá doména specifikována zvlášť pro každou proměnnou. Potom můžeme globální doménu vidět jako sjednocení domén jednotlivých proměnných s tím, že každá proměnná má přiřazenu unární podmínku, která z globální domény vybere příslušné hodnoty patřící do domény dané proměnné. Následující příklady ukazují, jak lze některé problémy zmíněné v úvodu formulovat jako problémy splňování omezujících podmínek.

*Příklad 1 (algebrogram)*
V rovnici SEND + MORE = MONEY přiřaďte každému písmenu nějakou cifru, tj. celé číslo z intervalu $\langle 0, 9\rangle$, tak, že příslušná rovnice platí, každé písmeno má přiřazeno jinou cifru a $S$ a $M$ jsou různé od 0.

Tento problém můžeme jednoduše formulovat jako problém splňování omezujících podmínek takto:
Proměnné: $\{S, E, N, D, M, O, R, Y\}$.
Doména: $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$.
Podmínky: $S \neq 0$, $M \neq 0$, $1000*S + 100*E + 10*N + D + 1000*M + 100*O + 10*R + E = 10000*M + 1000*O + 100*N + 10*E + Y$, $S \neq E$, $S \neq N$, $S \neq D$, $S \neq M$, $S \neq O$, …, $R \neq Y$.

*Příklad 2 (sudoku)*
Doplňte cifry 1, …, 9 do políček částečně vyplněné matice $9 \times 9$ tak, že žádná cifra se neopakuje v žádném řádku, sloupci, ani ve vyznačených čtvercích $3 \times 3$ (viz obrázek 6.1, šedé cifry představují řešení, tmavé cifry jsou předem vyplněné).
Formální popis problému pomocí omezujících podmínek může vypadat například takto:
Proměnné: $\{R_{i,j} \mid i, j = 1, \ldots, 9\}$.
Doména: $\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$.
Podmínky: $\forall i, j, k$: $R_{i,j} \neq R_{i+k,j}$ $R_{i,j} \neq R_{i,j+k}$, $R_{1,1} \neq R_{2,2}$, $R_{1,1} \neq R_{2,3}$, $R_{1,1} \neq R_{3,2}$, $R_{1,1} \neq R_{3,3}$, …

| 9 | 6 | 3 | 1 | 7 | 4 | 2 | 5 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 7 | 8 | 3 | 2 | 5 | 6 | 4 | 9 |
| 2 | 5 | 4 | 6 | 8 | 9 | 7 | 3 | 1 |
| 8 | 2 | 1 | 4 | 3 | 7 | 5 | 9 | 6 |
| 4 | 9 | 6 | 8 | 5 | 2 | 3 | 1 | 7 |
| 7 | 3 | 5 | 9 | 6 | 1 | 8 | 2 | 4 |
| 5 | 8 | 9 | 7 | 1 | 3 | 4 | 6 | 2 |
| 3 | 1 | 7 | 2 | 4 | 6 | 9 | 8 | 5 |
| 6 | 4 | 2 | 5 | 9 | 8 | 1 | 7 | 3 |

Obr. 6.1 Řešení hádanky sudoku, tmavé cifry jsou vyplněné, světlé cifry se hledají.

*Příklad 3 (N královen)*
Na šachovnici o rozměrech $N \times N$ umístěte $N$ figurek královen tak, aby se žádné dvě figurky neohrožovaly. Připomeňme, že dvě královny se ohrožují, pokud se nachází ve stejném sloupci, řádku nebo na stejné diagonále.

Typická formulace problému $N$ královen jako problému splňování podmínek předpokládá, že královny jsou umístěny do sloupců (tím je splněna jedna z podmínek neohrožování se) a hledá se pro ně umístění v řádcích:
Proměnné: $\{R_i \mid i = 1, \ldots, N\}$.
Doména: $\{1, \ldots, N\}$.
Podmínky: $\forall i, j, i < j$: $R_i \neq R_j, |R_i - R_j| \neq |i - j|$.

Jakmile je problém formálně popsán jako CSP, můžeme pro jeho řešení použít obecné řešící techniky, které podrobněji rozebereme v následujících oddílech. Splňování omezujících podmínek tak vlastně naplňuje myšlenku deklarativního programování – uživatel formálně popíše problém a počítač sám najde jeho řešení. V případě CSP je tímto řešením přiřazení hodnot z domény všem proměnným tak, aby všechny podmínky byly splněny. Takové přiřazení hodnot proměnným nazýváme *přípustné řešení.*

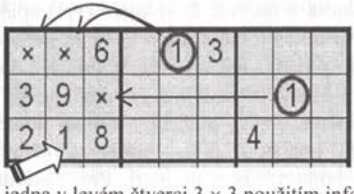
Přípustná řešení bývají někdy ještě dále uspořádána podle své kvality, například je preferováno řešení, kde vybraná proměnná má nejmenší hodnotu. Formálně to vypadá tak, že ke specifikaci CSP je přidána ještě objektivní funkce, tj. libovolná funkce definovaná na (pod)množině proměnných, která vrací celé číslo. Hovoříte potom o *optimalizačním problému splňování podmínek*, ve zkratce COP (z anglického Constrained Optimisation Problem). Řešením COP je přípustné řešení příslušného problému CSP, které minimalizuje (či maximalizuje) hodnotu objektivní funkce. Takovéto řešení nazýváme *optimální*. Jak uvidíme později, řešící techniky pro CSP lze přímo využít při řešení COP a opět zde platí princip deklarativního programování, tj. uživatel jen specifikuje problém a o jeho vyřešení se počítač postará sám.

Než popíšeme základní techniky pro řešení CSP i COP, podívejme se ještě podrobněji na to, co je to přesně *omezující podmínka*. V definici CSP jsme uvedli, že podmínka omezuje kombinace hodnot, které lze proměnným přiřadit. V příkladech jsme potom takové podmínky popisovali matematickou formulí (rovností nebo nerovností), což je

nejběžnější způsob zadání podmínky. Obecně může být podmínkou libovolná relace určující přípustné kombinace hodnot. Takovou relaci můžeme zadat buď právě aritmetickou či logickou formulí (například $A = B \vee A + 1 > C$) nebo prostě výčtem $n$-tic hodnot, které relaci splňují. Existující systémy pro řešení omezujících podmínek zpravidla podporují oba způsoby zadání podmínek. V případě zadání podmínky formulí se jednotlivé systémy liší tím, jaké typy formulí podporují.

## 6.3 Filtrační algoritmy

Vraťme se nyní k hádance sudoku, kterou jsme představili v úvodu, a podívejme se, jaké postupy se používají pro její řešení. Připomeňme, že úkolem je do každého políčka čtverce $9 \times 9$ přiřadit jednu cifru 1, ..., 9 tak, aby cifry v řádcích, sloupcích a vyznačených čtvercích $3 \times 3$ byly navzájem různé. Některá políčka jsou navíc předem vyplněna, což je to, co dělá celý problém netriviálním (jinak lze tabulku celkem jednoduše vyplnit postupným posouváním libovolné permutace čísel 1, ..., 9 po řádcích). Pro umístění příslušné cifry do políčka lze přirozeně využít logické dedukce vycházející z dostupných informací. Těmito informacemi jsou hodnoty v již obsazených políčkách a požadované vlastnosti, že hodnoty v řádcích, ve sloupcích a také v malých čtvercích $3 \times 3$ musí být navzájem různé. Obrázek 6.2 ukazuje dedukci vedoucí k umístění čísla 1 do průsečíku třetího řádku a druhého sloupce. Protože jednička už se nachází v prvním i druhém řádku, jsou volná pole na těchto řádcích v levém čtverci pro toto číslo zakázána. V levém čtverci tak zbývá jediné místo, kam lze jedničku umístit, a to je právě pozice na průsečíku třetího řádku a druhého sloupce.

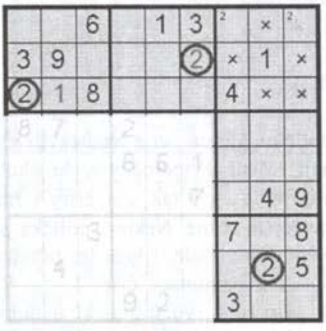



Obr. 6.2 Odvození pozice čísla jedna v levém čtverci $3 \times 3$ použitím informace o jeho existujícím umístění v prvním a druhém řádku.

Podobnou dedukci lze provádět samozřejmě také použitím informace ze sloupců. Co ale dělat v případě, že ani kombinace údajů z řádků a sloupců nedá jednoznačnou pozici čísla v nějakém políčku? V takovém případě můžeme číslo poznamenat do políček, ve kterých se může nacházet, jak to ukazuje obrázek 6.3. Políčka, ve kterých takto bude poznamenáno pouze jedno číslo, dostanou toto číslo přidělené (protože jiné číslo v nich být nemůže). Přidělením čísla do políčka získáme další informaci, kterou můžeme použít pro odvození čísel v dalších políčkách dokud celou hádanku nevyřešíme.

Podívejme se nyní na řešení sudoku z obecného pohledu, tedy přesněji z pohledu splňování omezujících podmínek. Pokud každé políčko reprezentujeme proměnnou s doménou $\{1, ..., 9\}$, jak jsme to udělali v příkladu 2, tak právě popsaný postup odpovídá vyškrtávání hodnot z domén proměnných odpovídajících příslušným políčkům.

Vyškrtnutí hodnoty znamená, že daná hodnota nemůže být pro políčko použita. Vyškrtávání je „řízeno" podmínkou mezi proměnnými, vyškrtnuta je vždy jen hodnota, která podmínku nesplňuje. Například, pokud se číslo 1 nachází v nějakém políčku prvního řádku, tak je vyškrtnuto z domén proměnných všech ostatních políček na prvním řádku. Obecně se tomuto postupu říká *filtrace domén.*



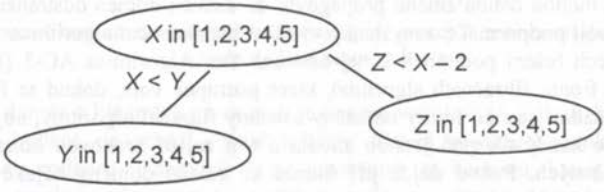
Obr. 6.3 Odvození možných pozic čísla dvě v pravém horním čtverci 3 × 3 použitím informace o jeho existujícím umístění v řádcích a sloupcích.

Filtrace domén obecně spočívá v odstraňování hodnot z domény proměnné, které nesplňují nějakou podmínku nad touto proměnnou. Uvažujme například proměnné $X$ a $Y$ s identickou doménou {1, 2, 3, 4, 5} a podmínku $X < Y$ mezi těmito proměnnými. Hodnotu 1 můžeme zřejmě vyřadit z domény proměnné $Y$, protože v doméně proměnné $X$ neexistuje žádná hodnota $v$ taková, že dvojice $(v,1)$ splňuje podmínku $X < Y$. Neexistuje tedy žádné řešení problému, ve kterém by byla zároveň hodnota 1 přiřazena do $Y$ a podmínka $X < Y$ byla splněna. Říkáme, že hodnota 1 nemá v doméně proměnné $X$ žádnou *podporu*. Podobně můžeme z domény proměnné $X$ vyřadit hodnotu 5, která nemá podporu v doméně proměnné $Y$. Pro popsanou filtraci domén tedy stačí, aby každá podmínka, kterou chceme při popisu problému používat, měla přiřazený filtrační algoritmus odstraňující z domén hodnoty bez podpor. Pro podmínky zadané extenzionálním způsobem, tj. jako seznam $n$-tic hodnot, které podmínku splňují, je zpravidla nutné pro každou hodnotu najít potřebnou podporu explicitně, tj. prozkoumáním příslušných $n$-tic hodnot. To lze udělat obecným algoritmem nezávislým na obsahu konkrétní relace. Pro podmínky zadané aritmetickou nebo logickou formulí bývá filtrační algoritmus často ušitý na míru příslušné podmínce, resp. typu podmínky a pro efektivní filtraci využívá sémantiky podmínky. Například filtrační algoritmus pro již zmíněnou podmínku $X < Y$ může pracovat takto: z domény proměnné $X$ vyřadí všechny hodnoty větší nebo rovné maximální hodnotě z aktuální domény proměnné $Y$ (tj. číslo 5 v našem příkladu) a z domény proměnné $Y$ vyřadí všechny hodnoty menší nebo rovné minimální hodnotě z aktuální domény proměnné $X$ (tj. číslo 1). Takovýto algoritmus tedy nemusí kontrolovat každou hodnotu zvlášť a zjišťovat, jestli má podporu v aktuálních doménách ostatních proměnných v podmínce nebo ne, ale filtraci provede „najednou" pro celou doménu. Formálně můžeme filtrační algoritmus podmínky $X < Y$ zapsat dvěma přiřazovacími pravidly:

$$\text{dom}(X) \leftarrow \{ v \mid v \in \text{dom}(X) \wedge v < \max(\text{dom}(Y)) \}$$
$$\text{dom}(Y) \leftarrow \{ v \mid v \in \text{dom}(Y) \wedge v > \min(\text{dom}(X)) \},$$

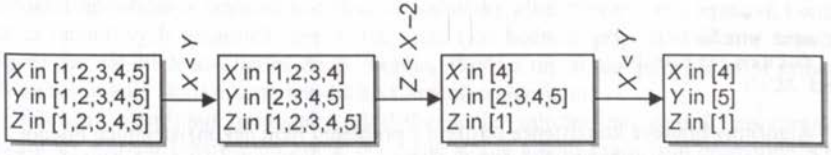
kde dom($X$) je doména proměnné $X$. Čtenář jistě nahlédne, že takto se odstraní právě jen hodnoty bez podpory.

Pokud doména proměnné $V$ obsahuje pouze hodnoty mající podporu vzhledem k dané podmínce C, hovoříme o *hranové konzistenci* (AC – arc consistency) proměnné $V$ vzhledem k podmínce C. Tento pojem vznikl z grafové reprezentace problému splňování omezujících podmínek, kde vrcholy představují proměnné a (hyper)hrany představují podmínky (viz obrázek 6.4). Filtrace domén tedy hledá největší podmnožinu domény proměnné tak, aby proměnná s touto zmenšenou doménou byla hranově konzistentní vzhledem k dané podmínce.




Obr. 6.4 Síť podmínek graficky reprezentuje problém splňování omezujících podmínek.

Pojem hranové konzistence můžeme nyní rozšířit na celý problém CSP – říkáme, že problém splňování omezujících podmínek je hranově konzistentní, pokud je každá proměnná hranově konzistentní ke všem podmínkám, které ji obsahují (Mackworth, 1977). Otázkou zůstává jak hranové konzistence problému dosáhnout. Zřejmě je potřeba spustit filtrační algoritmus každé podmínky, která se v problému vyskytuje. Jak ale ukazuje obrázek 6.5, nemusí to stačit, protože filtrací domény může dojít k tomu, že hodnota, která dosud byla podporována, svoji podporu ztratí. Například po filtraci domény proměnné $X$ podmínkou $Z < X - 2$ svoji podporu vzhledem k podmínce $X < Y$ ztratily hodnoty 2, 3 a 4 proměnné $Y$. Pro obnovení hranové konzistence proměnné $Y$ je tedy nutné tyto hodnoty z domény proměnné $Y$ odstranit opakovanou filtrací podle podmínky $X < Y$.




Obr. 6.5 Zmenšování domén proměnných opakovaným voláním filtračních algoritmů jednotlivých podmínek.

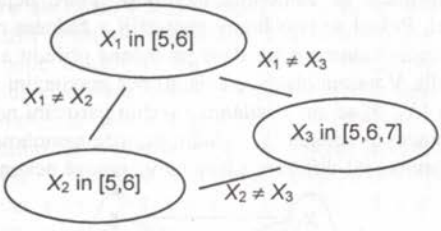
Pro dosažení hranové konzistence problému je obecně potřeba filtrační algoritmy volat opakovaně, dokud dochází ke zmenšování domén. Toto opakované volání filtračních algoritmů ukončíme ve chvíli, kdy v rámci jedné iterace (zavolání filtračních algoritmů všech podmínek) nedojde ke změně domén. Protože v dané iteraci žádný filtrační

algoritmus neodstranil žádnou hodnotu, jsou všechny proměnné hranově konzistentní ke všem podmínkám, a tedy i celý problém je hranově konzistentní. Postup pro dosažení (hranové) konzistence budeme obecně nazývat *konzistenční algoritmus*, někdy se také používá pojem *propagační algoritmus*. Konzistenční algoritmus, který v každé iteraci volá filtrační algoritmy všech podmínek, se nazývá AC-1 (Mackworth, 1977), nejedná se ale o v praxi používaný algoritmus. Zásadní nevýhodou algoritmu AC-1 je zbytečné volání filtračních algoritmů, které nevede k redukci domén. Pokud se totiž v rámci iterace změní například pouze doména jedné proměnné, řekněme $X$, nemá smysl opakovaně volat filtrační algoritmy podmínek, které proměnnou $X$ vůbec neobsahují (jako to dělá AC-1, který i po malé změně volá filtrační algoritmy všech podmínek). Obecně má cenu opakovat filtraci pouze u těch podmínek, kde se od posledního volání příslušného filtračního algoritmu změnila doména některé zúčastněné proměnné. Pouze tyto filtrační algoritmy totiž mohou danou změnu propagovat do dalších domén odstraněním hodnot, které ztratily svoji podporu. Této myšlenky využívá konzistenční algoritmus AC-3, který patří v systémech řešení podmínek k nejrozšířenějším. Algoritmus AC-3 (Mackworth, 1977) používá frontu filtračních algoritmů, které postupně volá, dokud se fronta nevyprázdní. Na začátku jsou do fronty umístěny všechny filtrační algoritmy, abychom měli zajištěno, že se každý alespoň jednou zavolá a tím zajistí hranovou konzistenci příslušných proměnných. Pokud dojde při filtraci ke změně domény nějaké proměnné, přidají (vrátí) se do fronty filtrační algoritmy podmínek, které tuto proměnnou obsahují. Tím je zaručeno, že změna domény bude propagována do dalších proměnných a že po doběhnutí algoritmu budou všechny proměnné hranově konzistentní ke všem podmínkám. Následující kód zobrazuje základní schéma algoritmu AC-3. Vstupem algoritmu je CSP problém (D reprezentuje množinu domén jednotlivých proměnných), výstupem jsou buď redukované domény hranově konzistentního problému, nebo indikace vyprázdnění nějaké domény, která znamená, že problém nemá řešení:

```
procedure AC-3(V,D,C)
  Q ← C
  while non-empty Q do
    select c from Q
    D' ← c.FILTER(D)
    if any domain in D' is empty then return (fail,D')
    Q ← Q ∪ {c'∈C | ∃x∈var(c') D'x≠Dx} - {c}
    D ← D'
  end while
  return (true,D)
end AC-3
```

Algoritmy hranové konzistence odstraní z problému řadu nekonzistentních hodnot, tj. hodnot, které nelze přiřadit příslušné proměnné v žádném přípustném řešení. Otázkou zůstává, jestli lze takto vyřešit libovolný problém splňování podmínek. Jak již bylo řečeno, pokud při filtraci dojde k vyprázdnění nějaké domény, víme, že žádné přípustné řešení neexistuje. Pokud po filtraci obsahují všechny domény právě jednu hodnotu, máme řešení, které splňuje všechny podmínky. V případě, že ve více doménách zůstaly dvě či více hodnot, nemáme bohužel garantováno, že tyto hodnoty lze v nějakém řešení použít a tedy ani to, že řešení existuje. Obrázek 6.6 ukazuje hranově konzistentní

problém, kde hodnoty 5 a 6 nelze přiřadit do proměnné $X_3$ přestože jsou hranově konzistentní. Pokud by v doméně této proměnné nebyla hodnota 7, problém by dokonce neměl žádné řešení, přestože je hranově konzistentní.



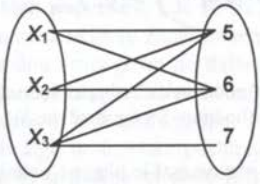

Obr. 6.6 Hranově konzistentní síť podmínek obsahující hodnoty, které nepatří do žádného řešení (hodnoty 5 a 6 proměnné $X_3$).

Filtrace domén u binárních nerovností je obecně velmi slabá. Pokud domény obsahují dvě či více hodnot, nedochází vlastně k žádné filtraci, protože libovolná hodnota má vždy nějakou podporu v doméně druhé proměnné (tu hodnotu, která se od ní liší). Podíváme-li se ale na obrázek 6.6 z nadhledu, můžeme odvodit, že hodnoty 5 a 6 nelze přiřadit proměnné $X_3$, protože tyto hodnoty už jsou „rezervovány" pro proměnné $X_1$ a $X_2$ (i když ještě není rozhodnuto, jaká hodnota bude do které proměnné přiřazena). Tento nadhled na části problému se skutečně reálně používá, a to formou tzv. *globálních podmínek*. Množinu jednodušších podmínek, v našem případě binárních nerovností, totiž můžeme nahradit jedinou podmínkou nad stejnými proměnnými, v našem případě ji nazveme all-different (požadujeme, aby hodnoty všech proměnných v podmínce byly různé). Zavedení globální podmínky má samozřejmě smysl pouze v případě, že jsme pro tuto podmínku schopni sestrojit efektivní filtrační algoritmus, který buď odstraní více nekonzistencí než při použití množiny jednodušších podmínek, nebo odstraní stejné nekonzistence, ale rychleji než by se toho dosáhlo obecným mechanismem AC-3 nad množinou jednodušších podmínek. V případě all-different existuje filtrační algoritmus, který dělá podmínku hranově konzistentní, a je tedy speciálně schopen odvodit, že v předchozím příkladě musí být hodnoty 5 a 6 vyřazeny z domény proměnné $X_3$ (uvědomme si, že tentokrát hovoříme o hranové konzistenci podmínky all-different, ne o hranové konzistenci jednotlivých binárních nerovností, která tyto hodnoty připouští). Tento filtrační algoritmus navrhl Jean-Charles Regin (Regin, 1994) a my si teď jeho základní princip představíme jako ukázku komplexnějšího filtračního algoritmu.

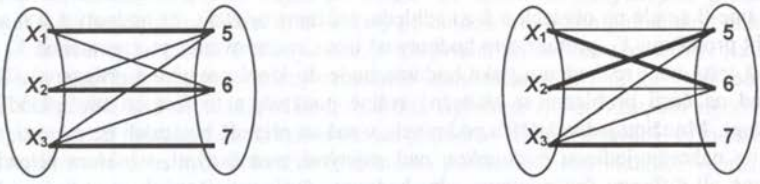
Filtrační algoritmus podmínky all-different je založen na vnitřní reprezentaci podmínky formou bipartitního grafu. Na jedné straně tohoto grafu jsou proměnné a na straně druhé hodnoty. Hrany vedou pouze mezi proměnnými a hodnotami (proto bipartitní graf), proměnná je spojena hranou s hodnotou právě tehdy, když je daná hodnota v doméně proměnné. Obrázek 6.7 ukazuje reprezentaci podmínky z předchozího příkladu.

Vlastní filtrační algoritmus je založen na hledání maximálního párování v grafu podmínky. Maximální párování je taková množina disjunktních hran (nemají společný vrchol), která je největší. Párování tak vlastně ukazuje možné přiřazení hodnot proměn-

ným tak, že různé proměnné mají různou hodnotu. V našem grafu by tedy mělo existovat maximální párování, jehož velikost je právě počet proměnných. Pokud existují pouze menší párování, nelze nalézt přípustné přiřazení proměnných a problém tedy nemá řešení. V opačném případě se zaměříme na hrany, které nejsou součástí žádného maximálního párování. Pokud se tyto hrany nepoužijí v žádném maximálním párování, znamená to, že příslušnou hodnotu nelze dané proměnné přiřadit a můžeme ji tedy z domény proměnné vyřadit. V našem příkladě existují dvě maximální párování (viz obrázek 6.8) a hrany ($X_3$, 5) a ($X_3$, 6) se ani v jednom z těchto párování nevyskytují. Hodnoty 5 a 6 tedy můžeme z domény proměnné $X_3$ vyřadit, protože nemohou být součástí žádného řešení splňujícího podmínku all-different a jsou tedy hranově nekonzistentní.




Obr. 6.7 Bipartitní graf reprezentující podmínku all-different.




Obr. 6.8 Možná maximální párování v bipartitním grafu podmínky all-different.

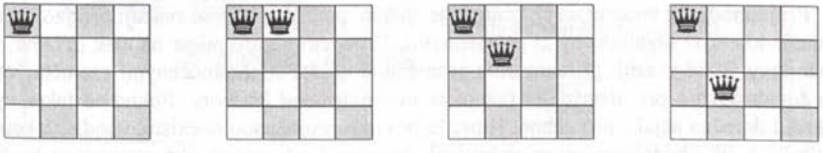
Podmínka all-different je typickou ukázkou, jak lze algoritmy z jiných oblastí, v tomto případě z teorie grafů, vhodně využít pro konstrukci efektivního filtračního algoritmu nějaké (globální) podmínky. Dnes existují desítky různých globálních podmínek ať už obecných podobného typu jako all-different, nebo specializovaných na konkrétní aplikační oblasti, jako jsou například podmínky modelující disjunktivní zdroje v rozvrhovacích problémech (Baptiste a kol., 2001). Množství těchto podmínek, které se navíc neustále zvětšuje, má bohužel negativní dopad na snadnost použití technologie splňování omezujících podmínek. Při tvorbě efektivních formálních modelů, o kterých budeme hovořit později, je totiž potřeba se v tomto množství dostupných podmínek dobře orientovat a vědět, jaké podmínky se pro popis daného problému hodí. Proto se v poslední době výzkumné aktivity, jako je třeba australský projekt G12, zaměřují na oblast automatického generování, resp. upravení omezujících podmínek z formální specifikace problému.

## 6.4 Prohledávací algoritmy

Filtrační algoritmy popisované v předchozí kapitole jsou schopny odstranit řadu nekonzistencí. Jak už jsme ale říkali, v řadě případů to pro úplné vyřešení problému nestačí, a to ani v případě použití globálních podmínek. Je sice teoreticky možné rozšířit pojem konzistence tak, že bude uvažovat více (potenciálně všechny) podmínek najednou, časová a prostorová složitost příslušných konzistenčních algoritmů ale rychle poroste až na exponenciální úroveň, což omezuje praktickou využitelnost tohoto přístupu. Konzistenční techniky vyššího řádu, jako je třeba konzistence po cestě (Mackworth, 1977), se proto využívají jen v určitých speciálních problémech, jako jsou jednoduché temporální sítě (Dechter a kol., 1991), kde je jejich časová a prostorová složitost pod kontrolou. Pro řešení obecných problémů se ale nehodí.

Pro řešení obecných problémů splňování omezujících podmínek se nejčastěji hranová konzistence kombinuje s nějakým prohledávacím algoritmem, který, krátce řečeno, rozhoduje zbývající disjunkce v problému, tj. přiřazení hodnot proměnným s doménami o více prvcích. Prohledávací algoritmy samozřejmě obecně také mají exponenciální časovou složitost, lze je ale jednodušeji implementovat a také zpravidla mají mnohem menší paměťové nároky než konzistenční techniky vyšších řádů. Spojení filtračních technik s prohledávacími algoritmy často vede k mylnému závěru, že technologie omezujících podmínek odpovídá jednoduché enumeraci. V dalším textu si ukážeme, že tomu tak není, že filtrační algoritmy mohou s prohledávacími algoritmy velmi dobře spolupracovat a že vzniklá kombinace je silnější než jednotlivé techniky použité odděleně. Řešící algoritmy tentokrát budeme prezentovat na problému *N* královen (viz Příklad 3).

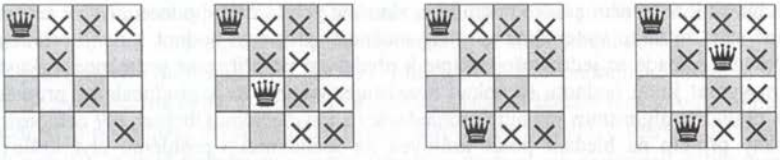
Naši cestu prohledávacími algoritmy začneme u klasického *prohledávání s navracením* (backtracking). Pro začátek zapomeneme na filtrační techniky, které odstraňují nekonzistentní hodnoty, a podíváme se na to jak řešit problém splňování podmínek pouze pomocí prohledávání prostoru možných ohodnocení proměnných. Naším cílem je najít hodnoty všech proměnných tak, aby byly splněny všechny podmínky. Jednoduchý způsob jak toho dosáhnout je postupně vybírat hodnoty pro proměnné tak, aby podmínky nad již ohodnocenými proměnnými byly splněny. Hodnoty budeme z domény systematicky vybírat v určitém pořadí a pokud přiřazení hodnoty do proměnné vede k neúspěchu, tj. k nesplnění nějaké podmínky, zkusíme prostě další hodnotu. Může se samozřejmě stát, že takto vyčerpáme všechny možnosti přiřazení hodnot k dané proměnné. V takovém případě se jednoduše vrátíme k předchozí již přiřazené proměnné a zkusíme pro ni vybrat jinou hodnotu (a pokud neexistuje, vrátíme se k předposlední proměnné atd.). Odtud se algoritmus jmenuje prohledávání s navracením. Obrázek 6.9 demonstruje popsaný princip na hledání pozic královen na šachovnici v problému čtyř královen. Přesněji řečeno, pro jednotlivé královny umístěné do sloupců hledáme umístění v řádku.



Obr. 6.9 Postup řešení problému čtyř královen jednoduchou enumerací.

Popsaným způsobem hledání řešení konstruujeme *strom prohledávání*, kdy postupně rozšiřujeme částečné ohodnocení proměnných, dokud nenalezneme úplné ohodnocení, které je řešením problému. Protože systematicky prozkoumáváme všechna „rozumná" ohodnocení proměnných, musíme řešení nalézt, pokud existuje. Rozumným ohodnocením zde rozumíme částečné přiřazení hodnot do proměnných takové, že podmínky nad ohodnocenými proměnnými jsou splněny. Popsaný algoritmus systematicky prochází domény jednotlivých proměnných a protože při tom pouze kontroluje splnění podmínek, můžeme ho nazvat jednoduchou enumerací. Jak už jsme se ale zmínili, splňování podmínek není jednoduchou enumerací a v následujících odstavcích ukážeme, jak mohou filtrační algoritmy výrazně zmenšit prohledávaný prostor a odlišit tak prohledávání použité při splňování podmínek od jednoduché enumerace.

Při jednoduché enumeraci se provádí test podmínek až po přiřazení proměnných. Algoritmus se tedy dívá jakoby zpět v ohodnocení, jestli nově vybraná hodnota je kompatibilní s již ohodnocenými proměnnými. V případě, že se objeví problém, řeší se opět pohledem zpět, tj. změnou hodnoty nějaké již přiřazené proměnné. Existují pokročilejší metody, které dokáží odhalit důvod problému (proč nelze danou proměnnou ohodnotit) a místo prostého kroku o jednu proměnnou zpět, skočí až na proměnnou, jejíž aktuální hodnota za tímto neúspěchem stojí. Tyto techniky se někdy souhrnně nazývají inteligentní prohledávání s navracením a patří sem metody, jako je backjumping (Gaschnig, 1979) a backmarking (Haralick, Elliot, 1985). Všechny tyto metody ale mají společné to, že se „pouze" snaží analyzovat důvod neúspěchu a potom ho opravit. Říká se jim proto *metody pohledu zpět* (look back methods). Obecně lepším způsobem řešení problémů je zřejmě předcházení neúspěchů místo zotavení se z nich. Neúspěchům můžeme předcházet jedině tak, že se při prohledávání budeme dívat vpřed, tj. na dosud neohodnocené proměnné, a pokusíme se odhadnout, jaký vliv bude mít vybraná hodnota na tyto proměnné. Zde již přichází na řadu filtrační algoritmy, které tím, že odfiltrují hodnoty nekompatibilní s aktuálním přiřazením, dokáží „dopředu" zjistit, že vybraná hodnota může v budoucnu způsobit nemožnost výběru hodnoty jiné proměnné (kvůli nesplnění nějaké podmínky). Obrázek 6.10 demonstruje, jak se taková filtrace dělá použitím podmínek obsahujících proměnnou, do které se právě přiřadila hodnota.

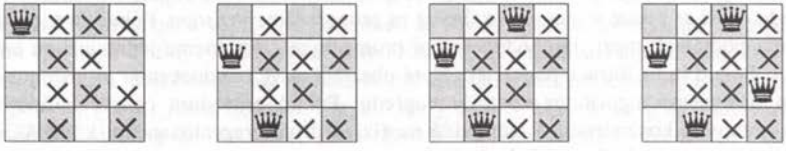


Obr. 6.10 Postup řešení problému čtyř královen kontrolou podmínek dopředu.

Při jednoduché enumeraci se kontroluje splnění podmínek, které svazují přiřazovanou proměnnou s již ohodnocenými proměnnými. V novém algoritmu se naopak díváme na podmínky, které svazují přiřazovanou proměnnou s ještě neohodnocenými proměnnými, a z domén těchto proměnných vyřazujeme nekonzistentní hodnoty. Pokud se takto vyprázdní doména nějaké proměnné, víme, že pro tuto proměnnou neexistuje hodnota kompatibilní s již ohodnocenými proměnnými. Na rozdíl od jednoduché enumerace to ale zjistíme dříve, ne až ve chvíli, kdy při prohledávání k této proměnné dojdeme. Protože

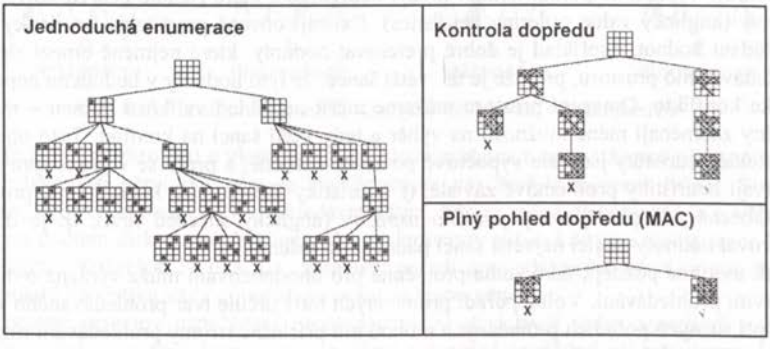
podmínky kontrolujeme dopředu a v doménách necháváme pouze hodnoty kompatibilní se všemi již přiřazenými proměnnými, není potřeba provádět kontrolu splnění podmínek mezi ohodnocenými proměnnými, kterou dělá jednoduchá enumerace. Podle tohoto principu se algoritmus nazývá *kontrola dopředu* (anglicky forward checking). Samozřejmě, pokud se algoritmus začne vracet, je potřeba odfiltrované hodnoty do domén zpátky vrátit.

Kontrola dopředu volá filtrační algoritmy pouze pro podmínky obsahující právě ohodnocovanou proměnnou a nějakou dosud neohodnocenou proměnnou. Nebere ale v úvahu podmínky, které právě ohodnocovanou proměnnou neobsahují. Samozřejmě silnější filtrace, a tedy i dřívějšího odhalení nějakého konfliktu, lze dosáhnout tím, že se zavolají filtrační algoritmy všech podmínek, dokud nedosáhneme plné hranové konzistence. Této technice se říká *(plný) pohled dopředu* (anglicky full look ahead) nebo také udržování *hranové konzistence* (MAC – Maintaining Arc Consistency). Obrázek 6.11 ukazuje, že se takto můžeme dobrat řešení v mnohem menším počtu kroků prohledávacího algoritmu.



Obr. 6.11 Postup řešení problému čtyř královen udržováním hranové konzistence.

Samozřejmě udržování konzistence něco stojí a jednotlivé kroky jsou tedy pomalejší, než je tomu u jednoduché enumerace. Vzhledem k výraznějšímu prořezání prohledávacího stromu se ale tato metoda ve většině případů vyplatí. Obrázek 6.12 srovnává stromy prohledávání jednoduché enumerace, kontroly dopředu a plného pohledu dopředu při řešení problému čtyř královen. Výhoda metody MAC oproti jednoduché enumeraci je zde zřejmá.




Obr. 6.12 Stromy prohledávání pro různé techniky řešení problému čtyř královen.

Algoritmus MAC, tj. kombinace prohledávání s udržování konzistence, je dnes hlavní technikou splňování omezujících podmínek. Formálně můžeme tento algoritmus popsat následujícím kódem. Algoritmus postupně ohodnocuje (značkuje) proměnné, odtud jeho anglický název labelling:

```
procedure labelling(V,D,C)
  if all variables from V are assigned then return V
  select not-yet assigned variable x from V
  for each value v from Dx do
    (TestOK,D') ← consistent(V,D,C∪{x=v})
    if TestOK=true then
      R ← labelling(V,D',C)
      if R ≠ fail then return R
  end for
  return fail
end labelling
```

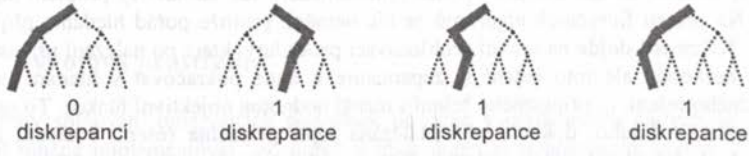
Uvědomme si, že stejnou strukturu mají i algoritmy jednoduché enumerace a kontroly dopředu – liší se pouze v tom, co se skrývá za procedurou *consistent*. Pokud je to pouze kontrola podmínek mezi již ohodnocenými proměnnými, dostaneme jednoduchou enumeraci. Pokud je to filtrace podmínek, které obsahují nově ohodnocenou proměnnou $x$, potom dostaneme algoritmus kontroly dopředu. Pokud procedura *consistent* ukrývá plnou hranovou konzistenci, tj. odpovídá například dříve prezentovanému kódu AC-3, získáme algoritmus plného pohledu dopředu.

Kromě volby procedury *consistent* obsahuje prezentovaný prohledávací algoritmus ještě dva další skryté parametry, které je potřeba při implementaci zvolit. Jedná se o volbu proměnné pro ohodnocování a určení pořadí hodnot, v jakém se budou do vybrané proměnné přiřazovat. Samozřejmě pořadí hodnot výrazně ovlivňuje efektivitu řešení. Pokud bychom na prvním místě vždy vybrali hodnotu, která je součástí řešení, potom by se prohledávací algoritmus nemusel vracet a přípustné řešení bychom našli jedním průchodem. To je ale pouze hypotetický případ, protože předpokládá, že známe řešení problému (známe-li vždy správnou hodnotu libovolné proměnné, známe i řešení problému). V praxi se pořadí hodnot určuje heuristikou, které říkáme *heuristika výběru hodnoty* (anglicky value ordering heuristics). Existují obecně použitelné heuristiky pro uspořádání hodnot, například je dobré preferovat hodnoty, které nejméně omezí zbytek prohledávaného prostoru, protože je tak větší šance, že tyto hodnoty v budoucnu nepovedou ke konfliktu. Omezení prostoru můžeme měřit například velikostí domén – menší domény znamenají méně možností na výběr a tedy větší šanci na konflikt. Tyto obecně použitelné heuristiky jsou ale výpočtově poměrně náročné, a proto se v praxi zpravidla používají heuristiky problémově závislé, tj. heuristiky šité na míru konkrétnímu problému. Obecně zde platí *princip prvního úspěchu* (anglicky succeed first), tj. je dobré preferovat hodnoty mající největší šanci padnout do řešení.

Jak uvidíme později, také volba proměnné pro ohodnocování může výrazně ovlivnit efektivitu prohledávání. Volba pořadí proměnných totiž určuje tvar prohledávaného stromu a při různých pořadích proměnných mohou mít příslušné stromy prohledávání různou velikost. Při volbě proměnných se obecně používá *princip prvního neúspěchu* (fail first) mající následující motivaci. Pokud ohodnocení nějaké proměnné povede ke konfliktu, tj.

neúspěchu v prohledávání, je dobré to zjistit na začátku prohledávání, protože tím zabráníme prozkoumávání řady kombinací hodnot, které by kvůli nemožnosti ohodnotit inkriminovanou proměnnou stejně vedly k neúspěchu. Jinou interpretací téhož principu je, že proměnné, pro které je těžké najít správnou hodnotu, by se měly ohodnocovat co nejdříve. Tyto principy se objevují v obecných *heuristikách výběru proměnné* (variable ordering heuristics), kdy jsou například preferovány proměnné mající menší doménu (menší šance na nalezení správné hodnoty) nebo účastnící se více podmínek (těžší nalezení hodnoty splňující podmínky). I při výběru proměnných je možné používat problémově závislé heuristiky, které například preferují ohodnocení klíčových proměnných. Na rozdíl od výběru hodnot se zde ale často využívají právě obecné heuristiky. Ve většině případů se preferují proměnné s menší doménou a v případě shody velikostí domén se dříve berou proměnné, které jsou ve více podmínkách.

Heuristiky jsou u reálných problémů důležitým průvodcem prohledávacího algoritmu. Základní vlastností dobrých heuristik je schopnost ve velkém počtu případů doporučit správný směr prohledávání (hovoříme teď o heuristikách výběru hodnoty). Na cestě vedoucí z kořene do řešícího listu prohledávaného stromu tedy bude jen málo „odboček", kdy heuristika nedoporučila správný směr. Tyto odbočky budeme nazývat *diskrepancemi*. Heuristiky tedy nejsou vševědoucí a mohou se mýlit. Otázkou nyní je, jak se z případného omylu heuristiky rychle vzpamatovat. Klasický prohledávací algoritmus, který jsme prezentovali v předchozích odstavcích, se prostě vrátí k předposlední přiřazené proměnné a zkusí pro ni jinou hodnotu. Jinými slovy takový algoritmus předpokládá, že se heuristika mýlí spíše ke konci prohledávání než na začátku. Je to ale správný předpoklad? Heuristiky „nevaří z vody", ale doporučují hodnotu na základě dostupných informací. Obecně lze očekávat, že více informací bude přítomných později v prohledávání než na jeho začátku, a tedy že heuristika je méně spolehlivá na začátku prohledávání než na jeho konci. Pokud na dané cestě nenajdeme řešení, chyba je spíše na začátku než na konci. Z pohledu heuristik se tak prohledávání s navracením chová iracionálně, protože se snaží nejdříve opravit chybu na konci cesty.

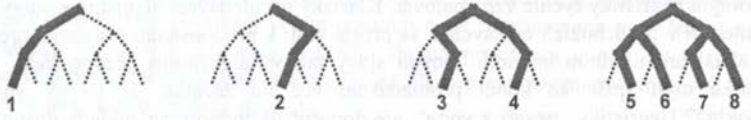



Obr. 6.13 Pořadí procházených větví při prohledávání s diskrepancemi.

Předchozí pozorování o vlastnostech heuristik můžeme nyní využít pro přeuspořádání prohledávaných větví tak, aby se zvýšila šance na dřívější nalezení řešení. Stručně řečeno, nejprve bychom měli prohledávat větve, kde je méně diskrepancí, a u větví se stejným počtem diskrepancí bychom měli preferovat ty větve, kde jsou diskrepance blíže k počátku. Obrázek 6.13 ukazuje „správné" pořadí prohledávaných větví (zleva doprava), pokud heuristika vždy doporučuje vydat se levou větví.

Popsané principy uspořádání prohledávaných větví tvoří jádro algoritmů *prohledávání s diskrepancemi*, které jsou populární a úspěšné například při řešení rozvrhovacích problémů. Prvním takovým algoritmem bylo prohledávání s omezenými diskre-

pancemi (LDS – Limited Discrepancy Search) (Harvey a Ginsberg, 1995), které pracuje v iteracích a v každé iteraci prohledává všechny větve mající maximálně daný počet diskrepancí. Tyto větve prochází v pořadí od větví, v nichž jsou diskrepance blíže k počátku. Algoritmus začíná s limitem na počet diskrepancí nula a v každé další iteraci tento limit o jednu diskrepanci zvětšuje. Nevýhodou LDS je to, že v každé další iteraci prochází také všechny větve z předchozí iterace, což přirozeně celkový běh při více iteracích zpomaluje. Richard Korf (Korf, 1996) proto navrhl vylepšenou verzi LDS, kdy se v každé iteraci procházejí pouze větve s přesně daným počtem diskrepancí. Větve v rámci iterace ale neuspořádává podle pozice diskrepancí, nýbrž je bere v klasickém pořadí prohledávání s navracením. Zajímavý přístup k diskrepancím navrhl Toby Walsh (Walsh, 1997) v podobě prohledávání s hloubkově omezenými diskrepancemi (DDS – Depth-bounded Discrepancy Search). Algoritmus opět pracuje v iteracích a v každé iteraci jsou diskrepance povoleny jen do určité hloubky. Navíc je na této hloubce diskrepance vyžadována, čímž se jednoduše zabrání tomu, aby se prohledávaly stejné větve jako v předchozí iteraci. Omezením hloubky se omezí maximální počet diskrepancí v iteraci a zároveň se diskrepance koncentrují v horní části prohledávaného stromu. Obrázek 6.14 ukazuje pořadí prozkoumávaných větví v jednotlivých iteracích DDS.




Obr. 6.14 Procházené větve při prohledávání s hloubkově omezenými diskrepancemi.

Až dosud jsme se zabývali pouze hledáním přípustných řešení problému splňování podmínek, řada reálných problémů má ale spíše *optimalizační charakter*. Jak bylo uvedeno v úvodu, formálně takový problém popíšeme tak, že ke specifikaci CSP přidáme objektivní funkci, jejíž hodnotu chceme maximalizovat nebo minimalizovat (pro jednoduchost budeme nadále uvažovat pouze minimalizaci). Jak ale takový problém budeme řešit? Na úrovni filtračních algoritmů se nic nemění, protože pořád hledáme přípustné řešení. Ke změně dojde na úrovni prohledávací procedury, která po nalezení přípustného řešení neskončí, ale toto řešení si zapamatuje a bude pokračovat v hledání lepšího přípustného řešení, tj. přípustného řešení s menší hodnotou objektivní funkce. To se bude opakovat tak dlouho, dokud budou nalézána lepší přípustná řešení. Poslední z nich potom bude řešení optimální. Tomuto přístupu se říká *metoda větví a mezí* (branch-and-bound).

Praktická realizace metody větví a mezí vypadá tak, že se zavede nová objektivní proměnná $V$ a objektivní funkce *Obj* nad množinou proměnných *Xs* se převede do nové tzv. objektivní podmínky $V = Obj(Xs)$, která se přidá k ostatním podmínkám. Při nalezení přípustného řešení má proměnná $V$ právě hodnotu objektivní funkce, označme ji *Bound*. Pro nalezení lepšího řešení stačí do množiny podmínek přidat podmínku $V < Bound$, čímž máme garantováno, že další přípustné řešení bude lepší než poslední nalezené řešení. Následující formální kód ukazuje, jak lze libovolný algoritmus Solve pro řešení CSP převést na optimalizační algoritmus řešící problém metodou větví a mezí:

```
procedure BB-Min(Variables, V, Constraints)
   Bound ← sup
   NewSolution ← fail
   repeat
      Solution ← NewSolution
      NewSolution ← Solve(Variables,Constraints ∪ {V<Bound})
      Bound ← value of V in NewSolution (if any)
   until NewSolution = fail
   return Solution
end BB-Min
```

Vraťme se ještě na chvíli k podmínce $V = Obj(Xs)$, která má, stejně jako jiné podmínky, přiřazen nějaký filtrační algoritmus. Také zde se kvalita filtrace odrazí v efektivitě prohledávání. Propagace směrem od $Xs$ k $V$ vlastně odhaduje hodnotu objektivní funkce na základě aktuálních domén proměnných $Xs$. Pokud je tento odhad dostatečně dobrý, můžeme při prohledávání rychleji objevit oblasti, kde se nenachází dobré řešení. Naopak propagace od $V$ směrem k $Xs$ umožňuje odfiltrovat hodnoty, které nejsou součástí dostatečně kvalitního řešení, a tím se oblastem, kde neleží dobré řešení, vyhnout (pokud máme k dispozici horní mez pro $V$). Získání dobré meze pro proměnnou $V$ je tedy důležitým předpokladem silnějšího prořezání prohledávaného prostoru a tedy rychlejšího výpočtu. Uvědomme si totiž, že při hledání optimálního řešení metodou větví a mezí se prochází víceméně celý prohledávaný prostor. Propagace klasických podmínek nám pouze umožní vyříznout některé oblasti, kde se nenachází přípustné řešení, zatímco propagace objektivní podmínky nám umožní vyříznout některé oblasti, kde se nenachází lepší než dosud nalezené řešení. To je také důvod, proč hledání optimálního řešení a zvláště pak dokázání jeho optimality je časově mnohem náročnější než nalezení přípustného řešení. V reálných problémech ale není optimalita často vyžadována a potom stačí algoritmus zastavit, jakmile nalezne přípustné řešení, s jehož kvalitou je uživatel spokojen.

## 6.5 Vývojová prostředí

Algoritmy splňování omezujících podmínek popsané v předchozích částech je samozřejmě možné implementovat „od nuly" v libovolném programovacím jazyku. V řadě případů by to ale byla ztráta času, protože pro mnoho programovacích jazyků již byly systémy pro splňování omezujících podmínek vyvinuty a formou knihoven do těchto jazyků integrovány. Snad nejpřirozenější je integrace omezujících podmínek do logického programování, tedy hlavně do programovacího jazyku Prolog. Jedním z nejrozšířenějších prologovských systémů je SICStus Prolog, který nabízí velké množství knihoven pro různé problémy včetně knihoven pro řešení omezujících podmínek. Je zde implementována řada globálních omezení a systém také poskytuje rozhraní pro integrování vlastních filtračních algoritmů. Podobnou knihovnu pro řešení omezujících podmínek poskytuje také prologovský systém ECLiPSe šířený jako Open Source produkt. Systém ECLiPSe se ale více soustředí na hybridní řešení problémů, v nichž se metody omezu-

jících podmínek kombinují s lokálním prohledáváním či technikami matematického programování (obsahuje rozhraní na externí simplexové řešiče XPRESS-MP a CPLEX).

Prolog přirozeně není jediným programovacím jazykem, v němž lze využívat technik splňování omezujících podmínek. Mezi aplikačními programátory je velmi populární jazyk C++ a samozřejmě i pro něj existuje řada knihoven implementujících techniky splňování omezujících podmínek. Komerčně velmi úspěšné je řešení ILOG CP francouzské firmy ILOG, která patří k největším dodavatelům optimalizačních řešení. Jedná se dnes patrně o nejrozsáhlejší (a také nejdražší) balík produktů z oblasti splňování omezujících podmínek. ILOG CP obsahuje vlastní jádro pro řešení podmínek (Solver), na které jsou navázané specializované moduly pro rozvrhování (Scheduler) a řešení transportačních problémů (Dispatcher). Volně dostupnou C++ knihovnou pro řešení omezujících podmínek je naopak systém Gecode poskytující otevřené prostředí pro tvorbu nových filtračních algoritmů i prohledávacích strategií.

Podobnou popularitu jako jazyk C++ zažívá i multiplatformní jazyk Java, který má také řadu dostupných knihoven pro řešení omezujících podmínek. Z komerčních produktů je možné jmenovat například systém Koalog, volně dostupný je naopak systém Choco, kde se autoři snaží nejen o řešení problémů s podmínkami, ale také o automatické hledání důvodů, proč se řešení problému případně nedaří, což je jeden z aktuálních směrů výzkumu. Autoři obou zmíněných systémů se ve Francii podíleli na popularizaci programování s omezujícími podmínkami právě demonstrováním toho, jak lehce lze touto technologií řešit velmi populární sudoku.

Z dalších systémů podporujících řešení omezujících podmínek stojí za zvláštní zmínku volně dostupný systém Mozart. Jedná se o vývojové prostředí založené na programovacím jazyku Oz, který v sobě kombinuje vlastnosti objektově-orientovaného a deklarativního programování spolu s řešením omezujících podmínek a paralelizací výpočtu.

Odkazy na řadu dalších existujících systémů pro splňování omezujících podmínek je možné najít na webových stránkách (Barták, 1998).

## 6.6 Modelování problémů

V předchozích oddílech jsme se několikrát zmínili o deklarativním charakteru problému splňování omezujících podmínek – uživatel formálně popíše problém jako CSP a počítač (s příslušným systémem pro řešení omezujících podmínek) ho sám vyřeší. V případě problému splňování omezujících podmínek nad konečnou doménou je počítač skutečně vždy schopen nalézt řešení nebo dokázat jeho neexistenci. Bohužel, někdy to trvá „trochu" déle.

Čas výpočtu pro nalezení řešení není dán jen použitou technologií (prohledávání), ale obecnou složitostí problému CSP, který patří mezi tzv. NP-úplné problémy (Garey, Johnson, 1979). V praxi to znamená, že pro některé problémy, které lze formulovat jako CSP, nejsou známy deterministické algoritmy řešící problém rychleji než v exponenciálním čase a obecně se předpokládá, že takové algoritmy ani neexistují. S takovými problémy asi „nic nenaděláme", ukazuje se ale, že také formulace problému formou CSP výraznou měrou ovlivňuje efektivitu řešení. Jinými slovy, efektivita řešení záleží na tom,

jakým způsobem je problém popsán jako problém splňování omezujících podmínek. Pro pozorného čtenáře to jistě není překvapivé. V předchozích kapitolách jsme například ukázali, že filtrace různých podmínek popisujících stejný problém se může lišit. Protože silnější filtrace zmenšuje prohledávaný prostor, může tak pozitivně ovlivnit efektivitu prohledávání a tedy celkový čas pro nalezení řešení.

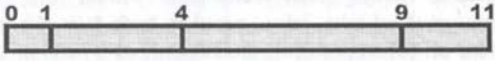
Uvědomme si, že pro daný kombinatorický problém často existuje řada jeho popisů formou proměnných a podmínek, které jsou ekvivalentní, pokud jde o definici množiny přípustných řešení, ale nemusí být ekvivalentní, pokud jde o efektivitu nalezení řešení. V tomto případě sehrává znalost řešicí technologie velkou roli a dobrá formulace problému je odvozena právě od toho, jak se problém bude řešit. To je jeden z hlavních důvodů, proč je vhodné při řešení problémů technikami splňování omezujících podmínek rozumět alespoň základním řešicím principům. Pochopení řešicích algoritmů totiž umožní vědomě konstruovat formální popisy problému, které mají šanci na efektivní vyřešení, popř. tyto formalizace inkrementálně upravovat tak, aby se zlepšila časová efektivita hledání řešení. O formulaci problému budeme nadále hovořit jako o *modelování* a formální popis problému formou CSP budeme nazývat *modelem*. V následujícím výkladu ukážeme několik technik, které se používají pro tvorbu efektivně řešitelných modelů.

Naši cestu za efektivním modelováním problémů opět začneme krátkou zastávkou u hádanky sudoku. V prvotním modelu (Příklad 2) jsme vlastnosti problému popsali sadou binárních nerovností. Jak jsme ale následně ukázali, takovýto model propaguje poměrně málo (téměř vůbec) a zanechává v doménách mnoho nekonzistentních hodnot. Vlastní řešení problému, tj. nalezení hodnot proměnných, tak ve velké míře leží na prohledávacím algoritmu. Pokud nepoužijeme nějakou chytrou heuristiku, stane se v tomto případě z prohledávání poměrně triviální enumerace možných ohodnocení, tedy postup řešení, který není příliš elegantní ani efektivní. Stačí ale jedna změna modelu a vše je jinak. Množiny binárních nerovností pro proměnné v řádcích, sloupcích a vyznačených čtvercích 3×3 nahradíme podmínkami all-different (Regin, 1994), které dokáží vyřadit mnohem více nekonzistencí. Potom stačí takovýto model udělat hranově konzistentní a většina běžně dostupných problémů sudoku je okamžitě vyřešena. Prohledávání v těchto případech ani není potřeba, protože hranová konzistence většinou zanechá v každé doméně právě jeden prvek, což odpovídá nalezení řešení. Pro nejtěžší problémy sudoku je potřeba použít o něco silnější verzi konzistence, které se říká *jednotková hranová konzistence* (SAC – singleton arc consistency) (Prosser a kol., 2000). V tomto případě se ale nejedná o plně deduktivní techniku – SAC je založen na zkoušení hodnot, které zhruba odpovídá prohledávání s mělkým navracením (při navracení v prohledávacím stromu jdeme zpět maximálně o jednu úroveň). Shrneme-li předchozí řádky, tak při znalosti technologie splňování omezujících podmínek je vytvoření programu pro automatické řešení problémů sudoku otázkou několika minut. Tedy pokud máme k dispozici řešič omezujících podmínek obsahující filtrační algoritmus podmínky all-different, jehož naprogramování rozhodně není triviální záležitost.

Podívejme se nyní na jiný problém, který na rozdíl od sudoku patří mezi těžké problémy kombinatorické optimalizace – problém hledání optimálního Golombova pravítka. Golombovo pravítko velikosti $M$ je pravítko, na kterém je umístěno $M$ značek tak, že vzdálenosti mezi všemi dvojicemi značek (nejen sousedními) se navzájem liší.

Optimální Golombovo pravítko je potom nejkratší takové pravítko. Obrázek 6.15 ukazuje optimální Golombovo pravítko velikosti 5.

Jak již bylo řečeno, hledání optimálního Golombova pravítka je těžký problém, a to i pro poměrně malé velikosti $M$. Ve skutečnosti není známo optimální řešení pro pravítka velikosti 24 a větší, přestože existují přípustná řešení pro pravítka až do velikosti 150 ovšem bez důkazu optimality. Problém optimálního Golombova pravítka není pouze těžkým teoretickým problémem, ale jeho řešení má také praktické využití například v radioastronomii.




Obr. 6.15 Optimální Golombovo pravítko velikosti 5.

Na příkladě hledání optimálního Golombova pravítka nyní ukážeme, že i drobné změny modelu CSP mohou mít velký vliv na efektivitu řešení. V následujících odstavcích nejprve budeme prezentovat základní model problému následovaný jeho možnými rozšířeními zlepšujícími efektivitu řešení. Nebude se v tomto případě sice jednat o model, který by se vyrovnal nejlepším speciálním technikám pro řešení tohoto problému (s navrženým modelem budeme v rozumném čase schopni najít optimální pravítka do velikosti 12), ale tento model nám umožní demonstrovat některé základní modelovací techniky pro zlepšení efektivity řešení. Poznamenejme jen, že nebudeme měnit obecnou techniku splňování podmínek, pouze budeme daný problém popisovat různými sadami podmínek.

Přirozeným způsobem, jak popsat problém Golombova pravítka jako problém splňování omezujících podmínek, je použití proměnných pro pozice jednotlivých značek. Pro pravítko velikosti $M$ tedy budeme mít $M$ základních proměnných $X_1,\ldots, X_M$ určujících, kde se příslušná značka nachází. Protože nás zajímá relativní vzdálenost první a poslední značky, můžeme první značku rovnou umístit do pozice 0 a za doménu pro ostatní proměnné zvolit přirozená čísla. Jako horní odhad maximální pozice můžeme použít hodnotu $M^2$ (nejlepší známá pravítka jsou vždy kratší než tato hodnota), čímž získáme konečné domény. Značky navíc můžeme uspořádat zleva doprava, abychom zbytečně nemuseli prozkoumávat různé permutace pozic. To lze jednoduše zařídit použitím podmínek tvaru $X_i < X_{i+1}$. Zbývá popsat základní vlastnost Golombova pravítka, totiž že vzdálenosti mezi dvojicemi značek jsou navzájem různé. Pro tuto vlastnost můžeme zavést nové distanční proměnné $D_{i,j}$ pro $i < j$, které jsou na základní proměnné napojeny podmínkami tvaru $D_{i,j} = X_j - X_i$. Poučeni modelem pro sudoku svážeme tyto distanční proměnné jednou podmínkou all-different. Celý základní model Golombova pravítka potom vypadá takto:

$X_1 = 0$
$X_1 < X_2 < \ldots < X_M$
$\forall i<j\ D_{i,j} = X_j - X_i$
all_different($\{D_{1,2}, D_{1,3}, \ldots, D_{M,M-1}\}$)

Základní model popisuje podmínky, které musí být splněny, abychom získali korektní Golombovo pravítko. Při hledání optimálního pravítka stačí minimalizovat hodnotu proměnné $X_M$. Jak ale bývá u těžších problémů běžné, základní model není příliš

efektivní, pokud jde o rychlost jeho vyřešení, a je vhodné tento model rozšířit o další podmínky, které efektivitu řešení zvýší.

Jednou z efektivních modelovacích technik je *odstranění symetrií* z problému. Symetrickým řešením rozumíme takové řešení, které lze získat jednoduchou konverzí z jiného řešení. Například Golombovo pravítko [0, 1, 4, 9, 11] je symetrické pravítku [0, 2, 7, 10, 11] – jedná se o jeho zrcadlový obraz. Symetrické řešení zde získáme jednoduše tak, že příslušné hodnoty pozic značek odečteme od čísla 11. V obou případech se jedná o přípustná řešení, která tedy musíme při hledání optimálního řešení vzít v úvahu. Jinými slovy, je potřeba prohledat symetrické části prostoru možných řešení, což zdržuje a na kvalitě řešení to nepřidá. Uvědomme si, že takto prohledáváme nejenom symetrické části, kde se nacházejí řešení, ale také symetrické části, kde se žádná řešení nenacházejí. Pokud například zjistíme, že vzdálenost 3 mezi prvními dvěma značkami není přípustná, prohledávací algoritmus musí totéž ověřit pro poslední dvě značky, i když z principu symetrie víme, že vzdálenost 3 není přípustná ani mezi posledními dvěma značkami, a je tedy zbytečné příslušnou část prostoru možných řešení explicitně prohledávat. Prohledávání symetrických částí prostoru možných řešení můžeme zabránit přidáním podmínky, která symetrická řešení zakáže – takové podmínce se říká *bořič symetrií* (symmetry breaking constraint). V případě Golombova pravítka může taková podmínka vyžadovat, aby vzdálenost mezi prvními dvěma značkami byla vždy menší než vzdálenost mezi posledními dvěma značkami, čímž se jednoduše odstraní zrcadlová řešení. Formálně tato podmínka vypadá takto:

$$D_{1,2} < D_{M-1,M}.$$

Přidáním podmínky rozbíjející symetrie můžeme zkrátit čas pro nalezení optimálního řešení o více než polovinu, jak ukazuje tabulka 6.1. Poznamenejme ještě, že velké množství symetrií jsme již odstranili v základním modelu, v němž jsme značky uspořádali zleva doprava použitím podmínek $X_i < X_{i+1}$ a že pro odstraňování symetrií z problému existují i další techniky (Smith, 2001).

Druhou efektivní modelovací technikou, kterou si zde představíme, je přidání *implicitních podmínek*. Implicitní podmínka je taková podmínka, kterou lze odvodit z existujících podmínek. Příkladem může být odvození podmínky $A = D + 1$ z podmínek $A = B + C + 1$ a $D = B + C$. Podmínka $A = D + 1$ je přirozeně redundantní, tj. jinými slovy, každé řešení splňující základní podmínky bude splňovat také implicitní podmínku. Tato podmínka nám ale může posílit propagaci tím, že umožní vyřadit více nekonzistentních hodnot. Představme si, že v našem minipříkladu mají proměnné $B$, $C$ a $D$ doménu $\{0, 1\}$. Použitím hranové konzistence pro podmínku $A = B + C + 1$ odvodíme, že konzistentní hodnoty pro $A$ jsou 1, 2, 3. Hranová konzistence totiž „vidí" podmínku $D = B + C$ samostatně a není tedy schopna odvodit, že maximální hodnota součtu proměnných $B + C$ v podmínce $A = B + C + 1$ je 1. Pokud ale přidáme odvozenou podmínku $A = D + 1$, budou pro proměnnou $A$ hranově konzistentní pouze hodnoty 1 a 2. Nově přidaná podmínka tedy posílila filtraci domén a tím přispěla ke zmenšení prohledávaného prostoru. Přirozeně ne každá odvozená podmínka má tuto vlastnost a při přidávání implicitních podmínek je potřeba být opatrný – více podmínek znamená pomalejší filtraci (je potřeba testovat konzistenci přidaných podmínek) a tyto zvýšené náklady je potřeba vyvážit silnější filtrací.

V případě Golombova pravítka je možné posílit filtraci domén distančních proměnných $D_{i,j}$ tím, že přidáním implicitních podmínek pro tyto proměnné odvodíme lepší horní a dolní mez jejich hodnoty. $D_{i,j}$ popisuje vzdálenost mezi značkami $i$ a $j$ a tato vzdálenost se vlastně skládá ze vzdáleností značek $(i, i+1)$, $(i+1, i+2)$ $\ldots(j-1, j)$. Můžeme tedy psát

$$D_{i,j} = D_{i,i+1} + D_{i+1,i+2} + \ldots + D_{j-1,j}.$$

Protože víme, že jednotlivé vzdálenosti se navzájem liší, můžeme zdola odhadnout hodnotu $D_{i,j}$ jako součet prvních $(j-i)$ přirozených čísel, tedy

$$D_{i,j} \geq \Sigma_{j-i} = (j-i)*(j-i+1)/2.$$

Podobný princip můžeme použít také pro odhad maximální hodnoty $D_{i,j}$. Tentokrát si uvědomíme, že vzdálenost značek $i$ a $j$ je jednou ze vzdáleností mezi značkou 1 a značkou $M$:

$$X_M = X_M - X_1 = D_{1,M} = D_{1,2} + D_{2,3} + \ldots + D_{i-1,i} + D_{i,j} + D_{j,j+1} + \ldots + D_{M-1,M},$$
$$D_{i,j} = X_M - (D_{1,2} + \ldots D_{i-1,i} + D_{j,j+1} + \ldots + D_{M-1,M}).$$

Opět použijeme vlastnosti, že všechny vzdálenosti se navzájem liší, a tedy také vzdálenosti značek $(1, 2)$, $\ldots$, $(i-1, i)$, $(j, j+1)$, $\ldots$, $(M-1, M)$ jsou různé. Dolní odhad jejich součtu je potom součtem prvních $(M-1-j+i)$ přirozených čísel, což nám umožní definovat horní odhad hodnoty proměnné $D_{i,j}$ takto:

$$D_{i,j} \leq X_M - (M-1-j+i)*(M-j+i)/2.$$

Přidáním těchto implicitních podmínek pro proměnné $D_{i,j}$ opět výrazně zlepšíme efektivitu řešení problému. Tabulka 6.1 ukazuje srovnání časů řešení pro jednotlivé modely a demonstruje tak, že různé modelovací přístupy mohou výrazně ovlivnit čas řešení problému.

Tabulka 6.1 Čas nalezení optimálního Golombova pravítka pro různé modely (v milisekundách na počítači Mobile Pentium 4-M 1,70 GHz, 768 MB RAM)

| velikost pravítka | základní model | základní model + bořič symetrií | základní model + bořič symetrií + implicitní podmínky |
|---|---|---|---|
| 7 | 220 | 80 | 30 |
| 8 | 1 462 | 611 | 190 |
| 9 | 13 690 | 5 438 | 1 001 |
| 10 | 120 363 | 49 971 | 7 011 |
| 11 | 2 480 216 | 985 237 | 170 495 |

Na závěr části o modelování problémů se ještě podívejme, jak efektivitu řešení ovlivňuje volba prohledávací strategie. Nejedná se sice přímo o modelovací techniku, tj. o volbu vhodných proměnných a podmínek pro popis problému, ale volba prohledávací strategie je svým způsobem také deklarativní informace, která příslušný model doprovází při řešení klasickou technikou kombinace propagace podmínek a prohledávání. Zůstaneme u problému hledání optimálního Golombova pravítka a porovnáme dva způsoby uspořádání proměnných pro prohledávání a tři způsoby větvení.

V případě uspořádání proměnných budeme porovnávat klasickou obecnou heuristiku prvního neúspěchu (fail first) se statickým uspořádáním proměnných pro pozici značek, v němž značky umisťujeme zleva doprava, tj. od nejmenšího indexu. Uvědomme si zde, že distanční proměnné $D_{i,j}$ se nemusí přímo účastnit prohledávání, protože určením hodnot proměnných $X_i$ a propagací přes podmínky $D_{i,j} = X_j - X_i$ budou hodnoty distančních proměnných automaticky určeny. Tím, že vyloučíme distanční proměnné $D_{i,j}$ z prohledávání, jsme výrazně zmenšili prohledávaný prostor a tím zvýšili efektivitu hledání řešení.

Nahoře zmíněná uspořádání proměnných zkombinujeme s následujícími větvícími strategiemi. *Enumerační strategie* postupně zkouší vybrané proměnné přiřadit hodnoty z její aktuální domény od nejmenší po největší. Pokud jsou například v doméně proměnné $X$ hodnoty 5, 6, 7 a 8, chová se tato strategie jako řešení disjunkce $X = 5 \vee X = 6 \vee X = 7 \vee X = 8$, kdy se nejprve zkusí přidat podmínka $X = 5$, v případě neúspěchu se zkusí přidat podmínka $X = 6$, v případě dalšího neúspěchu se zkusí podmínka $X = 7$ a nakonec se zkusí podmínka $X = 8$. Druhou strategií je tzv. *kroková strategie*, kdy se nejprve zkusí nejmenší hodnota z domény proměnné a v případě neúspěchu se před dalším větvením tato hodnota vyřadí z domény. Jinými slovy kroková strategie se řídí disjunkcí $X = 5 \vee X \neq 5$. Na první pohled vypadá kroková strategie stejně jako strategie enumerační s tím rozdílem, že původně $n$-ární větvení (kde $n$ je počet hodnot v doméně) se převede na $n - 1$ binárních větvení. V případě obyčejného prohledávání s navracením tomu tak skutečně je, u algoritmu MAC ale hraje svoji roli také propagace, která se spustí nejen přiřazením hodnoty do proměnné ($X = 5$), ale také vyřazením hodnoty z domény proměnné ($X \neq 5$). Přidáním podmínky $X \neq 5$ tak můžeme odvodit nekonzistenci, aniž bychom museli explicitně zkoušet přiřazení všech hodnot různých od 5, což ve svém důsledku může řešení problému urychlit. Experimenty s Golombovým pravítkem (tabulka 6.2) tuto hypotézu potvrzují. Třetí běžná větvící strategie, která je zobecněním krokové strategie, je *strategie půlení domény*. Místo, abychom u proměnné zkoušeli konkrétní hodnotu, dojdeme k této hodnotě postupným „půlením" domény příslušné proměnné. V našem minipříkladě to odpovídá řešení disjunkce $X \leq 6 \vee X > 6$. Motivace skrytá za touto strategií je stejná jako u krokové strategie. Zmenšená doména je propagována do dalších proměnných a můžeme tak rychleji zjistit, v jaké části domény se nachází hodnota patřící do řešení, popřípadě, že řešení neexistuje pro žádnou hodnotu.

Tabulka 6.2 ukazuje srovnání časů řešení pro různé prohledávací strategie. V případě hledání optimálního Golombova pravítka se jako nejefektivnější ukazuje strategie postupného umisťování značek zleva doprava metodou půlení domény. Výrazně lepší časy pro toto statické uspořádání proměnných jsou dány charakterem problému, kdy postupné usazování značek vede k lepší propagaci než usazování „nahodilé" a tedy k většímu prořezání prohledávaného prostoru. U jiných problémů může být vhodnější jiná prohledávací strategie, zpravidla se jako první možnost zkouší kombinace heuristiky prvního neúspěchu s krokovou strategií (zvláště, pokud je k dispozici dobrá heuristika doporučující hodnotu proměnné). Poznamenejme ještě, že v případě hledání optimálního Golombova pravítka se používala metoda větví a mezí, protože se jedná o optimalizační problém.

Tabulka 6.2 Čas nalezení optimálního Golombova pravítka pro různé prohledávací strategie (v milisekundách na počítači Mobile Pentium 4-M 1,70 GHz, 768 MB RAM)

| velikost pravítka | zleva doprava | | | první neúspěch | | |
|---|---|---|---|---|---|---|
| | enum | krok | půlení | enum | krok | půlení |
| 7 | 30 | 30 | 30 | 40 | 60 | 40 |
| 8 | 220 | 190 | 200 | 390 | 370 | 350 |
| 9 | 1 182 | 1 001 | 921 | 2 664 | 2 384 | 2 113 |
| 10 | 8 782 | 7 011 | 6 430 | 20 870 | 17 545 | 14 982 |
| 11 | 209 251 | 170 495 | 159 559 | 1 004 515 | 906 323 | 779 851 |

Určení efektivity modelu daného problému je netriviální problém, který vyžaduje znalosti technik splňování omezujících podmínek. Obecně lze říci, že nejlepší je takový model, který dříve a více propaguje, viz (Marriot, Stuckey, 1998). V předchozích oddílech jsme popsali některé techniky, které mohou k takovému modelu vést. Za prvé je to použití globálních podmínek, za druhé je to odstranění symetrií a za třetí přidání implicitních podmínek.

## 6.7 Aplikační oblasti

Výsledky z oblasti umělé inteligence bývají veřejnosti často prezentovány formou řešení různých hříček, připomeňme například známý šachový program DeepBlue. Ani v této kapitole jsme neudělali výjimku a na prezentaci technologie splňování omezujících podmínek jsme používali příklady, jako je populární sudoku nebo problém *N* královen. To by ale v žádném případě nemělo vzbuzovat pocit, že uvedená technologie je vhodná pouze na řešení kombinatorických hříček. Její skutečná síla se totiž projevuje právě v řešení reálných problémů, když ostatní technologie selhávají. Vzhledem k obecnosti popisu problému pomocí omezujících podmínek je záběr této technologie, pokud jde o reálné aplikace, skutečně široký a sahá od analýzy struktury DNA přes rozvrhování a konfiguraci až třeba k animaci lidské tváře nebo geografickým informačním systémům.

Asi nejčastěji nachází omezující podmínky uplatnění v různých přiřazovacích a rozvrhovacích problémech, jejichž kombinatorický charakter je pro techniky splňování omezujících podmínek zvláště vhodný. S aplikacemi založenými na omezujících podmínkách se můžeme setkat při alokaci stojánek pro letadla (Dincbas, Simonis, 1991), přiřazení odbavovacích pultů na letištích (Chow, Perett, 1997) nebo alokaci kotvišť ve velkých překladištích (Perett, 1991). Omezující podmínky najdeme i v pozadí přípravy rozvrhů směn v nemocnicích (Chan a kol., 1998) nebo přiřazení posádek k vlakovým spojům (Focacci a kol., 1997). Technik splňování podmínek se používá při rozvrhování výroby vojenských i obchodních letadel (Bellone a kol., 1992), při plánování vrtů na ropných plošinách (Johansen, Hasle, 1997) nebo při optimalizaci těžby dřeva (Adhikary a kol., 1997). Úspěšnou aplikací je také optimální rozmisťování základových stanic u bezdrátových telekomunikačních sítí (Frühwirth, Brisset, 1998). Z nových oblastí, kde se začíná CP objevovat, lze jmenovat například návrh (authoring) multimediálních děl, analýzu běhu programů nebo generování kódu pro testování čipů (Naveh a kol., 2006).

V názvu kapitoly jsme naznačili, že aplikace omezujících podmínek míří do vesmíru a tak tomu je nejen v přeneseném významu. 24. října 1998 odstartovala z mysu

Canaveral raketa Delta II nesoucí na palubě kosmickou loď Deep Space 1, jejímž cílem byla Borrellyova kometa. Skutečným cílem projektu ale bylo ověření použitelnosti dvanácti nových pokročilých technologií, mezi kterými byl i tzv. vzdálený agent (RA – Remote Agent). Úkolem vzdáleného agenta bylo plně autonomní řízení kosmické lodi zahrnující plánování, exekuci i monitorování akcí, které loď provádí. Jako jedna z technik použitá při implementaci tohoto agenta byla právě propagace omezujících podmínek, jejímž cílem bylo omezit množství alternativ při plánování. Celý experiment byl velmi úspěšný, i když ukázal dnes známé slabiny tohoto přístupu, jako je nepredikovatelný čas prohledávání s navracením (drobná změna zadání problému může mít významný vliv na dobu běhu řešícího algoritmu). Omezující podmínky se používají i v dalších kosmických aplikacích, kdy vlastní systém pro řešení podmínek sice zůstává na Zemi, ale výsledek řešení problému se aplikuje v kosmu. Takovou aplikací je například rozvrhování a správa satelitů pro snímkování zemského povrchu (Bensana a kol., 1999) nebo plánování a rozvrhování operací známých marťanských vozítek Spirit a Opportunity pomocí programu MAPGEN (Bresina a kol., 2005).

## 6.8 Další zdroje

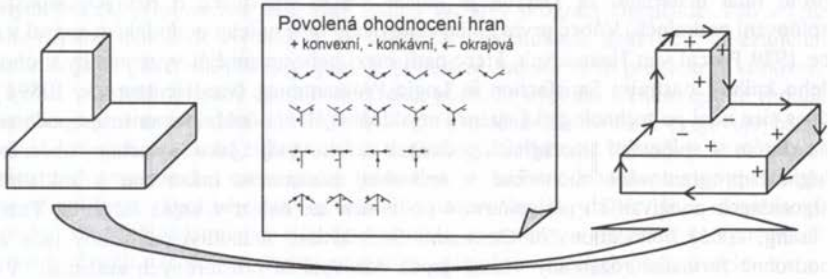
Technologie splňování omezujících podmínek již dosáhla věku jisté zralosti a existuje proto řada materiálů, ze kterých je možné čerpat informace o různých aspektech splňování podmínek. Vůbec první knihu věnovanou omezujícím podmínkám napsal v roce 1989 Pascal van Hentenryck, který patří mezi nejvýznamnější výzkumníky v oboru. Jeho kniha Constraint Satisfaction in Logic Programming (van Hentenryck, 1989) již dnes sice není po technologické stránce nejaktuálnější, ale dobře demonstruje podhoubí, na kterém se splňování omezujících podmínek začalo vyvíjet jako samostatný obor, totiž logické programování obohacené o splňování podmínek. Informace o základních algoritmech používaných při splňování podmínek lze nalézt v knize Edwarda Tsanga (Tsang, 2005) Foundations of Constraint Satisfaction. Jednotlivé algoritmy jsou zde podrobně formálně rozebrány včetně jejich časových a prostorových složitostí. Více z uživatelského pohledu se na omezující podmínky dívá kniha Programming with Constraints australských autorů Kima Marriotta a Petera Stuckeyho (Marriott a Stuckey, 1998). Čtenář zde najdu řadu příkladů jak modelovat problémy jako problémy splňování podmínek a kniha tak může posloužit jako dobrý úvodní text pro uživatele, kteří nepotřebují jít do hloubky řešících algoritmů, ale naopak potřebují okamžitě využít výhod technologie pro řešení svých aktuálních problémů. Čtenáři, které naopak zajímají právě techniky pro implementaci systémů řešících omezující podmínky, by určitě neměli opomenout knihu Christiana Schulteho: Programming Constraint Services (Schulte, 2002). Oblast efektivní implementace řešících algoritmů nebývá příliš zastoupena ve vědeckých článcích, přesto ale hraje nemalou roli v praktické použitelnosti celé technologie. Výborným výukovým materiálem pro studium omezujících podmínek je kniha Riny Dechterové: Constraint Processing (Dechter, 2003). Kniha přehledným stylem prezentuje základní témata celé oblasti, i když zpravidla nejde do takové hloubky jako Tsangova kniha. Zatím posledním přírůstkem do literatury je rozsáhlá sbírka Handbook of

Constraint Programming (Rossi a kol., 2006), která poskytuje podrobný pohled na různé oblasti splňování podmínek od autorů, kteří v daných oblastech aktivně působí.

Kromě tištěných materiálů existuje také řada on-line zdrojů informací o omezujících podmínkách. Odborná komunita má svůj webový portál CP-Online, kde jsou publikovány aktuální informace o dění v komunitě, jako jsou pořádané akce nebo pracovní příležitosti. Existuje také archiv Constraints Archive odkazů na materiály, jako jsou odborné články, sady benchmarků či existující implementace. Z on-line výukových materiálů je nejvíce odkazů na On-line Guide to Constraint Programming (Barták, 1998) popisující základní algoritmy splňování omezujících podmínek a obsahující odkazy na další materiály tutoriálového charakteru.

## 6.9 Závěr

Techniky splňování omezujících podmínek mají své počátky v sedmdesátých letech dvacátého století, kdy se výzkumníci jako David Waltz (Waltz, 1975) zabývali rozpoznáváním trojrozměrných objektů v dvourozměrných obrazech scény. Waltzova myšlenka hledání konzistentního označkování hran v obrazu scény (viz obrázek 6.16) vedla ke vzniku konzistenčního algoritmu, který je dnes známý jako AC-2.




Obr. 6.16 Použití konzistenčních technik při rozpoznávání obrazů.

Další výzkumníci v té době už řešili obecné problémy konzistence sítí podmínek (Montanary, 1974), (Mackworth, 1977) a z tohoto výzkumu pocházejí základy algoritmů a technik, jako jsou již zmíněné: hranová konzistence nebo konzistence po cestě. Větší rozvoj pak obor splňování omezujících podmínek začal po zjištění, že unifikace v logickém programování je vlastně speciálním případem splňování podmínek (Gallaire, 1985) – vzniklo *logické programování s omezujícími podmínkami* (CLP – Constraint Logic Programming) (Jaffar a Lasses, 1987). Později se omezující podmínky od logických programů opět oddělily a dnes je to zcela samostatný obor, jehož základy jsme představili v této kapitole. Soustředili jsme se především na hlavní proud řešících podmínek, tj. spojení prohledávání do hloubky s udržováním konzistence. Počátky v umělé inteligenci se zde nezapřou – prohledávání patří k jednomu ze základních řešících postupů UI. My jsme zde prezentovali i některé sofistikovanější prohledávací algoritmy, jako je

prohledávání s diskrepancemi. Omezující podmínky také ve stále větší míře vstřebávají výsledky dalších oborů, jako je diskrétní matematika a operační výzkum. To se projevuje například ve filtračních algoritmech pro globální podmínky, my jsme krátce představili podmínku all-different, nebo v hybridním přístupu k řešení problémů, o kterém jsme zde ale nehovořili (Jacquet-Lagreze, 1998). Celý obor splňování omezujících podmínek směřuje k cíli, kdy uživatel pouze formálně specifikuje problém a automatizovaný systém sám nalezne řešení. V řadě případů už to takto funguje, někdy ale musí uživatel počítači pomoci vhodným přeformulováním problému. Svatý grál deklarativního programování tedy zatím dosažen nebyl, omezující podmínky se mu ale v mnohém blíží.

### Literatura


Adhikary J., Hasle G., Misund G.: *Constraint Technology Applied to Forest Treatment Scheduling*. Proc. of Practical Application of Constraint Technology (PACT97), Londýn 1997.

Baptiste P., Le Pape C., Nuijten W.: *Constraint-based Scheduling: Applying Constraints to Scheduling Problems*. Kluwer Academic Publishers, Dordrecht 2001.

Barták R.: *On-line Guide to Constraint Programming*. Praha 1998. http://kti.mff.cuni.cz/~bartak/constraints/

Bellone J., Chamard A., Pradeless C.: *PLANE: – An Evolutive Planning System for Aircraft Production*. Proc. of Practical Application of Prolog (PAP92), Londýn 1992.

Bensana E., Lemaître M., Verfaillie G.: *Earth Observation Satellite Management*. Constraints: An International Journal, 4(3), 1999, s. 293–299.

Bresina J. L., Jónsson A. K., Morris P. H., Rajan K.: *Activity Planning for the Mars Exploration Revers*. Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS 2005), AAAI Press, 2005, s. 40–49.

*Constraints Archive:* http://4c.ucc.ie/web/archive/index.jsp

*CP Online:* http://slash.math.unipd.it/acp/

Dechter R., Meiri I., Pearl J.: *Temporal Constraint Networks*. Artificial Intelligence, 49, 1991, s. 61–95.

Dechter R.: *Constraint Processing*. Morgan Kaufmann, 2003.

Deep Space One project: http://nmp.nasa.gov/ds1/

Dincbas M., Simonis H.: *APACHE – A Constraint Based, Automated Stand Allocation Systems*. Proc. of Advanced Software Technology in Air Transport (ASTAIR91), Londýn 1991.

ECLiPSe, Prolog programming environment: http://www.eclipse-clp.org/

Focacci F., Lamma E., Mello P., Milano M.: *Constraint Logic Programming for the Crew Rostering Problem*. Proc. of Practical Application of Constraint Technology (PACT97), Londýn 1997.

Frühwirth T., Brisset P.: *Optimal Placement of Base Stations in Wireless Indoor Telecommunication*. Proc. Principles and Practice of Constraint Programming (CP98), Pisa, Itálie, 1998.

G12 project: http://www.g12.cs.mu.oz.au/

Gallaire H.: *Logic Programming: Further Developments*. IEEE Symposium on Logic Programming, Boston, IEEE, 1985.

Garey M. R., Johnson D. S.: *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman & Comp., San Francisco, 1979.

Gaschnig J.: *Performance Measurement and Analysis of Certain Search Algorithms*. Technical report CMU-CS-79-124, Carnegie-Mellon University, 1979

Gecode, C++ programming environment: http://www.gecode.org/

Golomb Ruler, results: http://www.research.ibm.com/people/s/shearer/grtab.html

Haralick R. M., Elliot G. L.: *Increasing tree search efficiency for constraint satisfaction problems.* Artificial Intelligence 14, 1980, s. 263–314.

Harvey W.D., Ginsberg M. L.: *Limited Discrepancy Search.* Proceedings of the Interna-tional Joint Conference on Artificial Intelligence, 1995, s. 607–613.

Chan P., Heus K., Weil G.: *Nurse Scheduling with Global Constraints in CHIP: GYMNASTE.* Proc of Practical Application of Constraint Technology (PACT98), Londýn 1998.

Chow K. P., Perett M.: *Airport Counter Allocation using Constraint Logic Programming.* Proc. of Practical Application of Constraint Technology (PACT97), Londýn 1997.

Choco, Java programming environment: http://choco.sourceforge.net/

ILOG CP, C++ programming environment: http://www.ilog.com/products/cp/

Jacquet-Lagreze E.: *Hybrid Methods for Large Scale Optimisation Problems: an OR Perspective.* Proc. of Practical Application of Constraint Technology (PACT98), Londýn 1998.

Jaffar J., Lassez J. L.: *Constraint Logic Programming.* Proc. The ACM Symposium on Principles of Programming Languages, ACM, 1987.

Koalog, Java programming environment: http://www.koalog.com/

Korf R. E.: *Improved Limited Discrepancy Search.* Proceedings of National Conference on Artificial Intelligence (AAAI-96), AAAI Press, 1996, s. 286–291.

Johansen B. S., Hasle G.: *Well Activity Scheduling – An Application of Constraint Reasoning.* Proc. of Practical Applications of Constraint Technology (PACT97), Londýn 1997.

Mackworth A. K.: *Consistency in Networks of Relations.* Artificial Intelligence 8(1), 1977, s. 99–118.

Marriot K., Stuckey P.: *Programming with Constraints: An Introduction.* The MIT Press, Cambridge, Mass., 1998.

Montanary U.: *Networks of constraints fundamental properties and applications to picture processing.* Information Sciences 7, 1974, s. 95–132.

Mozart, Oz programming environment: http://www.mozart-oz.org/

Naveh Y., Rimon M., Jaeger I., Katz Y., Vinov M., Marcus E., Shorek G.: *Constraint-Based Random Stimuli Generation for Hardware Verification.* Proceedings of IAAI, AAAI Press, 2006, s. 1720–1727.

Perett M.: *Using Constraint Logic Programming Techniques in Container Port Planning.* ICL Technical Journal, May, 1991, s. 537–545.

Prosser P., Stergiou K., Walsh T.: *Singleton Consistencies.* Proc. Principles and Practice of Constraint Programming (CP2000), Springer Verlag, 2000, s. 353–368.

Regin J.-C.: *A filtering algorithm for constraints of difference in CSPs.* Proceedings of 12th National Conference on Artificial Intelligence, AAAI Press, 1994, s. 362–367.

Rossi F., van Beek P., Walsh, T. (eds): *Handbook of Constraint Programming.* Elsevier, 2006.

Schulte C.: *Programming Constraint Services: High-Level Programming of Standard and New Constraint Services.* LNCS 2302, Springer-Verlag, 2002.

SICStus Prolog: http://www.sics.se/isl/sicstus.html

Smith B.: *Reducing Symmetry in a Combinatorial Design Problem.* Proceedings of CP-AI-OR2001, Wye College, Velká Británie, 2001, s. 351–359.

Tsang E.: *Foundations of Constraint Satisfaction.* Academic Press, Londýn 1995.

van Hentenryck P.: *Constraint Satisfaction in Logic Programming.* The MIT Press, Cambridge, Mass., 1989.

Walsh T.: *Depth-bounded Discrepancy Search.* Proceedings of 15th International Joint Conference on Artificial Intelligence, 1997, s. 1388–1393.

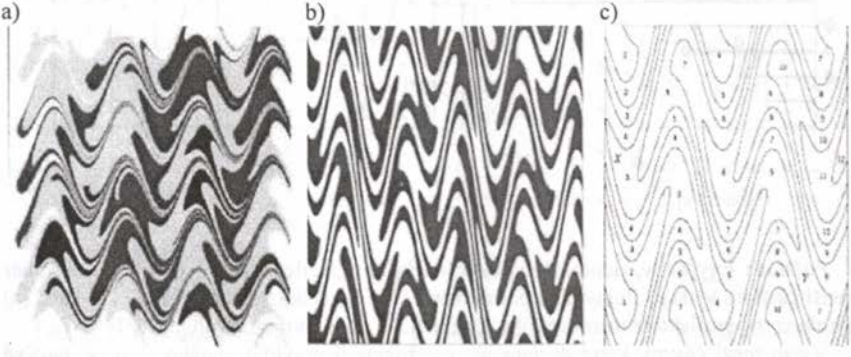
Waltz D. L.: *Understanding line drawings of scenes with shadows.* Psychology of Computer Vision, McGraw-Hill, New York 1975.

# 7 Lze popsat krásu matematickým výrazem?

## 7.1 Úvod

Matematika a umění se těší plodnému vztahu již od starověku. Zlatý řez, který se stal zásadním paradigmatem umění západní civilizace už v době starověkého Egypta, ale není jediným pojítkem mezi oběma disciplínami. Jen namátkou, orientální obklady, renesanční perspektiva nebo kartografické projekce – společně s dalšími příklady, které jsou popisovány a ilustrovány v tomto textu – mohou být považovány za předchůdce dnešních sofistikovaných matematických programů, z nichž mezi nejpopulárnější patří fraktální umění.

Je možné zachytit krásu do matematické formule? Odpověď na tuto otázku nemusí být jednoznačná. „Filosofové, kteří si myslí, že není krásy v matematice, se mýlí. Matematika je ve skutečnosti vrcholnou formou krásy", říká Aristoteles. „Umění začíná tam, kde končí výpočty", namítá Le Corbusier, vytyčuje tak jasnou hranici mezi oběma disciplínami.



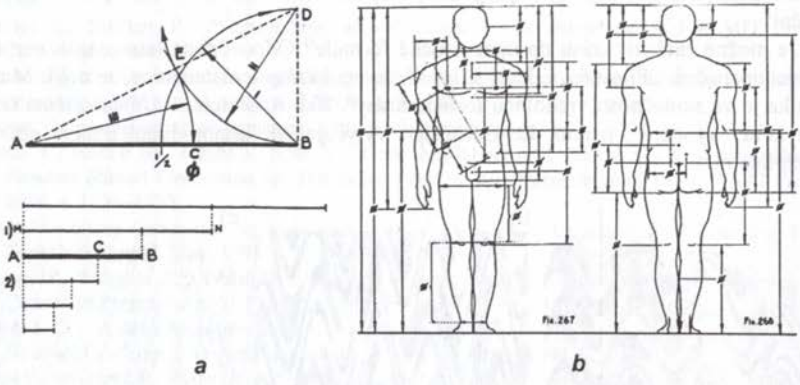

Obr. 7.1

Je přímka popsaná matematickým výrazem matematikou, pokud se nakreslí ve třídě na tabuli, ale změní se v umění, pokud je předložena veřejnosti na plátně? Je za rozdíl odpovědný úmysl toho, kdo přímku kreslí? Je rozdíl mezi matematikou a uměním závislý pouze na přístupu diváka? Obrázek 7.1 ilustruje popsanou situaci. První obrázek (a) je

umění nabízené na internetu. Druhé dva obrázky (b) a (c) pochází z knihy Structural Geology (Ragan, 1984) a zobrazují model vrásnění ve vrstvě usazenin. Kdokoli může vybarvit obrázek (c) a dostat tak obrázek (a). Vzor na obrázku (b) lze vytvořit pomocí jednoduchých matematických operací vycházejících z počátečních přímých a rovnoběžných pruhů. Tento postup může být proveden za pomoci počítače pod dohledem jak umělce, tak i vědce. Oba výsledky budou identické. Každý z nich se ovšem bude dívat na svůj výsledek jinýma očima. Divák bez znalosti původu „uměleckého díla" bude vnímat hlavně estetickou stránku. A bude-li se mu zamlouvat – umělecké dílo se mu bude líbit.

## 7.2 Proporce a zlatý řez: jak „krásně" rozdělit celek na části

Pokud uvažujeme bezrozměrný bod, je jako takový nedělitelný. V případě, že uvažujeme přímku, je jedno kam umístíme dělicí bod vzhledem k tomu, že obě části budou stejně dlouhé – nekonečné. Problémy nám nastanou v okamžiku, pokud chceme rozdělit konečnou část na dvě různé části: kde je nejlepší místo pro řez?




Obr. 7.2

Počínaje Egypťany, odpověď na tuto otázku je: „Nejlepší místo je tam, kde poměr menší části ku větší je stejný jako poměr větší části ku celku." Na obrázku 7.2 v části (a) je zobrazena grafická konstrukce toho, co umělci a architekti nazývají „zlatý řez".

Vztah mezi částmi, které dostaneme, použijeme-li pravidlo zlatého řezu, se nazývá „zlaté číslo". Jeho hodnota je 0,618 nebo 1,618 podle toho, jak je podíl definován. Stejný vztah můžeme najít i ve Fibonacciho posloupnosti, nazvané po italském středověkém matematikovi Leonardovi da Pisa[1]. Poměr každých dvou následujících čísel v posloupnosti (3, 5, 8, 11 atd.) se při zvyšujícím se $n$ blíží k hodnotě 1,618 nebo 0,618 – zlatému číslu.

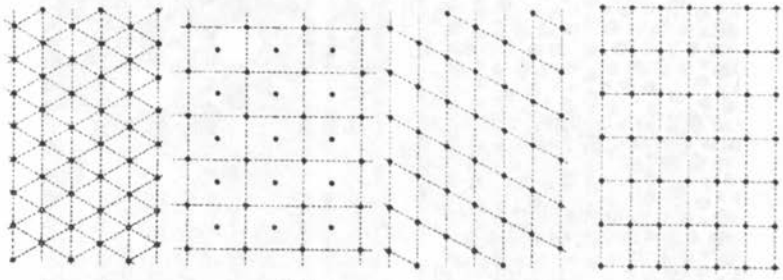
[1] Fibonacci byla jeho přezdívka.

Pravidlo zlatého řezu bylo následně rozšířeno do tvorby obrázků, těles a dokonce i barev a hudby na základě přesvědčení, že dokonalost – jeden z atributů KRÁSY po mnoho let – je silně svázána se striktním poměrem mezi částmi. Obrázek 7.2(b) ilustruje, jak sochař může vytvořit lidské tělo, když za pomoci zlatého čísla najde relativní délky všech jeho částí.

## 7.3 Vzory v ploše a prostoru: jednotvárnost a rytmus

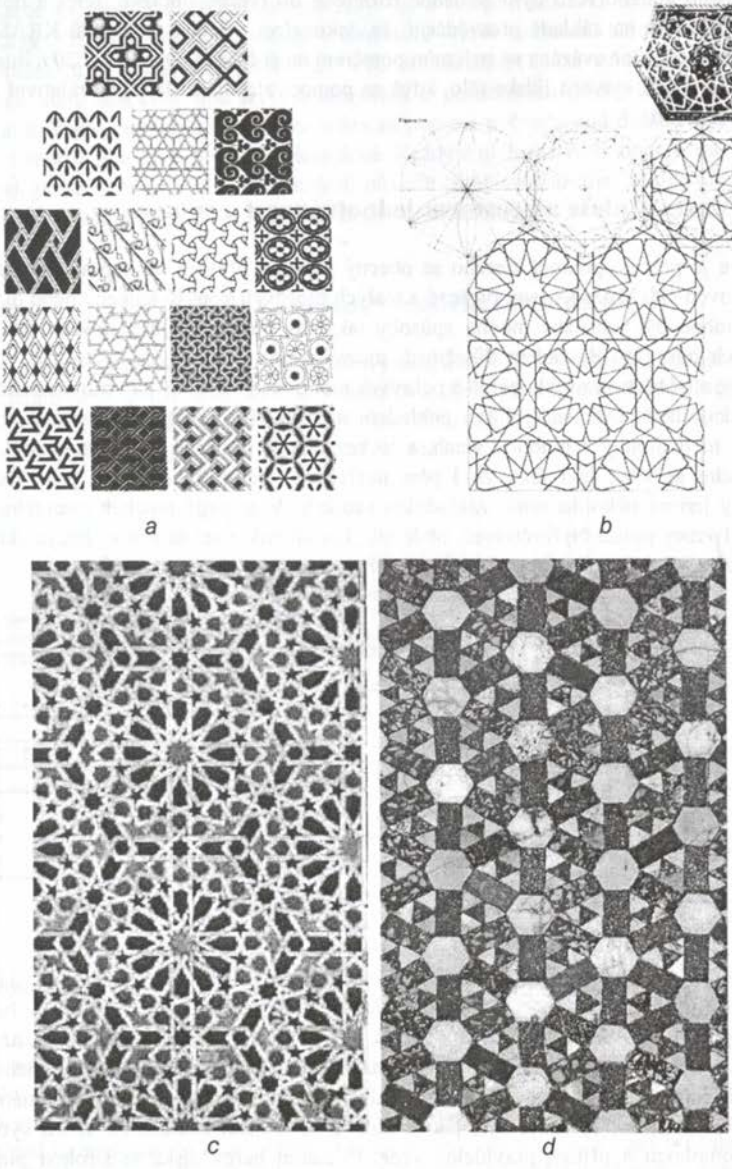
Teselace je proces, pomocí kterého se obecný polygon převádí na nepravidelnou trojúhelníkovou síť. Mozaiky jsou tvořené z malých plochých kousků kamene nebo dlaždic, které mohou být umístěny mnoha způsoby tak, aby pokryly nějaký povrch. Mozaiky z různých dob jsou dokladem důležitosti mozaiky pro architekty a výtvarníky. Nejen mozaiky, ale i různé vzory koberců a pokrývek mohou být připsány na vrub mozaikám.

Není obtížné, byť jen rychlým pohledem na tato díla zjistit, že principy mozaiky designu měli dlouhý a hluboký vztah, a to bez ohledu na dobu a umělecký styl. Překvapujícím se však jeví fakt, že i přes nepřeberné množství designů, jsou všechny založeny jen na několika málo základních vzorech. V případě plochého povrchu jsou základní vzory pouze čtyři: čtverec, obdélník, kosodélník a šestiúhelník. Jak je ukázáno na obrázku 7.3, jsou základem všech známých designů.




Obr. 7.3

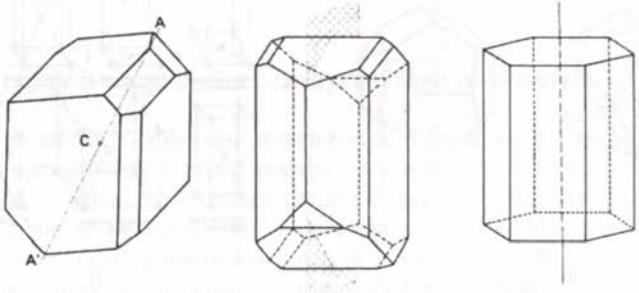
Designy na obrázku 7.4(a) ilustrují 17 základních vzorů, z nichž většina byla, až překvapivě, vlastní různým kulturám již od starověku. Orientální umění v zemích pod vlivem islámu, který zakazuje zobrazovat lidi i zvířata, ovládlo brilantně geometrický design. Obrázek 7.4(b) zobrazuje, jak je možné kombinací základních vzorů vytvořit velmi komplexní a přitom pravidelný vzor. Přidáním barev získáme širokou plejádu fascinujících designů. Obrázky 7.4(c) a 7.4(d) představují reprodukce výjimečných ukázek mozaiky orientálního a italského původu.




Obr. 7.4

## 7.4 Symetrie v ploše a prostoru: od zrcadel ke krystalům

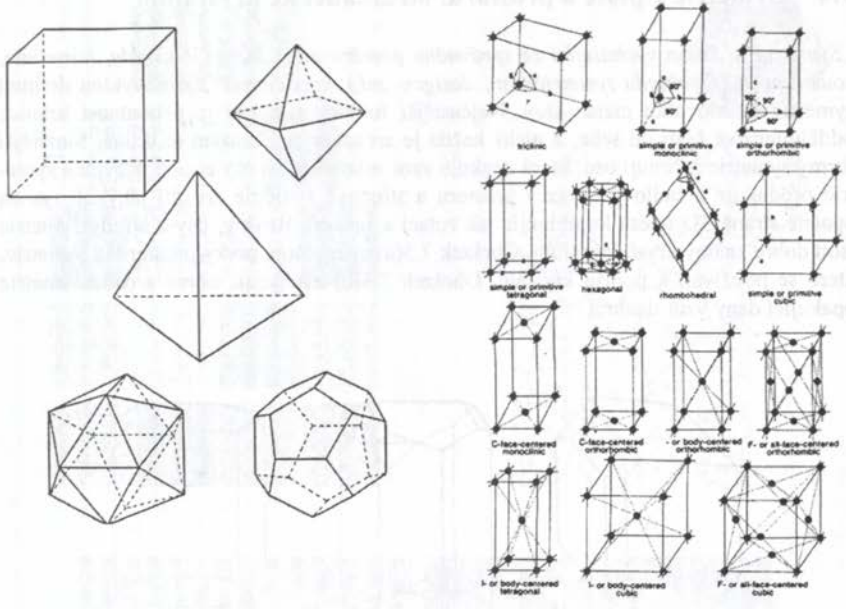
*„Symetrie je krása vycházející ze správného poměru mezi částmi – kvality, harmonie, souměrnosti (z pohledu symetričnosti, designu atd.) mezi částmi"*, je obvyklou definicí symetrie ve slovníku cizích slov. Nejčastější formou symetrie je přítomnost hranice oddělující dvě části od sebe, z nichž každá je zrcadlovým obrazem té druhé. Složitější formy symetrie zahrnují osu, která opakuje vzor *n*-krát kolem osy rotace. Rovinná symetrie produkuje zrcadlový obraz v prostoru a středová symetrie zrcadlí jakýkoli rys na opačné straně 3D tělesa kombinujíc tak rotaci a inverzi. Roviny, osy a středy symetrie jsou dobře známy krystalografům. Obrázek 7.5(a) znázorňuje prvky prostorové symetrie, které se používají k popisu krystalů. Obrázek 7.5(b) zachycuje obraz s osou symetrie opakující daný vzor dvakrát.




a

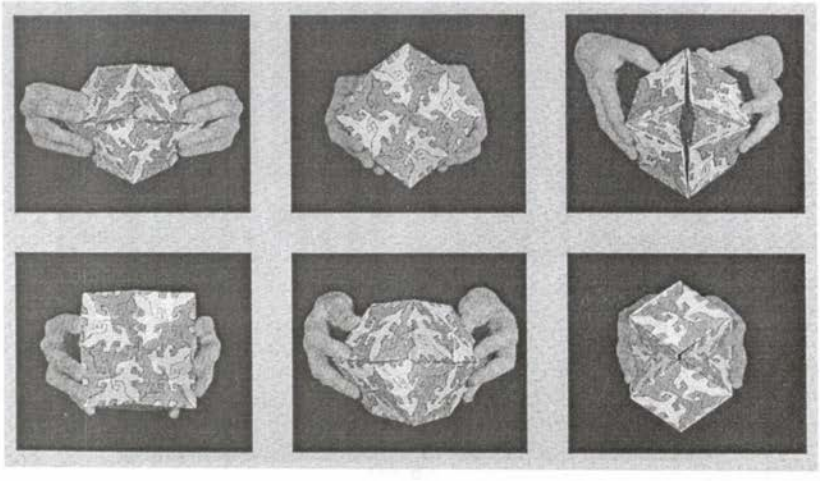
b

Obr. 7.5




a b

Obr. 7.6



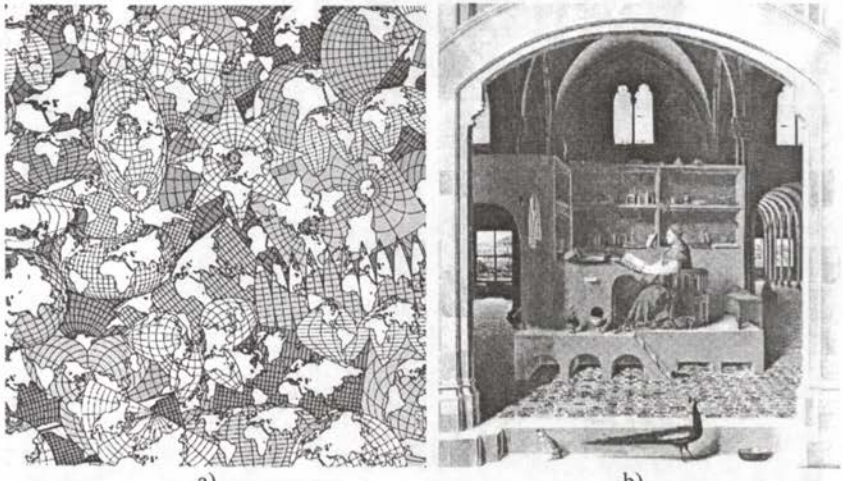
Obr. 7.7

Hledání perfektních mnohostěnů („Platonových těles") dosud nedospělo k jejich hojnému počtu, existuje jich pouze pět, viz obrázek 7.6(a). Dnes se zájem o normální (nejen platonovská) tělesa soustředil na třírozměrné řetězce atomů nebo molekul v přirozených i uměle vytvořených materiálech, jejichž fyzikální vlastnosti jsou přímým důsledkem pořadí. Překvapivě existuje pouze několik různých řetězců atomů, iontů a molekul. Obrázek 7.6(b) ukazuje 14 základních mřížek (tzv. „Bravaisových mřížek", pojmenovaných po krystalografovi, který je jako první popsal), jež dávají vzniknout šesti rozdílným krystalickým systémům a zahrnují všech 230 možných konstelací trojrozměrných krystalových mřížek.

Kaleidocykly, překvapující třírozměrné dynamické útvary vytvořené skládáním, umožňují přivést Escherovy designy (viz další text) k životu, jak je patrné z obrázku 7.7 (též barev. příl.). Vzor na každé stěně mnohostěnu navazuje na sousední stěny i v případě, že je celý objekt pootočen.

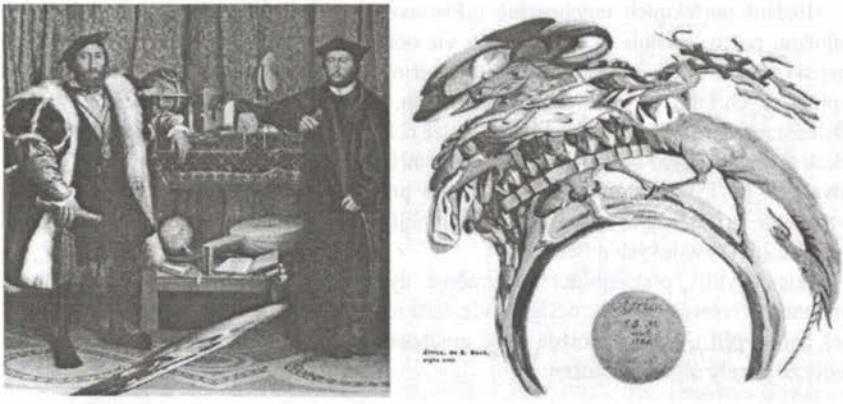
## 7.5 Perspektiva a projekce: zachycení třetího rozměru

Projekce je základem zobrazení třírozměrných objektů na ploše. Nejen rozdílné požadavky kartografů, ale i „hravé" podněty matematiků vyústily v mnoho různých typů projekcí, jak je velmi dobře vidět z obrázku 7.8(a). Obrázek 7.8(b) zachycuje, jaké pokroky učinilo umění perspektivy v době renesance. Zpočátku „camera obscura", následovaná striktními geometrickými pravidly, umožnila malířům vytvářet interiéry a perspektivy volného prostoru tak dokonale, že dosahovaly téměř fotografické kvality.



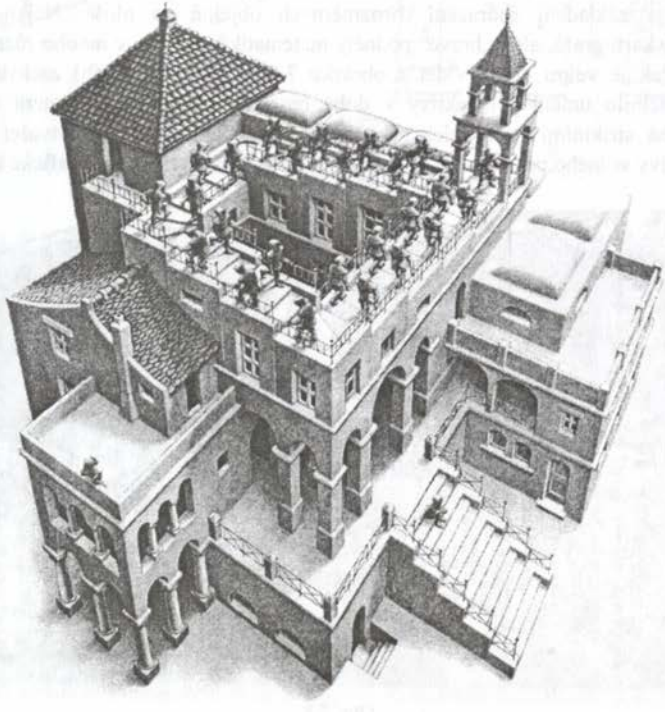
a) b)

Obr. 7.8



a b

Obr. 7.9

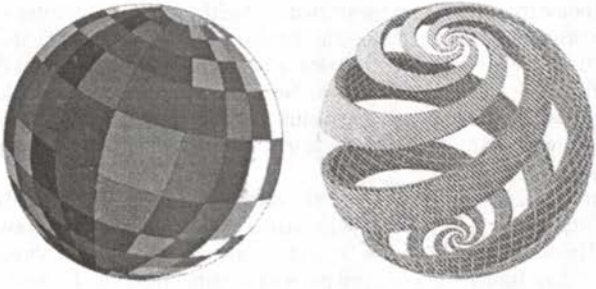


Obr. 7.10

Nové způsoby vytváření perspektivy, jako anamorfóza (proces změny) nebo deformace tvarů vycházející z velmi kosého úhlu pohledu, se posléze zapsaly do dějin umění. Příkladem může být obrázek 7.9(a). Komplexnější příklad, vyžadující cylindrická zrcadla k převedení do rozluštitelné formy, je na obrázku 7.9(b).

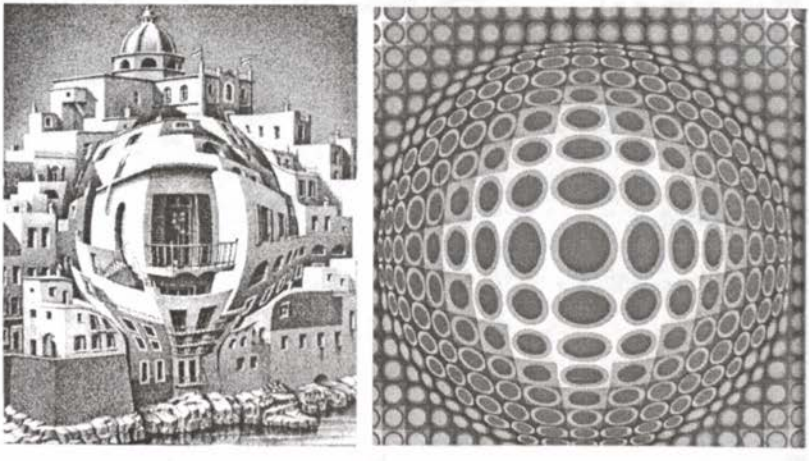
Escher ovládl umění perspektivy. Vytvořil úžasné pseudoperspektivy, které zaujmou oko a zmatou mozek do takové míry, že jen pečlivé prozkoumání kresby povede k odhalení triků, jež způsobují iluzi nikdy nekončících schodů, takových, ať jimi vojáci stoupají či sestupují, pokaždé dojdou opět do výchozího bodu. Escherovy perspektivy, jako například ty zobrazené na obrázku 7.10, jsou dvourozměrnými iluzemi, které by nikdy nemohly být vytvořeny ve třech rozměrech.

Obrázek 7.11 (též barev. příl.) ukazuje, jak lze vytvářet mapové projekce, tedy něco, co by mělo být daleko od zájmu malířů, ale přesto je to fascinuje. Obrázky 7.12 ilustrují



Obr. 7.11

a) b)



Obr. 7.12

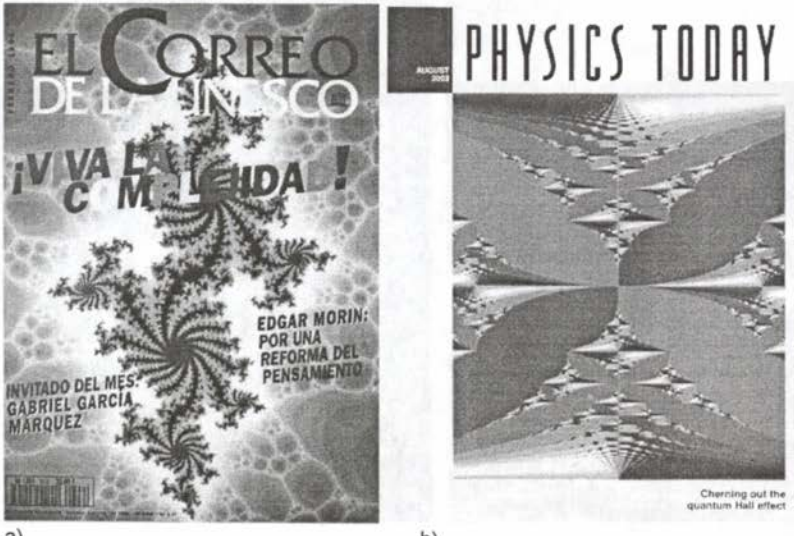
využití geometrické projekce k zdeformování vzorů (obvykle reprezentovaných plochými povrchy), pravděpodobně z důvodů zmatení pozorovatele, něco co umělci dvacátého století velmi rádi používají.

## 7.6 Geometrické a fraktální umění

Významná umělecká díla dvacátého století nabízejí mnoho příkladů využití geometrických útvarů. Tvarově jednoduché a často izolované okouzlují diváka iluzí dokonalosti. Fraktální umění, vytvářející atraktivní vzory na základě rovnic, se může stát novým přístupem k využití matematického aparátu pro design a tvorbu v umění. Počínaje devadesátými lety dvacátého století mnoho umělců před monitory svých počítačů a pouze s pomocí příslušného programového vybavení a s myší jako jediným nástrojem manipuluje barvami, profily a stínováním za účelem vytvoření dekorativního designu, krásného jen ze své podstaty. Fraktály mají nicméně využití i v jiných oblastech než v umění. Mnoho rysů v přírodě lze z matematického pohledu popsat fraktály. Tento fakt ilustruje obrázek 7.13(a), ukazující dekorativní umění, a obrázek 7.13(b) popisující fyzikální jev.

Zájem o fraktální umění rychle roste. Na mezinárodním matematickém kongresu v Madridu v roce 2006 byla zorganizována výstava fraktálních obrázků na počest dr. Mandelbrota, jednoho z otců fraktální teorie. Obrázek 7.14 (též barev. příl.) je ukázkou z této výstavy.

Velmi zajímavými jsou z pohledu historie vztahu umění a matematiky dvě veledíla na obrázku 7.15 (též barev. příl.). Prvním je známá „Velká vlna u Kanagawa" od japonského umělce Hokusaie. Druhým je ne tak známý, ale o to zajímavější obraz z patnáctého století, který využívá fraktály dávno před počátkem zájmu matematiků o ně.




a) b)

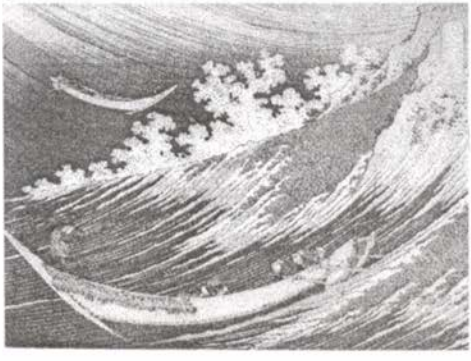
Obr. 7.13

a)

b)

Obr. 7.14

a)



b)

Obr. 7.15

## 7.7 Závěrečné poznámky

Matematika a umění se mohou jevit neinformovanému pozorovateli jako dvě oddělené součásti kultury. Uvedené příklady však poukazují na fakta, která tento náhled nepodporují. Počínaje intuitivně vytvořenými křivkami a obrázky prvních umělců, přes využívání matematických nástrojů k vylepšení a usnadnění práce umělců až po dnešní vytváření sofistikovaných algoritmů pro designéry byl vztah matematiky a umění dlouhý a plodný. Není žádný důvod se domnívat, že tomu tak nebude i v budoucnosti.

### Literatura

El-Said I., Parman A.: *Geometric Concepts in Islamic Art*. World of Islam Festival Publishing Company Ltd., Londýn 1976.

Ghyka M. C.: *Estética de las Proporciones en la Naturaleza y en las Artes*. Editorial Poseidón, Barcelona 1983.

Mercer I. F.: *Crystals*. Natural History Museum, Londýn 1990.

Mottana A., Crespi R., Liborio G.: *Rocks and Minerals*. Simon and Shuster Inc. Publishers, New York 1978.

Ragan D. M.: *Structural Geology*. John Wiley and Sons Inc., New York 1984.

Schattschneider D., Walker W.: *MC Escher Calidociclos*. Tascher Verlag, Kolín n.R. 1992.

Strossberg E.: *Art and Science*. Abbeville Press Publishers, New York 2001.

Tosto P.: *La composición Aúrea en las Artes Plásticas*. Ed. Hachette SA, Buenos Aires 1969.

# 8 Počítač jako laboratoř

*"Why," said Mr. Pickwick, "I may have formed some ideas upon the subject, but as I have never submitted them to the test of experience, I should be sorry if you were induced to regulate your proceedings by them."*

## 8.1 Úvod

Práce aplikovaného matematika je v dnešní době nemyslitelná bez použití výpočetní techniky. Na počítač nahlížíme jako na nástroj, který je pravděpodobně nejsilnější pomůckou aplikovaného matematika – a také mnoha dalších vědců. Díky počítačům byly navrženy, implementovány a využity úžasné matematické modely, rozšířila se nová odvětví všech matematických a statistických oblastí, podařilo se získat do té doby nedosažitelné výsledky a vznikla nová vědecká disciplína – počítačové vědy (computer sciences). Některá odvětví této nové vědy doznala nevídaného rozvoje, například umělá inteligence, dolování dat, expertní systémy a robotika. Počítač tak významně přispěl k převratnému rozvoji matematiky, vědy a techniky a nikoliv neprávem bychom jej mohli nazývat elektronickým Golemem – věrným, pracovitým a užitečným pomocníkem lidí. Vliv tohoto Golema nespatřujeme pouze v pokrocích matematiky, vědy a techniky. Počítač v jistém smyslu změnil to, co Thomas Kuhn nazval paradigma výzkumu v matematice. Aplikovaná matematika se stala, alespoň zčásti, experimentální vědou. Počítačové experimenty umožňují aplikovaným matematikům objevovat nové zákonitosti a charakteristiky reálného světa, ať již ve fyzice, meteorologii, biologii nebo i v sociálních vědách (zákonitosti a charakteristiky musí být samozřejmě testovány také tradičními experimenty nebo alespoň na pozorováních a měřeních reálného světa). Počítač se tak stává opravdovou laboratoří aplikovaného matematika. Prvním použitím počítače v roli laboratoře fyzika byl před více než padesáti lety počítačový model vodíkové bomby. Pravděpodobně nejslavnější novou fyzikální teorií, která má počátek v počítačových experimentech – a také posloužila jako základ nové matematické disciplíny, je teorie chaosu. V mnoha vědeckých oblastech není počítač pouhým nástrojem, ale také experimentálním prostředím. Nakonec i v klasické matematice nejsou počítače používány pouze k formulaci nových matematických teorií ani pouze pro rozvoj těchto teorií, jako je tomu u mnoha fyzikálních experimentů, ale k velkému překvapení také ke strojovým důkazům.

Uznávaný americký meteorolog Edward N. Lorenz v roce 1963 publikoval ve významném časopise zaměřeném na meteorologii článek (Lorenz, 1963). Význam této kapitoly je těžko možné docenit: Na jedné straně dal základ novému odvětví matematiky, teorii chaosu, na druhé straně změnil pohled na naše možnosti obecného předvídání fyzikálních jevů, zejména počasí, a vyvolal vážnou diskusi o vztahu mezi náhodností a determinismem.

Historie Lorenzovy kapitoly je dobře známá a lze si ji přečíst (a vychutnat) v populární knize (Gleick, 1986) nebo spolu s dalšími podrobnostmi a postřehy v publikaci (Lorenz, 1993). Lorenz zformuloval jakýsi „model hraček", pomocí kterého reprezentoval některé základní charakteristiky atmosférického proudění a který se skládal ze tří obecných diferenciálních rovnic. Dvě z nich byly nelineární. Tento model implementoval v počítačovém programu pro počítač Royal-McBee LGP-30, který byl na dnešní poměry poměrně primitivní. Při spuštění tohoto programu si uvědomil, že pokud jako počáteční podmínky použil výpis z určité doby simulace předchozího běhu, výsledky se diametrálně lišily od výsledků předchozího běhu programu. Byl si samozřejmě vědom, že se jeho počáteční podmínky nepatrně liší od hodnot minulého běhu, protože reálná čísla na výpisu měla nižší přesnost než odpovídající reálná čísla v paměti počítače. Nijak se tím však nezabýval, protože předpokládal platnost implicitního pravidla fyziky, které zatím již po dobu tří století tak dobře fungovalo: *Matematická reprezentace zákonitostí přírody je stabilní; malé změny v datech způsobí malé změny ve výsledcích.* Pokud měříme nějaký jev pomocí zařízení s vyšším rozlišením, nebo pokud použijeme počítač s odlišnou reprezentací reálných čísel (v tomto případě s omezením, že použité numerické metody jsou stabilní), nebudou se v důsledku tohoto pravidla výsledky významně lišit. Kdyby tomu tak nebylo, nebylo by možné nic předpovídat a neexistovala by moderní věda. V technickém žargonu lze říci, že moderní věda, zejména fyzika, by měla být v Hadamardově smyslu správně definována.

Příroda však není vždy stabilní. Klasická Hadamardova definice stability říká, že: *„model je stabilní, pokud má řešení pro hustou podmnožinu (v některé topologii) možných dat, potom je řešení jedinečné pro každou množinu dat a malé změny hodnot dat se projeví jako malé změny ve výsledcích"*. To implikuje, že v případě nestabilního modelu se malé změny v datech projeví jako velmi odlišné trajektorie. Lorenz objevil, že některé meteorologické jevy jsou však něco více než nestabilní: trajektorie směřují k podivným (např. fraktálním) atraktorům. Dostáváme se tak k deterministickému chaosu. I kdybychom měli k dispozici počítač s neomezenou přesností, nejpřesnější rovnice proudění a nejhustší síť meteorologických senzorů, nebudeme dopředu schopni předpovědět počasí na období delší než deset až patnáct dní. Je to tak svazující a neúprosné jako konstantní rychlost světla $c$ (přestože tu známe s mnohem menší přesností). Mimochodem, limit doby předpovědi ještě nebyl dosažen, takže si můžeme být jisti, že se dnes již poměrně přesné krátkodobé předpovědi v průběhu dalších let zpřesní – ale pouze na období deseti až patnácti dnů. Žádné zařízení umělé inteligence, žádný expertní systém ani žádný výborně naprogramovaný robot nebude nikdy schopen předpovědět počasí na delší období. Pozorování (a přijetí faktu), že v přírodě existuje deterministický chaos, vyvolal ve vědě radikální změnu paradigmatu ve smyslu Thomase Kuhna. Nebude asi překvapením, že deterministický chaos – anebo techničtěji, teorie špatně navržených nelineárních dynamických systémů, jejichž trajektorie směřují k neceločíselné Hausdorffově dimenzi – byl objeven meteorology. Deterministický chaos je ve svém smyslu čárou přes rozpočet našim ujištěním, že počasí je předpověditelné, že je jeho chování „racionální" a že je můžeme „řídit". Za objevení deterministického chaosu můžeme děkovat právě počítačům. Je ale ironií, že zařízení, které nám v mnoha oblastech dává obrovský prediktivní výkon, o kterém jsme ani nesnili a které je nástrojem umožňujícím vývoj umělé inteligence a robotiky, nám současně přináší svědectví o tom, že naše intelektuální a fyzikální schopnosti nejsou neomezené. Podobné limity ukázal i Gödelův teorém o neúplnosti omezující naši schopnost dedukce nebo existence Einsteinovy světelné konstanty $c$ ukázující omezení nejvyšší možné rychlosti našich vozů.

Nechceme zde detailně analyzovat teorii chaosu, která je sepsána v mnohem kompletnější bibliografii. Chceme se zaměřit na to, co již bylo popisováno jako *nástroj*, který byl kritickým při objevení něčeho, co změnilo naše vnímání přírody – na počítač. Elektronický Golem moderní doby, neúnavný a oddaný služebník, který šetří lidskou práci zpracováním velkých objemů dat neuvěřitelnou rychlostí a řešící vědecké, technické a administrativní problémy, jejichž řešení se před několika desetiletími zdálo nemožné, se najednou stává něčím víc než jen nástrojem. Jeho použití se začíná podobat laboratořím ve vědě v tom smyslu, že spuštění programu je srovnatelné s provedením experimentu. Lze pozorovat (možná překvapující) chování přírody, lze objevit nové zákonitosti, pro jejichž modelování někdy bývá nutné vytvořit nové matematické teorie. A více než to. Tento typ Golema pomáhá také klasickým matematikům, protože pomocí počítačů byly dokázány různé teorémy, například teorém čtyř barev, pomocí počítačů byly předpovězeny výsledky, které byly později matematicky potvrzeny, například Feigenbaumovy konstanty. Práci matematiků také výrazně zjednodušily symbolické manipulátory, například Mathematica a Maple. Význam elektronického Golema vzrostl nad všechna očekávání a je držen pod kontrolou. Vzpoura počítače Hal 9000 ve filmu *2001: Vesmírná Odysea* zůstává zatím pouze noční můrou a výzvou k ostražitosti.

## 8.2 První počítačové experimenty

Teorie chaosu je nejpopulárnějším a nejvíce udivujícím příkladem použití počítače v roli laboratoře aplikovaného matematika. Ale není prvním. Strogatz (Strogatz, 2003) ve svém excelentním článku píše o tom, jak v roce 1953 Enrico Fermi, John Pasta a Stanislaw Ulam (dále FPU) vynalezli koncepci „počítačového experimentu". Použili numerické simulace pro snazší pochopení entropie, viz (Fermi a kol., 1955). Při simulaci jednorozměrného dynamického systému se 64 částicemi, kdy síly mezi sousedy obsahovaly nelineární členy a primární energie byla umístěna v prvním uzlu (na rozdíl od termalizace, kdy je energie jakoby rovnoměrně rozdělena do všech uzlů), zůstala energie pouze v několika z nich. Jednalo se o neočekávaný výsledek, který byl později vysvětlen v práci (Zabusky, Kruskal, 1965), když autoři objevili limitu dlouhých vln modelu FPU. Jak by požadoval Dickensův pan Pickwick, byly tzv. „solitony" autorů Zabuskeho a Kruskala také pozorovány v přírodě. Počítač autorů FPU byl samozřejmě jednodušší než ten Lorenzův a jeho název (MANIAC) je pro Golema jistou symbolikou. A počítačový experiment, včetně překvapivých výsledků, fungoval stejně jako laboratorní experiment.

Experiment autorů Fermiho, Pasty a Ulama byl prvním *deterministickým* počítačovým experimentem. Předtím byly prováděny pouze *stochastické* experimenty. Několik let před tím tento obhroublý elektronický Golem „stvořil" náhodnost: Metoda Monte Carlo je technikou pro simulaci náhodné situace počítačem, která spočívá ve vytvoření řady, která „vypadá" náhodně – *pseudonáhodné řady*. Tento název vynalezl vědec jménem Metropolis a poprvé byl použit v publikaci (Metropolis, Ulam, 1949). Uvedená technika, která byla Ulamovým nápadem, byla kritizována Johnem von Neumannem. Z pohledu definice náhodnosti je velice zajímavé, že pseudonáhodné řady čísel lze generovat pomocí deterministického zařízení (počítače) a použít je pro simulaci náhodných situací, například příchodu klientů do fronty. Snad ještě zajímavější je generování

pseudonáhodných čísel pomocí deterministického zařízení tak, abychom se dostali do deterministické situace, například při počítání integrálů pomocí metody Monte Carlo. V obou případech nás elektronický Golem přivádí k otázce: Co je to náhodnost? (Některá pojednání o náhodnosti a determinismu lze nalézt v publikaci (Jacovkis, 1995).) Je možné vytvořit náhodnost pomocí determinismu? Pokud ne, můžeme používat deterministické řady (ony počítačem generované pseudonáhodné řády) pro simulaci náhodných jevů? Protože – světe div se – ono to funguje! Jediná přijatelná odpověď je následující: Pokud nikdo nepozná, že řada není náhodná, potom je pro všechny praktické účely náhodná dostatečně. Zajímavé je, že tento „pragmatický" přístup má velmi slavnou tradici. Alan Turing tuto metodu (že věci jsou tím, čím si myslíme, že jsou) v podstatě použil, když v roce 1950 ve svém klíčovém článku o umělé inteligenci (Turing, 1950) představil koncept, který byl později nazván Turingovým testem: „*O stroji si můžeme myslet, že je inteligentní v případě, že člověk není schopen rozlišit, zda komunikuje se strojem či s jiným člověkem.*" S tím se asi filosofové nespokojí, ale pokud to funguje, je těžké takovou myšlenku opustit, dokud ji někdo nenahradí něčím lepším.

## 8.3 Laboratoř meteorologa

Jak již bylo uvedeno v předchozím výkladu, náš elektronický Golem změnil různá paradigmata, zejména v meteorologii. Možná bychom také mohli mluvit o významu e-Golemů v epistemologii (nauce o poznání)). Spojení meteorologie a počítačů je v jistém smyslu historicky přirozené – prostřednictvím numerické analýzy: v roce 1904 Vilhelm Bjerknes navrhoval, že pro předpověď počasí bude nutné řešit diferenciální rovnice (a to mnoho let před tím, než to bylo technicky zvládnutelné). Od roku 1910 se touto myšlenkou zabýval Lewis Fry Richardson, který se rozhodl numericky – a ručně – řešit rovnice proudění, pomocí kterých se pokusil reprezentovat atmosférické proudění. Strávil několik měsíců samostatnou prací na přípravě šestihodinových předpovědí v okolí města Mnichova v Německu. Nutno podotknout, že jeho předpovědi byly chybné, protože jeho numerické metody byly nestabilní. Navzdory tomu se Richardson s nebývalou prozřetelností rozhodl, že tato myšlenka má smysl a v roce 1922 publikoval významnou knihu (Richardson, 1922). Richardson netušil, že se o dvě desetiletí později objeví počítače. Myslel si, že předpověď bude možné provést pomocí armády lidí čítající přibližně 64 000 jedinců, kteří budou současně ručně řešit rovnice, a kteří si informace budou vyměňovat telegraficky ještě před tím, než se ex post dozví zamítavé či potvrzující výsledky této předpovědi.

Bylo přirozené, že John von Neumann, pravděpodobně první osoba, která objevila potenciální využití počítače jako laboratoře matematika, byl svou genialitou a svým globálním vnímáním vědy prvním vědcem, který na začátku počítačového věku používal počítače jako svou laboratoř (zejména pro výpočty v meteorologii). Jak podotkl Macrae (Macrae, 1992), von Neumann se po své účasti na válečných projektech v průběhu druhé světové války začal zajímat o numerické řešení nelineárních parciálních diferenciálních rovnic, které reprezentovaly atmosférické proudění. J. von Neumann si záhy uvědomil, pravděpodobně ještě dříve než kdokoliv jiný, že počítače by mohly vyřešit některé obtížné vědecké problémy, od aerodynamiky až po nukleární fúzi a detekci a eliminaci

hurikánů. V mnoha hlediscích měl pravdu, některé z jeho myšlenek se však dnes zdají podivné: s Vladimirem Zworykinem, jedním z vynálezců moderní televize, viz (Zworykin, 1971), konzultoval možné použití atomových bomb pro likvidaci hurikánů. Nutno podotknout, že Ulam a další také přemýšleli o možnosti eliminace hurikánů pomocí atomových bomb. J. von Neumann komunikoval s meteorology již od roku 1942 (ještě před začátkem počítačového věku), když konzultoval s Carlem-Gustavem Rossbym. J. von Neumann a Jule Gregory Charney navázali v roce 1948 na Institute of Advanced Study v Princetonu velmi produktivní spolupráci. V roce 1950 Charneyho skupina úspěšně předpověděla počasí na 24 hodin dopředu a od té doby jsou předpovědi počasí rok od roku přesnější. Již tehdy, v roce 1950, onen elektronický Golem transformoval, či pomohl transformovat, meteorologii z empirické disciplíny na disciplínu teoreticky podloženou.

## 8.4 Další vědy

Název „výpočetní teorie proudění" (computational fluid dynamics) označuje odvětví fyziky, které se od 60. let 20. století stále vyvíjí a ve kterém počítač neoddiskutovatelně slouží jako laboratoř. Přináší obrovský užitek, protože experimenty s výpočtem proudění jsou obecně mnohem levnější než experimenty prováděné v klasických laboratořích. Nadto se animace zasloužila o zprostředkování pocitu, který bychom získali při „normálním" experimentu. Někdy však je počítačová role laboratoře aplikovaného matematika diskutabilní. Vědci si například povšimli (Kennedy, 1963), že v řekách a vodních tocích s pohyblivým podložím se při nadkritickém toku projeví antidunový jev. Rovnice, které reprezentují pohyby podloží, jsou v současné době empirické, neboť popisovaný fyzikální proces je příliš složitý a doposud nebyla vytvořena žádná teorie, která by jej zcela vysvětlovala. Před několika lety jsem potřeboval otestovat korektnost programu, který jsem vytvořil pro řešení rovnic řek a vodních toků s pohyblivým podložím. Proto jsem provedl simulaci několika nadkritických situací a na všech výstupech byla zřetelná přítomnost antiduny. Jaká byla v tomto případě role našeho elektronického Golema? Potvrzení správné funkce programu? Potvrzení, že empirické vzorce reprezentují fyzikální proces, stejně jako klasická laboratoř obecně potvrzuje empirické vzorce?

Mnoho aplikovaných matematiků a fyziků by mohlo zcela jistě vyprávět o některých jevech, které objevili pomocí počítače. Nemá smysl asi tolik lpět na tak obrovském významu počítačů v fyzice, s výjimkou následujících dvou velmi zajímavých situací: Thomas A. Witten a Leonard M. Sander při psaní článku v roce 1981 použili termín „difuzí omezené shlukování" (Diffusion Limited Aggregation, DLA), viz (Witten a Sander, 1981), ve kterém popisovali počítačové experimenty, jež sloužily k simulaci růstu shluků náhodně pohybujících se částic, a které jsou použitelné pro proces shlukování kovových částeček, jehož korelace byly předtím měřeny. Získané shluky měly jasně fraktální strukturu. Nebylo jim zcela jasné, zda existoval přirozený růst typu DLA, nebo zda šlo pouze o matematickou zajímavost. V roce 1984 však R. Brady a R. C. Ball vydali článek (Brady a Ball, 1984), ve kterém popisovali elektrické ukládání sulfátu mědi za podmínek omezených difuzí, čímž prokázali užitečnost Wittenova–Sandersova modelu při modelování tohoto jevu. Později bylo publikováno mnoho dalších článků,

v jednom z nich (Matsushita a kol., 1984) byly získané elektricky ukládané lístky zinku zřetelně podobné náhodným vzorům simulovaným počítačovým Wittenovým–Sandersovým modelem. Uvedený model provádí výpočet testu invariance měřítka korelací mezi funkcemi hustoty digitalizovaných obrázků a výpočtu střední hodnoty Hausdorffovy dimenze (přes velký počet vzorků), která s excelentní přesností odpovídá dvourozměrnému modelu DLA.

Jedná se o velice zajímavý příklad jevu, který byl *nejprve* pozorován v počítači a až poté v přírodě. V jistém smyslu bylo nutné nalézt v přírodě jev, který byl předtím pozorován v počítači. Což je přesně opačná situace oproti zmíněnému příkladu s antidunami.

Dalším zajímavým případem, kdy je počítač (jako měřicí zařízení) mnohem efektivnější než laboratoř fyzika, je teorie perkolace (percolation theory). Uvažujme dostatečně velikou množinu bodů distribuovaných podle předem daných geometrických pravidel v jedno-, dvou-, nebo třírozměrném prostoru tak, aby mohla být považována za nekonečnou. Pokud sousední body (s vhodnou definicí sousednosti) jsou s pravděpodobností $p$ spojeny či nespojeny hranou, potom (Stauffer, 1985), (Grimmet, 1999) existuje kritická pravděpodobnost $p_c$ taková, že pro $p < p_c$ je pravděpodobnost existence nekonečného shluku bodů nulová a pro $p > p_c$ je pravděpodobnost existence alespoň jednoho nekonečného shluku rovna jedné. Tento typ modelu je užitečný pro reprezentaci mnoha různých jevů, například difuze v neuspořádaných médiích, lesních požárech, šíření pověstí atd. Až na několik speciálních geometrických konfigurací je pro téměř všechny geometrické konfigurace ve dvou a třech dimenzích nemožné kritickou pravděpodobnost analyticky spočítat. Na druhou stranu lze pomocí počítačových simulací kritickou pravděpodobnost určit s vyšší přesností, než jaké je možné dosáhnout měřením v laboratoři – pokud je to vůbec proveditelné. Nutno podotknout, že stejná situace nastává v případě časových řad: často bývá vhodné určit fraktální dimenzi (nebo jednu z fraktálních dimenzí) grafu časové řady. Až na některé případy uměle vytvořených fraktálů, kdy lze fraktální dimenzi odvodit, je tuto dimenzi nutné numericky spočítat, neboť analýza v laboratoři nepřichází v úvahu.

Není příliš překvapující, že se počítače staly laboratoří fyziků (přinejmenším „výpočetních" fyziků). Je však až s podivem, do jaké míry je počítačové modelování využíváno v biologii. Termíny „biomatematika" a „bioinformatika", které se již staly nedílnou součástí vědeckého světa, vyjadřují použití matematických a výpočetních modelů v některých biologických oblastech. Jak ukazuje název „genetický algoritmus", byla i biologie zdrojem inspirace pro počítačové vědy. Neméně zajímavé je podívat se na opačné situace, kdy počítač, umožňující matematické modelování, pracuje v biologii jako laboratoř. Uveďme některé příklady.

Slavný biolog Richard Dawkins ve své knize (Dawkins, 1996) napsané v roce 1991 v příloze II uvádí: „Dokud jsem si nezačal hrát s rozšířeným programem Blind Watchmaker (Slepý hodinář), nebylo mi až tolik zřejmé, že nejsou všechny embryologické skupiny stejně *plodné* ve vztahu k budoucí evoluci." Rozšířený počítačový program Blind Watchmaker si Dawkins vytvořil pro simulaci biologické evoluce. Znamená to, že se k vědecké myšlence v biologii dostal od výsledků počítačového matematického modelu.

To je také případ matematického modelu, který navrhli Cohen a Gürtler (Cohen, Gürtler, 2001), díky kterému došli k důležitým závěrům a doporučením týkajících se strategie řízení přenosu Chagasovy nemoci (konkrétně, že riziko lidské nákazy lze významně snížit odstraněním domácích zvířat, zejména nakažených psů, z ložnice). V jistém smyslu je přístup tohoto článku shodný s přístupem článku (Cohen, 2004), který lze částečně shrnout názvem tohoto článku: „Matematika je dalším mikroskopem biologie; ...“ Dodal bych jen, že je to díky našemu elektronickému Golemovi, počítači. Díky biomatematice a bioinformatice došlo v posledních letech k rozvoji používání matematických modelů v biologii. A užitečnost matematických modelů se neustále zvyšuje. Počítač může fyzikální model nahradit (nebo předvídat), viz například (Mackenzie, 2004). V mnoha případech není z etických důvodů možné fyzikální model použít. Přehled některých užití jednoduchých matematických modelů s komplexní odezvou je k dispozici v publikaci (May, 1976), v níž je opět popsán fascinující jev chaosu, v tomto případě ukazující někdy nepředvídatelnou dynamiku populačního růstu: velikost populace se může dočista „zbláznit“.

Hledání sekvencí lidského genomu je dalším pozoruhodným příkladem použití počítače v roli vědecké laboratoře. Počítače významně zvýšily uskutečnitelnost tohoto projektu. Například autoři knihy (Venter a kol., 2001) uvádějí: „Vzrůstající počet lidských sekvencí EST (Expressed Sequence Tag) si vyžádal vývoj nových počítačových algoritmů, které by byly schopny analyzovat velké objemy dat těchto sekvencí, a v roce 1993 byl v ústavu TIGR (The Institute of Genomic Research) vyvinut algoritmus, který dovoloval sestavení a analýzu stovek tisíců sekvencí EST...“. Z této věty by někdo mohl usuzovat, že počítač zde hrál svou tradiční roli mocného nástroje pro výpočty a manipulaci s daty, tentokrát samozřejmě v neuvěřitelně velkém měřítku. Abychom rozptýlili veškeré pochyby o aktuálním významu počítače, dodejme, že Venter se svými kolegy ve stejném článku popisuje expertní systém založený na pravidlech, nazvaný Otto, který vyvinuli právě pro identifikaci a charakterizaci genů v lidském genomu. To, že se v oblastech spojených s výpočetní biologií objeví expertní systémy, tedy obecněji systémy umělé inteligence, je asi nevyhnutelné.

A s umělou inteligencí také přirozeně souvisí příklad patřící do oblasti výpočetní biologie (nebo bioinformatiky), kdy počítač je nejen laboratoří, ale kdy pravidla pro jeho chování jsou navržena jako adaptace biologického procesu – oblast umělých neuronových sítí. Tato oblast výzkumu odstartovala svou existenci spolu s počítačovou vědou (a zároveň nezávisle na ní) klíčovou prací (McCulloch, Pitts, 1943).

Zde je asi vhodné poznamenat, že počítače umožnily vytvořit některé disciplíny, které sice nelze jednoduše zařadit mezi tradiční experimentální vědy (experimentální v tom smyslu, že používají počítač jako ekvivalent laboratoře fyzika, chemika nebo biologa), ale pro které je počítač cosi mnohem zásadnějšího než pouhé zařízení pro výpočty a zpracování dat. Tyto disciplíny zahrnují mnoho oblastí, které se vztahují k umělé inteligenci, například dobývání dat (nebo extrakci znalostí) – vědu o získávání užitečných informací z rozsáhlých množin dat, viz například (Hegland, 2001), (Berka, 2003). Tento vědní obor se stává stále více a více důležitým. Technologický pokrok umožnil lidem, společnostem a zemím (zejména rozvinutým zemím) nashromáždit nepředstavitelné množství všech možných typů dat. Často však nevíme co s nimi, jak v nich nalézt struktury nebo vzory. Dobývání dat (data mining) nalézá v původních datech skryté poklady.

Tuto část nelze ukončit bez zmínky o tom, že i v geologii byl počítač použit jako laboratoř. Existuje implementace modelu konečných prvků (finite element model) pro numerickou studii vývoje topografie a deformace horní slupky nekolizních orogenů v zóně vyjmutí, která byla vytvořena kůrou oceánu a která je odnímána (Quinteros a kol., 2004), (Quinteros a kol., 2006). Získané výsledky jsou velmi povzbudivé. Pokud lze (numerické) studium minulosti použít pro (numericky) lepší pochopení současnosti, potom může simulace geologické minulosti významnou měrou přispět k lepšímu poznání geologické současnosti. Ve světě, kde jsou zdroje energie stále více a více omezené, to bude nelehký úkol.

## 8.5 Modelování v sociálních vědách

V sociálních vědách nalezneme počítače zejména v demografii. Demografické modely částečně patří do ekonomiky (zejména v genetice a epidemiologii nebo v případě, že analyzovaným druhem je *Homo sapiens*) a částečně do biologie (dynamika populace). Demografické modely jsou mocným nástrojem při studiu velice důležitých problémů, které nastanou v blízké budoucnosti, ale bez těchto modelů je nelze předpovědět – například budoucí rozložení věku, oblastí nebo tříd populace – a jejichž následky mohou být katastrofické. Uveďme pár příkladů: Oscar Varsavsky zkombinoval demografické modely typu Leslie-matrix s učením a Leontievovými vstupně-výstupní modely pro analýzu chování a budoucnosti latinskoamerických společností, viz (Varsavsky, 1969a). Počítač se mu opravdu stal laboratoří sociálních věd (Varsavsky, 1969b). Mimochodem také jedinou možnou laboratoří. Je také důležité podotknout, že výsledky modelů, které předpovídají, co se stane za dvacet či třicet let, a které v nějakém smyslu ovlivní svět, vždy bývají předmětem diskusí. Nezřídka se jejich předpovědi ukáží jako chybné – jako tomu ostatně bývá i v případě některých laboratorních experimentů. Ale ať již jsou správné nebo chybné, diskuse nad socioekonomickými modely jsou velmi důležité. V tak složitém světě, v jakém žijeme, nám pouze holistická vize budoucnosti sestavená pomocí počítačových matematických modelů může nabídnout představu o trendech, nebezpečích a příležitostech takové budoucnosti a země, které je používají, budou při strategických návrzích vždy ve výhodě.

Stejná myšlenka, tj. sestavení komplexního počítačového modelu pro studii komplikovaných sociálních jevů, pochází z teorie komplexních systémů a návrhu modelů světa. Slavný technik, Jay Forrester, který vynalezl feritovou paměť a později v MIT (Massachusetts Institute of Technology) vytvořil počítačový simulační jazyk DYNAMO, navrhl model světa (Forrester, 1971). D. H. Meadows poté se svými spolupracovníky implementoval model světa (Meadows a kol., 1972), který maltuziánským způsobem předpověděl budoucí katastrofu, která nastane v případě, že nedojde k zastavení růstu. Název knihy, ve kterých své výsledky shrnují, je značně provokativní – *The limits of growth* (Meze růstu). Model byl tvrdě kritizován z důvodů ideologických (pokud ustane vývoj společnosti, všichni ustrnou jako na fotografii a ti, kterým se aktuální situace nelíbí, by se měli navždy vzdát jakékoliv naděje na změnu) i z důvodů technických (model byl podstatně nestabilní, numericky řešil systém běžných diferenciálních rovnic modelu Eulerovou metodou, takže jeho užitná hodnota je sporná; ponecháváme stranou, že není

jakkoliv jisté, zda systém obecných diferenciálních rovnic tento jev pokrývá). Kritiku tohoto modelu lze nalézt v publikaci (Scolnik, 1979). Kritiky jsou oprávněné. Za pozornost však stojí intelektuální troufalost autorů, kteří jako první zformulovali takto náročný model. Hovoříme zde o tom, že nebezpečného Golema je nutné stále pečlivě hlídat.

Diskuse o tomto modelu měla za následek návrh a implementaci dalších alternativních modelů, zejména Mesarovicův–Pestelův model (Mesarovic, 1979) a Latin American World Model (Bariloche Model, viz (Herrera a kol., 1976)). Nejzajímavějším rysem těchto modelů byla jejich interdisciplinárnost. Na modelu Bariloche se například podíleli matematici, ekonomové, geologové, biologové, architekti a další osoby. Tuto charakteristiku si zachovávají institucionální projekty „komplexních věd", zejména ve dvou nejznámějších ústavech, ve kterých je na toto téma prováděn výzkum: Institut IIASA (International Institute for Applied Systems Analysis) v Laxenburgu v Rakousku, založený 1972, a zejména institut Santa Fe Institute v Santa Fe v Novém Mexiku, který je v provozu od roku 1984.

Když počítače otevřely cestu formalizace vědním oborům, jejichž charakteristika je odlišná od běžných charakteristik fyziky popsaných komplexními a většinou nediferencovatelnými nelineárními systémy, zcela nevyhnutelně se objevily některé přehnané naděje i teorie o cestě k vyřešení všech problémů (zejména sociálních) pomocí matematických modelů. Spousta lidí začala elektronického Golema považovat za univerzální všelék, který je schopen nalézt řešení všech dosud vzdorujících problémů. Asi nejznámější případ nesprávného a přehnaného použití matematického modelování, které nesplnilo očekávání, byla zřejmě teorie katastrof, velice diskutovaná a v 70. létech 20. století mnohými obdivovaná. Dnes už o ní téměř nikdo nemluví. Ničivá a z pohledu nauky o poznání velmi zajímavá kritika teorie katastrof je uvedena v publikacích (Sussmann, 1975) a (Sussmann, Zahler, 1978). Méně kritický, nicméně stejně jasný popis lze nalézt v malé knize (Arnold, 1986). Závěrem budiž řečeno, že je důležité odolat pokušení vytvořit všemocného elektronického Golema. Zdá se, že v blízké budoucnosti se jedná o úlohu zcela nemožnou.

## 8.6 Počítač a klasičtí matematikové

V matematice lze na vliv teorie poznání nahlížet třemi různými způsoby. Velice známý problém čtyř barev (mapu je vždy možné obarvit pomocí čtyř barev tak, aby státy se společnou hranicí nebyly vyznačeny stejnou barvou) byl vyřešen pomocí počítačového programu (Appel, Haken, 1977), (Appel a kol., 1977): Problém byl redukován na důkaz, že pro několik tisíc případů postačují čtyři barvy, a následnou kontrolu těchto případů pomocí počítačového programu. Toto řešení bylo trnem v oku některým matematikům (z pohledu klasického matematika se problém redukuje na případ formální verifikace správné funkce počítačového programu). Pozdější jednodušší důkaz, který byl získán o několik let později, také v hojné míře vyžaduje použití počítače (Robertson a kol., 1997). Jedná se o překvapivé proniknutí počítačů do klasické matematiky. Známe však i další případy: Po mnoha počítačových experimentech byl Mitchell Feigenbaum v roce 1975 přesvědčen, že pro určitou třídu problémů v teorii chaosu existuje univerzální konstanta s určitými charakteristikami. O několik let později byla existence této

konstanty dokázána a její hodnota byla určena (Feigenbaum, 1979), (Lanford, 1982), (Lanford, 1984). To znamená, že počítač byl použit jako laboratoř *klasického matematika*. V jistém smyslu je mnoho takových charakteristik přítomno v symbolických manipulátorech, jakými jsou například programy Mathematica a Maple. Pokud například pomocí programu Mathematica počítáme přesné kořeny polynomu třetího stupně, je tento symbolický manipulátor mocným nástrojem klasického matematika. Pokud však pomocí symbolického manipulátoru zjednodušíme velmi složitý vzorec na překvapivě jednoduchý výsledek, nebo pokud testujeme rovnost dvou dlouhých komplexních výrazů, stává se počítač skrze symbolický manipulátor laboratoří, která klasickému matematikovi pomáhá dokázat jistý teorém. Náš mocný e-Golem dokáže také pracovat jako matematik. Krásná kniha autorů Petkovseka a kol. (Petkovsek a kol., 1996) uvádí: „computers not only find proofs of known identities, they also find completely new identities" (nejenže počítače nacházejí důkazy známých rovností, ale dokáží také nalézt zcela nové). Autoři ukazují, že v současné době jsou počítače schopné odhalit, zda daný hypergeometrický součet je možné vyjádřit v jednoduché „uzavřené formě", a pokud je vyjádřitelný, mohou tuto formu nalézt; pokud vyjádřitelný není, jsou schopny dokázat, že vyjádřitelný není. Bressound (Bressound, 2001), který rozlišuje mezi důkazy založenými na počítači (například důkaz problému čtyř barev), kdy počítač používáme jako nástroj, a mezi zcela automatickými počítačem generovanými důkazy (například důkazy problémů, uváděných ve zmíněné knize autora Petkovseka a jeho kolegů Wilfa a Zeilbergera), o těchto problémech prohlašuje: „Nyní se dostáváme na dosah problémům, které byly kdysi nedosažitelné." Počítačem generované důkazy v jistém smyslu spadají do oblasti umělé inteligence.

Byl také založen časopis *Experimental Mathematics*, který na své webové stránce [1] publikuje: „formální výsledky inspirované experimenty, teorie získané experimenty, popisy algoritmů a software pro matematický výzkum, přehled oblastí matematiky z experimentálního pohledu a tematicky zajímavé články". Na toto téma byly vydány některé knihy, například (Borwein, Bailey, 2004).

Nakonec je důležité se zmínit o Zeilbergerovi, který je spolu s Wilfem (Zeilberger, 2002) spolutvůrcem metody WZ pro automatické zpracování hypergeometrických řad, jež představuje velmi zajímavý a vtipný přístup. Zeilberger se dostává až za hranici tvrzení, že počítač je laboratoří matematika, říká totiž, že při vytváření softwaru pro svůj projekt pracoval jako technik-inženýr.

## 8.7 Závěr

Matematické modely dovolují počítači stát se čímsi jiným: Může se stát profesionálním nástrojem, kdy jsou meteorologické předpovědi vytvářeny pomocí numerických modelů, nebo kdy se spouští hydrodynamický model, abychom zjistili, zda režim provozu nádrže způsobí či nezpůsobí poškození dále po proudu. Může se také stát pomocným výzkumným nástrojem, například při testování hypotézy nebo při ověření experimentu. Počítačové testování fyzikálního jevu v tomto případě zdůrazňuje důvěru ve (někdy empirické) vzorce, které by měly jev pokrývat. A počítač může být nakonec i hlavním výzkumným nástrojem, jak tomu bylo například v pokusech autorů Fermiho–Pasta–

Ulama nebo v Lorenzově případě. Může také sloužit jako nástroj klasickému matematikovi – pomocí počítače může matematik dokazovat teorémy. Ať již je nástrojem nebo laboratoří, je jeho užitečnost imponující: náš e-Golem je velmi účinný a svědomitý. Neměli bychom však tomuto Golemovi slepě věřit. Někdy může být příroda složitější, než bychom očekávali. Někdy i pro to, že věříme, že něco víme. Například v roce 2004 hurikán Charlie nejenže změnil svou dráhu, ale navíc se změnil z kategorie 2 na kategorii 4 – obě tyto změny byly zcela neočekávané (a krajně nepříhodné).

## Literatura


Appel K., Haken W.: *Every planar map is four colorable. Part I. Discharging*. Illinois J. Math., 21, 1977, s. 429–490.

Apple K., Haken W., Koch J.: *Every planar map is four colorable. Part II. Reducibility*. Illinois J. Math., 21, 1977, s. 491–567.

Arnold V. I.: *Catastrophe theory*. Springer, Berlín 1986.

Berka P.: *Dobávýní znalostí z databází*. Academia, Praha 2003.

Borwein J., Bailey D.: *Mathematics by experiment: plausible reasoning in the 21st century*. A K Peters, Wellesley, MA, 2004.

Brady R., Ball R. C.: *Fractal growth of copper electrodeposits*. Nature, 309, 1984, s. 225–229.

Bressoud D. M.: Computer-generated proofs of mathematical theorems, 2001, http://www.macalester.edu/~bressoud/pub/cgp.pdf.

Cohen J. E.: *Mathematics is biology's next microscope, only better; biology is mathematics' next physics, only better*. PLoS Biol, 2(12), e439, 2004, s. 2017–2023.

Cohen J. E., Gürtler R. E.: *Modeling household transmission of American Trypanosomiasis*, Science, 293, 2001, s. 694–698.

Dawkins R.: The blind watchmaker. 2nd edition, W. W. Norton & Company, New York 1996.

Dickens Ch.: *The Pickwick Papers*. The New American Library, Inc., 1964, s. 360.

Feigenbaum M.: *The universal metric properties of nonlinear transformations*. J. Stat. Phys., 21, 1979, s. 669–706.

Fermi E., Pasta J., Ulam S.: *Studies in nonlinear problems, I*. Los Alamos Report LA 1940. Reproduced in: A. C. Newell (ed.), *Nonlinear wave motion*, Ame. Math. Soc., Providence, RI, 1974.

Forrester J.: *World dynamics*. Wright-Allen Press, Cambridge, MA, 1971.

Gleick J.: *Chaos*. William Heinemann, Londýn, Reprinted several times by Abacus, 1986.

Grimmet G.: *Percolation*. Springer-Verlag, 2nd edition, Berlín 1999.

Hegland M.: *Data mining techniques*. Acta Numerica, 10, 2001, s. 313–355.

Herrera A., Scolnik H. D., Chichilnisky G., Gallopin G., Hardoy J., Mosovich D., Oteiza E., Romero Brest G., Suárez C., Talavera L.: *Catastrophe or new society?* International Development Research Center (IDRC), Ottawa 1976.

Jacovkis P. M.: *Computación, azar y determinismo*. Ciencia Hoy, 5, No. 28, 1995, s. 44–50.

Kennedy J. F.: *The mechanics of dunes and antidunes in erodible-bed channels*. J. Fluid Mech., 16, 1963, s. 521–544.

Lanford III O. E.: *A computer-assisted proof of the Feigenbaum conjectures*. Bull. Ame. Math. Soc., 6, 1982, s. 427–431.

Lanford III O. E.: *A shorter proof of the existence of the Feigenbaum fixed-point*. Commun. Math. Phys., 96, 1984, s. 521–538.

Lorenz E. N.: *Deterministic nonperiodic flow*. J. Atmos. Sci. 20, 1963, s. 130–141.

Lorenz E. N.: *The essence of chaos*, University of Washington Press, Seattle 1993.

Mackenzie D.: *Mathematical modeling and cancer*. SIAM News, 37:1, 2004, s. 1–3.

Macrae N.: *John von Neumann*. Pantheon Books, New York 1992. Reprinted by the American Mathematical Society in 1999.

Matsushita M., Sano M., Hayakawa Y., Honjo H., Sawada Y.: *Fractal structures of zinc metal leaves grown by electrodeposition.* Phys. Rev Lett., 53, 1984, s. 286–289.

May R. M.: *Simple mathematical models with very complicated dynamics.* Nature, 261, 1976, s. 459–467.

McCulloch W. S., Pitts W.: *A logical calculus of the ideas immanent in nervous activity.* Bulletin of Mathematical Biophysics, 5, 1943, s 115–133. Reprinted in Anderson J. A., Rosenfeld E. (eds) *Neurocomputing: foundations of research*, MIT Press, Cambridge, MA 1988, s. 18–28.

Meadows D. H., Meadows D. C., Randers J., Behrens W. W.: *The limits of growth.* Universe Books, New York 1972.

Mesarovic M.: *Practical application of global modeling.* in: Lazarevic B. (ed.), *Global and large scale system models.* Springer, Berlín, 1979, s. 58–80.

Metropolis N., Ulam S.: *The Monte Carlo method.* Journal of the American Statistical Association, 44, 1949, s. 335–341.

Petkovsek M., Wilf H. S., Zeilberger D.: *A=B*, A K Peters, Wellesley, MA, 1996.

Richardson L. F.: *Weather prediction by numerical process.* Cambridge University Press, Cambridge, UK. Reprinted by Dover, New York 1956.

Robertson N., Sanders D. P., Seymour P. D., Thomas R.: *The four color theorem.* J. Combin. Theory Ser. B, 70, 1997, s. 2–44.

Quinteros J., Jacovkis P. M., Ramos V. A.: *Modelado numérico del levantamiento de un orógeno y su relación con clima y erosión.* in: G. C. Buscaglia E. A. Dari and Zamonsky O. M. (eds), *Mecánica computacional.* Vol. XXIII (Proceedings of ENIEF 2004, Bariloche), 2004, s. 2923–2932.

Quinteros J., Jacovkis P. M., Ramos V. A.: *Evolution of the upper crustal deformation in subduction zones.* to be published in ASME Journal of Applied Mechanics, 2006.

Scolnik H. D.: *A critical review of some global models.* in: B. Lazarevic (ed.), *Global and large scale system models.* Springer, Berlín, 1979, s. 58–80.

Stauffer D.: *Introduction to percolation theory.* Taylor & Frances, Londýn 1985.

Strogatz S.: *The real scientific hero of 1953.* The New York Times, March 4, 2003.

Sussmann H. J.: *Catastrophe theory.* Synthèse, 31, 1975, s. 229–270.

Sussmann H. J., Zahler R.: *Catastrophe theory as applied to the social and biological sciences: a critique.* Synthèse, 37, 1978, s. 117–216.

Turing A.: *Computing machinery and intelligence.* Mind, 59, 1950, s. 433–460.

Varsavsky O., Annex I, in: E. de Gortari, T. Garza H., C. Dagum,. J. Hodara and O. Varsavsky, *El problema de la predicción en ciencias sociales.* Instituto de Investigaciones Sociales, Universidad Nacional Autónoma de México, México, D. F., 1969a, s. 115–190.

Varsavsky O.: *Los modelos matemáticos y la predicción en ciencias sociales.* in: E. de Gortari, T. Garza H., C. Dagum,. J. Hodara and O. Varsavsky, *El problema de la predicción en ciencias sociales.* Instituto de Investigaciones Sociales, Universidad Nacional Autónoma de México, México, D. F., 1969b, s. 97–114.

Venter J. C. a kol.: *The sequence of the human genome.* Science, 291, 2001, s. 1304–1351.

Witten T. A., Sander L. M.: *Diffusion-limited aggregation, a kinetic critical pheno-menon.* Phys. Rev. Lett., 47, 1981, s. 1400–1403.

Zabusky N. J., Kruskal M. D.: *Interaction of "solitons" in a collisionless plasma and the recurrence of initial states.* Phys. Rev. Lett. 15, 1965, s. 240–243.

Zeilberger D.: *Determinants and computer algebra.* edited version of the transcript of a talk given at the Max-Planck Institute, Lipsko, August 5, 2002, available from http://www.math.rutgers.edu/~zeilberg/mamarim/mamarimPDF/opa.pdf.

Zworykin V. K., Olessi F.: *Iconoscope. An autobiography of Vladimir Zworykin*, unpublished manuscript, Princeton, NJ, 1971; available from http://www.davidsarnoff.org/ vkz.htm..

[1] http://www.expmath.org

# 9 Úloha znalostí v inženýrském designu

## 9.1 Úvod

Podstatná část lidské činnosti je zaměřena na hledání a uskutečňování akcí, které mění prostředí tak, aby lépe vyhovovalo naším představám a požadavkům. Úlohy, jejichž cílem je navrhnout takovéto změny, budeme souhrně označovat jako *úlohy designu*, *konstrukční úlohy* nebo jen krátce *design*. Z uvedených charakteristik se nabízí definice designu jako úlohy s cílem navrhnout posloupnost akcí, které převedou prostředí z počátečního stavu do předepsaného cílového stavu s tím, že nebudou porušena zadaná omezení a bude optimalizováno zvolené kritérium kvality. Jak ukážeme, reálné problémy designu jsou složitější, silně závisí na využití znalostí a tato definice je dostatečně necharakterizuje.

Tato kapitola si klade čtyři cíle:

1. vysvětlit, proč hrají znalosti klíčovou roli při řešení úloh designu,
2. ukázat, jak lze organizovat znalosti podle použitého modelu řešení,
3. objasnit roli explicitních a implicitních znalostí,
4. navrhnout způsoby reprezentace, sdílení a předávání znalostí v kontextu úloh designu.

Není asi třeba příliš vysvětlovat, že design je důležitou součástí inženýrských disciplín. Návrh nového automobilu, řídicího systému v elektrárně, konstrukce nového obráběcího stroje nebo plán na stavbu domu jsou příklady inženýrského designu. Herbert Simon ale ukazuje, že i mnoho jiných disciplín řeší úlohy, jejichž cílem je změnit současný stav prostředí na jiný, požadovaný (Simon, 1996). Například lékař hledá takové zásahy do organismu, které pacienta vyléčí, pedagog se snaží zorganizovat látku tak, aby student nabyl znalosti z dané problematiky, architekt navrhuje lidská sídla tak, aby se v nich příjemně žilo, obchodník se snaží vytvořit poptávku po určitém výrobku. Ve všech případech se jedná o cílevědomý zásah do prostředí tak, aby byl dosažen požadovaný stav (zdraví pacienta, nové znalosti, jednotlivý dům nebo urbanistický celek, prodej výrobku). Podobných příkladů z různých oblastí lidské činnosti lze nalézt mnoho. Podle Simona lze všechny tyto činnosti chápat jako úlohy designu, protože řešitel vždy hledá takové zásahy do prostředí, které změní současný stav na nový, požadovaný. V anglosaské literatuře se takovéto disciplíny někdy označují jako *professions*. Opakem jsou přírodovědné disciplíny, jako fyzika, matematika nebo biologie, které se snaží svět analyzovat, poznat, pochopit a popsat. Ty se potom nazývají *sciences*. Obě skupiny disciplín se svým obsahem samozřejmě prolínají a navzájem doplňují. V profesních úlohách je zapotřebí poznávat prostředí a jeho vlastnosti, naopak přírodovědci často

potřebují ke svému bádání cílevědomě připravené zásahy do zkoumaného prostředí. V historii inženýrského studia lze nalézt období, kdy se přírodovědné disciplíny neúměrně preferovaly na úkor inženýrských, a naopak, kdy se v inženýrském studiu příliš nevyžadovaly znalosti přírodních věd (Schön, 2003).

## 9.2 Dobře a špatně strukturované problémy

Úlohy designu patří do třídy *špatně strukturovaných problémů*. V literatuře lze tyto problémy nalézt prod názvem *ill-structured problems* (Simon, 1973). Rittel a Webber nazývají podobnou skupina problémů *wicked problems* (Rittel a Webber, 1973). Opakem špatně strukturovaných problémů jsou *dobře strukturované problémy*, v originálu nazývané *well-structured problems* (Simon, 1973) nebo *tame problems* (Rittel, Webber, 1973)[1].

Špatně strukturované problémy jsou vymezeny jako doplněk k dobře strukturovaným problémům.

Dobře strukturované problémy jsou potom charakterizované následujícími podmínkami:

1. Existuje úplná výchozí specifikace problému. To znamená, že existuje stavový prostor úlohy, ve kterém lze definovat výchozí a cílový stav.
2. Existuje konečné kritérium, které by umožňovalo posoudit zda navržené řešení vyhovuje zadání.
3. Existuje kritérium, které by dovolovalo optimalizovat řešení.
4. Existuje explicitní reprezentace přechodů mezi stavy ve stavovém prostoru úlohy.
5. Znalosti spojené s řešením lze vyjádřit jako součást stavového prostoru úlohy.
6. Uvedené podmínky lze efektivně vyhodnotit existujícími výpočetními prostředky.

Uvedený seznam nelze pokládat za úplný, postihuje jenom ty nejdůležitější vlastnosti. Například Rittel a Webber ukazují, že špatně strukturované problémy obsahují často také důležité sociální hledisko: v úloze existuje více agentů („zákazníků" a „řešitelů"), jejichž partikulární zájmy jsou vzájemně rozporné a řešení je musí nějak uspokojit.

Pokud některá z podmínek není splněna, řekneme, že problém je špatně strukturovaný. Rozdělení problémů na dobře a špatně strukturované však není ostré, mezi dobře a špatně strukturovanými úlohami existuje celé spektrum možností. Vždy je třeba ověřit, jak jsou nebo nejsou jednotlivé podmínky splněny a podle toho posoudit do jaké míry je daný problém dobře nebo špatně strukturovaný. Podrobnější diskusi vlastností špatně strukturovaných problémů lze najít v (Rittel, Webber, 1973).

Zabývejme se detailněji jednotlivými podmínkami, které rozdělují úlohy na dobře nebo špatně strukturované.

Podmínky 1, 2, 3 a 4 se týkají specifikace problému a jeho řešení. U špatně strukturovaných problémů nelze dopředu přesně popsat problém. Zadání se upřesňuje jako součást procesu řešení úlohy. Informace potřebné k detailnímu pochopení zadání závisejí

[1] Výrazy „wicked" a „tame" jsou zde odborné termíny, který však mají v doslovném překladu silnou emotivní konotaci. Proto jsou do češtiny obtížně přeložitelné. Po konzultaci s pracovníky Ústavu translatologie Filosofické fakulty Univerzity Karlovy v Praze budeme v textu používat neutrální české termíny „dobře a špatně strukturované problémy".

na způsobu řešení a nelze je získat dopředu. Teprve v okamžiku, kdy řešitel získá výsledek, se kterým je spokojen, může zformulovat konečnou podobu zadání.

Jako jednoduchý případ ukážeme návrh simulačního modelu pérování osobního auta. Cílem bude posoudit jízdní komfort při provozu na silnici. Úlohu můžeme konceptualizovat mnoha různými způsoby. Začněme například modelováním pérování jednoho kola – vytvoříme tzv. model „čtvrtauta". Model celého auta potom získáme spojením čtyř modelů čtvrtauta. Jaký způsob spojení ale použijeme? Modely čtvrtauta můžeme spojit tuhými nosníky, nebo můžeme nosníky modelovat jako 3 hmoty spojené dvěma pružinami (tříhmotový model). Jinou alternativou může být modelování nosníků pomocí metody konečných prvků. Rovněž model čtvrtauta může mít mnoho variant. Je pružina lineární? Jak budeme modelovat tlumič (pro jednoduchost předpokládejme, že je pasivní)? Jaký model použijeme pro rozložení hmot vozidla? Navíc spojení čtyř stejných modelů nemusí, s ohledem na cíl modelování, odpovídat dostatečně přesně realitě. Dynamické vlastnosti přední a zadní nápravy mohou být totiž pro náš účel příliš odlišné. Existuje mnoho dalších otázek, které musíme zodpovědět. Jak zahrneme dynamické vlastnosti pneumatik? Jak popíšeme povrch silnice a jak by měla vypadat zkušební jízda, při které srovnáme simulační model pérování s reálným autem? Je zapotřebí získat výsledek simulace v reálném čase? Všechny tyto otázky se dotýkají jednak specifikace úlohy, jednak kritéria, kdy lze prohlásit problém za vyřešený, protože je už model pro náš účel dostatečně dobrý. S tím úzce souvisí i třetí podmínka. Simon tvrdí, že v reálných úlohách neexistují optimální, ale jenom *dostatečně dobrá* nebo *špatná* řešení. Hledání takovýchto řešení nazývá *satisficing* (Simon, 1996).

Pátá podmínka v uvedeném výčtu se týká znalostí, které jsou k dispozici pro řešení úlohy. Splnění podmínky 5 by znamenalo, že při hledání řešení nemůžeme použít analogií, metafor a implicintních znalostí. Právě roli implicitních znalostí v inženýrských úlohach se budeme věnovat podrobněji v dalším textu. O modelu designu procesem, ve kterém se opakovaně střídají kroky *specifikace problému* a *řešení dané specifikace*, je diskutováno v (Džbor, Zdráhal, 2002).

Šestá podmínka vyžaduje, aby problém byl v praxi řešitelný. V uvedeném příkladu návrhu simulačního modelu auta nelze tedy například vyzkoušet pro všechny možné konceptualizace úlohy všechny možné modely navržených konceptualizací ve všech přípustných podmínkách, protože počet alternativ, které by musely být uvažovány, je příliš veliký. Použitý příklad má sice všechny charakteristiky špatně strukturovaného problému, ale je v rámci této třídy relativně jednoduchý. Úlohy, jako je plán nové dopravní infrastruktury města, návrh nového auta, které by uspělo na trhu, nebo návrh nových studijních plánů pro zvolený univerzitní obor jsou podstatně složitější, jejich možné konceptualizace jsou daleko méně omezené a řešení musí vzít v úvahu často protichůdné zájmy mnoha účastníků řešení.

Porovnejmě řešení úloh z hlediska použitelných metod řešení. Každá metoda má předpoklady, za kterých může být aplikovatelná, a výsledky, které za těchto předpokladů poskytuje. Obecně lze říci, že čím jsou předpoklady přísnější, to znamená, čím metoda požaduje více informací o řešené úloze, tím lepší výsledky poskytuje. Intuitivně řekneme, že metoda poskytuje „horší" výsledky například, když nachází řešení, která splňují jenom nejnutnější kritéria, neumožňuje poskytnout optimální nebo alespoň suboptimální řešení, pokud optimalizuje nenachází globální, ale jenom lokální extrémy, pomalu konverguje, nebo neumí stanovit podmínky řešitelnosti. Předpokládejme, že řešíme

úlohu, ve které jsme schopni pouze ve stavovém prostoru generovat různá potenciální řešení a máme k dispozici kritérium, které dovoluje otestovat, zda je vygenerovaný kandidát řešením úlohy (viz podmínka 2). Jediná dostupná metoda spočívá v generování dosud neověřených kandidátů a jejich postupném testování. Uvedená metoda se nazývá *generate-and-test*. Pokud budou ještě k dispozici operátory, které popisují přechody mezi stavy (viz podmínka 4), lze metodu vylepšit tím, že se nebude generovat dvakrát stejný kandidát. Ještě lepší výsledky dostaneme, když budeme umět ohodnotit kritériem jednotlivé kroky generované pomocí operátorů. Pokud bude navíc toto kritérium splňovat vlastosti vzdálenosti a umožní lokální vyhodnocování, lze využít k řešení *gradientní metody*. Pokud bychom navíc uměli definovat počáteční a cílový stav a uměli měřit vzdálenosti v celém stavovém prostoru úlohy, lze zkonstruovat řešení *algoritmem A** přímo, aniž bychom testovali stavy, které se na řešení podílet nebudou. Je zřejmé, čím více toho o úloze víme, tím lepší výsledky může poskytnout použitelná metoda řešení. Metody, které požadují málo informace o úloze a taky poskytují jenom omezené výsledky, se nazývají *slabými metodami* (weak methods). Naopak, pokud existuje dostatečně veliká informace o úloze, lze použít silnější metody, např. takové, které převedou úlohu na řešení diferencialních nebo algebraických rovnic a tak poskytnou daleko lepší výsledky. Špatně strukturované problémy jsou charakteristické tím, že pro ně existují jenom slabé metody řešení (Newell, 1973). Protože většinou požadujeme daleko lepší výsledky než ty, které mohou poskytnout slabé metody řešení, je nezbytné transformovat špatně strukturované problémy na dobře strukturované.

Ukážeme, že právě v tomto procesu hrají znalosti rozhodující roli. S tím souvisejí následující dvě otázky:

1. Proč jsou úlohy designu špatně strukturované?
2. Jsou všechny úlohy inženýrského designu špatně strukturované, nebo existuje také dobře strukturovaný design?

Jednou z definičních vlastností špatně strukturovaných problémů je jejich neurčitost, jak je zřejmé z prvních čtyř definičních podmínek dobře a špatně strukturovaných problémů. Tato neurčitost však neznamená, že dosud nebyla provedena potřebná měření nebo pozorování nutná k popisu problému, formulaci kritérií nebo definici stavových přechodů. Jedná se spíše o „neurčitelnost", protože bez doplňujících předpokladů potřebné údaje získat nelze. Důvodem je, že úloha designu sama o sobě má velmi obecné a univerzální vymezení, ale nemá žadný vlastní obsah (Buchanan, 1992). Tím sice zahrnuje mnoho různých a velmi odlišných úloh (viz Simonovo chápání designu), ale neodkazuje k nějaké konkrétní problémové oblasti. Tím nezapojuje do úlohy další vědní disciplíny, které by jednak umožnily specifikovat problém designu v rámci této disciplíny a navíc poskytly silné metody řešení. Schön říká, že je zapotřebí vyjmenovat (*name*) objekty, s kterými budeme pracovat, a vymezit (*frame*) jejich kontext (Schön, 2003). Teprve když se řešitel rozhodne interpretovat úlohu designu a dát ji konkrétní náplň, může využít příslušné znalosti. Říkáme, že je problém špatně strukturovaný „ve velkém" ale dobře strukturovaný „v malém" (terminologická analogie s pojmy, které se týkají stability nelieárních systémů). Úměrně tomu, jak znalosti pomohou splnit podmínky 1–6, posune se úloha směrem k dobře strukturovanému konci spektra.

Pokud se tedy například rozhodneme modelovat pérování automobilu jako čtvrtauto s lineární pružinou, s tlumičem, jehož tlumicí síla je úměrná rychlosti, a s konstantní hmotností, převedeme původní špatně strukturovaný problém na dobře strukturovaný.

Rozhodnutím, dát problému právě tuto strukturu, interpretujeme původní úlohu designu a tím přivádíme do řešení znalosti z příslušné vědní oblasti, v tomto případě fyziky a matematiky. Tyto znalosti dovolí upřesnit zadání problému sestavením odpovídajících rovnic, určením jejich parametrů a vyřešit úlohu použitím některé ze standardních metod pro řešení diferenciálních rovnic. K řešení původní úlohy je tedy zapotřebí dvou „typů" znalostí: znalostí na vytvoření struktury pro špatně strukturovaný problém a znalostí na vyřešení výsledného dobře strukturovaného problému. Nejrůznější metody řešení dobře strukturovaných problémů jsou popsány v učebnicích příslušné vědní disciplíny. Základní informace o tom jak vytvořit strukturu pro špatně strukturované problémy by měly být součástí inženýrského studia. Protože jsou však špatně strukturované problémy jedinečné, neexistují obecná pravidla a expertní znalosti se získávají zejména zkušeností, napodobováním úspěšných řešení a poučením se z chyb. Protože z principu neexistuje kritérium, které by ohodnotilo, zda je navržená struktura optimální, nelze ani nalézt kritérium pro posouzení výsledku dobře strukturované úlohy jako řešení úlohy původní. V úloze o pérování automobilu to může znamenat, že sice správně sestavíme rovnice, nalezneme příslušné konstanty a rovnice vyřešíme, ale model pérování pomocí čtvrtauta nedostačuje k posouzení komfortu jízdy v některých situacích. V takovém případě je třeba zvolit jinou strukturalizaci problému a celý proces opakovat. Úlohu prohlásíme za vyřešenou, když rozhodneme, že je nalezené řešení je dostatečně dobré. Zadání úlohy se tedy vytváří v průběhu řešení a jeho konečná podoba je nalezená, až když je úloha vyřešená.

Existují úlohy designu, které lze označit za dobře strukturované? Úlohy designu můžeme zhruba rozdělit do tří základních skupin:

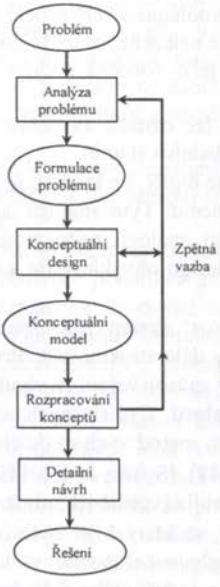
- *Kreativním designem* rozumíme úlohy, ve kterých je k dispozici velmi málo znalostí potřebných ke strukturování problému. Tyto znalosti jsou většinou implicitní, metody řešení často vycházející z hledání analogií nebo metafor (Altshuller, 1984), (Dym, 1994), (Schön, 2003). Kreativní design obvykle vede k objevení nových principů nebo ke vzniku zcela nových artefaktů.
- Úlohy označené jako *inovativní design* jsou špatně strukturované, ale existují znalosti, které dovolují tyto úlohy dále strukturovat. Protože je každá úloha jedinečná a nelze dopředu rozhodnout, který způsob řešení je vhodný. Po zvolené konceptualizaci úlohy však existuje dostatek znalostí z příslušných vědních oborů, které umožňují pokračovat v řešení, často použitím metod vycházejících ze základních principů přírodních věd (Dym, 1994), (Schön, 1988), (Schön, 2003). Příkladem může být návrh nového domu nebo nového modelu automobilu (včetně již zmíněné úlohy pérování).
- *Rutinním designem* jsou úlohy, ve kterých již bylo o struktuře do značné míry rozhodnuto. Příkladem může být *konfigurační design*, ve kterém existuje konečný soubor částí, pravidla pro jejich skládání ve větší celky a funkce, kterou má výsledný artefakt splňovat. Jedná se tedy o kombinatorickou úlohu. Problémem je obvykle návrh metody řešení, které by splnilo zadané kritérium. Jinou verzí rutinního designu je *parametrický design*, ve kterém jsou již i možné konfigurace dány a úlohou je optimalizovat hodnoty parametrů vzhledem k zadanému kritériu (Motta a kol., 1996).

Kreativní a inovativní design jsou špatně strukturované problémy s různou mírou a typem znalostí, které lze využít pro konceptualizaci. Naopak, rutinní design je v podstatě dobře strukturovaný problém, ve kterém je pouze třeba najít metodu, která s da-

nými výpočetními prostředky poskytne výsledek v přijatelném čase. Pokud je výpočetní komplexnost příliš veliká, řeší se úloha aplikováním vhodných heuristik.

## 9.3 Design jako proces

Z uvedeného výkladu je zřejmé, že znalosti hrají v úlohách designu velmi významnou roli. V první fázi pomáhají interpretovat úlohu tak, aby ji šlo posléze řešit metodami, které známe z fyziky, matematiky, biologie a ostatních přírodních věd. Tím, že převedeme původní špatně strukturovaný problém na dobře strukturovaný, umožníme potom použití mnohem silnějších metod řešení, které fyzika, matematika a další vědní disciplíny nabízejí a které poskytují přesnější, určitější a podrobnější výsledky nebo zkracují dobu potřebnou k řešení. Obě fáze se opakují tak dlouho, dokud nezískáme přijatelný výsledek. Jak již bylo řečeno, v mnoha případech neexistuje konečné kritérium, které by dovolovalo jednoznačně stanovit, že výsledek splňuje zadání. Důležitá je tedy také znalost, která umožňuje rozhodnout, kdy už není zapotřebí výsledek zlepšovat.




Obr. 9.1 Model designu jako procesu.

V literatuře existuje několik modelů procesu, které popisují řešení konstrukčních úloh (French, 1971), (Dym, 1994). Často používaný Frenchův model je uveden na obr. 9.1 (French, 1971).

Model popisuje úlohu designu jako proces se čtyřmi kroky: analýzu problému, konceptuální design, rozpracování zvolených konceptů a detailní návrh. Tyto kroky se však můžou mnohokrát opakovat.

Řešení úlohy designu budeme demonstrovat na příkladu návrhu mostu. Smyslem příkladu je ukázat, jak se znalosti z různých vědních oborů postupně zapojují do řešení v závislosti na předchozích rozhodnutích konstruktéra a jak se tím stává problém postupně lépe a lépe strukturovaný.

### 9.3.1 Analýza problému

Prvním krokem je analýza daného problému vedoucí k počáteční specifikaci úlohy. V této fázi se zejména popisují funkce, které by měl navrhovaný artefakt zabezpečovat, kritéria kvality a podmínky, které by měl splňovat a ve kterých by měl tyto funkce vykonávat. Přestože je pravděpodobné, že se bude specifikace v průběhu řešení upřesňovat, je žádoucí, aby již v první fázi co nejpřesněji charakterizovala problém.

V příkladu návrhu mostu zaznamenáme, jaké prostředí se má mostem překlenout (řeka nebo údolí?), co se bude po mostě přepravovat (vlaky, automobilová doprava, armáda, chodci, materiál – např. plyn v potrubí) a jak často, jak budou přepravované náklady těžké, v jakých klimatických podmínkách má být most provozován a mnoho podobných otázek týkajících se předpokládané funkce mostu. V neposlední řadě nás bude zajímat očekávaná cena řešení.

### 9.3.2 Konceptuální design

Na základě specifikace navrhne konstruktér konceptuální model, který bude dále rozpracovávat. Zvolí takovou filosofii řešení, od které očekává, že bude vhodným kompromisem mezi navzájem protichůdnými požadavky. Konceptuální model by měl obsahovat všechny komponenty a jejich strukturální a prostorové vztahy. Komponenty by měly být zvoleny tak, aby zabezpečily všechny požadované funkce. V této fázi má konstruktér největší volnost, protože není ještě vázán konkrétními omezeními přírodovědních disciplín. Ty zahrne až jako důsledek zvoleného konceptuálního řešení. Tato volnost ale současně znamená, že musí být vybaven dostatečnými znalostmi, má-li jeho rozhodování konvergovat k dobrému řešení. Jeho znalosti totiž ovlivňují, zda bude řešení originální (kreativní), nebo zda a do jaké míry bude vycházet z řešení dříve vyvinutého. Rovněž se bude rozhodovat, zda bude proces hledání řešení ryche konvergovat, nebo zda bude řešitel zkoušet mnoho variant. V celé úloze designu jsou znalosti potřebné na tento krok nejkritičtější a přitom jsou většinou založené pouze na zkušenostech. Tyto znalosti také nejvíce přispívají k rozdílům mezi řešením experta a začátečníka. V souladu s předpokladem, že expert je vybaven příslušnými znalostmi a ty mu dovolují správně a rychle konceptualizovat úlohu, bychom očekávali, že expertní řešení bude potřebovat méně kroků na to, aby úloze dodalo strukturu. Některé studie však ukazují, že expert záměrně ponechává déle problém špatně strukturovaný, aby si ponechal prostor pro kreativní myšlení. Chováním začátečníka a experta při řešení úlohy designu se zabývá přehledový článek (Cross, 2004), který uvádí i další reference.

Při návrhu mostu se v této fázi rozhodnuje konstruktér, zda zvolí konstrukci mostu jako nosníku, podepřeného na obou stranách, zavěšeného na jednom nebo více pylonech, nebo zda navrhne třeba most jako pontonový. Výsledné rozhodnutí závisí na plánovaném použití mostu, předpokládané době, po kterou bude používán, nákladech na výstavbu, začlenění mostu do zbývající dopravní infrastruktury a do okolní krajiny. Pokud se konstruktér rozhodne pro most podepřený na obou koncích , bude využívat při dalším

řešení poznatky z oblasti geologie a statiky. V případě, že se rozhodne pro pontonový most, bude se další řešení opírat o mechaniku tekutin. Znalosti z těchto vědních oborů využije jednak k upřesnění specifikace problému, jednak potom aplikuje existující metody známé pro řešení této specifikace.

### 9.3.3 Rozpracování konceptů

Koncepty, navržené v předchozím kroku je třeba naplnit konkrétním obsahem. Pro zvolené koncepty určí konstruktér, jak budou realizovány a jak budou vzájemně prostorově uspořádány tak, aby řešení splňovalo všechny požadované funkce. Dále se v této fázi upřesní struktura konceptuálního řešení s využitím znalostí vědních disciplín, které jsou s vybranými koncepty spojené.

Pokud se při návrhu mostu rozhodl konstruktér pro pontonový most, je náplní této etapy spočítat velikost jednotlivých pontonů s ohledem na jejich nosnost a předpokládaný provoz, navrhnout jejich uspořádání, navrhnout vozovku a rozpracovat nájezdy a výjezdy z mostu v obou směrech.

### 9.3.4 Detailní návrh

Cílem detailního návrhu je určit všechny zbývající parametry a dokončit návrh tak, aby všechny součásti řešení byly plně specifikovány. Znalosti potřebné k detailnímu návrhu mají často podobu vzorců nebo vztahů a lze je nalézt v tabulkách nebo v inženýrských příručkách.

Při návrhu mostu by detailní návrh obsahoval podrobné rozpracování konstrukce mostu, např. na úroveň standardní technické dokumentace potřebné ke stavbě.

Na modelu designu podle obr. 9.1 si všimněme několika jeho vlastností:

- Základní charakteristikou modelu je změna reprezentace mezi jednotlivými kroky. Výsledkem konceptuálního designu je řešení popsané prostřednictvím základních komponent. V dalším kroku řešitel navrhne jak realizovat komponenty pomocí objektů, které jsou k tomu vhodné. Většinou existuje více alternativ, jak lze danou komponentu realizovat. V posledním kroku se navrhne detailní popis a konkrétní podoba těchto objektů. Celý proces designu lze chápat jako několikrát opakovanou změnu reprezentace řešení. Výchozí specifikace se transformuje na konceptuální řešení, to se změní na popis jednotlivých konceptů a ty se nakonec přemění na konečné řešení. Z hlediska transformace reprezentací úloha zdánlivě připomíná klasický překlad počítačového programu, např. v programovacích jazycích Fortran nebo C. Zdrojový program odpovídá zadání problému. První průchod překladačem připomíná vytvoření konceptuálního řešení. Program je vyjádřen v nějakém mezijazyce – problém designu je popsán konceptuálním řešením. Případný druhý průchod je analogií rozpracování konceptů ve Frenchově modelu. Mezijazyk je přeložen do relativního kódu – koncepty jsou vyjádřeny pomocí objektů, kterými by měly být realizované. V posledním kroku jsou k relativnímu kódu připojeny knihovny a výsledkem je absolutní kód – objekty v úloze designu jsou reprezentovány svými konkrétními parametry. Toto srovnání má ale dvě zásadní odlišnosti. Především je zadáním zdrojového programu jednoznačně určena výsledná podoba spustitelného kódu. Programátor většinou nikde během celého procesu nezasahuje a nerozhoduje jak realizovat další krok. Naopak designér v každém kroku rozhoduje jak bude postupovat dál a formuluje předpoklady pro své rozhodnutí. Každý designérův krok je podmíněný.

Druhým rozdílem je, že překlad programu postupuje podle tzv. vodopádového modelu (waterfall model). To znamená, že se jednotlivé dílčí etapy ukončí a v dalším zpracování se k ním už nevracíme. Protože jsou však úlohy designu šatně strukturované, očekáváme, že bude zapotřebí revidovat minulá rozhodnutí, měnit konceptualizaci, popř. realizaci jednotlivých konceptů. V uvedené analogii srovnáváme překlad zdrojového kódu, nikoliv vývoj programu, který je sám úlohou designu a proto může mít charakteristiky špatně strukturovaného problému.

- V modelu existuje několik zpětných vazeb. Pokud stávající formulace problému nedovolí navrhnout vhodnou konceptualizaci, je nutné opakovat analýzu problému a úlohu částečně přeformulovat. Podobně může nastat situace, že zvolené konceptualní řešení nelze dále rozpracovat, aniž by se porušila omezení kritická pro přijatelnost výsledku. V takovém případě je nutné buď hledat jiné konceptuální řešení, nebo se dokonce opět vrátit a opakovat analýzu problému. Nabízí se otázka, proč konstruktér neudělal dostatečně podrobnou analýzu problému nebo nezvolil správnou konceptualizaci hned napoprvé, proč se musí předchozí výsledek korigovat? K tomu, abychom tuto otázku objasnili, se musíme vrátit k základním vlastnostem úlohy designu. Protože je design špatně strukturovaný problém, jeho specifikace se upřesňuje v průběhu řešení, a naopak, metoda řešení závisí na momentální specifikaci. V prostředí značné neurčitosti mají rozhodnutí konstruktéra jak předpokládaný, tak ale i nepředpokládaný efekt. Právě neočekávané důsledky přijatých rozhodnutí mohou vést ke zmíněným konfliktům. Schön říká, že konstruktér musí při řešení úlohy „konverzovat se situací", protože situace mu při řešení „odpovídá" (Schön, 2003). Model tohoto procesu je popsán v (Džbor a Zdráhal, 2002).
- Ačkoliv French svůj model koncipoval pro použití v inženýrském designu (French, 1971) a všechny příklady, které uvádí, jsou zaměřeny na konstruování ve stojírenství, lze tento model snadno aplikovat na obecnou úlohu designu tak, jak to navrhuje Simon.
- V obrázku 9.1 není zpětná vazba od detailního návrhu. V této fázi je totiž už problém obvykle dobře strukturován a detailní návrh je úlohou rutinního designu. V původním Frenchově obrázku také není explicitně znázorněna reprezentace rozpracovaných konceptů, což jsme zachovali.

## 9.4 Podpora rozhodování ve špatně strukturovaných úlohách

Frenchův model rozděluje řešení úloh designu na posloupnost na sebe navazujících kroků. Každý z nich je rozhodovací procedurou, která na základě vstupů a cílů určuje na dané úrovni řešení úlohy. Zpětná vazba umožňuje návrat k předchozím rozhodnutím, pokud jejich výsledek neumožňuje pokračovat. Rozhodovací procedury musejí brát často v úvahu konfliktní kritéria. Frenchův model popisuje proces řešení úlohy, ale neříká, jak v jednotlivých krocích řešení probíhalo, které alternativy byly vzaty v úvahu a proč se řešitel nakonec přiklonil k vybranému kompromisu mezi protichůdnými zájmy jednotlivých stran. Například architekt při navrhování nového domu se snaží skloubit mnoho požadavků:

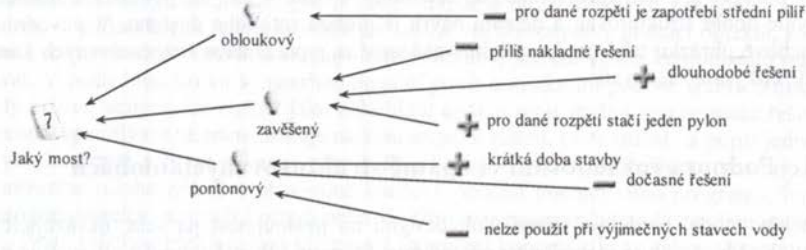
- přání klienta, klient chce, aby velikost a funkce jednotlivých místností odpovídaly jeho potřebám a cena domu byla přijatelná,

- své představy, architekt sám může mít jiný názor na esteticku a funkčnost než jeho zákazník, navíc si je vědom omezení vyplývajících z územních plánů,
- zájmy stavební firmy, ta chce, aby návrh umožňoval použití technologií, které má k dispozici, čímž by dosáhla největší zisk,
- přání majitelů sousedních pozemků, ti nechtějí mít zakrytý výhled, a proto by byli nejraději, kdyby se na parcele nestavělo vůbec.

Všechny tyto požadavky zjevně nelze současně uspokojit. Hledání vhodného kompromisu je proces sestávající z velkého počtu dílčích rozhodnutí, která vyžadují různé typy znalostí. Je proto žádoucí celý proces zaznamenat s tím, že se budou vhodně reprezentovat jednotlivá rozhodnutí, jejich ospravedlnění, důvody pro a proti, žádoucí i nežádoucí důsledky, problémy, které dílčí rozhodnutí vyřešilo, a ty, které naopak vytvořilo.

Rittel a jeho spolupracovníci nejen definovali špatně strukturované problémy a popsali jejich vlastnosti, ale také navrhli způsob, jak tyto problémy řešit (Kunz a Rittel, 1970). Výsledkem je metodologie umožňující zapsat diskutované otázky, uvažované odpovědi a jejich zdůvodnění užité při řešení špatně strukturovaného problému. Z této metodologie, nazvané IBIS (Issue-Based Information System), vychází mnoha dalších metod pro zaznamenávání rozhodovacích procesů spojených s konceptualizací úlohy. U většiny z nich grafické prostředí a podpůrné nástroje umožňují uživateli zaznamenat probíhající diskusi, orientovat se v argumentaci a uložit výsledky pro budoucí použití.

Základními pojmy v systému IBIS jsou *sporný bod* (issue), *možná odpověď* (idea) a *vysvětlení této odpovědi* (argument). Řešení začíná nalezením prvního sporného bodu a jeho vyjádřením ve formě otázky. Způsob formulace otázky je důležitý, protože cílem je, aby otázka umožňovala nabídnout co nejrůznější odpovědi a tak umožnila různé konceptualizace problému. Otázky, na které lze odpovědět jen ano/ne jsou nevhodné, protože omezují počet variant a nemotivují tvůrčí přístup k řešení. Účastníci rozhodovacího procesu mohou nabídnout různé odpovědi spolu s jejich vysvětlením. Příklad možné argumentace při rozhodování o stavbě mostu je uveden na obr. 9.2.

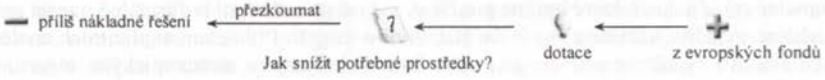



Obr. 9.2 Ukázka záznamu argumentace v systému IBIS.

Na obr. 9.2 je otázka znázorněna ikonou s otazníkem, odpovědi znázorňuje žárovka a argumenty pro nebo proti znázorňuje znaménko + nebo –. Kromě těchto základních relací existují další, které umožňují rozšířit argumentaci. Lze například podrobit přezkoumání každý přednesený argument, jak je znázorněno na obr. 9.3. Podobně lze vyjádřit relace zobecnění a specializaci odpovědi, rozšíření otázky a jiné užitečné relace používané v diskusi.

V případě, znázorněném na obrázku, byl argument „příliš nákladné řešení" zpochybněn účastníkem diskuse, který si uvědomil, že na stavbu mostu lze získat dotaci

z evropských fondů. To je reprezentováno jako nová otázka svázaná relací „přezkoumat" s původním tvrzením. Kromě uvedené odpovědi může tato otázka vést k dalšímu větvení vzhledem k alternativním návrhům. Jiný účastník může třeba dotace odmítat jako příliš nejisté. Takto se graf rozrůstá a zachycuje všechny názory, které se objevily během procesu řešení.




Obr. 9.3 Přezkoumání argumentace.

Pro některé třídy úloh byl rozšířen soubor relací tak, aby dobře popisoval vztahy často se vyskytující v těchto úlohách. Přehled metod, které rozšiřují jazyk IBIS tak, aby lépe vyhovovaly úlohám designu, lze nalézt v (Louritas, Loucopoulos, 2000).

Reprezentace rozhodovacího procesu založená na metodologii IBIS je jednoduchá, nevyžaduje zvláštní znalosti a dobře popisuje argumenty vznesené v průběhu řešení. Tím usnadňuje řešení špatně strukturovaných problémů. Navíc zachovává zdůvodnění všech rozhodnutí a tak umožňuje později plně zrekapitulovat původní řešení a případně ho upravit, pokud se vyskytly nové skutečnosti. Podporuje tedy zpětnou vazbu znázorněnou na obr. 9.1. Zásadní problém špatně strukturovaných problémů však neřeší, nebo jej řeší jen v omezené míře. Protože je IBIS problémově nezávislá metoda, neobsahuje znalosti specifické pro řešenou oblast problémů. Proto nepřináší do řešení další potřebné znalosti jako důsledek přijaté konceptualizace. Tedy například, rozhodne-li se řešitel pro stavbu zavěšeného mostu, IBIS mu nezprostředkuje přístup k pojmům a metodám řešení z teorie pružnosti a pevnosti pro výpočet lan nebo ke znalostem z geologie potřebným pro návrh pylonu. V následujících oddílech ukážeme jaký typ znalostí se účastní procesu rozhodování a jakým způsobem lze takovéto znalosti sdílet.

## 9.5 Explicitní a imlicitní znalosti

Všimli jsme si, že znalosti hrají v různých fázích procesu designu různou roli a mají různou formu. Při konceptualním designu řešitel na základě svých znalostí navrhuje a vybírá jednu z možných alternativ a tím, kromě jiného, rozhoduje, jaké vědní obory budou potřebné k dalšímu řešení. Zatímco při konceptualizaci srovná nejčastěji řešitel stávající úlohu s tím, co zná ze zkušenosti, a na základě podobnosti se rozhodne jak pokračovat, při detailním návrhu bude používat známé poznatky z příslušné oblasti matematiky, fyziky, chemie atd. V prvním případě jsou znalosti velice osobní, ve druhém případě jsou obecně dostupné. Znalosti jsou však vždy vázány na nějakého agenta – řešitele, který je může využívat k rozhodování o svých akcích. Říkáme, že agent *vlastní* nebo má znalosti. Vlastnictví znalostí nesouvisí s tím, zda je agent momentálně využívá – konstruktér má znalosti potřebné ke návrhu mostu, i když právě spí nebo řídí auto. V úlohách inženýrského designu je agentem obvykle člověk (konstruktér) nebo skupina lidí – tým, který se podílí na návrhu artefaktu. Chápání znalostí jako něčeho, co lze vlastnit, dovoluje jednoduše interpretovat pojmy *sdílení znalostí* (knowledge sharing) a jejich *znovupoužití* (knowledge reuse). Sdílení je poskytnutí znalostí jednoho agenta

jinému agentovi, znovupoužití znalostí nastává v čase a znalosti může použít jak původní vlastník, tak i jiní agenti. Vzhledem k různým formám znalostí však procesy sdílení a znovupoužití nejsou zcela jednoduché. Problém je v tom, že lidé obvykle nejsou schopni popsat, co znají, a naopak neumějí aplikovat vše, co jsou schopni popsat. Znalosti, které umíme popsat slovy, matematickými formulemi, pravidly – obecně jakýmkoliv neformálním nebo formálním způsobem, nazýváme *explicitní* (explicit knowledge). Znalosti, které umíme používat, ale nejsme schopni je formálně popsat nebo verbálně vyjádřit, nazveme *implicitní* (tacit knowledge)[2]. Příkladem explicitních znalostí jsou znalosti vyjádřené přirozenými nebo formálními jazyky, matematickými algoritmy, logickými pravidly, heuristikami, vzorci apod. Implicitní znalosti takto zapsat nelze. Projevují se činnostmi, při kterých byly použity. Příkladem mohou být znalosti potřebné k vykonání nějaké pohybové akce. Mnoho lidí umí jezdit na kole, ale většina z nich není schopna vysvětlit, jak otočit řidítka, když se kolo příliš naklání na jednu stranu. Podobně operátor portálového jeřábu neumí vysvětlit jak pojíždět s vozíkem jeřábu tak, aby se náklad nekýval. I kdyby existovaly podrobné instrukce v podobě textu nebo obrázků, cyklista ani operátor se nejsou schopni podle nich v reálné situaci rozhodovat. V uvedených příkladech řídí na základě svých implicitních znalostí motorickou činnost. Existují ale i příklady, kdy implicitní znalosti řídí kognitivní činnosti. Lékař je schopen stanovit diagnózu, aniž by uměl vyjmenovat všechny příznaky, které ho vedly k danému závěru. Šachový velmistr dokáže vyhodnotit koncovku, ve které je na šachovnici jen několik figur, aniž by zkoumal všechny možné tahy. Podobné příklady lze najít i v ostatních profesních disciplínách včetně inženýrského designu.

Z hlediska sdílení a znovupoužití mají explicitní a implicitní znalosti různé vlastnosti. Explicitní znalosti lze jednoduše uschovávat, kopírovat nebo předávat. Všechny tyto operace totiž pomáhají zprostředkovat média, na kterých jsou znalosti zapsané. Implicitní znalosti se jeví jako obtížněji uchopitelné, nepřenositelné a nepřístupné exaktnímu zkoumání. Jejich základní vlastností je, že jsou vázány na nějakou akci a jen jejím prostřednictvím se projevují. Zdálo by se, že by mělo být naším cílem převést všechny znalosti do explicitní formy a tak usnadnit jejich další přenos. Je ale vůbec možné transformovat implicitní znalosti na explicitní a naopak? Explicitní a implicitní znalosti jsou dvě různé formy, které mohou existovat nezávisle na sobě. Implicitní znalosti nemůžeme „převést" na explicitní a naopak, lze však použít jednoho typu znalostí jako pomůcku k *vytvoření* typu druhého. Tento krok je *vždy* zprostředkován činností, na kterou se implicitní znalosti váží (Cook a Brown, 1999). Příklad, jak lze podporovat vytváření implicitních znalostí v architektuře, uvádí (Haylighen a kol., 2006).

## 9.6 Příklady vytváření znalostí

Nonaka uvádí jako příklad demonstrující vzájemné vztahy explicitních a implicitních znalostí následující projekt (Nonaka, 1998). Firma Matsushita Electronic Company se rozhodla, že vyvine a uvede na trh kuchyňský stroj na domácí pečení chleba. Úkol navrhnout takový stroj byl zadán vývojovému oddělení firmy. Přes veškeré úsilí se však

[2] „Tacit knowledge" je další termín, který je do češtiny obtížně přeložitelný. Kromě zde použitého překladu „implicitní znalosti" lze najít v české literatuře také „skryté", „tiché", „nevyřčené" nebo dokonce „tacitní" znalosti. Termín „implicit knowledge" je používán v psychologii, viz např. (Dienes, Fahey, 1998).

dlouho nedařilo nalézt řešení, které by se kvalitou chleba vyrovnalo profesionálním pekařům. Problém byl zjevně v technologii hnětení těsta. Nicméně ani nejmodernějšími technickými prostředky se nepodařilo dostatečně popsat v čem je problém a následně ho uspokojivě vyřešit. Firma proto nechala zaměstnat jednu ze svých softwarových vývojářek v pekárně velkého hotelu, který byl známý vynikající kvalitou svého chleba. Tato pracovnice pracovala jako pekařka a učila se jak správně hníst těsto. Všimla si, že hlavní pekař používá svůj zvláštní způsob „vytahování těsta". Současně s prací v hotelové pekárně spolupracovala s vývojovým oddělením firmy Matsushita na návrhu stroje. Po ročním úsilí se podařilo vyvinout jedinečnou metodou „kroucení těsta" a navrhnout kuchyňský stroj, který produkoval chléb v požadované kvalitě. Během prvního roku od uvedení na trh vytvořil tento stroj v kategorii kuchyňských spotřebičů nový rekord v počtu prodaných kusů.

V popsaném projektu ukazuje Nonaka následující vztahy mezi implicitními a explicitními znalostmi:

- Softwarová vývojářka si získala implicitní znalosti o hnětení těsta vytahováním od profesionálního pekaře tím, že spolu s ním pracovala na přípravě těsta. Výsledkem bylo, že tato pracovnice uměla správně hníst těsto (ne však nutně popsat tento proces). Pro vytváření implicitních znalostí napodobováním činnosti někoho jiného používá Nonaka termín *socializace*. Podstatné je zde získávání znalostí vlastní činností.
- Softwarová vývojářka také spolupracovala s pracovníky vývojového oddělení firmy Matsushita na předávání znalostí o hnětení těsta. Pravděpodobně jednak předávala své implicitní znalostí také socializací, a to tím, že se spolupodílela na vývoji nového stroje, jednak vyjadřovala své znalosti potřebné ke specifikaci stroje explicitně. Vytváření explicitních znalostí popisováním činnosti, která používá implicitní znalosti, nazývá Nonaka *externalizací*.
- Kromě toho jistě vývojoví pracovnící shromažďovali a uspořádávali své explicitní znalosti z oboru, které hodlali použít k návrhu stroje. Vytváření explicitních znalostí z jiných explicitních znalostí nazývá Nonaka *kombinací*.
- Prací na návrhu stroje si pracovníci vývojového týmu z explicitních znalostí pravděpodobně vytvořili své implicitní znalosti, které jim umožní provádět konstrukční práce podobného charakteru. Vytváření implicitních znalostí z explicitních nazývá Nonaka *internalizací*.

Tento příklad ukazuje, jak se implicitní znalosti získávají prostřednictvím činnosti, pro jejíž vykonávání jsou zapotřebí.

Jiný příklad popisují D. MacKenzie a G. Spinardi, kteří studovali význam implicitních znalostí pro vývoj jaderných zbraní. Jejich studie (MacKenzie, Spinardi, 1995) vychází z analýzy velkého množství veřejně dostupných materiálů. Autoři ukazují, že poté, co vědci a technici USA vyvinuli první dva typy atomové bomby (bomba shozená na Nagasaki používala jiné konstrukční řešení než bomba shozená na Hirošimu), byly potřebné explicitní znalosti roku 1945 zveřejněny ve zprávě vládní agentury USA. Přestože explicitní znalosti byly veřejně dostupné, potřebovaly všechny další státy, které atomovou bombu vyrobily na její vývoj alespoň 4 roky. Autoři studie docházejí k názoru, že tato doba je zapotřebí k tomu, aby si jednotlivé týmy vytvořily implicitní znalosti potřebné zejména na vyřešení technologických problémů. V terminologii Nonaky lze implicitní znalosti získat buď socializací, nebo internalizací. Protože práce jednotlivých týmů probíhaly izolovaně, bez výměny znalostí s ostatní vědeckou a technickou

komunitou, přichází v úvahu pouze internalizace. Jistě lze předpokládat, že na jaderných programech jednotlivých zemí se podíleli nejlepší odborníci z každé země a že tyto programy nebyly zdržovány nedostatekm finančních prostředků. Za těchto okolností je pozoruhodné, že všechny významnější jaderné mocnosti potřebovaly na vývoj zbraně stejně dlouhou doby. Vytváření a získávání znalostí činností bývá také nazýváno *learning by doing*.

Je zajímavé studovat, jak lze podporovat vytvoření explicitních znalostí na základě znalostí implicitních. V Nonakově terminologii se jedná o externalizaci znalostí. Připomeneme, že implicitní znalosti nelze na znalosti explicitní transformovat, překódovat nebo jinak změnit. Externalizací se explictní znalosti vytvoří a implicitní znalosti se zachovají. To ukazuje následující studie.

Oxfordští psychologové Berry a Broadbent připravili experiment, pomocí kterého zkoumali vytváření implicitních znalostí a jejich vyjádření pomocí slovního reprezentace (Berry, Broadbent, 1984). Tří experimentů se účastnila vždy skupina 24 testovaných osob – dobrovolníků z řad studentů Oxford University. Úkolem testovaných osob bylo naučit se řídit jednoduchý dynamický systém jen na základě znalosti vstupu a výstupu a potom popsat slovy svoji řídicí strategii. Matematický model systému testované osoby neznaly a ani nebylo cílem jej identifikovat. Systém byl popsán následujícími rovnicemi:

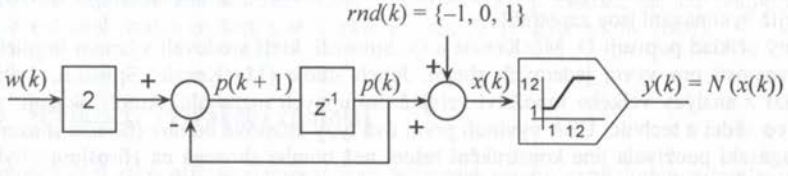
$$p(k+1) = 2w(k) - p(k),$$
$$y(k) = N(p(k) + rnd(k)),$$

kde $N(x) = 1$ pro $x \leq 1$, $N(x) = x$ pro $1 \leq x \leq 12$ a $N(x) = 12$ pro $x \geq 1$, $rnd(k) = \{-1, 0, 1\}$ je náhodný proces se třemi hodnotami –1, 0, 1.

Tento systém můžeme znázornit blokovým diagramem dle obr. 9.4.

Systém obsahuje zpoždění 1. řádu, generátor náhodných poruch $rnd(k)$, který způsobuje, že výstup není nikdy v ustáleném stavu, a nelinearitu typu nasycení, která zabezpečuje, že výstup systému $y(k)$ nebude mít nikdy hodnotu menší než 1 a větší než 12. Hodnoty všech proměnných jsou celočíselné.

Navenek byla úloha prezentována účastníkům experimentu dvěma rozdílnými způsoby: jednak jako řízení výroby cukrovaru a jednak jako udržování stylu konverzace mezi dvěma osobami.




Obr. 9.4 Řízený systém.

V případě, že systém modeloval cukrovar, byl vstupem počet zaměstanaců cukrovaru a výstupem množství vyrobeného cukru. Hodnotě $w = 1$ odpovídalo 100 zaměstnanců, $w = 2$ znamenalo 200 zaměstnanců atd. až hodnotě $w = 12$ odpovídalo 1200 zaměstnanců cukrovaru. Výstup byl interpretován jako množství vyrobeného cukru: $y = 1$ znamená

1000 tun, $y = 2$ znamená 2000 tun a podobným způsobem se interpretují ostatní hodnoty, až $y = 12$ odpovídá 12 000 tunám cukru. Testovaná osoba komunikovala se systémem tak, že zadávala počet pracovníků a dostávala jako odezvu množství cukru. Úkolem bylo naučit se řídit cukrovar tak, aby se výroba udržovala na 9000 tunách cukru, přičemž za správnou se pokládalo takové řízení, které dosáhlo výroby mezi 8000 a 10 000 tunami cukru. Vlivem působení náhodného generátoru je to vlastně nejlepší možný výsledek.

V druhém případě byla úloha vyjádřena jako udržování požadovaného stylu konverzace. Systém modeloval chování fiktivního člověka jménem Clegg. Testovaná osoba „komunikovala" s Cleggem prostřednictvím terminálu tím, že říkala, jakým stylem bude s Cleggem v následujícím kroku jednat, a Clegg „odpovídal", jak na toto jednání bude reagovat. Jak testovaná osoba, tak i Clegg si mohli vybrat vždy jeden z následujících 12 stylů {*velmi hrubý, hrubý, velmi chladný, chladný, indiferentní, zdvořilý, velmi zdvořilý, přátelský, velmi přátelský, laskavý, velmi laskavý a láskyplný*}. V modelu tyto styly odpovídaly hodnotám vstupu i výstupu $w$, $y = 1, 2, ..., 12$. Testovaná osoba tedy například zvolila styl *velmi zdvořilý* a Clegg na to sdělil, že jeho odpověď bude mít styl *přátelský*. Když byl například následující styl tesované osoby *velmi přátelský*, Clegg zvolil styl *laskavý* atd. Komunikace probíhala vzájemným „sdělováním adjektiv" prostřednictvím počítačového terminálu. Cílem bylo přivést Clegga na styl *velmi přátelský* a udržet ho v tomto stavu s přípustnou chybou jednoho stylu. Lze se snadno přesvědčit, že tato úloha je izomorfní s úlohou řízení cukrovaru – v obou případech se jedná o nalezení takového řízení, které bude držet výstup na hodnotě $y = 9$ s chybou +1 nebo –1.

Autoři popisují řadu experimentů, které měly ověřit, jak se testované osoby učí získat implicitní znalosti potřebné k řešení úlohy a jak jsou schopny tyto znalosti vyjádřit explicitně. Především výsledky ukázaly, že obě iterpretace úlohy řízení dávají v podstatě stejné výsledky, pro testované osoby tedy není podstatné, zda řídí výrobu cukru nebo dialog s Cleggem. V první sadě experimentů se ukázalo, že testovaní jedinci si po zhruba 60 krocích vytvořili potřebné implicitní znalosti a byli schopni udržet výstupní proměnnou v požadovaných mezích, a to přesto, že neznali, na jakých principech funguje soustava, kterou řídí. Své znalosti však neuměli popsat – uváděli například, že k řízení používali „svou intuici", nebo že získali „správný pocit pro řízení". V druhé sadě experimentů nejdříve testované osoby chvíli řídily soustavu a potom jim experimentátor přečetl návod s kvalitativními pravidly řízeni. Instrukce vysvětlovaly kromě jiného, že odezva závisí na vzdálenosti současného stavu od požadovaného, že výstup je trvale zatěžován náhodnou chybou, a byly doprovázeny několika ukázkami. Poté testované osoby pokračovaly v nabývání implicitních znalostí řízením soustavy a nakonec opět měly popsat svoji strategii řízení. Po vyhodnocení výsledků této sady experimentů se ukázalo, že verbální instrukce nezlepšily proces učení, ale testované osoby byly lépe schopny vysvětlit svoji strategii řízení. Převedeno do terminologie znalostního inženýrství a Nonaky, poskytnutím ontologie popisující problém se nepodpořil proces učení nebo internalizace znalostí, ale zlepšila se jejich externalizace. Třetí sada experimentů rozšiřovala druhou sadu o požadavek komentovat nahlas akce současně s jejich prováděním v procesu řízení. Výsledky ukázaly výrazné zlepšení při řízení systému, ale vliv na následné vysvětlení použité strategie nebyl zcela přesvědčivý.

V kontextu úlohy designu se podobným problémem zabývali Lloyd a jeho kolegové (Lloyd a kol., 1995). Jejich výzkum se opírá o analýzu protokolu, který zachycuje aktivity konstruktéra Dana, který řeší úlohu návrhu nosiče pro jízdní kolo. Podobně jako

v experimentech s řízením „cukrovaru" Dan řeší úlohu a přitom verbalizuje svoje myšlení – „myslí nahlas". Měřítkem verbalizace je frekvence slov, kterými Dan popisuje svoji činnost. Studiem videa, na kterém je zaznamenána Danova činnost, a porovnáním s frekvencí slov, kterými k daném okamžiku svoji činnost komentoval, lze v protokolu identifikovat čtyři rozdílné fáze:

- *Neverbální myšlení při řešení úloh designu.* Ve chvílích, kdy Dan dělal inženýrský náčrtek, kterým graficky navrhoval možné řešení, jeho verbalizace se výrazně zpomalila. Tento jev je připisován střídavému zapojování různých částí mozku do řešení. Je známé, že řeč a logické myšlení se odehrává v levé mozkové hemisféře zatímco prostorové vnímání a představivost probíhá v pravé hemisféře. Lze se domnívat, že tato fáze odpovídá vytváření konceptualizace úlohy, tedy vytváření struktury problému.
- *Verbální myšlení při řešení úloh designu.* Jestliže je problém dostatečně podrobně konceptualizován a je tedy dobře strukturován, k jeho řešení Dan využívá logické postupy nebo výpočty vyplývající ze znalosti základních vědních disciplín, např. fyziky. V této fázi Dan řeší úlohu a současně popisuje svoje řešení. Frekvence jeho slov je výrazně vyšší než u předchozí fáze.
- *Pochopení úlohy designu.* V této fázi si Dan vytváří vnitřní představu o o tom, co je úlohou v rámci zkušeností a svého vnímání světa. Jedná se o vytvoření určité abstrakce problému. Danovo chování je podobné fázi neverbálního myšlení. Také tuto fázi můžeme zařadit do konceptualizace úlohy, a to časově dříve, než začal Dan kreslit náčrtky.
- *Okamžiky náhlého pochopení.* V této fázi náhle Dan plně pochopil své řešení a začlenil ho do kontextu svých znalostí. Danův verbální projev nemá podstatné pauzy. Někdy se tato fáze nazývá „Aha!" okamžik. Dan je schopen nejen hovořit o svém řešení, ale také ho navázat na další explicitní znalosti. Podle Nonaky bychom klasifikovali tuto fázi jako *externalizaci* znalostí s možností *kombinace.*

Tuto dualitu v řešení složitých úloh na základě implicitních a explicitních znalostí modeluje Michie v (Michie, 2000). K objasnění a vzájemnému porovnání metod řešení používá následující metaforu:

*„Na řešení se podílejí dva agenti. Jeden z nich je schopen řešit úlohy logicky, sestavovat plány a hlavně komunikovat. Tento agent se jmenuje v metaforickém modelu Člověk. Druhý agent, který je zodpovědný za vnímání, abstraktní myšlení a má daleko rychlejší odezvy se nazývá Zombie. Na řešení úloh se podílejí jak Člověk, tak i Zombie, každý podle své specializace."*

Existence těchto agentů je ukázána na případech lidí, v jejchž myšlení jeden „agent" kvůli dysfunkci části mozku plně nepracuje. V tom případě totiž převáží funkce zbývajícího agenta a jeho působení se stává zřetelnějším. Když oba agenti fungují normálně a podílejí se v rámci svých kompetencí na řešení úloh, je mezi nimi omezená komunikace. Zombie řeší úlohy, ale jen obtížně sděluje Člověku, jak to dělá. Přitom jenom Člověk umí komunikovat s vnějším světem a tedy sdělit to ostatním. Podobné potíže ma i Člověk, když má instruovat Zombie. Když náhle začne Zombie a Člověk komunikovat, nastane okamžik náhlého pochopení (viz případ konstruktéra Dana). Michie popisuje případ francouzského matematika Henryho Poincarého. Ten pracoval asi čtrnáct dní úsilovně, ale bez výsledku, na důkazu, že neexistuje to, co se nazývá nyní automorfní funkce. Jednoho večera si dal šálek černé kávy a nemohl usnout. Vtom se mu začaly myšlenky spojovat a organizovat do ligického celku. Zbývalo pouze teorii automorfních funkcí zapsat na papír.

Ve všech uvedených případech se ukazuje, že znalosti se vyskytují ve dvou formách: implicitní, která podporuje činnost, a explicitní, která umožňuje komunikaci. Obě dvě formy jsou potřebné pro řešení úloh a každá hraje svoji roli. Vzájemné vztahy mezi těmito formami nejsou jednoduché.

## 9.7 Stručné shrnutí a požadavky na podporu inženýrského designu

Zopakujme tedy základní fakta o úloze inženýrského designu, znalostech potřebných k řešení a ukažme, jaké vlastnosti by měl mít systém, který by práci inženýra-designera podporoval.

1. Úloha (inženýrského) designu je špatně strukturovaná.

2. K řešení špatně strukturovaných lze použít jen slabé metody, ty však dávají jenom horší výsledky.

3. Úloha designu je špatně strukturovaná, protože nemá sama o sobě žádný obsah. Ten jí dodávají další znalosti, které umožní navrhnout konceptuální řešení, tím vytvoří úloze strukturu a se strukturou přinesou do úlohy další, specializované znalosti a silné metody řešení. Znalosti jsou tedy důležitým činitelem při řešení špatně strukturovaných úloh.

4. Jednotlivé kroky řešení znázorňuje například Frenchův model. Na rozhodování v každém kroku jsou zapotřebí znalosti. V počátečních krocích jsou znalosti spíše heuristické nebo odvozené z již vyřešených případů, v pozdějších krocích se uplatňují zejména znalosti základních disciplín, jako je fyzika, chemie atd.

5. K řešení špatně strukturovaných problémů lze použít metod odvozených z metody IBIS. Ty sice umožňují zaznamenat argumentaci a rozhodnutí spojené s jednotlivými kroky, ale nepřinášejí do řešení ani znalosti potřebné ke konceptualizaci, ani znalosti, které jsou použitelné jako důsledek této konceptualizace.

6. Znalosti existují ve dvou formách, jako implicitní a explicitní. Implicitní znalosti jsou vázány na činnost, ke které jsou zapotřebí, a lze je vytvářet a sdílet jenom vykonáváním této činnosti. Explicitní znalosti lze vyjádřit formálně (vzorce, formule, algoritmy) nebo neformálně (přirozený jazyk).

7. Lze předpokládat, že implicitní znalosti budou častěji používány v úlohách konceptuálního designu a méně často v pozdějších fázích.

Na základě těchto poznatků můžeme popsat požadavky na systém pro podporu inženýrského designu.

1. Strukrura systému bude podporovat Frenchův model. Protože se však jiné modely (Cross, 1994) liší zejména počtem kroků, do kterého je design rozdělen, bylo by užitečné, kdyby systém byl v tomto ohledu flexibilní. V praktických aplikacích by potom bylo možné do modelu v závislosti na konkrétní úloze zahrnout i další kroky životního cyklu artefaktu, jako výrobu, instalaci, údržbu během provozu a nakonec likvidaci.

2. Řešení úlohy designu je svojí podstatou postupnou transformací reprezentace problému. Znalosti by měly být organizovány tak, aby podporovaly očekávané nebo možné kroky transformace.

3. Je zapotřebí podporovat jak reprezentaci explicitních znalostí, tak i prostředí na vytváření implicitních znalostí. Vzhledem k povaze implicitních znalostí se nabízí jako možná varianta intergrace znalostí, používaných k designu, s reprezentací inženýr-

ské práce, tedy s produktem těchto znalostí. Ty mají nejčastěji podobu skic, nákresů, modelů atd.

4. V případě explicitních znalostí je potřeba reprezentovat jak formální znalosti (pravidla, ontologie, taxonomie, vzorce), tak i neformální anotaci textem (vysvětlení, předpoklady, komentáře).

5. Formální a neformální znalosti by měly umožňovat anotování reprezentace inženýrské práce. Tím se, v mezích možností, ukáže spojitost. Schön říká: „Nákresy a slovní popis dohromady tvoří 'jazyk konstruování'" (Schön, 2003) explicitních a implicitních znalostí.

6. Při řesení by systém měl podporovat formu dialogu tak, jak to zavádí systém IBIS.

## 9.8 Podpora inženýrského designu: projekt Clockwork

Představy o znalostní podpoře inženýrského designu, naznačené v předchozích oddílech, byly základem filosofie, architektury a implementace softwarového systému vyvinutého v rámci projektu s názvem „Creating Learning Organisations with Contextualised Knowledge-rich Work Artefacts" a akronymem Clockwork. Projekt byl podporován Evropskou komisí, měl 7 partnerů, trval 39 měsíců a stanovil si tři základní cíle:

- Podporu týmů řešitelů při sdílení simulačních modelů dynamických systémů obohacených znalostmi.
- Podporu znovupoužití znalostí z oboru simulací a modelování v rámci organizací.
- Podporu „opatrného" sdílení znalostí mezi organizacemi.

Tyto cíle byly záhy rozšířeny tak, aby se netýkaly pouze návrhu simulačních modelů, ale obecně inženýrského designu. Třetí cíl se bezprostředně netýká témat popsaných v této kapitole, a proto jen stručně vysvětlíme jeho podstatu: Existují situace, kdy inženýrské organizace spolupracují na vývoji nějakého produktu, ale současně se částí své expertizy překrývají a vzájemně si konkurují na trhu. Spolupráce na vývoji vyžaduje sdílet znalosti, konkurenční prostředí nutí je tajit. Toto je jednak právní problém ochrany duševního vlastnictví, jednak technický problém minimálního nutného sdílení znalostí, což byl právě jeden z cílů projektu Clockwork.

Aplikační oblastí, na kterou byl projekt původně zaměřen, bylo modelování a simulace dynamických systémů. Příkladem takové úlohy může být vytvoření simulačního modelu pérování auta a jeho implementace v prostředí, které poskytuje MATLAB™ a SIMULINK™. Možné zadání této úlohy bylo popsáno v úvodu. Oproti obecné úloze designu má návrh simulačního modelu jeden rozdíl. Cílem není vypracovat detailní návrh simulačního modelu, ale vytvořit tento model a ověřit jeho funkce. Detailní návrh tedy splývá s realizací – „vyrobením" artefaktu podle navrženého plánu. V terminologii Clockworku nazýváme jednotlivé fáze procesu designu *„světy"* (world). Pokud bychom tedy „přejmenovali" Frenchův model terminologií Clockworku, hovořili bychom o světech zadání problému, konceptuálního designu, rozpracování konceptů a detailního návrhu. Každý z těchto světů má své vlastní objekty, znalosti a pravidla jak s objekty zacházet. Protože je simulace a modelování již dobře zavedenou disciplínou, má kromě speciálních znalostí také svou vlastní terminologii. Proto v Clockworku hovoříme o reálném světě, konceptuálním světě, světě modelů a simulačním světě.

Vývoj simulačního modelu je tedy úlohou designu, jejíž cílem je transformovat zadání popsané v reálném světě na simulační model, což je objekt v simulačním světě. Clockwork nabízí metodologii, jak postupovat při řešení úlohy, a vývojové prostředí nazvané Clockwork Knowledge Manager (CKM), ve kterém lze tuto metodologii realizovat. Základem metodologie je anotování důležitých inženýrských znalostí vznikajících v průběhu řešení úlohy. Inženýrské modely mají obvyklou podobu náčtků, schémat, grafů, popř. jiných dokumentů, které designer používá ke své práci. Clockwork archivuje výsledky každého kroku, který designer udělá. Navíc dovoluje připojit anotaci ve formě znalostí, které byly k dosažení výsledku zapotřebí. Znalostní popis se ale vytváří pouze tehdy, pokud může přinést něco nového pro pochopení nebo znovupoužití v budoucnosti. Běžně známé inženýrské znalosti se neanotují. V každém světě tedy existují vedle sebe obvyklá inženýrská dokumentace řešení a odpovídající znalostní model.

Formální znalosti jsou vyjádřeny pomocí pojmů uspořádaných do ontologií. Každý svět je popsán alespoň jednou ontologií. Ontologie obsahují popis objektů, které se v daném světě vyskytují, to znamená, že definují „jazyk", ve kterém se vyjadřují formální znalosti v daném světě. Pokud chce řešitel anotovat nějaký objekt v reprezentaci své inženýrské práce (náčrtku), vytvoří v ontologii instanci třídy, do které objekt patří, vyplní konkrétní hodnoty jeho vlastností (např. rozměry) a jednoznačně přiřadí znalostní popis objektu v inženýrské reprezentaci. Instance znalostního popisu se nazývají v Clockworku sémantické indexy, soubor všech sémantických indexů tvoří bázi znalostí. Kromě indexů, které popisují jednotlivé objekty, lze také popisovat relace mezi objekty stejného světa a relace mezi objekty dvou sousedních světů.

Předpoklady nebo podmínky, za jakých daný krok řešení platí jsou většinou popsány textem v přirozeném jazyce. Komentář se může vztahovat k jednotlivým objektům světa, jejich skupinám nebo k transformaci mezi dvěma světy. V tomto smyslu se jedná o analogii textových komentářů, které používá IBIS. Anotace inženýrského modelu je znázorněna na obrázku 9.5.

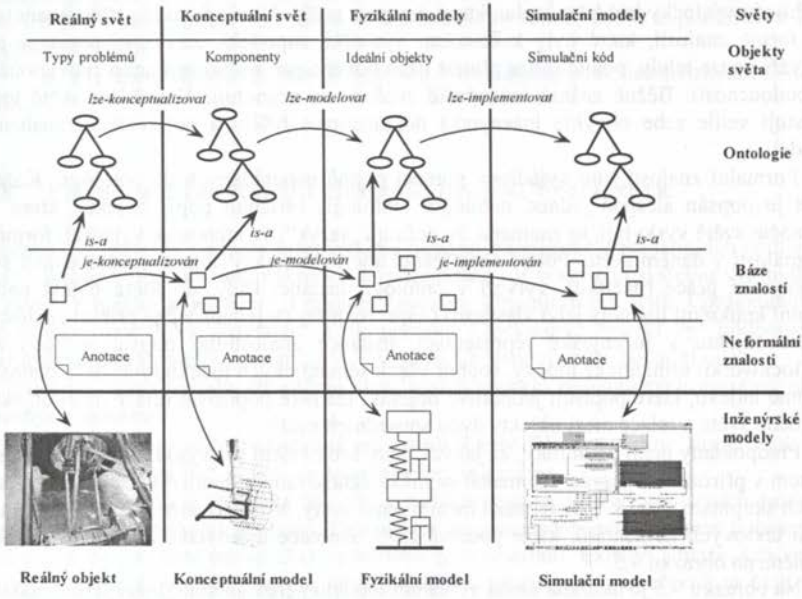
Na obrázku 9.5 je ukázána úloha vytváření modelu pérování auta. Inženýrské modely tak běžně používané v práci konstruktéra jsou ve spodní části obrázku. V horní části je formální a neformální anotace pomocí znalostních modelů. Tyto anotace vyjadřují explicitní znalosti řešitele. Jak jsme ukázali dříve, implicitní znalosti lze získat pouze prováděním činnosti, ve které jsou obsaženy. Formální znalosti a textová anotace slouží k identifikaci důležitých rozhodnutí.Tím, že jsou tyto znalosti integrovány s inženýrskými modely, umožňují rychou orientaci, poskytují instrukce jak s modely pracovat a vysvětlují, jak byly tyto modely vytvořeny. Získávání implicitních znalostí je analogické se způsobem použitým v experimentu s řízením cukrovaru.

Jakmile je úloha převedena do světa fyzikálních modelů, je už obvykle dobře strukturovaná. Je k dispozici dost znalostí, jaký simulační kód pro jednotlivé fyzikální modely použít. Přiřazení je často jednoznačné, nebo exkistuje jen několik málo variant. Výrobci simulačních programových balíků proto nabízejí grafické prostředí, ve kterém lze simulační model vytvářet. Vybráním a vzájemným propojením ikon představujících objekty fyzikálního světa se automaticky začlení k modelu odpovídající simulační kód. To je umožněno tím, že dobře strukturovaná úloha je pevně zakotvená v příslušných částech fyziky a matematiky, které poskytují algoritmy řešení.

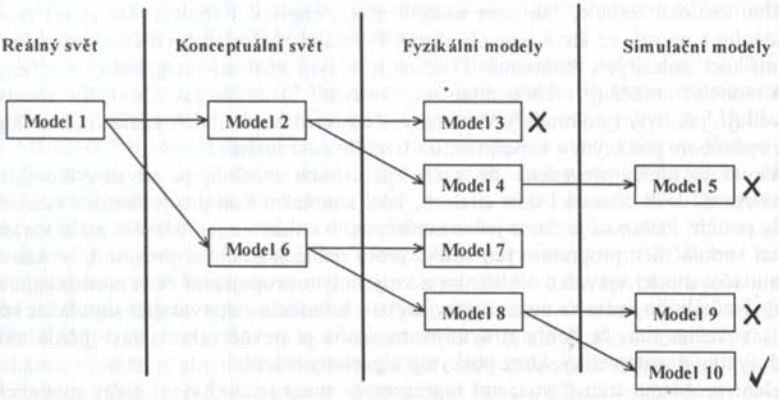
Řešení problému transformacemi reprezentace mezi jednotlivými světy metodicky řešitele vede. Současně integrování inženýrských a znalostních modelů, anotování

předpokladů zvolených transformací a možnost návratu k předchozím dílčím výsledkům poskytuje možnosti blízké metodám IBIS. Příklad stromu modelů, který vznikl řešením úlohy, je ukázán na obr. 9.6.

Na obrázku 9.6 jsou modely číslovány vzestupně tak, jak byly konstruovány. Modely 3, 5 a 9 řešitel zamítl, Model 7 řešitel zatím dále nerozpracovával a Model 10 přijal jako řešení úlohy.



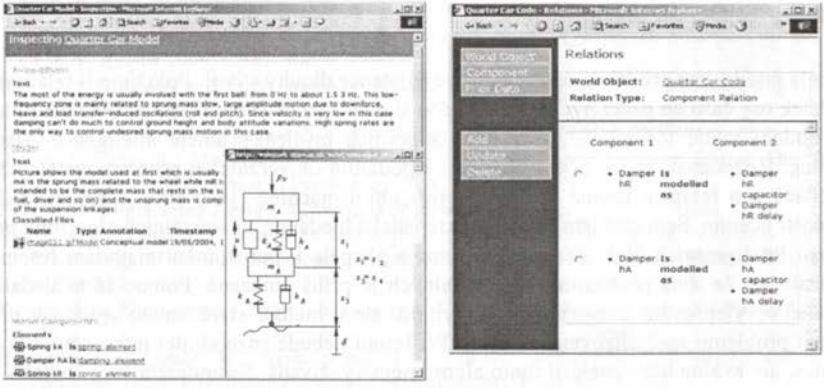

Obr. 9.5 Clockwork: inženýrské a znalostní modely.




Obr. 9.6. Příklad stromu řešení úlohy.

Příklad, jak prostředí Clockworku umožňují neformálně popsat předpoklady a formalizovat definice relací mezi sémantickými indexy, jsou ukázány na obrázku 9.7.

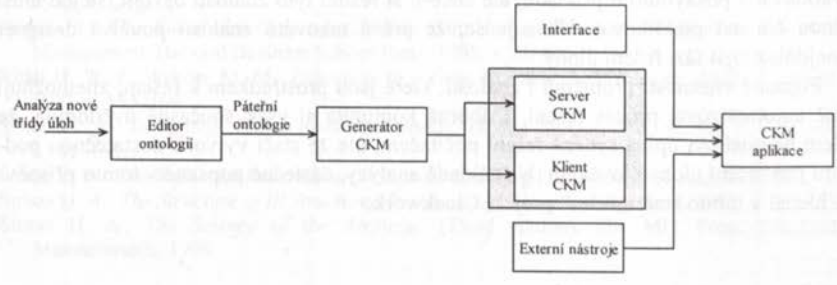
CKM obsahuje také několik nástrojů na sémantické hledání v bázi znalostí a databázi. Konstruktér tedy může vyhledat podobné vyřešené problémy, najít v nich příbuzná řešení, seznámit se s použitými inženýrskými modely, pokud mají podobu spustitelného simulačního kódu, vyzkoušet je na různých datech. V případě, že je výsledek použitelný v řešené úloze, lze vybraný model nebo jeho části zařadit do nového řešení.




Obr. 9.7. Clockwork: neformální a formální reprezentace znalostí.

Kromě podpory vývoje modelů pro modelování a simulaci dynamických systémů, byly metodologie a softwarové nástroje vyvinuté v projektu Clockwork využity při vývoji průmyslových technologií na indukční ohřev. Zde byla využita možnost rozšířit úlohu o světy popisující instalaci a servis u zákazníka. Znalosti získané anotací modelů v těchto světech slouží jako vstupní informace při vývoji nové technologie.

Zkušenosti ukázaly, že CKM, založený na organizování problémů na světy a řešení problémů jako transformací mezi světy, je vhodný nástroj pro řešení celé třídy problémů. Každou novou aplikaci lze chápat jako instanci této třídy, která má své specifické světy, v nich příslušné ontologie a metody řešení. Lze proto vytvořit ontologii která popisuje světy, jejich řazení, typ znalostí, které svět používá, možné relace uvnitř světů a mezi




Obr. 9.8 Generování CKM pro novou třídu úloh.

světy. Tuto ontologii nazýváme *páteřní ontologií* aplikace. Pokud nově připravovanou aplikaci, např. návrh tepelných technologií, popíšeme pomocí páteřní ontologie, může být programem vygenerován jak server, tak i klient, vč. nového CKM. Ke kompletaci nového nástroje je pouze zapotřebí implementovat uživatelký interface a integrovat externí nástroje používané v nové aplikační oblasti, viz obr. 9.8. Tímto způsobem byly vyvíjeny další aplikace.

## 9.9 Závěrečné poznámky

Umělá inteligence prodělala od začátku své existence dlouhý vývoj. Položíme-li relativní počátek osy času do padesátých let dvacátého století a prohlásíme-li vše, co bylo předtím, za období mýtů, formuloval jako jeden z prvních myšlenky umělé inteligence Alan Turing. Rozpoznal, že lze očekávat řešení s použitím univerzálních programovatelných počítačů pro řešení a kromě dalšího navrhl „child machine" jako stroj, který získává znalosti učením. Šedesátá léta byla charakteristická hledáním univerzálních algoritmů na řešení inteligentních úloh. Přestože významně přispěla k porozumění metodám řešení, ukázalo se, že třída problémů takto řešitelných je příliš omezená. Pomoc se nehledala u dalšího vylepšování univerzálních algoritmů, ale u metod, které budou aplikovat při řešení problému specializované znalosti. Při řešení nebude rozhodující inferenční algoritmus, ale kvalita báze znalostí tímto algoritmem využívaná. Sedmdesátá léta, charakteristická v umělé inteligenci rozvojem expertních systémů, označuje Schön jako éru „technické racionality" (Schön, 2003). Zhruba ve stejném období se však začalo ukazovat, že existuje třída problémů, které se vymykají tomuto přístupu, kromě jiného proto, že je nejsme schopni podrobně definovat, aniž bychom učinili předpoklady o tom, jak je řešit. Úlohy designu patří do této třídy. Specifikace úlohy se upřesňuje v závislosti na navrhnutém řešení. V řešení lze pokračovat až po upřesnění. Protože celý proces je založen na předpokladech jak dál postupovat a ty se definitivně přijmou (nikoliv potvrdí), až když se přijme celé řešení, je velmi důležité vědět, jaké předpoklady mohou vést k řešení. Úloha silně závisí na znalostech, které má řešitel. Koncem šedesátých let vyslovil Michael Polanyi názor, že existují znalosti, které z principu neumíme artikulovat a označil tyto znalosti jako „tacit knowledge". Tyto znalosti se nabývají vykonáváním nějaké činnosti. Jejich získávání lze usnadnit vytvořením návodu jak činnost vykonávat a poskytnutím příkladů, ale chce-li si řešitel tyto znalosti osvojit, stejně musí danou činnost praktikovat. Ukazuje se, že právě takovéto znalosti používá designer v nejdůležitější fázi řešení úlohy.

Popsané vlastnosti problémů i znalostí, které jsou prostředkem k řešení, znemožňují plně automatizovat proces řešení. Odborná komunita si však současně uvědomila, že cílem nemusí být úplná synéza řešení počítačem, ale že stačí vytvořit dostatečnou podporu pro řešení úlohy člověkem. Na základě analýzy, částečně popsané v tomto příspěvku hledal v tomto směru řešení projekt Clockwork.

## Literatura


Altshuller G. S.: *Creativity As an Exact Science*. Gordon a Breach, New York 1984.

Berry D. C., Broadbent D. E.: *On Relationship between Task Performance and Associated Verbalizable Knowledge*. The Quarterly Journal of Experimental Psychology. 36A, 1984, s. 209–231.

Buchanan R.: *Wicked Problems in Design Thinking*. Design Issues, Vol.8, No. 2, 1992, s. 5–12. Also in The Idea of Design (V. Margolin and R. Buchanan eds).The MIT Press, Cambridge Mass., Londýn 1995, s. 3–20.

Cook S. D. N., Brown J. S.: *Bridging Epistemology: The Generative Dance Between Organizational Knowledge and Organizational Knowing*. Organization Science Vol. 10, No. 4. July–August 1999, s. 381–400.

Cross N.: *Engineering Design Methods*. Second Edition, Strategies for Product Design. John Wiley a Sons, 1994.

Cross N.: *Expertise in Design: an Overview*. Design Studies 25, 2004, s. 427–441.

Dienes Z., Fahey R.: *The Role of Implicit Memory in Controlling a Dynamic System*. The Quarterly Journal of Experimental Psychology. 51A, 1998, s. 593–614.

Dym C. L.: *Engineering Design, A Synthesis of Views*. Cambridge University Press. 1994.

Džbor M., Zdrahal Z.: *Design as Interactions of Problem Framing and Problem Solving*. Proceedings of the 15th European Conference on Artificial Intelligence (ECAI-2002), 2002, s. 210–214.

French M. J.: *Engineering Design: The Conceptual Stage*. Heinemann Educational Books, Londýn 1971.

Haylighen A., Casaer M., Neukermans H.: *UnaWare: supporting tacit design knowledge exchange*. Int. J. Web Based Communities. Vol. 2., No. 1, 2006, s. 31–44.

Kunz W., Rittel H. W. J.: *Issues as Elements of Information Systems*. Working paper No. 131. Center for Planning and Development Research, University of Califomia, Berkely 1970.

Lloyd P., Lawson B., Scott P.: *Can concurrent verbalization reveal design cognition?* Design Studies 16, 1995, s. 237–259.

Louritas P., Loucopoulos P.: *A Generic Model for Reflective Design*. ACM Transactions on Software Engineering and Methodology. Vol. 9, No. 2, April 2000, s. 199–237.

MacKenzie D., Spinardi G.: *Tacit Knowledge, Weapons Design, and the Uninvention of Nuclear Weapons*. American Journal of Sociology. Vol. 101, Nr. 1, July 1995, s. 44–99.

Michie D.: *It Takes Two (At Least) to Tango*. Linköping Electronic Articles in Computer and Information Science: Vol. 5, 2000, No. 030 URL: http://www.ep.liu.se/ea/cis/ 2000/030/

Motta E., Stutt A., Zdrahal Z., O'Hara K., Shadbolt N.: *Solving VT in VITAL: a study in model construction and knowledge reuse*. Int. J. Human-Computer Studies, 44, 1996, s. 333–371.

Newell A.: *Artificial Intelligence and the Concept of Mind*. In Computer Models of Thought and Language (R. C. Schank and K. M. Colby eds). W. H.Freeman and Comp. San Francisco, 1973, s. 1–60.

Nonaka I.: *The Knowledge-Creating Company*. In Harvard Business Review on Knowledge Management. Harvard Business School Press, 1998, s. 21–45.

Rittel H. W. J., Webber M. M.: *Dilemmas in a General Theory of Planning*. Policy Sciences 4, 1973, s. 155–169.

Schön D. A.: *Designing: Rules, types and worlds*. Design Studies. Vol. 9, No. 3., July 1988, s.181–190.

Schön D. A.: *The Reflective Practitioner*. Ashgate Publishing Ltd., Aldershot, Anglie, 2003.

Simon H. A.: *The Structure of Ill Structured Problems*. Artificial Intelligence 4, 1973, s. 181–201.

Simon H. A.: *The Science of the Artificial*. (Third edition). The MIT Press, Cambridge Massachusetts, 1996.

# 10 Mechatronika – navrhování inteligentních strojů

## 10.1 Úvod

Mechatronika představuje v dnešním denním životě obor od přístrojů v domácnosti přes automobily po průmysl. Jedna z jejích zajímavých definicí je „mechatronika je navrhování inteligentních strojů". Inteligence ve smyslu definice umělé inteligence, že jisté chování u strojů neočekáváme a u člověka bychom ho označili za projev inteligence, je přítomná u všech mechatronických výrobků. Mechatronické výrobky jsou však stroje v dnešní reálné praxi, a tak není míra jejich inteligence tak velká, jak si lze pod „e-Golemem" (Mařík a kol., 2005) představovat. Jejich realita je mnohdy prostá a šedivá, kdy i neočekávaná přítomnost zpětnovazebního řízení je označována za inteligenci (např. inteligentní materiály, chytré struktury, mechatronické komponenty apod.). Mechatronika však představuje inteligentní stroje v dnešní praxi. Příkladem budiž automatická pračka, která si střídavými pohyby nejdříve urovná rozložení prádla v bubnu, aby potom „neskákala". Je to projev (umělé) inteligence???

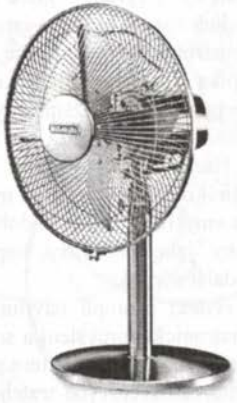
## 10.2 Co je mechatronika

Mechatronika (Valášek, 1996), (Buur, 1989), (Buur, 1990), (Isii, 1985), (Miu, 1993), (Bradley a kol., 1991) je interdisciplinární obor, který se zabývá spojením inteligentního počítačového řízení a rozhodování prostřednictvím elektroniky se strojí a mechanickými systémy. Velmi vhodná definice je, že mechatronika je synergistická integrace strojů s elektronikou a inteligentním počítačovým řízením při návrhu a výrobě výrobků a procesů. Zde je kladen důraz na synergii jako pojem zdůrazňující, že mechatronika přináší více než jen pouhý součet částí. Návrhová integrace vede k produktu, který zlepšuje předchozí úrovně výkonu více než jen zlepšením, ale přidáním nové dimenze funkčnosti. V definicích pro 21. století (Tomizuka, 2000), (Isermann, 2005) se zobecňují stroje a mechanické systémy na jakékoli fyzikální systémy, elektronika na informační technologie a řízení na složité rozhodovací procesy. Mechatronika se tak stává základem všech nových strojů, všech nových výrobků.

S mechatronikou se všichni setkáváme na každém kroku. Je obsažena v naprosté většině všech výrobků. Nejmechatroničtějším výrobkem je dnes zřejmě automobil (Isermann, 2005). Mnoho tradičních soustav je nahrazeno mechatronickým řešením na bázi mechanické funkčnosti spojené s elektronikou a mikroprocesorovým řízením, které přináší nové vlastnosti a nové nečekané funkčnosti. Lidská fantazie je zhmotněna. Místo karburátoru je řízené vstřikování, které výrazně snižuje spotřebu paliva a exhalací. Místo

tradiční hydraulické brzdové soustavy je systém antiblokovacího brzdového systému ABS nebo dokonce elektronického stabilizačního systému ESP, který zabrání smyku a neovladatelnosti při brzdění, stabilitu při zatáčení přivede na mez fyzikálních zákonů. Mechatronika se projevuje ale i v komunikaci palubního počítače s řidičem, kde je poskytnuta samodiagnostika, nová funkce inteligentního chování. Mechatronika se projevuje v uzamykání vozidla, kde je zajišťována bezpečnost posádky jak před vloupáním, přepadením, tak při vyprošťování při havárii.

S mechatronikou se setkáváme stále častěji. V domácnosti je to popisovaná automatická pračka, ale i jen komunikace sporáku a trouby s uživatelem. V našem běžném životě jsou to digitální hodinky, jejich verze pro fyzický trénink amatérských i profesionálních sportovců a další. Daleko masivnějšímu projevu mechatroniky v našem životě brání jen ekonomické důvody, meze fyzikálních zákonů, jakoby v softwarovém řešení funkčnosti mechatronických výrobků neexistovaly.



Obr. 10.1 Stolní ventilátor.

Jeden z nejjednodušších příkladů převratu mechatronickým přístupem je stolní ventilátor (Valášek a kol., 1996). Vybavíme-li obyčejný stolní ventilátor (obr. 10.1) čidlem teploty okolí, čidlem citlivým na přítomnost osob v místnosti a mikroprocesorovým řízením, dostaneme zařízení s adaptivitou, inteligencí a schopností úspory energie. Od teploty okolí totiž můžeme odvodit intenzitu jeho otáček, od čidla citlivého na přítomnost osob v místnosti jeho vypínání při prázdné místnosti, a tak jeho adaptivní přizpůsobení teplotě místnosti s „inteligentním chováním" a s úsporou energie.

## 10.3 Vznik a historie mechatroniky

Mechatronika je slovo, které vzniklo v Japonsku v polovině 70. let a v únoru 1976 se objevilo na titulní straně časopisu zabývajícího se stavem japonského průmyslu.

Dosavadní historii mechatroniky můžeme přibližně rozdělit na čtyři období. Vývoj mechatroniky je úzce spojen s vývojem NC obráběcích strojů. Problém velikosti a spo-

lehlivosti řídicího systému vedl posléze k použití mikroprocesorů, jejichž první skutečně velmi úspěšné řešení na bázi 8bitového procesoru Intel 8080 bylo realizováno v roce 1974. Velikost a cena řídicího systému NC stroje dramaticky poklesla, spolehlivost a množství funkcí, které mohl plnit, prudce vzrostla. Bylo možné vestavět řídicí systém přímo do stroje. Když se vestavěný typ řídicího systému stal běžným, tvořila se nová slova jako elektromechanické stroje nebo mechanoelektronické stroje pro označení strojů s vestavěným malým řídicím systémem. Tato slova neznějí dobře, a tak teprve slovo mechatronika se ujalo pro označení stroje s řízeným mechanismem s elektronickým obvodem. Mechatronický návrh snižuje váhu a cenu výrobku, zvyšuje jeho spolehlivost a navíc i jeho funkčnost. Často jsou totiž užitečné funkce nerealizovatelné pouze mechanickými prvky doplněny elektronikou při mechatronickém návrhu. Vestavěný mikroprocesor má většinou přebytek výkonu, který lze právě využít pro další funkce realizovatelné podle vyměnitelného programu. Proto se návrhový koncept mechatroniky rychle rozšířil. Dalšími mechatronickými výrobky, které se především rozšířily po celém světě, byly v Japonsku roboty. Jejich rozvoj byl také umožněn použitím mikroprocesorů jako jejich řídicích systémů. Mechatronika je v Japonsku často chápána jako synonymum robotiky. Tato první mechatronika 70. let vzniklá kombinací slov MECHAnismus a elekTRONIKA je nazývána mechatronikou v užším smyslu.

Koncept mechatroniky se v 80. létech rozšířil a byl úspěšně použit při návrhu řady výrobků, nejen pro elektronicky řízené mechanismy NC strojů a robotů. Pojem mechatroniky se zobecnil a označoval širší oblast na rozhraní mezi mechanikou a elektronikou. Mluvíme o mechatronice v širším smyslu. V tomto období zvláště japonské firmy uvedly na trh klasické high-tech výrobky, jako jsou faxy, kopírky, automatické fotoaparáty, videokamery, videopřehrávače a další výrobky.

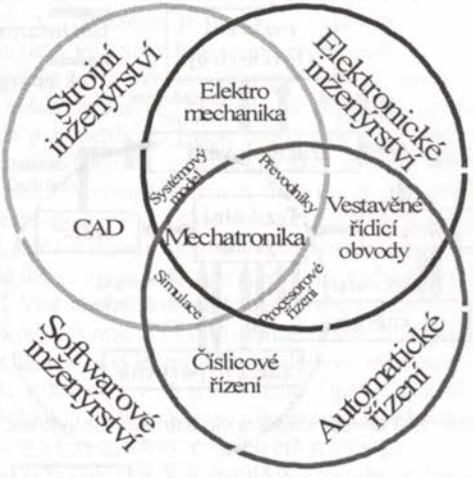
Koncem 80. let dochází k reflexi postupů návrhu mechatronických výrobků do prvních obecných závěrů o mechatronickém myšlení a současně dochází k jeho aplikaci na nejširší technické systémy. Japonské výrobky, které můžeme zahrnout pod aplikace mechatroniky, byly tak úspěšné na celosvětových trzích, že vlády v Evropě a USA se začaly mechatronikou vážně zabývat. Evropské univerzity zavedly výuku mechatroniky, vznikly výzkumné programy financované vládami a firmami, začaly vycházet specializované časopisy a pořádat se konference. V tomto třetím období se mechatronika zobecňuje na metodu návrhu výrobků, který kombinuje použití více technologií (obr. 10.2).

Konečně na prahu 21. století se nejen mechatronice přisuzuje jedno z klíčových míst dalšího rozvoje inženýrství, ale i obsah mechatroniky se zobecňuje. Místo jen o mechanických systémech se uvažuje o jakýchkoli fyzikálních systémech, elektronika je chápána jako informační technologie a řízení jako složité rozhodovací procesy. Tím je konkretizována představa kombinace různých technologií v jednom výrobku.

## 10.4 Současná mechatronika

Mechatronické výrobky se postupně prosazují ve všech výrobcích. Běžně jsou to komponenty strojů, vlastní stroje a zařízení, ale již celé systémy, jako je navigace, doprava, zásobování, armáda. Vedle strojů proniká mechatronika do budov a dalších oblastí.

Obecná struktura mechatronického systému je znázorněna na obr. 10.3. Ukazuje spojení jednotlivých technologií – komponent z obr. 10.2. Obsahuje mechanicko-elektricko-optickou komponentu ve spojení pohony–fyzikální systém–čidla, elektronickou komponentu v pohonech a čidlech, řídicí komponentu v bloku rozhodování a počítačově-informační komponentu v bloku rozhodování a rozhraní člověk–stroj. Každý, kdo se seznámí s principy a příklady mechatronických výrobků, je vystaven pokušení zjednodušujícího pohledu na mechatronické aplikace. Na první pohled se zdá, že vše je náhle prostě jednoduché. Stačí vzít jakýkoli fyzikální systém (obr. 10.4), doplnit ho čidly, pohony a řízením a můžeme získat báječné aplikace, kdy lze vše řídit, zlepšit, učinit optimálním, vše ovlivnit, vše vytvořit. To je prvotní mechatronický entusiasmus. Záhy se však ukáže, že vytvoření mechatronické aplikace není zdaleka tak jednoduché.

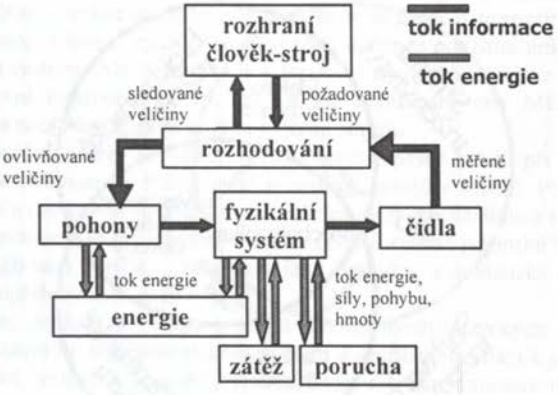



Obr. 10.2 Mechatronika jako kombinace technologií.

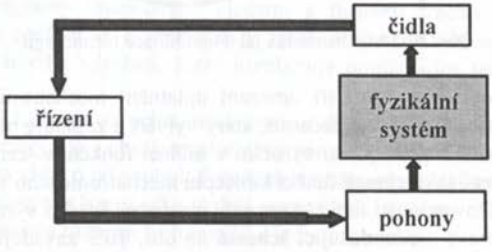
Z obecného pohledu existují tři omezení uplatnění mechatroniky. Je to koncept celého výrobku z hlediska jeho funkčnosti, který vytváří a zaplňuje bílé místo na trhu, je to ekonomické hledisko prodejnosti výrobku v bilanci funkčnost–cena–poptávka–kupní síla a je to otázka realizovatelnosti funkcí konceptu mechatronického výrobku.

Z hlediska realizovatelnosti funkcí konceptu mechatronického výrobku a ekonomické relace na trhu je právě zjednodušující schéma na obr. 10.3 zavádějící. Ukazuje se, že většina mechatronických aplikací narazí na přerušení smyčky znázorněné na obrázcích 10.4 a 10.3 tím, že něco schází nebo není realizovatelné. Tok energie nebo informace je přerušen. Chybějí čidla nebo pohony, nebo scházejí jiné vlastnosti fyzikálního systému. Tento nedostatek je nutné nejdříve odstranit úpravou fyzikálního systému. To je také základní rozdíl mezi dřívějšími elektromechanickými systémy a dnešními mechatronickými systémy. Většina mechatronických aplikací je založena na patentu, který obvykle odstraňuje tento nedostatek a chrání úpravu fyzikálního systému.

Konkrétních příkladů je celá řada. Chybějící čidla je třeba buď vyvinout (např. kvadrantovou diodu pro měření zaostření obrazu), nebo nahradit jinými (např. protiblokovacím brzdovým systémem ABS vozidla, které nahrazuje čidlo skluzu kola po vozovce čidlem relativního otáčení kola vůči podvozku vozidla) nebo se bez nich obejít (příkladem je čidlo polohy objektu v prostoru u dále popsaného nekývajícího jeřábu). Chybějící pohony je nutné buď vyvinout (např. tlumiče s proměnnou tlumicí silou pro řízené pérování vozidel) nebo zbytek systému upravit tak, aby existující pohony bylo možné použít (např. přídavnou hmotou nebo přídavnou strukturou pro setrvačný nebo poddajný opěrný bod k vyvinutí síly podle zákona akce a reakce). Konečně celý fyzický systém je třeba upravit pro použití mechatronického schématu na obr. 10.4. Příkladem jsou paralelní kinematické struktury, které je třeba doplnit nadbytečnými rameny s redundantními pohony, aby mohly být odstraněny singulární polohy mechanické struktury.




Obr. 10.3 Obecná struktura mechatronického systému.




Obr. 10.4 Zjednodušující schéma mechatronického systému.

Nejmarkantnějším příkladem jsou vojenská letadla s technologií fly-by-wire. Takové letadlo je mechanicky upraveno tak, že bez zpětnovazebního řízení je nestabilní. Zpětnovazební řízení letadlo stabilizuje a poskytuje mu požadované vlastnosti. Úprava letadla do pasivně mechanicky nestabilního systému totiž poskytuje letadlu podstatně větší dynamické vlastnosti a tedy lepší manévrovatelnost. Pasivně stabilní mechanický systém

musí mít značnou setrvačnost, aby její kapacita vyrovnávala poruchy. Velká setrvačnost (kapacita) brání rychlé změně stavu. Je-li setrvačnost (kapacita) malá, může být změna stavu systému rychlá. Nejrychlejší změnu dosahuje nestabilní systém.

Meze uplatnění mechatroniky jsou dány omezeními ekonomickými, nikoliv technologickými. Důležitým rysem také je, že mechatronický návrh se vztahuje k dostupným výrobním technologiím než k realizovatelnosti podle mezí fyzikálních zákonů. Velmi důležitým rysem mechatroniky je její orientace na výrobek. Mění se role inženýra-návrháře a postoj celého průmyslu ve společnosti. Návrh celého výrobku je podstatně více vázán na počáteční průzkum trhu, výběr koncepce budoucího výrobku a volba jeho vlastností se odvíjí od nalezení volného místa na trhu. Výrobek má vedle vlastností nutných pro daný výrobek vlastnosti, které dělají jeho úspěch (snadnost komunikace s uživatelem, samodiagnostiku, nepředpokládané vlastnosti). Mnohé úspěšné mechatronické výrobky vznikly tak, že si samy vytvořily svoji potřebu a svůj trh. Vytvořily ve společnosti „životní styl". Příkladem je walkman. Jejich výrobci provádějí nejen průzkum trhu, ale sociologický výzkum a formování společnosti.

Elektronika ovlivňuje systémovou koncepci stroje nejen tím, že nahrazuje mechanické funkce, avšak především, že doplňuje inteligenci. Inteligencí se zde chápe schopnost použít řadu parametrů pro řízení funkcí. Jde o možnost programovatelnosti, komunikace, samoregulovatelnosti, samodiagnostiky, samoopravitelnosti, adaptivnosti, učení a samoorganizace. Připomeňme jen u tiskárny změnu fontů písma, u elektronického šicího stroje programování stehu, u kopírky komunikaci o počtu kopií a typu formátu, diagnostiku, kde uvízl papír, automatické nastavení kontrastu atd.

Z uvedeného výkladu je zřejmé, že tvořivost hraje klíčovou roli při návrhu mechatronického výrobku. Většina mechatronických výrobků vznikla při kreativním nebo inovativním návrhu a jejich vznik byl závislý na tvořivosti návrháře. Tvořivost je jedním z klíčových rysů inteligence. Zvýšenou míru inteligence mechatronika nepřináší jen ve vlastním výrobku, ale vyžaduje ji i při jeho návrhu. Opět se ukazuje, že přesně nevíme, co je inteligence, natož jak ji realizovat nebo alespoň uměle nahradit.

Hlavními odlišujícími rysy mechatronického inženýrství je

- interdisciplinarita vyžadující znalost výrobků a výroby z širokého výběru technologií,
- systémové myšlení dávající unifikovaný postup ke kombinaci nepravděpodobných technologií pro vytvoření optimálního výrobku,
- týmová práce nutná při spolupráci inženýrů různých oborů,
- tvořivost, nutná odvaha k návrhu a experimentování s dříve neznámými kombinacemi technologií,
- obchodní hledisko nezbytné pro vyhodnocení konceptu v konkurenčním prostředí trhu.

## 10.5 Inteligence v mechatronice

Z pohledu na historii mechatroniky se zdá, že právě inteligence je nejdůležitějším důsledkem technologických změn vedoucích k mechatronice. Jedna pozoruhodná definice mechatroniky lakonicky konstatuje, že mechatronika je navrhování inteligent-

ních strojů. Tato definice klade důraz na velmi typickou vlastnost mechatronických výrobků – jistou míru jejich inteligence. Inteligence se v mechatronice chápe jako programovatelnost, samoregulovatelnost, adaptivnost, samodiagnostika, přívětivá komunikace s uživatelem atd. Na první pohled se zdá, že je to odlišné pojetí inteligence od jejího chápání v umělé inteligenci. Projevy této inteligence v mechatronických výrobcích můžeme rozdělit tak, jak uvádíme v následujících odstavcích.

### 10.5.1 Komunikační inteligence

Mechatronický výrobek vede s uživatelem komunikaci o svém stavu, svých vlastnostech a nabízených funkcích. Mechatronický výrobek se vyskytuje v mrtvém a živém stavu. Aby mohl fungovat podle vytvořeného konceptu, musí být nejdříve převeden z mrtvého stavu do živého stavu. Jeho funkčnost musí být aktivována. Potom mechatronický výrobek komunikuje s uživatelem o svém stavu. Přirozeně je doplněn řadou čidel pro diagnostiku a posuzování svého stavu. Na tomto základě komunikuje s uživatelem.

Příkladem je kopírka. Po zapnutí nejdříve provede diagnostiku svého stavu a na jeho základě řídí přechod do aktivního stavu. Pokud není papír v zásobníku, někde po cestě v kopírce je papír zadržen, vyvolávací zařízení není zahřáto na pracovní teplotu, některá čidla nepracují správně, je uživatel informován a kopírka není převedena do stavu připravena, nelze kopírovat. V takovém případě je uživateli často poskytnut návod dalšího postupu pro odstranění závady.

Na této bázi lze vytvářet celou řadu konceptů mechatronických výrobků. Příkladem může být třeba nábytek, řekněme židle. Židli můžeme vybavit čidly, pohony, řízením, inteligencí. Je možné vytvářet různé scénáře. Člověk k židli přijde, ta se sama odsune od stolu, aby si do ní člověk mohl posadit, po té se s ním ke stolu přisune, nastaví výšku, reguluje svoji teplotu, poddajnost, vibrace, s člověkem hlasově komunikuje atd. Není technologický problém vyvinout a vyrobit mechatronický (robotický) nábytek. Problémem však je takový výrobek uvést na trh, tj. správně nastavit jeho užitné vlastnosti a jeho cenu.

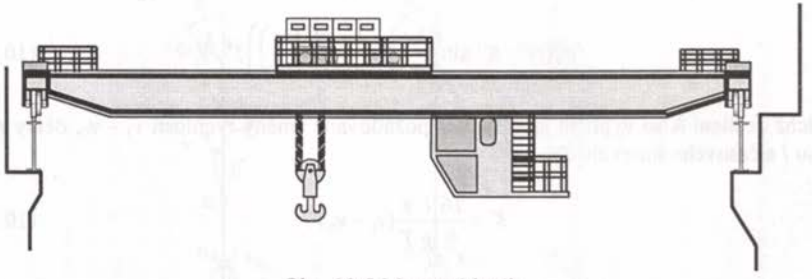
Zde je zajímavé srovnání s tradičními definicemi umělé inteligence. Turingova definice staví na iluzi komunikace s člověkem. Mechatronické výrobky této iluze sice nedosahují, ale důraz na komunikaci s uživatelem na základě autonomnosti dosažené informacemi z čidel a rozlišování svých různých stavů je zřetelný. Druhá definice, Kotkova, stavějící na vytváření vnitřního modelu okolního světa, má u mechatronických výrobků v jednoduché podobě plnou realizaci. Odlišení stavů mrtvý–živý, funkční–nefunkční a mnoha dalších je zřetelné vytváření vnitřního modelu.

### 10.5.2 Funkční inteligence

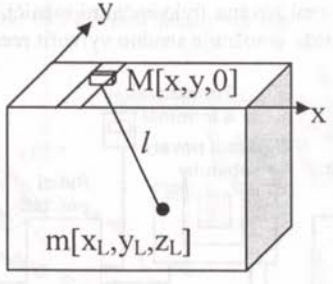
Daleko méně zřetelným a diskutabilnějším rysem inteligence lze mechatronickým výrobkům přisoudit z proklamovaného inteligentního počítačového řízení. Protože jde většinou o vymezení funkcí mechatronického výrobku, lze mluvit o funkční inteligenci. Málokdo by rys automatického řízení spojoval s umělou inteligencí. Přesto u mechatronických výrobků vzhledem k jejich předchozím funkčním vymezením není takový projev očekáván, a tak dochází k naplňování další tradiční Minského definice umělé inteligence, kdy realizace jistých funkcí člověkem by byla označena jako projev jeho inteligence.

Velmi jednoduchým příkladem může být stolní ventilátor. Jestliže stolní ventilátor opatříme čidly teploty okolí, přítomnosti člověka v místnosti a mikroprocesorovým řízením otáček ventilátoru, dostaneme zařízení s adaptivitou, inteligencí a schopností úspory energie a zlepšení pohody prostředí. Od teploty okolí odvodíme intenzitu jeho otáček, od čidla přítomnosti člověka v místnosti vypínání při prázdné místnosti, a tak jeho adaptivní přizpůsobení stavu okolí vytváří inteligentní zařízení. Uživatel u celkem známého jednoduchého zařízení je překvapen vyšší funkčností jeho chování, kterou označí jako projev inteligence.

Jiným příkladem je nekývající jeřáb (Valasek a kol., 1996). Jeřábníci jsou zvyklí na problém reziduálního kývání jeřábu a nutnosti jejího předcházení modifikací vstupních příkazů. Řídicí systém nekývajícího jeřábu modifikuje jakýkoli vstupní příkaz tak, aby k reziduálnímu kývání jeřábu nedošlo. Tento projev jeřábu oproti očekávané vlastnosti reziduálního kývání zřetelně vytváří vjem umělé inteligence.



Obr. 10.5 Mostový jeřáb.




Obr. 10.6 Model mostového jeřábu.

Za funkčností nekývajícího jeřábu není nic složitého, i když jde o netradiční použití automatického řízení. Pro tradiční aplikaci automatického řízení jako zpětnovazebního systému by bylo třeba čidlo polohy manipulovaného objektu v prostoru. Takové čidlo je buď omezující či nerobustní, nebo je drahé. Čidlo lze však nahradit dopředným řízením na základě časově proměnného modelu jeřábu měřením délky lana závěsu. Dopředné řízení je tzv. řízení tvarováním vstupu (input shaping control).

Mostový jeřáb (obr. 10.5) lze modelovat jako matematické kyvadlo s pohybujícím se závěsem (obr. 10.6). Jeho dynamický model ve směru osy $x$ lze upravit do tvaru

$$\ddot{x} = \ddot{\varphi}_x + \frac{g}{l}\varphi_x, \tag{10.1}$$

kde $\varphi_x = x - x_L$ je rozdíl mezi polohou jeřábové kočky a břemene, $g$ značí gravitační zrychlení a $l$ udává délku závěsu (lana). Pro stabilní pohyb bez reziduálního kývání platí

$$\begin{aligned} &x(0) = 0,\ \dot{x}(0) = v_0,\ \varphi_x(0) = 0,\ \dot{\varphi}_x(0) = 0, \\ &x(T) = d,\ \dot{x}(T) = v_1,\ \varphi_x(T) = 0,\ \dot{\varphi}_x(T) = 0, \end{aligned} \tag{10.2}$$

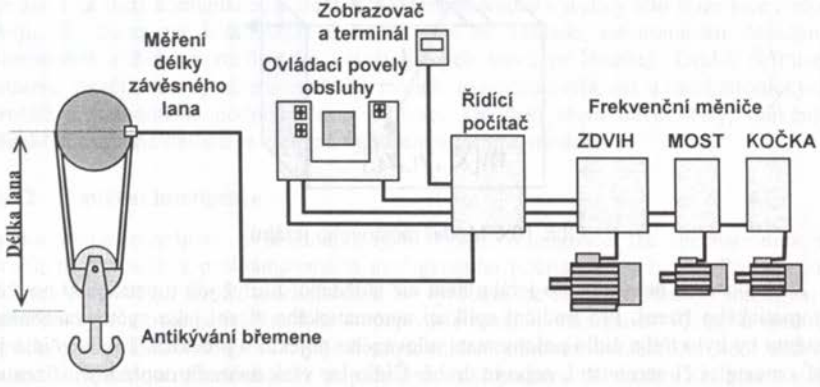
mezi výchozí $v_0$ a koncovou požadovanou rychlostí $v_1$ za čas $T$ na délce $d$ je možné nalézt třídu funkcí

$$\varphi_x(t) = K\left(\sin\left(\frac{\pi t}{T}\right) - \frac{1}{3}\sin\left(\frac{3\pi t}{T}\right)\right), \tag{10.3}$$

jejichž zesílení $K$ se vypočte jako funkce požadované změny rychlosti $v_1 - v_0$, délky závěsu $l$ a časového intervalu $T$:

$$K = \frac{16}{9}\frac{l}{g}\frac{\pi}{T}(v_1 - v_0). \tag{10.4}$$

Délka časového intervalu $T$ pro realizaci pohybu je funkcí přípustného zrychlení a maximální rychlosti. Ty vyplývají z realizovatelného přenosu hnací síly třením mezi koly a kolejnicemi pojezdu jeřábu. Z rovnice (10.1) pak vypočteme rychlostní profil pojezdu jeřábové kočky, která je realizována frekvenčními měniči. Zde je užitečné použít profil pohybu podle (10.3), protože umožňuje snadno vytvořit reentrantní řízení.

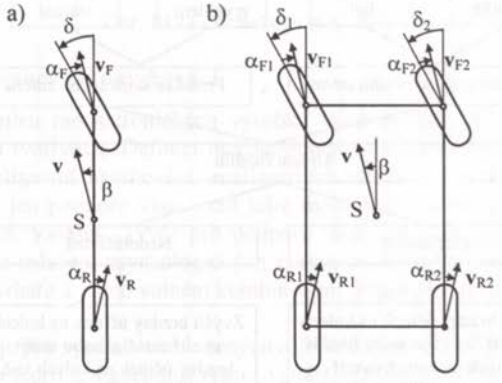



Obr. 10.7 Schéma řízení nekývajícího jeřábu.

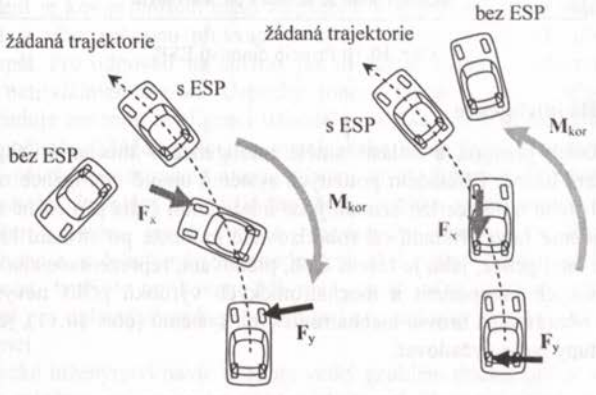
Jeřábník ovládá jeřáb tlačítky jako dříve. Jen mezi jeho ovládání a frekvenční měniče pohonů jeřábu je vložen mikroprocesor (obr. 10.7), který převádí požadavky jeřábníka do třídy funkcí (10.3). Tradiční stroj tak nabývá nových vlastností, které lze ve srovnání s původním chováním označit za inteligentní.

Jiným obdobným příkladem je stabilizace automobilu při zatáčení pomocí přibrzďování jednotlivých kol (na trhu známo pod zkratkami ESP, DSP aj.) (Isermann, 2006). Takové řízení, kdy v zatáčce jsou jednotlivá vnitřní nebo vnější kola přibrzďována, by člověk vůbec nedokázal. Pro označení inteligentní je však důležitý spíše rozdíl mezi očekávaným tradičním chováním a ovládáním automobilu a jeho novými vlastnostmi, které člověk neočekává a jimž pak přisuzuje jistou míru inteligence.

Na tomto systému je zajímavé, že řídicí systém pracuje se dvěma modely. Jeden model (tzv. jednostopý rovinný model uvedený na obr. 10.8(a)), je užit pro predikci




Obr. 10.8 Rovinné modely vozidla.




Obr. 10.9 Stabilizace stáčení vozidla korekčním momentem $\mathbf{M}_{kor}$ při příliš velké boční síle $\mathbf{F}_y$ pomocí brzdné síly $\mathbf{F}_x$.

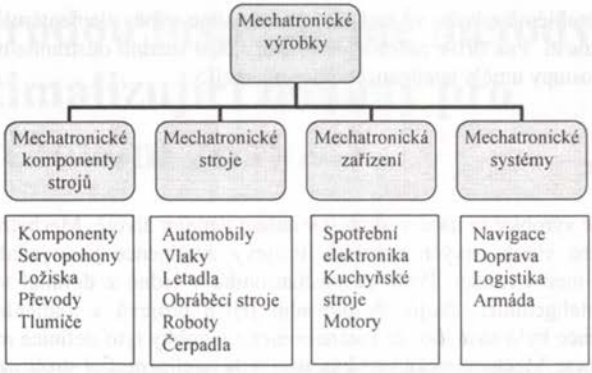
požadované a skutečné polohy vozidla (obr. 10.9). Druhý model (tzv. dvoustopý rovinný model znázorněný na obr. 10.8(b)) je užit pro výpočet intervence řídicího systému ve vozidle. Zjednodušená logika činnosti systému ESP je na obr. 10.10.

Z uvedeného výkladu vyplývá odpověď na otázku v úvodu. Automatická pračka je inteligentní stroj, protože opět tradiční pračka nabývá nových vlastností, které lze ve srovnání s původním chováním označit za inteligentní. Oporu pro tento závěr bychom nalezli u několika definic umělé inteligence, i když to zní zvláštně.

Obr. 10.10 Princip činnosti ESP.

### 10.5.3 Umělá inteligence

Použití tradičních postupů a oblastí umělé inteligence v mechatronických výrobcích překvapivě není běžné. Příkladem použitých systémů umělé inteligence může být počítačové vidění. Jeho aplikace lze označit jako inteligentní čidla pro různé mechatronické výrobky. Najdeme řadu příkladů od robotizované montáže po střihání krajek. Ale jiné oblasti umělé inteligence, jako je řešení úloh, plánování, reprezentace znalostí a další, se u mechatronických komponent a mechatronických výrobků příliš nevyskytují. Jejich uplatnění lze očekávat na úrovni mechatronických systémů (obr. 10.11), jejichž složitost si takové postupy bude vyžadovat.




Obr. 10.11 Dnešní mechatronické výrobky

### 10.5.4 Inteligence při navrhování

Při popisu vzniku mechatronických výrobků jsem ukázali, že jejich návrh vyžaduje zvýšenou míru tvořivosti. Definici mechatroniky z úvodu můžeme rozšířit na „Mechatronika je inteligentní navrhování inteligentních strojů." O podstatě tvořivosti a jejím modelování či jen podpoře víme snad ještě méně než o inteligenci. Existuje sice řada postupů (Beneš, Valášek, 1996) pro podporu tvořivosti, ale jejich naprostá většina je založena jen na oslabení psychologických zábran uvolnění již existujících vnitřních idejí a konceptů návrháře a jejich volném kombinování. Příkladem je brainstorming, synektika apod.

Výjimkou je jen vynálezecké inženýrství (Altshuller, 1984), (Beneš, Valášek, 2007), které vytvořilo teorii a algoritmus řešení vynálezeckých úloh. Jeho základní myšlenkou je, že podstatou řešení technického problému je nalezení řešení rozporu. Rozpor zjednodušeně znamená, že zlepšením jedné vlastnosti systému se jiná vlastnost zhorší a naopak. Tradiční řešení je kompromisem mezi zlepšením a zhoršením obou vlastností. Vynálezecké řešení je však schopno překvapivě nalézt takové řešení, kdy obě vlastnosti se současně zlepší. Pro odpověď na otázku, jak to udělat, přináší vynálezecké inženýrství celou řadu netriviálních metod. Úspěchy tohoto postupu jsou značné, přesto jeho aplikace vyžaduje značnou inteligenci uživatele a v žádném případě nejde o algoritmus pro dnešní počítače. Můžeme říci, že vynálezecké inženýrství poskytuje připraveným a kreativním návrhářům asociace a analogie možných řešení k nalézaným problémům stávajícího návrhu. Problém začíná již tím, že na začátku kreativního návrhu neexistuje žádný navržený systém, natož dvojice rozporných vlastností. Klíčová fáze tvoření prvotního návrhu z „ničeho" tedy není podpořena. A i v následných fázích je nutné dílčí kroky postupu tvořivě aplikovat. Návrhář musí být vybaven značnou dávkou představivosti pro uplatnění analogií, tvořivostí pro nalézání spojovacích bodů asociací, tedy inteligencí.

Vynálezecké inženýrství navíc ukazuje velký problém dnešní umělé inteligence. Její metody jsou založeny na apriorním výběru vlastností, které považujeme za elementární ať již pro měření, nebo pro logické úvahy. Vynálezecké inženýrství však ukazuje, že vel-

ké množství problémů návrhu se řeší tím, že změníme výběr elementárních, dále nedělitelných vlastností. Pak dříve neřešitelné rozpory jsou snadno odstranitelné. Něco takového dnešní postupy umělé inteligence vůbec neumějí.

## 10.6 Závěr

Mechatronické výrobky se dnes vyskytují v našem denním životě. Mechatronika se stává atributem skoro všech nových výrobků. Projevy inteligence jsou jedním z nejdůležitějších rysů mechatroniky. Proto se mechatronika v jedné z definicí vymezuje jako navrhování inteligentních strojů. Porovnáním jejích projevů s tradičními definicemi umělé inteligence bylo ukázáno, že mechatronické výrobky tyto definice mohou alespoň částečně splňovat. Mechatronické výrobky jsou tedy inteligentními stroji v dnešní praxi.

Přitom jejich inteligence je často založena na relativně jednoduché aplikaci počítačového řízení. Jeho atributem však je vjem uživatele, že výrobek se chová způsobem, kterému by přisoudil jistou míru inteligence. Mnohdy jde o právě o rozdíl mezi chováním tradičního výrobku, na jehož ovládání byl uživatel zvyklý, a chováním nového výrobku, který je počítačovým řízením vybaven. Atribut strojové inteligence tak do dnešních výrobků proniká po malých krůčcích, ale vytrvale.

### Literatura


Altshuller G. S.: *Creativity As an Exact Science*. Gordon & Breach, New York 1984.

Beneš P., Valášek M.: *Metody tvůrčí práce*. Vydavatelství ČVUT, Praha 1996.

Beneš P., Valášek M.: *Tvůrčí tým*. Ben, Praha 2007 (v tisku).

Bradley D. A., Dawson D., Burd N. C., Loader A. J.: *Mechatronics*. Chapman and Hall, Londýn 1991.

Buur J.: *A Theoretical Approach to Mechatronics Design*. Institute for Engineering Design, Technical University of Denmark, Lyngby 1990.

Buur J.: *Mechatronics Design in Japan*. Institute for Engineering Design, Technical University of Denmark, Lyngby 1989.

Isermann R. (ed.): *Fahrdynamik-Regelung*. Vieweg, Wiesbaden 2006.

Isermann R.: *Mechatronic Systems – Innovative Products with Embedded Control*. In: Horacek P. Simandl M., Zitek P. (eds): Proc. of 16th IFAC World Congress, Selected Plenaries, Milestones and Surveys, Praha 2005, s. 52–68.

Isii T., Simojama I., Inoue H., Hirose M. and Nakadzima N.: *Mechatronika*. Iwanami Shoten Publishers, Tokio 1985.

Mařík, V. a kol. (eds): *Interdisciplinary Aspects of Human-Machine Co-existence and Co-operation*. Proc. of Czech-Argentine Biennale Workshop „e-Golems", CTU, Praha 2005.

Miu D. K.: *Mechatronics*. Springer Verlag, New York 1993.

Tomizuka M.: *Mechatronics: From the 20th to 21st Century*. In: Isermann R. (ed.): Proc. of 1st IFAC Conference on Mechatronic Systems, Darmstadt 2000, s. 1–10.

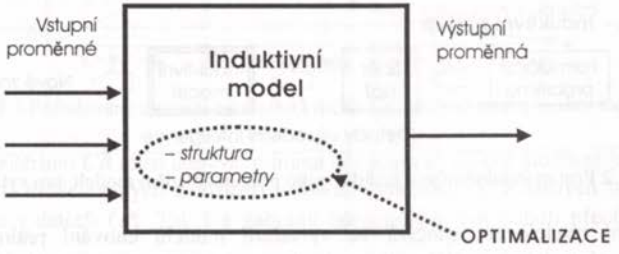
Valasek M. et al.: *Position and Velocity Control of Gantry Crane*. In: Proc of Mechatronics 96, Guimaraes 1996, s. I-203–I-208.

Valášek M. a kol.: *Mechatronika*. Vydavatelství ČVUT, Praha 1996.

# 11 Přírodou inspirované metody optimalizující modely pro vytěžování znalostí

## 11.1 Úvod

Motivací pro metody, které dále v této kapitole popíšeme, je hledání nejlepších řešení metodami inspirovanými přírodou. Nejlepší řešení (v našem případě) závisí na „nějakých" spojitých parametrech a právě jejich optimální hodnoty hledáme. Konkrétně jde o optimalizaci parametrů automaticky generovaných modelů reálných systémů, které používáme k vytěžování (Data Mining). Vytěžováním získáváme o původních systémech nové znalosti. Vedle optimalizace spojitých parametrů modelů popíšeme také optimalizaci struktury těchto modelů.




Obr. 11.1 Optimalizace struktury a parametrů induktivních modelů.

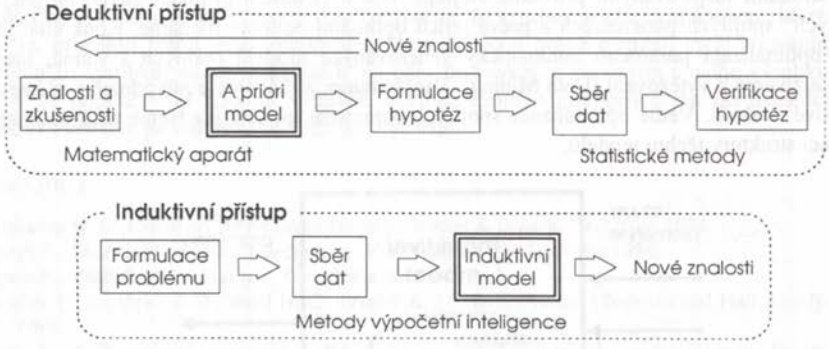
Naše modely vytváříme samoorganizací z funkčních bloků (jednotek nebo neuronů) pomocí evolučních technik. Samoorganizací generované modely mění svou strukturu, tedy počet vrstev a počet jednotek v nich, a dokonce i typ a strukturu přenosových funkcí, nejen váhy jednotlivých neuronů, jak je tomu u základních neuronových sítí. Procesem optimalizace se učí (s učitelem) fungovat tak, aby co nejlépe napodobovaly chování původních systémů.

Problémů je v tomto případě celá řada. Asi hlavní spočívá v tom, že při návrhu modelů je mnoho stupňů volnosti (mohou být použity mnohodimenzionální vstupy, různé typy jednotek a jednotky mohou být různě pospojovány). Klasické metody optimalizace nejsou přímo použitelné. Námi používané a v tomto článku popisované metody patří do arzenálu vědecké disciplíny „výpočetní inteligence" (Computational Intelligence).

## 11.2 Deduktivní a induktivní modelování

Naším úkolem je najít optimální model systému (např. pro predikci, aproximaci, klasifikaci atd.). Při jeho hledání lze postupovat dvěma způsoby:

- *dedukcí*, na počátku na základě zkušeností a poznatků vytvoříme obecný model, ten pak ověříme na reálných datech, popř. přizpůsobíme jeho parametry, aby lépe vystihoval chování reálného systému. Tímto postupem se dále zabývat nebudeme,
- *indukcí*, na počátku máme několik základních stavebních jednotek, ze kterých se během učicí fáze snažíme vytvořit (odvodit) model, který nejlépe požadované chování realizuje. Máme tedy k dispozici pouze data, nevyužíváme žádné externí znalosti o povaze modelovaného systému. Tento přístup je typický například pro neuronové sítě typu GMDH (Group Metod of Data Handling) a pro algoritmy, které jsou na tomto paradigmatu založeny.



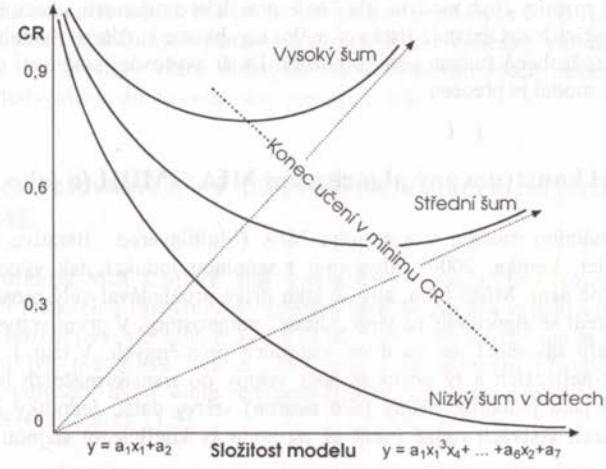

Obr. 11.2 Postup induktivního a deduktivního přístupu hledání modelů pro vytěžování dat

Indukční modelování spočívá ve vytváření modelů chování reálných systémů, u kterých známe jejich vnější projevy (reakce výstupů na vstupy). Přitom je nám ale vnitřní struktura těchto reálných systémů neznámá, nebo je poznatelná jen velmi obtížně. Induční přístup při tvorbě modelů v literatuře popisuje řada autorů, např. (Madala a Ivachněnko, 1994). Všichni ale museli navrhnout nějakou heuristiku, protože kombinatorický přístup k tvorbě modelů a vyhledávání nejlepšího řešení hrubou silou nejsou použitelné. Heuristika GMDH je založena na samoorganizaci a podrobněji ji popíšeme v dalším výkladu. O její programovou implementaci se zejména zasloužili spolupracovníci a doktorandi prof. Ivachněnka, např. J. Müller a F. Lemke, viz (Müller, Lemke, 2000), ale i jiní. Zajímavé informace můžete získat na webovské adrese (www.knowledgeminer.net).

Hledání optimálního modelu indukčním přístupem v GMDH (Müller, Lemke, 2000) lze připodobnit přírodnímu principu výběru nejsilnějšího jedince. Zde se za nejsilnějšího jedince považuje model, který je „nejlepší" ze všech modelů, které během hledání (učení) vytváříme. Matematicky lze selekci nejlepšího modelu popsat následujícím způsobem

$$m^* = \arg\min CR(m), \text{ kde } m \in M \text{ a } CR(m) = f(C,\sigma,T),$$

kde $m^*$ představuje optimální (hledaný) model, $M$ množinu všech zkoumaných modelů a $CR$ externí kritérium, na základě kterého se posuzuje kvalita modelu. Parametry externího kritéria jsou v tomto případě složitost modelu $C$, rozptyl šumu v datech $\sigma$ a typ transformace dat $T$.




Obr. 11.3 Požadované chování externího kritéria CR pro hodnocení kvality modelů.

Externí kritérium $CR$ musí nabývat minima pro modely, jejichž složitost jednak odpovídá složitosti modelovaných závislostí v datech (složitost $T$), a zároveň bere v úvahu rozptyl šumu v datech ($\sigma$). Tak lze zabránit přespecifikování neboli přeučení modelů (overfitting).

## 11.3 Model konstruovaný algoritmem COMBI GMDH (a jeho slabiny)

Proces generování modelů je v tomto případě proces s úplnou indukcí. Začíná s modely nejjednoduššími, s modely, jejichž výstupní hodnota závisí na jediné vstupní proměnné (a obvykle vystačí s lineární transformační funkcí).

Takto se vygenerují všechny možné modely v první vrstvě. S ohledem na potřeby dalšího výkladu použijeme i pro takto vzniklé modely termín populace modelů, i když tuto terminologii původní literatura neuvádí. Modely v dalších vrstvách pak závisí na dvou, třech a tak dále vstupních proměnných, ale i ty jsou stejného stupně jako transformační funkce v 1. vrstvě. Takto vzniklé modely tvoří 2. a vyšší populace modelů. Přenosová funkce je v každé z vyšších vrstev jiná, ale modely v každé vrstvě mají stejný stupeň složitosti. Z každé generace modelů (vrstvy) se pak podle zadaného externího

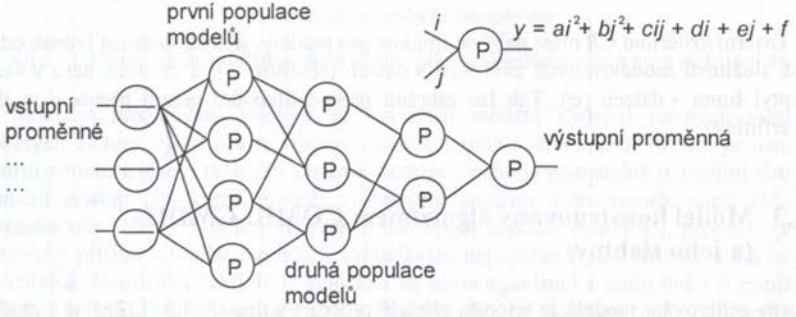
kritéria vybírá několik nejúspěšnějších a mezi všemi takto vybranými se pak hledá ten úplně nejlepší.

Hlavní slabinou kombinatorického algoritmu GMDH je jeho výpočetní a paměťová náročnost. Optimální model se hledá úplným prohledáváním prostoru možných řešení, takže s velkými výpočetními i paměťovými nároky. Tento kombinatorický algoritmus postupně generuje modely závisející na všech možných vstupních proměnných. Pro dobrý výsledek kupodivu stačí i jednoduchá přenosová funkce.

Teoretické rozbory chyb modelu, ale i naše praktické zkušenosti, ukazují, že pro každou množinu učicích dat existuje jistá optimální kombinace složitosti modelu (topologie) a chybovosti způsobená šumem učicí množiny. Další zvyšování složitosti modelu zlepšení nepřináší, model je přeučen.

## 11.4 Model konstruovaný algoritmem MIA GMDH (a jeho slabiny)

Hledání optimálního modelu v algoritmu MIA (Multilayered Iterative Alghorithm) GMDH (Müller, Lemke, 2000), algoritmu s neúplnou indukcí, tak výpočetně a paměťově náročné není. Místo toho, aby se jako dříve prohledával celý prostor možných modelů, soustředí se algoritmus na jeho „slibné" podprostory. V první vrstvě se generují všechny modely závisející jen na dvou vstupních proměnných. V této 1. populaci se najde několik nejlepších a ty poslouží jako vstupy do transformačních bloků (někdy označovaných jako jednotka, někdy jako neuron) vrstvy další. Jednotky jsou v tomto případě ve všech vrstvách stejné (mají až na hodnoty koeficientů stejnou přenosovou funkci).




Obr. 11.4 MIA GMDH model je tvořen polynomickými jednotkami s dvěma vstupy.

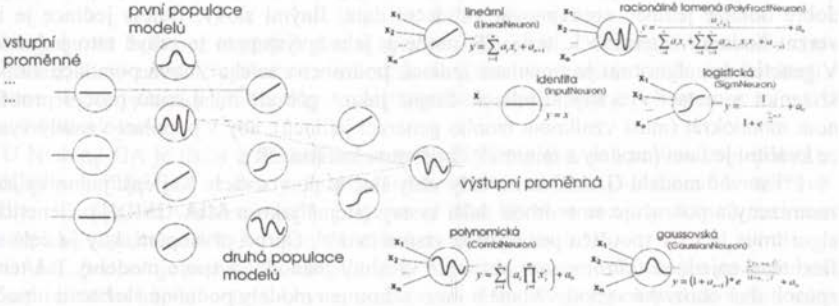
V algoritmech MIA GMDH (viz obr. 11.4) se tedy model tvoří tak, že se podle zvolené transformační funkce (např. polynom 2. stupně vstupních proměnných *i*, *j*) vytvoří kombinace všech vstupů aktuální vrstvy a z nich se vybere ten model (popř. těch několik modelů), který je nejlepší vzhledem k zadanému kritériu kvality (má např. nejmenší chybu na testovací množině, na validačních datech). Splňuje-li navíc i zadané externí kritérium *CR*, optimální model byl nalezen. Není-li tomu tak, opakuje se postup vytváření nové aktuální vrstvy, nové populace modelů.

Slabinou tohoto algoritmu je příliš velké omezení prostoru prohledávaných topologií. Transformační bloky (jednotky) mají pouze dva vstupy, stejnou přenosovou funkci a mohou být připojeny pouze k jednotkám předcházející vrstvy. Další nevýhodou je neefektivní prohledávání prostoru topologií – hrubou silou (procházení všech možných kombinací vstupů u jednotek).

Popsané metody hledaly optimální model hrubou silou, testovaly všechny možné kombinace vstupních proměnných. Je jasné, že tento postup vytváření aktuální vrstvy má své meze a že ho nelze od jistého počtu vstupních proměnných a od jisté mohutnosti množiny vstupních proměnných použít. Cestu k lepšímu výsledku vidíme v inteligentním prohledávání prostoru všech možných řešení (prostoru všech možných topologií modelu). Metody, které nám to umožňují, popíšeme dále.

## 11.5 Další zlepšování algoritmů pro tvorbu modelu, algoritmus GAME

Klasický algoritmus MIA GMDH jsme vylepšili hybridizací. Tou rozumíme aplikaci heterogenních jednotek s různými typy přenosových funkcí *V* a využití vstupů jednotek nejen z předchozí vrstvy, viz např. (Kordík a kol., 2004). Neomezujeme se v jedné vrstvě jen na jeden typ přenosové funkce, jako byl v popisovaném případě kvadratický Ivachněnkův polynom dvou vstupních proměnných, ale škála přenosových funkcí je mnohem větší (viz obrázek 11.5).




Obr. 11.5 Model GAME je tvořen jednotkami vhodného typu v závislosti na datech.

Pozměněná je i strategie tvorby vrstev: počet vstupů jednotky nemusí být omezen. I zde ale platí, že před tvorbou další aktuální vrstvy se vyberou jen ti nejlepší jedinci a ostatní, ti méně perspektivní, se prostě zahodí. Ukázalo se, a budeme na to ještě dále upozorňovat, že pro jednodušší problémy přežívají v síti jednotky s jednoduchou přenosovou funkcí, kdežto úspěšné modely složitějších systémů obsahují zejména složité jednotky.

Popisované vylepšení heuristiky GMDH jsme implementovali a vytvořili jsme softwarový nástroj pro automatické generování modelů GAME. Tento nástroj je k dispozici už od roku 2003 a stále jej inovujeme (http://neuron.felk.cvut.cz/game). Ukázalo se, a není to tak překvapivé, že pro každý soubor vstupních dat je automaticky

generovaný model jiný. Je složen z jiných přenosových funkcí a má i jinou topologii. Stejně jako ve dříve zmíněných původních algoritmech GMDH vzniká i tady nejlepší model samoorganizací. Implementaci automatického generátoru modelů výrazně pomohlo využití evolučních technik, které pracují s chromozomy, ve kterých jsou zakódovány jak vstupy, tak i typy jednotek a tvar jejich přenosových funkcí. Námi použité evoluční techniky jsou předmětem popisu v dalších oddílech.

## 11.6 Automatizace generování modelů v GAME, využití evolučních výpočetních technik

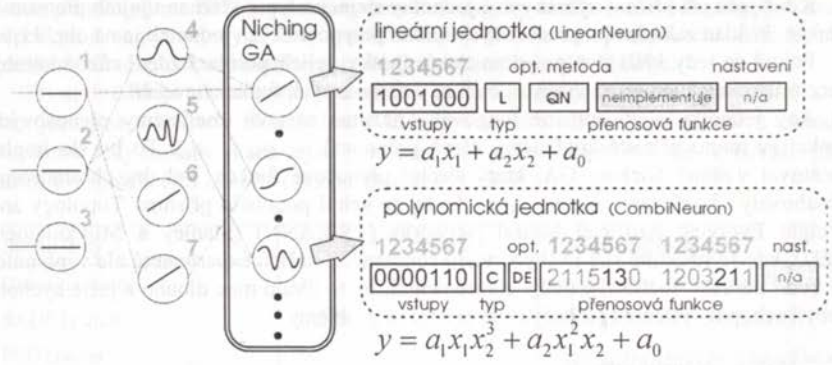
Prostor všech možných topologií modelů GAME je velmi rozsáhlý, protože modely lze tvořit z heterogenních jednotek, jednotek s různými přenosovými funkcemi (obr. 11.5). Jednotky mohou mít navíc libovolný počet vstupů připojených k libovolným jednotkám z předchozích vrstev. Další dimenze prostoru přidává typ a nastavení optimalizační metody, která přizpůsobuje hodnoty koeficientů přenosových funkcí. Je zřejmé, že prohledávání takto rozsáhlého prostoru hrubou silou (jako je tomu u COMBI GMDH) by bylo nejen neefektivní, ale přímo nemožné. Proto GAME využívá k hledání optimální topologie modelů metody výpočetní inteligence, konkrétně přírodou inspirovaný genetický algoritmus (GA).

Genetický algoritmus hledá optimum topologie GAME modelů pomocí populace jedinců, z nichž každý představuje nějaké řešení. Toto řešení je zakódováno v chromozomu jedince (jednotky, viz obr. 11.6). Kvalitu jedince určuje v našem případě to, jak dobře dokáže jedinec aproximovat validační data. Jinými slovy: fitness jedince je inverzní hodnota externího kritéria CR modelu, jehož výstupem je právě tato jednotka. V genetickém algoritmu je populace jedinců podrobena selekci, nová populace vzniká křížením a mutací vybraných jedinců. Stejně jako v přírodě musí tento proces proběhnout mnohokrát (musí vzniknout mnoho generací jedinců), aby v populaci vznikly vysoce kvalitní jedinci (modely s minimální hodnotou kritéria CR).

Při stavbě modelu GAME se modely vždy šlechtí po vrstvách. Nejlepší jednotky jsou zamrazeny a pokračuje se tvorbou další vrstvy (stejně jako u MIA GMDH). Genetický algoritmus je tedy spouštěn pro každou vrstvu zvlášť. Oproti přístupům, kdy je celá síť šlechtěna najednou (chromozom obsahuje všechny jednotky a spoje modelu), má tento způsob dvě obrovské výhody. Soutěží mezi sebou jen modely podobné složitosti a počet genů chromozomu je výrazně redukován.

Konkrétní příklad je uveden na obr. 11.6. Genetický algoritmus šlechtí modely v druhé vrstvě. Čtyři vyšlechtěné jednotky v první vrstvě již byly zamrazeny a nebude se s nimi dále manipulovat. Jednotky v aktuální vrstvě se mohou připojit buď na tři vstupní jednotky, nebo na jednotky z první vrstvy. První část chromozómů tedy kóduje vstupy jednotek. Dále následuje kód typu přenosové funkce, optimalizační metoda pro nastavení koeficientů přenosové funkce, tvar přenosové funkce a nastavení optimalizační metody.

Pokud se kříží jednotky různých typů, pracuje se jen s první částí chromozomů, která kóduje vstupy jednotek. Pokud se při křížení potkají 2 jednotky stejného typu (např. jednotky CombiNeuronu s polynomickou přenosovou funkcí), kříží se i další část chromozomu, která kóduje přenosové funkce jednotek.




Obr. 11.6 Typické zakódování jednotek GAME do chromozomů. Označení „neimplementuje" a „n/a" znamená, že jednotky daného typu tuto část chromozomu nevyužívají.

Typické zakódování lineárních a polynomiálních jednotek vidíte na obr. 11.6. Všimněte si, že koeficienty přenosových funkcí $a_0$, $a_1$, ..., $a_n$ nejsou v chromozomech zakódovány. Nastavují se optimalizačními metodami popsanými v další výkladu.

## 11.7 Zachování různorodosti jednotek v GAME

Zjistili jsme, že lepší než obyčejný genetický algoritmus je tzv. Niching GA (rozdělování do druhů), který umožní vyšlechtit v síti skupinu jednotek, které se vzájemně doplňují. K tomu, aby se jednotky „dobře doplňovaly", je třeba zajistit, aby jejich výstupy byly co nejméně korelované.

U Niching GA je třeba znát vzdálenost jedinců (běžný GA se vzdáleností nepočítá). Vzdálenost určuje, jak se dva jedinci od sebe liší. Vzdálenost jedinců může mít dvě složky. Genotypická vzdálenost udává, jak se od sebe liší chromozomy jedinců, zatímco fenotypická vzdálenost určuje různost vnějších atributů jedinců.

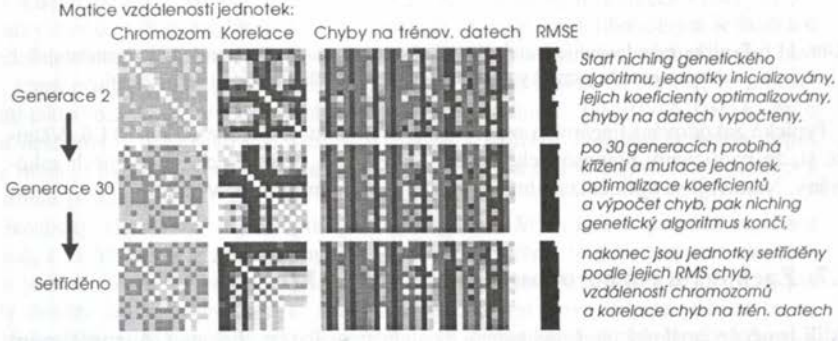
GAME podporuje niching na základě obou těchto vzdáleností. Záleží jen na tom, zda je daná jednotka implementuje. U všech jednotek lze vypočítat genotypickou vzdálenost jako Hammingovu vzdálenost zakódování vstupů. Kříží-li se dvě jednotky polynomického typu (CombiNeuron), genotypická vzdálenost má ještě další složku, a to vzdálenost zakódování přenosových funkcí. Fenotypická vzdálenost jednotek se vypočte jako záporná hodnota korelace chyb jednotek na trénovací množině. Celková vzdálenost polynomických jednotek je tedy dána jako (vzdálenost zakódování vstupů) + (vzdálenost zakódování přenosových funkcí) – (korelace chyb jednotek na trénovací množině).

Za běhu Niching GA tedy počítáme vzdálenost jednotek a o místo v další generaci bojují vždy jednotky sobě podobnější. Technik pro zachování různorodosti v populaci genetického algoritmu existuje celá řada. My používáme tzv. deterministické zalidňování (Deterministic Crowding) (Mahfoud, 1995), kde o místo v další generaci bojuje potomek se svým podobnějším rodičem.

Když jsou při křížení vybrány dvě jednotky stejného typu, kříží se i jejich přenosové funkce. Příklad zakódování přenosových funkcí polynomických jednotek je na obr. 11.6.

Pokud se tedy kříží dvě polynomické jednotky, jejich potomci zdědí různé kombinace polynomických členů vyskytujících se v přenosových funkcích rodičů.

Aby jednotky v síti správně fungovaly, musíme nastavit koeficienty přenosových funkcí (v tomto případě koeficienty členů polynomů $a_0$, $a_1$, ..., $a_n$). To by šlo implementovat v rámci Niching GA, který šlechtí přenosové funkce. Pak by chromozomy obsahovaly i koeficienty $a_0$, $a_1$, ..., $a_n$. To by se velmi podobalo přístupu Topology and Weight Evolving Artificial Neural Networks (TWEANN) (Stanley a Miikkulainen, 2002), kde se struktura sítě i váhy vyvíjejí současně. Z našich experimentů ale vyplynulo, že kvůli nárůstu složitosti a délky chromozomů, by to trvalo moc dlouho a ještě bychom nebyli schopni vyšlechtit složitější sítě pro těžké problémy.




Obr. 11.7 Různorodost jednotek v populaci Niching GA během tvorby vrstvy modelu GAME. V populaci je 15 jednotek. Z jejich chyb na trénovacích a validačních datech se vypočítá matice korelace a také chyba RMSE. Spolu s maticí genotypických vzdáleností poskytuje tato vizualizace poměrně dobrý přehled o různorodosti jednotek v populaci. Vrstvu modelu nakonec utvoří jednotky, které mají nejnižší chybu RMSE na validačních datech a zároveň si nejsou podobné.

V GAME tedy používáme tedy jiný přístup – šlechtíme jen architekturu a přenosové funkce, koeficienty optimalizujeme zvlášť.

## 11.8 Optimalizace koeficientů přenosových funkcí jednotek v GAME

V tomto případě prohledáváme stavový prostor všech možných modelů GAME s ohledem na koeficienty přenosových funkcí. Zatímco topologie modelů se mění nespojitě, koeficienty jsou reálná čísla. K jejich nastavení jsou zapotřebí metody pro spojitou optimalizaci.

Koeficienty přenosových funkcí jednotek v GAME optimalizujeme způsobem uvedeným na obrázku 11.8. Efektivnější je varianta b), kdy optimalizační metoda pracuje se znalostí gradientu chybové funkce. V případě, že gradient neznáme, nebo optimalizační metoda explicitní znalost gradientu nevyžaduje, použijeme méně efektivní variantu a).

Tento princip nám umožní vypočítat optimální hodnoty koeficientů i pro jednotky, které neposkytují analyticky vypočtený gradient chyby. Pak je na optimalizační metodě, aby gradient odhadla, pokud ho při výpočtech hodnot koeficientů vyžaduje.

Při optimalizaci ale vždy vznikne otázka: jakou optimalizační metodu použít? Každá data jsou jiná a lze tedy předpokládat, že je pro ně vhodná jiná optimalizační metoda. Zkoušíme tedy metody různé – jak gradientní, tak přírodou inspirované. V GAME jsou zatím k dispozici tyto metody optimalizace koeficientů:

| Jméno třídy | Zkratka | Hledání | Optimalizační metoda |
|---|---|---|---|
| UncminTrainer | QN | gradientní | kvazinewtonova metoda |
| SADETrainer | SADE | genetické | genetický algoritmus SADE |
| PSOTrainer | PSO | hejno | optimalizace pomocí hejna částic |
| HGAPSOTrainer | HGAPSO | hybridní | hybrid GA a PSO |
| PALDifferentialEvolutionTr. | PalDE | genetické | diferenciální evoluce verze 1 |
| DifferentialEvolutionTrainer | DE | genetické | diferenciální evoluce verze 2 |
| StochasticOSearchTrainer | SOS | jednoduché | stochastické ortogonální prohledávání |
| OrthogonalSearchTrainer | OS | jednoduché | ortogonální prohledávání |
| ConjugateGradientTrainer | CG | gradientní | metoda sdružených směrů |
| ACOTrainer | ACO | hejno | mravenčí kolonie |
| CACOTrainer | CACO | hejno | mravenčí kolonie, optimizalizace |

Obr. 11.8 Schéma optimalizace koeficientů přenosových funkcí jednotek s výpočtem analytického gradientu a bez něho.

Koeficienty jednotek sítě může nastavovat libovolná z nich. To, jaká metoda je použita, je zakódováno v chromozomu jednotky niching genetického algoritmu (obr. 11.6). Potomek jednotky má padesátiprocentní šanci, že zdědí optimalizační metodu po svých rodičích, jinak mu je přiřazena náhodně. Naše experimenty ukázaly (Kordík, 2006), že toto je nejlepší způsob výběru vhodné optimalizační metody.

V následujících oddílech princip jednotlivých metod optimalizace koeficientů přenosových funkcí stručně popíšeme.

## 11.9 Gradientní metody optimalizace koeficientů jednotek sítě

Chyba modelu je kromě topologie funkcí spojitých hodnot koeficientů přenosových funkcí. Pokud je možné vyjádřit závislost chyby na těchto parametrech analyticky, je možné vypočítat i derivaci a k hledání optima se zvláště hodí gradientní metody. V případě, že gradient nelze analyticky vyjádřit, tak jej musí gradientní metoda umět odhadnout.

### 11.9.1 Kvazinewtonova metoda (Quasi-Newton method QN)

V oblasti nelineární optimalizace je kvazinewtonova metoda jednou z nejpopulárnějších metod. Měla by dosahovat kvadratické konvergence, ale vždy záleží na problému. Směr hledání je vypočten pomocí Hesseho matice (matice druhých derivací funkce). Protože výpočet druhých derivací je velmi náročný, Hesseho matice je počítána pouze přibližně, iteračně pomocí doplňků, např. Davidonova–Fletcherova–Powellova, DFP, (Salane a Tewarson, 1980) nebo BFGS (Avriel a Mordecai, 2003).

### 11.9.2 Metoda sdružených směrů (Conjugate Gradients)

Metoda sdružených směrů je založena na myšlence, že konvergence k řešení může být zrychlena, pokud při určení aktuálního směru hledání uvažujeme i všechny předchozí směry, ve kterých jsme se vydali. V ideálním případě by směry hledání měly být na sobě lineárně nezávislé. Pak bychom řešení lineárního systému měli nalézt v maximálně v $n$ krocích, kde $n$ je počet neznámých v řešeném systému.

Metodu sdružených směrů lze nasadit i na řešení nelineárních systémů. Vlastnosti metody zlepšuje restartování (zapomenutí předchozích směrů).

## 11.10 Přírodou inspirované metody optimalizace koeficientů – evoluce

### 11.10.1 Diferenciální evoluce (DE)

Diferenciální evoluce je genetický algoritmus se speciálním reprodukčním mechanismem. Potomek vzniká tak, že se vážený rozdíl chromozomů dvou rodičů přičte k chromozomu třetího rodiče. Do chromozomu potomka se postupně vybírají buď geny z výsledného chromozomu, nebo geny čtvrtého rodiče. Potomek nahrazuje čtvrtého rodiče v populaci, pokud je kvalitnější (má lepší fitness). Podrobný popis tohoto algoritmu najdete v (Mařík a kol., 2001).

### 11.10.2 Zjednodušená atavistická diferenciální evoluce (Simplified Atavistic Differential Evolution – SADE)

Genetický algoritmus SADE (Hrstka a Kučerová, 2004) je navržen tak, aby omezil nežádoucí jev mnoha optimalizačních metod – předčasnou konvergenci do lokálního minima. Jako jeden se svých operátorů křížení používá reprodukční mechanismus diferenciální evoluce. Zajímavým prvkem algoritmu SADE jsou tzv. radiační pole. Tato pole jsou oblastmi prohledávaného prostoru, ve kterých existuje zvýšená pravděpodobnost mutace. Pokud se nějaký jedinec to takového pole dostane, je pravděpodobné, že bude

silně zmutován a přesune se do jiné části prostoru. Vstup jedince do radiačního pole zároveň oslabí intenzitu radiace (poloměr pole se zmenšuje). Radiační pole se zřizují v místech lokálních minim, aby zabránila předčasné konvergenci. Globální minimum je nalezeno, když poloměr nějakého radiačního pole poklesne k nule.

## 11.11 Přírodou inspirované metody optimalizace koeficientů – hejna

### 11.11.1 Optimalizace pomocí hejna částic (Particle Swarm Optimization)

Tento algoritmus je inspirován chováním hejna ptáků (ryb). Hejno se pohybuje prostorem za potravou a jednotlivé částice (ptáci nebo ryby) odvozují směr svého dalšího pohybu z polohy ostatních částic. Hejno se také často vrací za potravou do míst, kde se v minulosti hojně vyskytovala.

Algoritmus PSO simuluje toto chování pomocí iterační korekce směrů a rychlostí částic. Pohyb částic se odvozuje jednak od nejlepšího řešení, které částice dosud nalezla, ale také od zatím nejlepšího řešení nalezeného celým hejnem. Je si tedy třeba pamatovat a aktualizovat nejlepší nalezené řešení pro každou částici a pro celé hejno. Každá částice se vzhledem k náhodně inicializovaným koeficientům chová trochu odlišně od ostatních částic. Korekce směru k lokálnímu a globálnímu nejlepšímu řešení tedy bude pro každou částici jiná (Kennedy a Eberhart, 1995).

### 11.11.2 Mravenčí kolonie* (Ant Colony Optimization*)

Optimalizace pomocí mravenčích kolonií (ACO) se běžně používá v kombinatorických problémech (obchodní cestující, směrování paketů v síti). Teprve nedávno se klony této metody začaly uplatňovat také při řešení spojitých problémů. ACO je metoda vyvinutá pro trénování dopředných neuronových sítí (http://aco.wz.cz), (Blum a Socha, 2005), (Socha, 2004). Symbol „*" v názvu metody značí, že se jedná o modifikovanou verzi původní metody určenou pro spojitou optimalizaci. Připomíná paralelní simulované žíhání. Každý mravenec určuje bod v prohledávaném prostoru a další mravenci jsou generováni náhodně, přičemž některý z již existujících mravenců určuje střední hodnoty souřadnic a z jeho vzdálenosti od ostatních vypočteme odchylku normálního rozdělení, pomocí něhož je nová náhodná pozice generována.

### 11.11.3 Spojitá optimalizace pomocí mravenčí kolonie (Continuous Ant Colony Optimization)

Ve středu prohledávané plochy je hnízdo, ze kterého vychází konečný počet vektorů. K těmto vektorům je přiřazen feromon, jehož množství ovlivňuje výběr tohoto směru mravenci lezoucími z hnízda. Mravenec si vybere jeden z vektorů a přemístí se na pozici dosud nejlepšího mravence lezoucího tímto směrem. Potom se přemístí náhodným směrem do vzdálenosti omezené poloměrem prohledávání, který se v průběhu výpočtu zmenšuje.

Feromon k vektoru přidává pouze mravenec, který najde lepší řešení než dosavadní a změní tak polohu vektoru. Feromon se odpařuje a jeho množství je zdola omezeno minimální hodnotou. Mravenci, kteří si vyberou směr s minimální hodnotou feromonu,

nemusí dodržet omezení poloměrem prohledávání – to umožňuje únik z lokálního minima většími skoky (starving) (http://aco.wz.cz), (Kuhn, 2002).

## 11.12 Hybridní a ostatní metody optimalizace koeficientů

### 11.12.1 Hybrid GA a PSO (HGAPSO)

Algoritmus HGAPSO (Kuhn, 2002) vychází z následující úvahy: jak genetický algoritmus (GA), tak PSO pracují s populací jedinců, proto je není obtížné zkombinovat. V PSO se předpokládá, že všichni jedinci jsou stejně staří – existuje pouze jedna generace. Naproti tomu v GA se jedinci generaci po generaci šlechtí a neuvažuje se, že by se měnili (např. učením) v průběhu jedné generace.

V přírodě je běžné, že se jedinci zlepšují a přizpůsobují prostředí v rámci jedné generace. Proto byl genetický algoritmus obohacen o fázi přiučování jedinců pomocí PSO. Výhodou je, že PSO má charakter iteračního prohledávání, a proto vhodně doplňuje GA, který prohledává prostor víceméně nespojitě.

### 11.12.2 Ortogonální prohledávání (Orthogonal Search OS) a stochastické ortogonální prohledávání (Stochastic Orthogonal Search SOS)

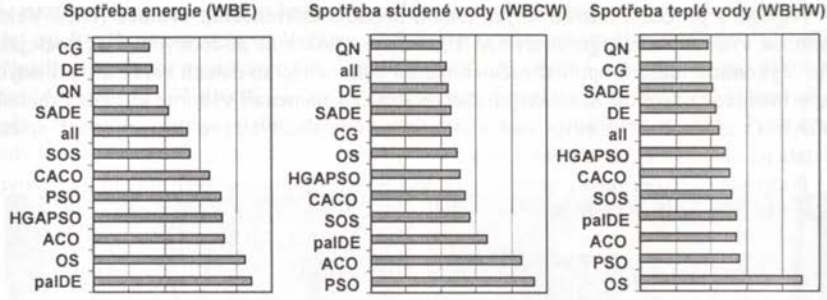
Ortogonální prohledávání (OS) optimalizuje mnoharozměrný problém tak, že vybírá postupně dimenzi po dimenzi a minimalizuje chybu v těchto dimenzích. V implementaci algoritmu (www.cebl.auckland.ac.nz/pal-project/), který jsme použili my, není využit gradient prostoru chyby.

Stochastické ortogonální prohledávání (SOS) se od popsaného algoritmu liší pouze tím, že jsou dimenze pro optimalizaci vybírány náhodně.

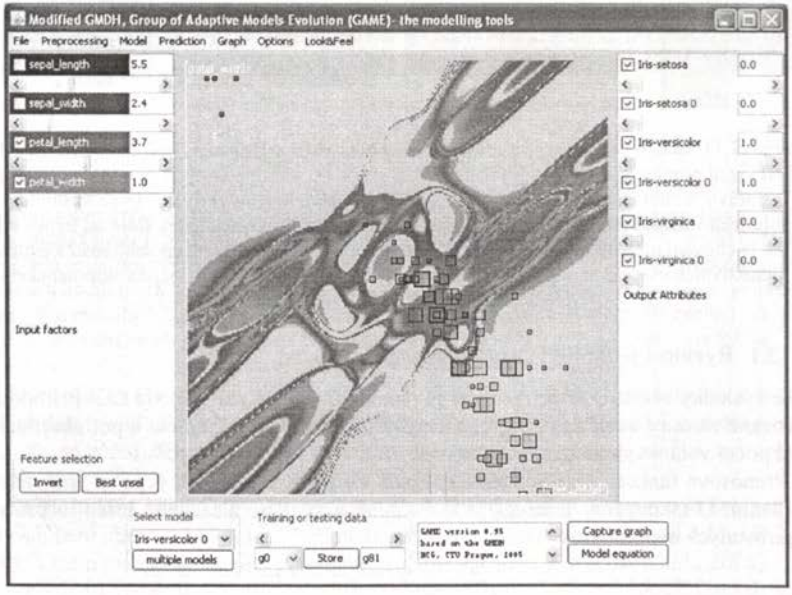
## 11.13 Porovnání výsledků optimalizačních metod

Řadu popsaných, přírodou inspirovaných optimalizačních metod jsme implementovali v programu GAME. S jeho pomocí jsme provedli řadu experimentů, například predikci spotřeby energie (WBE), teplé (WBHW) a studené vody (WBHW) v bytovém domě v závislosti na venkovní teplotě, rychlosti větru, vlhkosti vzduchu a míře slunečního záření – data „Budovy" získaná z práce (Prechelt, 1994). Výsledky experimentů znázorněné na obr. 11.9 uvádějí chybu modelů získaných jednotlivými metodami. Další aplikaci ukazuje obr. 11.10 (též barev. příl.).

Nejlépe dopadla kvazinewtonova metoda, která vytvořila modely v průměru s nejmenší chybou. Jen při modelování spotřeby energie, která se ze vstupních dat špatně odvozuje, dopadly lépe metody CG a DE. To by ukazovalo na fakt, že se metoda QN na zašuměných datech nepatrně přeučuje. Bude však třeba provést další experimenty, aby se tato domněnka prokázala. Dále jsme při tvorbě modelů zjistili, že metoda QN se lépe uplatňuje v prvních vrstvách modelu a pro šlechtění jednotek hlouběji v sítích se lépe hodí metoda CG. V hlubších vrstvách je prostor chyby složitější, proto lze předpokládat, že metodě CG vyhovují více složité optimalizační problémy.



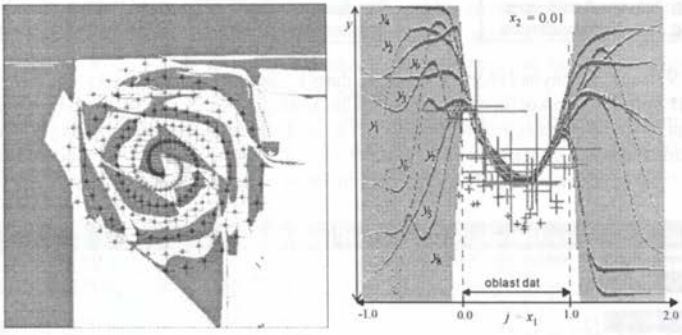

Obr. 11.9 Průměrná chyba GAME modelů na datech „Budovy" (vodorovná osa), v závislosti na metodě optimalizace koeficientů jednotek. Čím je sloupec delší, tím větší chyby dosahovaly modely na testovacích datech. Pro optimalizaci koeficientů přenosových funkcí byla použita vždy pouze jedna metoda (označení na svislé ose). Popis „all" značí, že při tvorbě modelu byly použity všechny metody a výběr vhodných zajistí evoluční algoritmus.




Obr. 11.10: Aplikace GAME je volně ke stažení na http://neuron.felk.cvut.cz/game. Můžete si sami vyzkoušet, jakých výsledků dosáhnete na vašich datech. Na obrázku vidíte klasifikaci kosatců do tří tříd v závislosti na parametrech rostliny. Vizualizační metody umožňují lépe rozpoznat, zda jsou modely přeučeny. Zde vidíte, že induktivní model založený na gaussovských jednotkách se evidentně přeučil, ačkoliv je jeho chyba při desetinásobném použití křížové validace minimální.

Nejlépe z přírodou inspirovaných metod dopadla diferenciální evoluce (DE). Velmi podobné výsledky měl algoritmus SADE, který je založen na podobných principech jako DE. Výkonnost dalších optimalizačních metod byla na těchto datech horší, ale při jiných experimentech na velmi zašuměných datech jsme zaznamenali výborné výsledky metody HGAPSO. „Mravenčí" metody zase dobře fungují na složitých problémech (např. spirální data na obr. 11.11, viz též bar. příl.)

Podrobnější experimenty srovnávající výkonnost optimalizačních metod na různých datech lze najít v (Kordík, 2006).




Obr. 11.11 Induktivní modely vyšlechtěné pomocí GAME zvládnou vyřešit i poměrně složité klasifikační problémy, jako je například rozlišení dvou do sebe zavinutých spirál (levý obrázek). Napravo je zobrazeno chování sdružených induktivních modelů (křivky), které se shoduje v oblastech vstupního prostoru, kde je k dispozici dostatek trénovacích dat. Dále od trénovacích dat je chování modelů náhodné. Tohoto faktu jsme využili k určování spolehlivosti výstupu induktivních modelů, a také k definici zajímavých oblastí v datech při jejich automatickém vyhledávání (obrázek 11.12, viz též bar. příl.).

### 11.13.1 Rychlost jednotlivých optimalizačních metod

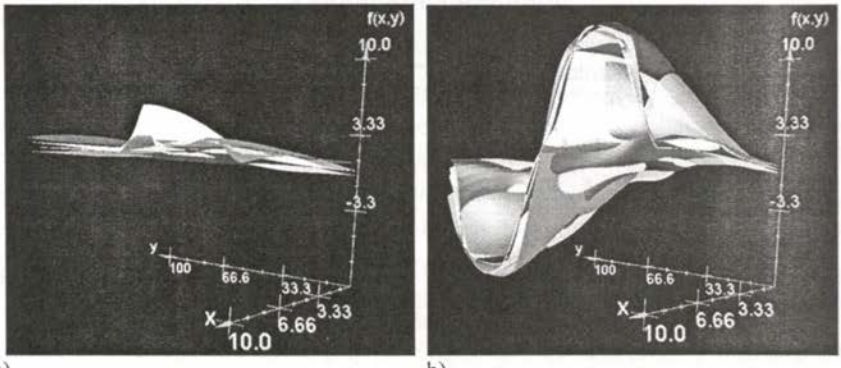
Naše výsledky ukazují, že nejrychlejší je metoda QN, dále pak metoda CG. Přírodou inspirované metody trvají déle, protože nevyužívají analytický gradient a potřebují daleko větší počet volání vyhodnocení chyby.

Přenosové funkce některých jednotek jsou ale příliš složité pro výpočet analytického gradientu. U takovýchto jednotek je doba běhu a kvalita nalezeného řešení u přírodou inspirovaných metod i gradientních metod srovnatelná.

## 11.14 Závěr

V této kapitole jsme popsali několik zajímavých přírodou inspirovaných optimalizačních metod. Naše výsledky ukazují, že některé jsou velmi dobře použitelné pro spojitou optimalizaci. Největší pole působnosti bezesporu najdou při složitých optimalizacích, jako je například hledání vhodné topologie induktivních modelů, kde nelze vypočítat gradient chybové funkce. I v případě, že gradient určit lze, se přírodou inspirované meto-

dy neztratí. Jsou sice o něco horší než oblíbené kvazinewtonovy optimalizační metody, ale jejich výkonnost velmi záleží na charakteru optimalizovaného problému (v našem případě charakteru modelovaných dat). Takzvaný „No Free Lunch Theorem" (Wolpert and Macready (1997)) tvrdí, že když vyladíme algoritmus tak, aby fungoval optimálně na jednu třídu dat, zhoršíme tím jeho výkonnost na datech jiného charakteru. Neexistuje tedy jeden algoritmus, který by fungoval optimálně na jakýchkoli datech.




a) b)

Obr. 11.12a): Umělá data mají tři vstupy $x,y,z$. Výstup $f(x,y)$ je závislý na vstupech $x$ a $y$, ale jen pro jisté hodnoty vstupu $z$. Data jsme použili k natrénování induktivních modelů metodou GAME. Pro zobrazení namodelovaných závislostí v datech jsme vyvinuli techniku 3D řezů. Na tomto obrázku vidíte řez, ve kterém je zobrazeno chování induktivních modelů pro nevhodně zvolenou hodnotu vstupu $z$. b): Pokud je hodnota vstupu $z$ nastavena optimálně, chování induktivních modelů je daleko zajímavější. Reálná data mají velké množství vstupů, proto je těžké nastavit jejich hodnoty tak, aby bylo chování modelů v řezu zajímavé. Pro tento účel jsme vyvinuli speciální evoluční algoritmus, který zajímavé oblasti hledá. Animaci ukazující evoluci 3D řezů naleznete na stránkách projektu FAKE GAME (http://neuron.felk.cvut.cz/game), viz též bar. příl.

Proto má smysl mít mnoho algoritmů založených na různých principech a pro neznámá data vybrat kombinaci nejvhodnějších optimalizačních algoritmů pomocí evoluce tak, jak se tomu děje v GAME. V našem dalším výzkumu se zaměříme zejména na využití analytického gradientu v přírodou inspirovaných metodách.

Další úspěšné využití přírodou inspirovaných metod v našem výzkumu je evoluce zajímavých řezů chování induktivních modelů pro vytěžování znalostí (obr. 11.11, též v bar. příl.). Více o tomto zajímavém problému najdete například v (Kordík a kol., 2006).

Ve vývoji systému GAME nadále pokračujeme. Nový koncept FAKE GAME slouží k automatizovanému vytěžování znalostí z dat prostřednictvím induktivních modelů. Pokud vás problematika zajímá, můžete se sami na vývoji systému podílet (http://neuron.felk.cvut.cz/game). Uvítáme také data různého charakteru, abychom mohli systém vyladit na co nejširší škálu dat.

Většina přírodou inspirovaných optimalizačních metod, které jsme popsali a implementovali v GAME, patří do arzenálu poměrně nové vědecké disciplíny – výpočetní inteligence (CI – Computational Intelligence). Není naším cílem podrobně vysvětlovat

rozdíly mezi výpočetní a umělou inteligencí. Chceme jen upozornit na fakt, že termín výpočetní inteligence poprvé uvedli před 10 lety známí a v oboru umělé inteligence pracující odborníci J. Bezdek a L. A. Zadeh. Obor se bouřlivě a úspěšně rozvíjí. Vzájemný vztah CI a AI snad nejlépe vystihuje obrázek 11.13, pro který jsme našli inspiraci v (Pasquier, www.c2i.ntu.edu.sg).

Obr. 11.13 Výpočetní inteligence a její vztah k ostatním vědeckým disciplínám.

Metodami výpočetní inteligence jsou neuronové sítě, fuzzy systémy a právě většinou v biologii inspirované evoluční výpočty (evoluční algoritmy – například genetické, hejna – jako třeba mravenčí kolonie apod.). Pro aplikaci metod tohoto oboru jsou vhodné problémové oblasti s nesymbolickou reprezentací znalostí. Jejich aplikací lze získávat z dat nové znalosti.

Oboru výpočetní inteligence se ujala zejména organizace IEEE. Od roku 1994 se pravidelně konají prestižní vědecké konference. Ta poslední v roce 2006 měla dokonce formát světového kongresu (WCCI 2006) a konala se ve Vancouveru v Kanadě. Bylo velmi zajímavé vidět jak metody, ze kterých jsme vyšli i my (např. NEAT), poskytují vynikající výsledky například ve vývoji nových algoritmů počítačových her (Miikkulainen a kol., 2006).

**Literatura**

Avriel and Mordecai: Nonlinear Programming: Analysis and Methods. Dover Publishing, 2003.

Blum Ch., Socha K.: *Training feed-forward neural networks with ant colony optimization: An application to pattern classification.* Proceedings of Hybrid Intelligent Systems Conference, Rio de Janeiro, Brazílie, listopad 6–9, 2005.

Hrstka O., Kučerová A.: *Improvements of real coded genetic algorithms based on differential operators preventing premature convergence.* Advances in Engineering Software, 35(3–4), březen–duben, 2004, s. 237–246.

Ivachnenko A. G.: *Polynomial theory of complex systems*. IEEE Transactions on Systems, Man, and Cybernetics, SMC-1(1):364–378, 1971.

Juang C.-F., Liou Y.-C.: *On the hybrid of genetic algorithm and particle swarm optimization for evolving recurrent neural network*. In Proceedings of the IEEE International Joint Conference on Neural Networks, volume 3, s. 2285–2289, Dept. Of Electr. Eng., Nat. Chung-Hsing Univ., Tchaj-pej, Tchajwan, červenec 2004, s. 25–29.

Kennedy J., Eberhart R.: *Particle swarm optimization*. in Proc. of the IEEE Int. Conf. on Neural Networks, Piscataway, NJ, 1995, s. 1942–1948.

KnowledgeMiner software http://www.knowledgeminer.net/

Kordík P., Saidl J., Šnorek M.: *Evolutionary Search for Interesting Behavior of Neural Network Ensembles*. to appear in Proceeding of 2006 IEEE World Congress on Computational Intelligence, červenec 2006, Vancouver, Kanada.

Kordík P., Šnorek M., Genyk-Berezovskyj M.: *Hybrid Inductive Models: Deterministic Crowding Employed*. Proceedings of the International Joint Conference on Neural Networks; Piscataway: IEEE, 2004, s. 2343–2346.

Kordík P.: *Fully Automated Knowledge Extraction using Group of Adaptive Models Evolution*, PhD thesis, FEE, CTU Prague, September 2006. Available online: http://neuron.felk.cvut.cz/game

Kuhn L.: *Ant Colony Optimization for Continuous Spaces*. A thesis submitted to The Department of Information Technology and Electrical Engineering The University of Queensland, October 2002.

Madala H., Ivakhnenko A.: *Inductive Learning Algorithm for Complex System Modelling*. CRC Press, 1994. Boca Raton.

Mahfoud S. W.: *A comparison of parallel and sequential niching methods*. In Sixth International Conference on Genetic Algorithms, 1995, s. 136–143.

Mařík V. a kol.: *Umělá inteligence III*. Academia, Praha 2001.

Miikkulainen R., a kol.: *Computational Intelligence in Games*. Computational Intelligence: Principles and Practice, IEEE Computational Intelligence Society, 2006, s. 155–191.

Müller J. A., Lemke F.: *Self-Organising Data Mining*. Berlín, 2000.

Oficiální webová stránka uživatelů metody GMDH http://www.gmdh.net

PAL Development Core Team, http://www.cebl.auckland.ac.nz/pal-project/

Pasquier M.: *Centre for Computational Intelligence*. Nanyang Technological University, http://www.c2i.ntu.edu.sg.

Prechelt L.: *A set of neural network benchmark problems and rules*. Technical Report 21/94, Karlsruhe, Německo, 1994.

Salane and Tewarson: *A uniffed derivation of symmetric quasi-newton update formulas*. Applied Math, 25:29, 36, 1980.

Socha K.: *ACO for Continuous and Mixed-Variable Optimization*. Proceedings of 4th International Workshop on Ant Colony Optimization and Swarm Intelligence (ANTS'2004), Brusel, Belgie, září 5–8, 2004.

Stanley K. O., Miikkulainen R.: *Evolving neural networks through augmenting topo-logies*. Evolutionary Computation, 10(2), Massachusetts Institute of Technology, 2002, s. 99–127.

Stránky o optimalizaci pomocí mravenčích kolonií – aco-web http://aco.wz.cz

Webová stránka projektu FAKE GAME http://neuron.felk.cvut.cz/game

# 12 O empirickém výzkumu algoritmického objevování

Rozvoj algoritmů strojového učení a data miningu umožnil částečně automatizovat základní badatelský úkon, kterým je tvorba hypotéz vysvětlujících vzorek empirických pozorování (obr. 12.1 vlevo). Nedávno započatý výzkum tzv. strojového učení v uzavřené smyčce (closed-loop machine learning) vyústil přímo v konstrukci vědce-robota, který bádá v oblasti funkční genomiky. Článek (King a kol., 2004) o tomto robotu dokonce otiskl časopis Nature, což je v oblasti umělé inteligence nevídaný úspěch. Podobně jako vědec-smrtelník i tento robot vymýšlí hypotézy tak, aby co nejlépe odrážely pozorovaná data. Formálně tyto hypotézy jsou programy v jazyce Prolog, tedy teoriemi v predikátové logice. Není tedy divu, že k jejich konstrukci robot využívá prostředků induktivního logického programování, se kterým se čtenář mohl seznámit v předešlých svazcích Umělé inteligence (2 a 4). Oproti těmto systémům ale robot umí robot jednu podstatnou věc navíc. Aby se mohl rozhodnout mezi několika hypotézami, které se zatím zdají podobně kvalitní, sám navrhne a fyzicky uskuteční příslušné in-vitro experimenty. Dá se říci, že lidští výzkumníci, který robota postavili, jsou odsunuti ze základní smyčky daného vědeckého bádání a postaveni do pozice pouhých metabadatelů. Ti již primárně nezkoumají funkce genů, ale principy robota, který tak činí za ně.

Obr. 12.1 Vlevo: tradiční vědecké bádání – člověk tvoří hypotézy vysvětlující pozorování přírodního jevu. Vpravo: počítačem podporované bádání využívající algoritmů strojového učení.

Opusťme nyní zajímavé téma robotova reálného experimentování a zaměřme se na obecnější aspekt algoritmického objevování, který plyne z toho, že oproti lidem mohou algoritmy umělé inteligence plodit hypotézy v mnohem masovějším měřítku, a to za dobře kontrolovatelných podmínek. Tak může metabadatel získat empirický vzorek dat popisujících proces objevování. Tato data sama o sobě mohou být podrobena statistické analýze, jako kterýkoliv přírodní jev (obr. 12.3). Taková analýza toho může mnoho odhalit o počítačovém objevování, ale potenciálně i o metodách experimentálního bádání jako takového. V tomto kontextu se v nedávné době zformoval zajímavý výzkumný směr (Williamson, 2001) snažící se prozkoumat souvislosti mezi strojovým učením na jedné straně a filosofií vědy na straně druhé. V tomto textu se ovšem zdržíme filosofických

úvah o zmíněných souvislostech a raději vystopujeme některé jejich důsledky okamžitého praktického významu.

V následujících dvou oddílech se nejprve zaměříme na dva významné objevy empirického zkoumání algoritmů pro řešení obtížných kombinatorických problémů, jejichž důsledky se týkají i problému konstrukce hypotéz z pozorování. Prvním z nich jsou tzv. fázové přechody v kombinatorických problémech a druhý se týká „podivných" pravděpodobnostních rozložení doby řešení úloh souvisejících s fraktální povahou prostorů přípustných řešení. Další dva oddíly se potom soustředí na plody empirických studií samotných algoritmů strojového učení. První z nich se týká empirických poznatků o užitečnosti známého principu Ockhamovy břitvy – pevného dogmatu seriózních vědeckých disciplín. Druhý souvisí s principem strojového metaučení: stejně jako základní výzkumný proces, i metaobjevitelská činnost může být v principu automatizována (obr. 12.4). V tomto případě algoritmus metaučení navrhuje hypotézy o jiných učících se algoritmech.

## 12.1 Fázové přechody

Zaměřme se nyní na typ experimentu, jehož příklad je znázorněn vpravo na obr. 12.3 a v němž je „badatelská činnost" počítače studována člověkem. První empirické studie v umělé inteligenci, které přinesly zásadní poznatky o chování algoritmů na reálných problémech, se netýkají výhradně strojového učení, ale širšího oboru řešení složitých kombinatorických problémů. Jak dále uvidíme, právě řešení tohoto typu problémů tvoří jádro inteligentních algoritmů pro konstrukci hypotéz ve srozumitelném symbolickém jazyce.

Algoritmy pro řešení problémů, jak toto téma chápe umělá inteligence, mají společný celkem prostý princip. I když jsou konkrétní reálné problémy rozličných povah, ve formalizaci umělé inteligence jde o to se $n$-krát správně rozhodnut mezi $m$ danými možnostmi („ingrediencemi řešení") a těmito volbami návrh na řešení „sestavit" (blíže viz Umělá inteligence (1)). Zda bylo konkrétní rozhodnutí správné ovšem vyplyne, až když je $n$-krát rozhodnuto a lze zjistit, zda je tak problém opravdu vyřešen. Naivnímu algoritmu *generuj a testuj* tak nezbude než vyzkoušet všech $m^n$ možností úplných návrhů řešení. S rostoucím $n$, tedy velikostí problému, je naivní algoritmus tímto exponenciálním nárůstem zcela diskvalifikován. A právě od inteligentního algoritmu se očekává, že již v časné fázi rozhodování „vytuší", jakou má která konkrétní volba naději na úspěch, a bude tedy zkoušet mnohem méně než $m^n$ řešení, a to především těch nadějných.

Problémy charakterizované exponenciálním nárůstem kandidátů na řešení, u nichž však lze efektivně (v čase stoupajícím s $n$ pouze polynomicky) ověřit, zda je kandidát opravdu správným řešením, patří do tzv. časově-složitostní třídy NP.[1] Do NP patří například problém splnitelnosti výrokových klauzulí, který se označuje jako SAT (z angl. satisfiability). Zadány jsou klauzule, tedy disjunkce jako například $v_1 \vee v_2 \vee v_3$ nebo $v_1 \vee v_3$ atd., a problém spočívá v přiřazení pravdivostní hodnoty každé proměnné $v_1, ..., v_n$ tak, aby všechny zadané klauzule byly splněny, tedy aby alespoň jedna proměnná

[1] Z angl. nondeterministic polynomial, blíže viz např. v (http://e.wikipedia.org/wiki/ Nondeterministic_polynomial_time).

v každé klauzuli měla hodnotu *pravda* (není-li negována), resp. *nepravda* (je-li negována). Prostor všech možných návrhů na řešení zde tedy má $2^n$ prvků. Jakkoliv se SAT může zdát odtažitý od „opravdových" problémů umělé inteligence, jeho význam je zásadní, neboť jakýkoliv jiný problém v NP lze efektivně *převést* právě na řešení SAT! Z toho důvodu se SAT (spolu s některými dalšími problémy) označuje jako problém NP-*úplný*.

Inteligentní algoritmus, který by jakýkoliv případ problému SAT řešil v čase stoupajícím pouze polynomicky s počtem proměnných, by tedy zároveň mohl efektivně řešit problémy obrovského dopadu ekonomického (např. plánování optimálních cest, návrh konfigurace výrobku), bezpečnostního (problém dešifrování bez privátního klíče) atd. Takový algoritmus zatím nikdo nevymyslel a všeobecně se věří (i když to není *dokázáno*), že ani neexistuje.

Skutečnost, že nelze *kterýkoliv* případ problému SAT řešit v polynomickém čase, nám ovšem mnoho neříká o typické obtížnosti problému. Jsou opravdu *všechny* případy SAT extrémně obtížné, nebo je takových jen nevýznamná většina? Na takovou otázku lze těžko odpovědět matematickým důkazem. S. Kirkpatrick a B. Selman z Cornellovy univerzity se k získání odpovědi vydali směrem v informatice dost neobvyklým – podnikli empirickou studii. Princip je triviální: *vygenerovat* velké množství náhodných případů problému SAT a zjistit, kolik a případně které z nich jsou opravdu obtížné. Ovšem jeden případ problému může být obtížný pro jeden algoritmus a snadný pro druhý. Výsledky v principu závisejí na zvoleném algoritmu a předmětem empirické studie je tedy doba běhu daného algoritmu na případech daného problému.

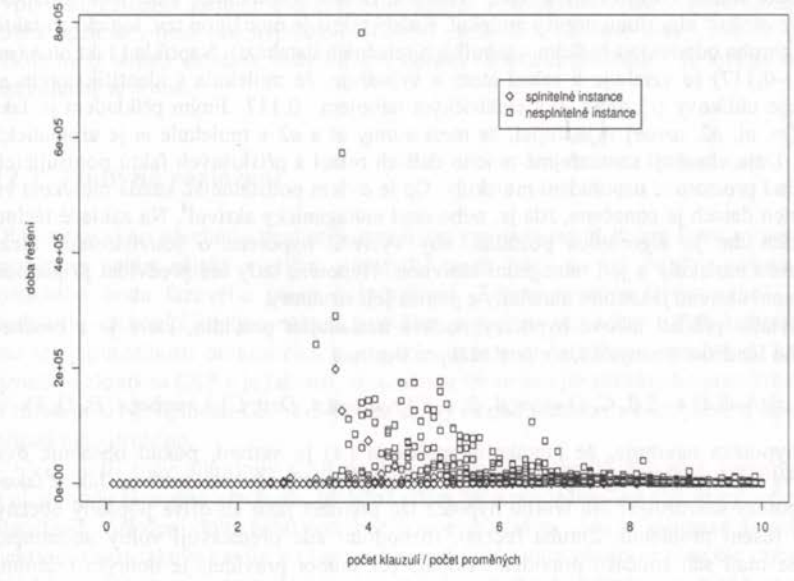
Tímto algoritmem byla tzv. Davisova–Putnamova metoda používající určitých inteligentních heuristik při řešení SAT, ale jejíž časová náročnost je v nejhorším případě stejně jako u naivního algoritmu *generuj a testuj* samozřejmě exponenciální. Studie vyústila velmi zajímavými výsledky. Za prvé, většinu případů vyřešil Davisův–Putnamův algoritmus rychle, jen výjimečně se objevily opravdu složité případy. Druhé pozorování se objeví při grafickém znázornění doby řešení[2] daného případu SAT proti veličině $c/n$, což je poměr počtu klauzulí a počtu proměnných v tomto případu. Tento graf vidíme na obr. 12.2 a plyne z něho značná nestejnoměrnost: těžké případy jsou soustředěny v okolí bodu $c/n \approx 4{,}5$.

Vysvětlení nabízí jev tzv. *fázového přechodu* v principu podobný stejnojmennému jevu ve fyzice. K jeho pochopení je třeba si uvědomit, že klauzule v SAT představují vlastně *omezení* kladená na hodnoty proměnných. Čím je omezení větší, tím je těžší najít hodnoty proměnných vyhovující všem omezením. Naopak čím je více proměnných, tím má algoritmus „volnější ruce" k tomu, aby omezením vyhověl. Na grafu vpravo od kritického bodu převládají případy s relativně velkým počtem klauzulí a malým počtem proměnných, tedy tzv. přeurčené (overconstrained) případy SAT. Je-li algoritmus dostatečně inteligentní, alespoň jako ten Davisův–Putnamův, včas „odhalí", je-li takový případ nesplnitelný.[3] Vlevo od kritického bodu se pak vyskytují zejména případy podurčené (underconstrained). Takové případy mají obvykle spoustu správných řešení a algoritmus má velkou šanci rychle alespoň jedno najít. Ale právě okolo kritického bodu nastává přechod mezi fázemi přeurčenosti a podurčenosti, v němž je často extrémně obtížné jak najít správné řešení, tak zjistit, že je daný případ SAT nesplnitelný.

[2] měřené počtem kroků uskutečněných algoritmem v prohledávacím prostoru

[3] Přesněji řečeno, na přeurčených případech vzroste efektivita strategií prořezávání stavového prostoru.

Vzhledem k převoditelnosti všech problémů NP na SAT se také všech problémů NP týká objevený jev fázového přechodu. V principu lze tedy z popisu případu libovolného problému NP zjistit, jak je blízko oblasti fázového přechodu, a odhadnout tak, zda se jedná o opravdu obtížný případ. Objev má tedy poměrně dalekosáhlé důsledky. Není tedy překvapením, že byl zveřejněn prestižním časopise Science (Kirkpatrick a Selman, 1994), viz též populárnější výklad v (Iiayes, 1997).




Obr. 12.2 Fázový přechod. Každý bod přísluší řešení náhodného případu problému SAT heuristickým Davisovým–Putnamovým algoritmem. Kosočtverečky znázorňují splnitelné případy, čtverečky případy nesplnitelné. Vertikální osa odpovídá době řešení, horizontální potom určenosti problému (vlevo: přeurčené, vpravo: podurčené, uprostřed: oblast fázového přechodu s ostrým nárůstem průměrné doby řešení). Testování hypotéz v predikátové logice lze za určitých podmínek převést na problém SAT. Podle empirických studií (Botta a kol., 2003), (Serra a kol., 2001) systémy induktivního logického programování typicky generují hypotézy, jejichž SAT protějšky leží právě v oblasti fázového přechodu.

Důsledky jevu fázového přechodu sahají i do strojového učení už z toho důvodu, že mnohé jeho algoritmy se musejí potýkat s NP-úplnými problémy. Jedno ze „zjevení" fázových přechodů ve strojovém učení je ovšem zvláště zajímavé a ovlivnilo náhled na obecné principy konstrukce hypotéz z dat.

Jak již bylo zmíněno, systémy induktivního logického programování konstruují hypotézy v predikátové logice (viz první svazek UI), nejčastěji v její podmnožině, tzv. relační logice. Ta jim umožňuje popsat komplexní vlastnosti objektů a zejména jejich vzájemné vztahy (relace). Tvorba hypotéz v predikátové logice se v jednom zásadním aspektu liší od běžnějších metod strojového učení založených pouze na atributových

hodnotách objektů (viz např. kapitola o strojovém učení v prvním svazku). Jazyk predikátové logiky je totiž dostatečně silný (expresivní) na to, aby zakódoval jakoukoliv teorii či algoritmus z libovolného jiného jazyka (např. přirozeného jazyka vědecké disciplíny, programovacího jazyka atd.). Konstrukce hypotéz v predikátové logice má tedy poměrně blízko k jejich vyjadřování při přirozeném bádání. Jazyk predikátové logiky využívá například robotický vědec představený v úvodu kapitoly. Konkrétní ukázkou automatické formulace teorií v predikátové logice v přírodních vědách je například studie (Srinivasan a kol., 1996) z oblasti biochemie. Zde jsou vstupem „objevovacího" algoritmu popisy molekul. Každý popis je množinou tzv. logických faktů (fakta zhruba odpovídají řádkům v tabulkách relačních databází). Například fakt *atom*(*m*, *a*1, *c*, –0,117) se vztahuje k relaci atom a vyjadřuje, že molekula s identifikátorem *m* obsahuje uhlíkový (*c*) atom *a*l s elektrickým nábojem –0,117. Jiným příkladem je fakt *vazba*(*m*, *a*l, *a*2, *arom*) vyjadřující, že mezi atomy *a*l a *a*2 v molekule *m* je aromatická vazba. Data obsahují samozřejmě mnoho dalších relací a příslušných faktů popisujících například prostorové uspořádání molekuly. Co je ovšem podstatnější, každá molekula ve vstupních datech je označena, zda je, nebo není mutagenicky aktivní[4]. Na základě těchto vstupních dat je algoritmus požádán, aby vytvořil hypotézu o souvislostech mezi strukturou molekuly a její mutagenní aktivitou. Hypotéza tedy má předvídat případnou mutagenní aktivitu jakékoliv molekuly z popisu její struktury.

Triviální příklad takové hypotézy podává následující pravidlo, které je z biochemického hlediska nesmyslné, ale poslouží pro ilustraci

$$\forall A\ aktivní(A) \leftarrow \exists B, C, D\ atom(A, B, c, C) \wedge atom(A, D, c, C) \wedge vazba(A, B, D, 1).$$

Tato hypotéza navrhuje, že kterákoliv molekula (*A*) je aktivní, pokud obsahuje dva uhlíkové (*c*) atomy (*B*, *D*) stejného náboje (*C*) spojené jednoduchou vazbou. Jak se takové hypotézy konstruují? Na tvorbu hypotéz lze pohlížet jako na dříve popsaný obecný případ řešení problémů. Zhruba řečeno, rozhodnutí zde představují volby podmínek, které se mají stát součástí pravidla. Pravidlo (či soubor pravidel) je dobrým řešením, pokud správně klasifikuje, tedy jeho podmínky jsou splněny právě pro ty molekuly, které jsou skutečně mutagenní.

Při hledání hypotéz v predikátové logice však volba správných podmínek nepředstavuje jediný problém. Samotné testování, zda aktuálně uvažovaný soubor podmínek platí pro konkrétní příklad v datech (v našem příkladě pro popis chemické molekuly), je docela složitá úloha. To je zřejmé už na první pohled – pokud se pravidlo zmiňuje o *n* atomech (v našem příkladě $n = 2$), je třeba zkoušet různé *n*-tice atomů v dané molekule, dokud pro některou neplatí předpoklady pravidla, nebo dokud nejsou vyzkoušeny všechny. Při splnění určitých obecných podmínek[5] je takový test NP-úplným problémem. Tedy každý pár *příklad* × *pravidlo* je převoditelný na případ problému SAT se všemi z toho vyplývajícími důsledky, včetně jevu fázového přechodu. Lze tedy očekávat, že fázové přechody složitosti budou hrát svou roli i v testování hypotéz.

Skutečně, nedávné studie (Serra a kol., 2001) a (Botta, 2003) toto očekávání potvrdily. Ty zkoumaly chování algoritmů induktivního logického programování použitých na velkých souborech náhodně generovaných problémů učení právě se zvláštní pozorností

[4] To jest potenciálně může poškodit genetický materiál. Mutagenita má silnou souvislost s karcinogenitou.

[5] Ty zaručují *rozhodnutelnost*, tedy, že pro takový test lze vůbec sestrojit algoritmus.

k jevu fázového přechodu a dospěly k zajímavému zjištění. Heuristické prohledávání prostoru možných hypotéz algoritmem ILP obvykle skončí v oblasti těsného okolí bodu fázového přechodu složitosti testování těchto hypotéz. Jinými slovy, oblast fázového přechodu složitosti tvoří jakýsi atraktor pro trajektorie prohledávání hypotézového prostoru. V rámci teorie fázových přechodů byl tak vlastně zformulován celkem přirozený princip. Hypotézu v oblasti podurčenosti, která si klade velmi málo podmínek, je sice snadné otestovat, ale nejspíš nebude příliš užitečná – v našem příkladě by z ní vyplývalo, že téměř každá molekula je mutagenní. Analogická úvaha, i když na „opačné straně spektra", platí pro hypotézy v oblasti přeurčení. Kvalitní tedy budou hypotézy někde mezi těmito extrémy, tedy blíže k oblasti fázového přechodu. Ty ovšem nebývá jednoduché testovat.

## 12.2 Podivná rozložení

Oblast fázového přechodu není atraktorem jen pro systémy ILP, ale i pro výzkumníky snažící se nalézt nějaký výstižný statistický popis toho, co se „děje" v oblasti okolí kritického bodu fázového přechodu složitosti. Zajímavá práce (Frost a kol., 1997) podrobila statistické analýze některé populární algoritmy pro řešení NP-úplného problému tzv. splnitelnosti omezujících podmínek (constraint satisfaction problem), který se označuje zkratkou CSP a je jakousi zobecněnou formulací již zmíněného problému SAT. Vzhledem k NP-úplnosti CSP a SAT lze každý případ jednoho z nich převést na nějaký případ toho druhého.

Ve studii byly algoritmy CSP použity pro řešení případů náhodně vzorkovaných z okolí bodu fázového přechodu, přičemž cílem bylo pro jednotlivé algoritmy určit jejich statistické rozložení doby běhu potřebné k nalezení řešení. Zde je zajímavé si uvědomit neortodoxnost takové studie v rámci informatiky. V tomto oboru se obvykle doba běhu nějakého algoritmu vyšetřuje zkoumáním jeho vnitřních principů, z nichž vyplynou asymptotické odhady doby běhu v nejhorším případě. Zkoumat typické chování algoritmu empiricky, tedy jako „černou skříňku", podobně jako jsou obvykle zkoumány přírodní jevy, může informatikovi připadat nedostatečně seriózní. Jak ale vzápětí uvidíme, je-li takový výzkum dostatečně systematický a jeho cíl přesně formulován, výsledky mohou o algoritmech prozradit mnoho nového a užitečného.

Statistickým modelem doby běhu je tzv. kumulativní distribuční funkce $F(t)$, pro kterou platí, že $F(t) = Pr(T \leq t)$ je odhadnutá pravděpodobnost, že daný algoritmus nalezne řešení náhodně vybrané CSP instance v nějakém čase $T$ nejvýše rovném $t$. V zájmu nezávislosti na konkrétním hardwaru počítače se čas obvykle měří jako počet nějakých základních kroků algoritmu, u nichž lze očekávat, že trvají zhruba stejně dlouho. Zajímavý závěr studie algoritmů CSP v oblasti fázového přechodu se netýká přímo rozložení $F(t)$, ale veličiny $h(t) = f(t)/(1 - F(t))$, kde $f(t) = \mathrm{d}F(t)/\mathrm{d}t$ je tzv. hustota pravděpodobnosti. Význam $h(t)$ je následující: pokud není řešení případu problému ještě nalezeno v okamžiku $t$, pak $h(t)\Delta t$ je pravděpodobnost, že bude nalezeno v intervalu mezi $t$ a $t + \Delta t$. Ve většině případů studovaných v (Frost a kol., 1997) $h(t)$ se stoupajícím $t$ nejprve roste a potom klesá.

Čím je tento závěr zajímavý? Sám o sobě ničím, jeho význam vyplývá až z toho, že zůstane v platnosti i při jisté „zrcadlové změně" v experimentálním režimu. V původním experimentu byl jeden pevný algoritmus použit pro řešení náhodně vzorkovaných případů problému. Nyní zafixujeme jeden náhodně zvolený případ problému a pro jeho řešení použijme postupně „náhodně vzorkované algoritmy". To jednoduše znamená, že je na jednom případu problému opakovaně spuštěn původní algoritmus, který ovšem obsahuje navíc náhodnou složku. Tou může být například náhodné rozhodování mezi volbami se stejnou heuristickou kvalitou, obecně pak takové znáhodnění, které neoslabuje heuristiky algoritmem používané. Z tohoto experimentu lze odhadnout rozložení $F(t)$ a funkci $h(t)$ stejně jako z původního uspořádání. Skutečnost, že hodnota $h(t)$ i zde od určitého $t$ trvale klesá, má nyní následující důsledek. Užitečná strategie by měla nyní přirozeně spočívat v tom, že znáhodněný algoritmus po jisté době $t$, kdy nenalezl správné řešení a kdy $h(t) < h(0)$, zastavíme a spustíme znovu od začátku.

Taková znáhodněná restartovací strategie odporuje intuici hned ze dvou důvodů. Za prvé se nezdá přirozené, že by znáhodněním deterministického principu bylo možné jakkoliv vylepšit efektivitu. Za druhé, restartovat algoritmus znamená samozřejmě i vymazat jeho pamět obsahující například všechny již vyhodnocené kandidáty na řešení, čímž je zapomenuta určitá „vybudovaná" informace o problému.

Strategie byla ale podrobně prozkoumána rozsáhlými empirickými studiemi (Gomes a kol., 2000) a její úspěšnost potvrdil fakt, že s ní byly vyřešeny případy kombinatorických problémů, jejichž řešení nebylo do té doby nalezeno žádnou jinou metodou. Carla Gomes (Gomes a kol., 2000) má pro tento jev empirické vysvětlení: řešení mnoha typů složitých reálných kombinatorických úloh algoritmy umělé inteligence vykazují tzv. mocninná rozložení doby běhu $F(t) \approx ct^{\alpha}$, $0 < \alpha < 2$, $c > 0$. Tato rozložení jsou „podivná" v tom smyslu, že hodnota $1 - F(t)$ klesá s $t$ mnohem pomaleji než exponenciálně, tedy než „obvyklá" rozložení, např. rozložení normální. Graf této funkce má vpravo (pro velká $t$) tzv. „těžký ohon" (heavy tail). Tento typ rozložení je znám například z ekonomie jako přibližný model rozdělení příjmů ve společnosti, z oblasti modelování chaotických jevů a fraktálů (Mandelbrot, 1983) nebo tzv. bezeškálových grafů, kterým je například Internet (Barabási, 1999). Zmíněná „podivná" rozložení doby řešení úloh potom souvisejí se skutečností, že z kombinace dostatečně složité kombinatorické úlohy s dostatečně inteligentním algoritmem (tj. takovým, který používá silných heuristik) vyplývá prohledávací prostor řešení často s fraktální strukturou (Gomes, Selman, 1999).

Mocninná rozložení vykazují některé poutavé vlastnosti. Jednou z nich je jejich nekonvergující střední hodnota pro určité hodnoty $\alpha$. Očekávaná doba běhu algoritmu s takovým časovým rozložením je tedy nekonečná. Přesto lze ale obvykle očekávanou hodnotu „stlačit" na konečnou hodnotu obyčejným restartováním algoritmu! Stačí jakkoliv malá šance, že algoritmus najde řešení v nějakém konečném počtu kroků $k$, tj. $F(k) > 0$, a rozložení (v jednotkách počtu restartů $N$) pro příslušný algoritmus restartovaný vždy po $k$ krocích je $F_k(N) = 1 - (1 - F(k))^N$. To je skutečně exponenciální rozložení s konečnou střední hodnotou.

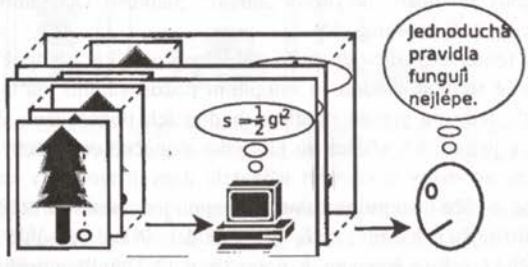
Je přirozené očekávat, že tyto závěry budou mít přímé důsledky v otázkách složitosti konstrukce hypotéz, podobně jako tomu bylo v případě fázových přechodů. Zda se podivnými rozloženími vyznačuje dříve popsaný NP-úplný problém testování hypotézy, a tedy zda by bylo možné jej řešit efektivněji náhodným restartovaným algoritmem,

zatím nebylo zodpovězeno. Věc by ovšem měla přinejmenším jeden háček: takový algoritmus by sice mohl v průměru rychleji dokazovat, že hypotéza danému pozorování vyhovuje, ale bez dalších úprav je neúplný. To znamená, že ať poběží jakkoliv dlouho bez výsledku, nikdy nebude jisté, že hypotéza danému pozorování nevyhovuje.

Důsledky empirického výzkumu znáhodněných restartovaných algoritmů ovšem do strojového učení pronikly. Nikoliv na úrovni testování hypotéz, ale v souvislosti se samotným problémem jejich hledání. Studie (Železný a kol., 2003) totiž ukázala, že rozložení s „těžkým ohonem" za určitých podmínek charakterizují i algoritmy prohledávání prostoru hypotéz v predikátové logice, jejichž čas je měřen v jednotkách počtu testovaných hypotéz, a znáhodněným restartováním prohledávacího algoritmu lze dosáhnout až řádového zvýšení efektivity.

## 12.3 Ockhamova břitva

Zatím jsme se zabývali souvislostmi mezi výsledky empirických výzkumů algoritmů umělé inteligence a procesem automatického formování hypotéz, a to především z hlediska složitosti, resp. efektivity tohoto procesu. Zůstaňme ještě u typu experimentů znázorněném na obr. 12.3. Vedle časové složitosti tvorby hypotéz mohou tyto experimenty odhalit další, ještě důležitější aspekty týkající se konkrétně predikční přesnosti formovaných hypotéz. Tak v biochemickém příkladu v dřívější části textu by predikční přesnost byla měřena počtem správně, popř. nesprávně, klasifikovaných molekul, které nebyly součástí vstupních pozorování pro tvorbu hypotézy – tzv. trénovacích dat. Algoritmus konstruující hypotézy s vysokou predikční přesností má dobrou schopnost zobecňovat z konkrétního vzorku pozorování na obecnější principy. Experimentální uspořádání z obr. 12.3 umožnilo empiricky vyhodnotit relevanci popu-




Obr. 12.3 Člověk zkoumající automatickou tvorbu hypotéz algoritmy strojového učení jako empirický jev. Tento typ výzkumu přinesl například nové pohledy na užitečnost principu tzv. Ockhamovy břitvy a zejména zhodnocení jeho relevance k predikční přesnosti hypotéz. Obecně empirický výzkum algoritmického řešení složitých kombinatorických v umělé inteligenci vedl k důležitým objevům, jako jsou fázové přechody v NP-úplných problémech, či zvláštní pravděpodobnostní rozložení doby běhu některých heuristických algoritmů na velmi složitých problémech. Tyto objevy následně vedly k lepšímu pochopení chování některých algoritmů strojového učení.

lární heuristiky využitelné při tvorbě hypotéz – tzv. Ockhamovy břitvy[6] – k zobecňovacím schopnostem „objevovacích" algoritmů.

Strojové učení se tradičně drží volné interpretace Ockhamova dogmatu podle níž, pokud je k dispozici více hypotéz stejně nebo podobně konzistentních s pozorováními, je třeba zvolit tu nejjednodušší – například tu s nejméně parametry, nejméně pravidly, obecně takovou, jíž lze vyjádřit nejkratším algoritmem. Zaujetí pro jednoduchost je na jednu stranu motivováno usnadněním interpretace hypotézy člověkem. Na druhou stranu je také všeobecně přijímáno, že za stejných ostatních podmínek dosáhne jednodušší hypotéza vyšší predikční přesnosti než hypotéza složitější.

Na první pohled má druhá, silnější interpretace zastání na několika frontách. Předními advokáty jsou tzv. princip kompromisu mezi odchylkou a variancí (bias-variance trade-off) (Hastie, 2001) známý ze statistické teorie učení a teorie tzv. PAC-učení (probably aproximately correct learning) (Mitchell, 1997). Oba z těchto formalismů v podstatě dokazují princip, který lze neformálně vyjádřit následovně. Příliš velká „svoboda" při výběru hypotéz, tedy příliš velká množina, z níž je hypotéza volena[7], jde na úkor očekávané predikční schopnosti zvolené hypotézy, a to v důsledku zvýšeného nebezpečí tzv. přeučení (overfitting (Hastie, 2001)). Ovšem tyto principy ve skutečnosti nijak Ockhamovu tezi nedokazují, neboť v zásadě velikost (kapacita) hypotézového prostoru nemusí nijak souviset s popisnou složitostí hypotéz, které obsahuje.

Z teorie tzv. solomonovské „univerzální predikce" (Li a Vitanyi, 1997), která vychází z principů kolmogorovovské složitosti algoritmů a konceptu tzv. algoritmické pravděpodobnosti, jsou odvozeny některé široce užívané techniky v induktivní inferenci, např. princip minimální popisné délky – minimum description length (Li a Vitanyi, 1997). Rozšířenou představou je, že touto teorií je Ockhamovo pravidlo rigorózně odůvodněno. Ve skutečnosti tato teorie sama Ockhamovův princip předpokládá tím, že postuluje apriorní pravděpodobnost algoritmů (které zde zastupují hypotézy) klesající s jejich (kolmogorovovskou) složitostí. Jakýkoliv „důkaz" platnosti Ockhamovy teze z tohoto teoretického rámce je tedy tautologický.

Přes chybějící teoretické zdůvodnění je stále dosti obvyklé předpokládat, že ze dvou navržených teorií se stejnou shodou se vstupními pozorováními má ta jednodušší větší šanci „na úspěch", tedy na přesné predikce budoucích pozorování. Preference jednoduchých hypotéz v přírodních vědách se jeví jako skutečně osvědčený recept. Ta může ale vyplývat pouze ze snahy o komfort uživatele daných teorií (ty komplikované jsou méně srozumitelné a hůře manipulovatelné než jejich jednodušší protějšky) než z opravdové tendence složitějších hypotéz hůře předpovídat. Právě rozsáhlé empirické studie procesů automatické tvorby a testování hypotéz (obr. 12.3) mohou poskytnout dostatečný vzorek k posouzení, zda zvyšováním složitosti nad nezbytnou mez trpí predikční přesnost.

---

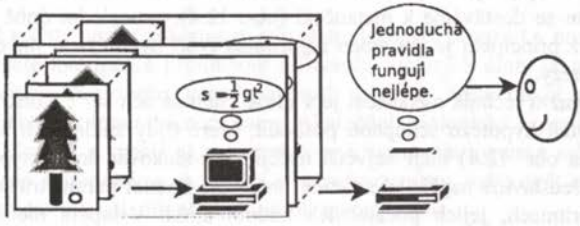
[6] „Numquam ponenda est pluralitas sine necessitate", tj. „nikdy nepostuluj mnohost bezdůvodně" – teze anglického logika Williama z Ockhamu (14. stol.) interpretovatelná jako „je-li více řešení, vyber to nejjednodušší". V případě hledání vysvětlení pozorovaných jevů se mnohost týká počtu předpokladů, které hypotéza zavádí.

[7] Přitom u nekonečných množin hypotéz, se kterými běžně strojové učení pracuje, se přílišnou velikostí míní přílišná kapacita této množiny, měřená tzv. Vapnikovou–Červoněnkisovou dimenzí (Vapnik, 1995).

P. Domingos (Domingos, 1995) podává přehled empirických indicií plynoucích ze zkušeností s algoritmy pro konstrukci hypotéz, a to jak pro Ockhamovo pravidlo, tak proti němu. Na straně „pro" autor uvádí tradiční argumenty:

- Techniku prořezávání klasifikátorů často ve strojovém učení používanou pro vylepšení predikční přesnosti. Prořezávaní spočívá ve zjednodušení popisu klasifikační hypotézy, obykle za cenu snížení přesnosti na trénovacích datech.
- Obvykle dobrou predikční přesnost hypotéz generovaných stabilními algoritmy. Stabilita zde zhruba znamená, že pro různé vzorky pozorování téhož jevu bude algoritmus konstruovat podobné hypotézy. Podobnost hypotéz zde není myšlena v deskriptivním smyslu, ale jako míra shody jejich předpovědí. Stabilní algoritmy cíleně snižují varianční složku již zmíněného kompromisu odchylky a variance a obecně se má za to, že stabilní algoritmy tvoří jednoduché hypotézy.
- Vysokou predikční přesnost velmi jednoduchých hypotéz (zde pravidel) zjištěnou (Holte, 1993) na některých populárních „soutěžních úlohách" strojového učení a data-miningu.

Podle P. Dominga jsou tato klasická „poučení" irelevantní k principu Ockhamovy břitvy. Úspěchy technik prořezávání a stabilních algoritmů jsou ve skutečnosti důsledkem omezení velikosti hypotézového prostoru a je jen technickým detailem, že prostor je obvykle omezen právě na jednoduché hypotézy. Pokud se týká velmi často citované studie (Holte, 1993), ta je podle P. Dominga nesprávně interpretována: hlavní závěr článku („velmi jednoduchá pravidla fungují dobře...") je v rozporu se skutečnými číselnými výsledky experimentů této studie.




Obr. 12.4 Strojové metaučení: algoritmus tvoří hypotézy o jiných algoritmech tvořících hypotézy. Metaučení se poměrně výrazně uplatnilo v praxi data-miningu.

Spojování jednoduchosti hypotéz s vyšší predikční přesností naopak některé empirické výzkumy zpochybňují: (Domingos, 1995) podává přehled experimentů různých autorů shodujících se na závěru, že jednoduché hypotézy nevykazují o nic vyšší přesnost, než ty složité. Například práce (Quinlan, Cameron-Jones, 1995) uvádí příklady postupného zvětšování hypotézového prostoru, jehož prohledávání vede k výsledným hypotézám, které jsou postupně stále jednodušší a méně přesné.

Některé nedávno úspěšné metody strojového učení zmíněné „predikční" interpretaci Ockhamovy břitvy doslova odporují. Jedná se zejména o metody *souborné klasifikace* (ensemble classification) (Hastie a kol., 2001) spočívající v kombinování většího množství klasifikačních hypotéz – jejich predikce jsou kombinovány například prostřednictvím „hlasování", průměrováním jejich číselných predikcí apod. Princip „zvážení

mnoha možných hypotéz" za účelem predikce obecně snižuje variační složku kompromisu pro odchylku a varianci. Souborné metody přitom zavádějí „nadbytečnou" složitost, a to hned dvakrát. Za prvé je pro kvalitní souborný klasifikátor třeba dostatečné variability mezi jeho jednotlivými složkovými klasifikátory. Čím vyšší je jejich přípustná složitost (i nad mez nutnou pro shodu se vstupními pozorováními), tím vyšší variability lze dosáhnout. Za druhé je výsledný soubor klasifikátorů samozřejmě deskriptivně složitější než jeho složkové klasifikátory. Výmluvná ilustrace je podána v (Domingos, 1997), kde jsou soubory klasifikátorů převáděny na ekvivalentní jednosložkové (nesouborné) klasifikátory, které jsou ve většině případů mnohem složitější než jednotlivé složky původního souboru, a přitom mají vyšší predikční přesnost.

Empirické studie algoritmů strojového učení na reálných úlohách konstrukce hypotéz z pozorování tedy naznačují, že interpretace Ockhamovy břitvy jakožto nástroje ke zvýšení predikční schopnosti hypotéz je mylná, a zdá se, že jediným odůvodněným účelem jejího použití je zlepšení srozumitelnosti teorií pro člověka. Snad ještě zajímavější je, že výzkum strojového učení ukázal případy, kdy je v zájmu predikční přesnosti tvořit hypotézy *nadbytečně* složité, a tak bez skrupulí Ockhamův princip porušovat – úkon na hranici rouhání v tradičních vědách!

## 12.4 Metaučení

V kontextu empirického výzkumu algoritmického objevování je velmi významnou skupina technik vyplývající z propuštění lidského badatele z obr. 12.3 a jeho nahrazení počítačem. Tím se dostáváme k metaučení (obr. 12.4), v poslední době aktivně zkoumanému, jehož principem je, že jeden algoritmus tvoří hypotézy o jiných algoritmech tvořících hypotézy.

Tradiční využití technik metaučení je v úloze „učit se učit se"[8]. Konkrétním cílem je obvykle vyvinout hypotézu schopnou posoudit, které třídy základních algoritmů (těch v rámečcích na obr. 12.4) mají největší naději vyprodukovat dobré hypotézy. Mírnou záměnu cíle představuje například navržení intervalů hodnot parametrů použitých v základních algoritmech, jejich počátečních hodnot apod. Vstupem metahypotézy jsou obvykle metadata, která popisují charakter dat základních (např. počet a datové typy veličin v každém pozorování, počet pozorování apod.).

Příkladem použití metaučení je například článek (Kalousis a kol., 2004). Zde autoři používají metaalgoritmus pro vytvoření tříd („shluků") základních algoritmů. Tyto třídy shromažďují algoritmy s podobnou výkonností na datech daného charakteru. Pomocí metahypotézy pak uživatel pro svůj konkrétní problém identifikuje tu nejlepší třídu a z ní potom zvolí svůj oblíbený algoritmus. Příkladem zmíněné variace úlohy pro určení parametrů je (Soares a kol., 2004), když metahypotéza navrhuje jeden důležitý parametr v základním algoritmu mechanismu podpůrných vektorů (support vector machine) – jedné z populárních metod statistického strojového učení.

Vedle těchto základních přístupů k metaučení se objevily i některé méně ortodoxní strategie související s tématy dříve diskutovanými v oddílech 12.1 a 12.3. Konkrétně

[8] Na rozdíl od Uljanovovy výzvy zde nejsou slovesa spojena souřadně.

práce (Maloberti, Sebag, 2004) je zajímavá svým přístupem k popisu základních dat. Namísto obvyklých metadatových deskriptorů (viz uvedené příklady) používají autoři s úspěchem některé odvozené veličiny, které popisují polohu problému vzhledem k oblasti fázového přechodu. Autoři ukazují, že takto získaná informace je vysoce relevantní k volbě nejkompetentnějšího základního algoritmu pro daný problém. V práci (Bensusan. 1998) se zase metaalgoritmus učí rozpoznat, jak extenzivně má základní algoritmus používat techniky prořezávání, tedy do jaké míry by na daném základním problému měla být „uplatněna" Ockhamova břitva.

Zcela nedávným příkladem metaučení uzavíráme v jistém smyslu vazbu mezi empirickým výzkumem induktivního učení na jedné straně a deduktivního uvažování na straně druhé. Vliv naposledy jmenovaného na strojové učení jsme diskutovali v oddílech 12.1 a 12.2. Technika tzv. *usměrňování* (streamlining) (Gomes, Sellmann, 2004) je naopak příkladem využití učení pro obecné řešení kombinatorických problémů. Na základní úrovni zde vystupuje algoritmus pro řešení daného problému a strojové učení je použito až na metaúrovni: příslušný algoritmus objevuje podstruktury často se objevující v řešeních nalezených základním algoritmem. Ty jsou pak zpětně na základní úroveň převedeny ve formě *heuristik* pro účely následného řešení složitějších případů problému: základní algoritmus přednostně vyhodnocuje potenciální řešení, která obsahují podstruktury nalezené na metaúrovni.

## 12.5 Závěr

Exponenciálně rychlý nárůst výkonnosti výpočetní techniky umožnil v posledních letech algoritmům umělé inteligence proniknout do natolik složitých úloh, jako je badatelská činnost. Algoritmy strojového učení navrhují hypotézy vysvětlující experimentální pozorování a některé k tomu dokonce samy uskutečňují biologické experimenty. Umělointeligentní uvažování, a speciálně automatizované formování hypotéz, může samo o sobě být podrobeno empirickému zkoumání ať už přirozenému, nebo opět automatickému. V textu jsme představili několik příkladů studií tohoto druhu.

Významnost jejich výsledků naznačuje, že empirické metody výzkumu chování systémů strojového učení, a obecně algoritmů umělé inteligence na rozsáhlých kolekcích reálných problémů, mají potenciál odhalit o nich mnohé z toho, co je před tradičními deduktivními metodami informatiky ukryto.

### Literatura

Barabási A-L.: *Emergence of scaling in random networks*. Science, 286, 1999, s. 509–512.

Bensusan H.: *God doesn't always shave with Occam's razor – learning when and how to prune*. In: Proceedings of the 10th European Conference on Machine Learning, Springer-Verlag, 1998, s. 119–124.

Botta M., Giordana A., Saitta L., Sebag M.: *Relational learning as search in a critical region*. Journal of Machine Learning Research, 4, 2003, s. 431–463.

Domingos P.: *Why does bagging work? a Bayesian account and its implications*. In: 3rd International Conference on Knowledge Discovery and Data Mining,. AAAI Press, 1997, s. 155–158.

Domingos P.: *The role of Occam's razor in knowledge discovery*. Data Mining and Knowledge Discovery, 3, 1999, s. 409–425.

Frost D., Rish I., Vila L.: *Summarizing CSP hardness with continuous probability distributions*. In: Matwin S., Sammut C., (eds), Proceedings of the Fourteenth National Conference on Artificial Intelligence, AAAI Press, 1997, s. 327–334.

Gomes C. P., Selman B.: *On the fine structure of large search spaces*. In: 11th IEEE International Conference on Tools with Artificial Intelligence, IEEE Computer Society, 1999, s. 197–201.

Gomes C. P., Sellmann M.: *Streamlined Constrained Reasoning*. In: Proc. of 10th Conf. on Principles and Practice of Constraint Programming, 2004.

Gomes C. P., Selman B., Crato N., Kautz H. A.: *Heavy-tailed phenomena in satisfiability and constraint satisfaction problems*. Journal of Automated Reasoning, 24(1/2), 2000, s. 67–100.

Hastie T., Tibshirani R., H. Friedman J.: *The Elements of Statistical Learning, Data Mining, Inference and Prediction*. Springer, 2001.

Holte R. C.: *Very simple classification rules perform well on most commonly used datasets*. Machine Learning, 20, 1993, s. 93–91.

Iiayes B.:. *Can't get no satisfaction*. American Scientist, 85(2), 1997, s. 108–112.

Kalousis A., Gama J., Hilario M.: *On data and algorithms: Understanding inductive performance*. Machine Learning. Special Issue on MetaLearning. 54, 2004.

King R. D., Whelan K. E., Jones F. M., Reiser P. K. G., Bryant C. H., Muggleton S. H., Kell D. B., Oliver S. G.: *Functional genomic hypothesis generation and experimentation by a robot scientist*. Nature, 427, 2004, s. 247–252.

Kirkpatrick S., Selman B.: *Critical behavior in the satisfiability of random boolean expressions*. Science, 234, 1994, s. 1297–1301.

Li M., Vitanyi P.: *An Introduction to Kolmogorov Complexity and Its Applications*. McGraw-Hill, 1997.

Maloberti J., Sebag M.: *Fast theta-subsumption with constraint satisfaction algorithms*. Machine Learning, 55(2), 2004, s. 137–174.

Mandelbrot B.: *The Fractal Geometry of Nature*. Freeman, New York 1983.

Mitchell T.: *Machine Learning*. McGraw-Hill, 1997.

Quinlan J. R., Cameron-Jones R. M.: *Oversearching and layered search in empirical learning*. In: Proceedings of the 11th International Joint Conference on Artificial Intelligence, 1995, s. 1019–1024.

Serra A., Giordana A., Saitta L.: *Learning on the phase transition edge*. In: Proceedings of the 17th International Joint Conference on Artificial Intelligence, s. 921–926. Morgan Kaufmann, 2001.

Soares C., Brazdil P., Kuba P.: *A meta-learning method to select the kernel width in support vector regression*. Machine Learning, Special Issue on Meta-Learning, 54, 2004.

Srinivasan A., Muggleton S., Sternberg M. J. E., King R. D.: *Theories for mutagenicity: A study in first-order and feature-based induction*. Artificial Intelligence, 85(1–2), 1996, s. 277–299.

Vapnik V. N.: *The Nature of Statistical Learning Theory*. Springer, 1995.

Williamson J.: *Machine learning and the philosophy of science: a dynamic interaction*. In: ECML'2001 Workshop on Machine Leaning as Experimental Philosophy of Science, 2001.

Železný F., Srinivasan A., Page D.: *Randomised Restarted Search in ILP*. Machine Learning 64(1–3), Springer, 2006, s. 183–208.

http://en.wikipedia.org/wiki/Nondeterministic_polynomial_time

# 13 Vliv prostředí v posilovaném učení

## 13.1 Úvod

Je známo, že zvláštnosti prostředí ovlivňují chování, které si lidé a zvířata vytvoří (Brooks, 1991). Tento vliv neovlivňuje pouze způsob, *jakým* zvířata provádějí určité činnosti, ale určuje také *které* činnosti jsou důležité k přežití a také které činnosti budou součástí repertoáru činností zvířete, jenž řeší nejběžnější a nejdůležitější úlohy. Při laboratorních pokusech se může krysa například naučit, jak má zatáhnout za páčku, za což je odměněna potravou (a svou úspěšnost časem zlepšovat). Stejná krysa se může naučit nalézt únikovou cestu z bludiště (a při každém pokusu bude rychlejší), pokud se nachází uvnitř bludiště a potrava je umístěna vně bludiště. Jak je však možné, že se krysa tyto dovednosti naučí? Je možné, aby se krysa naučila svému chování sama, pouhou interakcí s prostředím, vedena pomocí svých smyslů a stimulována svým potravním instinktem?

Proces učení využívající interakce s prostředím se nazývá *posilované učení* (Reinforcement Learning, RL). Tento typ učení byl důkladně zkoumán a psychologické učení nazvané *behaviorismus* jej prohlašuje za hlavního původce vývoje a formování chování lidí a zvířat. Významnými představiteli behavioristického učení byli zejména Ivan P. Pavlov, který zkoumal klasické podmiňování, Edward Thorndike, který postuloval tzv. „zákon efektu" (law of effect), John B. Watson, který odmítal introspektivní metody a pokoušel se omezit psychologii na experimentální metody, a také B. F. Skinner, který se zabýval výzkumem operačního podmiňování.

Za prvního behavioristu lze považovat Ivana P. Pavlova (1849–1936), ruského fyziologa a psychologa. V prvních dekádách 20. století vyslovil principy, které nyní známe pod názvem *klasické podmiňování*: Máme-li (přirozený vztah) mezi podnětem a odezvou (nepodmíněný reflex), můžeme odezvu spojit s dalším neutrálním podnětem tak, že mezi těmito dvěma podněty vytvoříme vazbu, čímž se vyvine podmíněný reflex. Nejslavnějším příkladem klasického podmiňování je vznik slinicího reflexu Pavlovových psů. Pokud je psům předložena potrava, začnou psi spontánně produkovat sliny. Proto Pavlov vazbu mezi podnětem (potrava) a nepodmíněnou odezvou (produkce slin) nazval nepodmíněný reflex. Vytvořil předpoklad, že pokud bude u psa při podávání potravy opakovaně přítomen další podnět, potom bude tento podnět spojován s potravou a bude mít u psa za následek produkci slin. Americký psycholog Edward Thorndike (1874–1949) šel ještě dále a při studiu *volního* chování a pravděpodobnosti jeho výskytu zavedl termín *operační podmiňování* (operand conditioning) a postuloval svůj známý „zákon efektu": Odezvy, které jsou úzce spjaty s pocitem uspokojení, se s touto situací asociují

a pokud se bude situace později opakovat, nastanou znovu s vyšší pravděpodobností. Pokud jsou naopak odezvy spojeny s nepříjemnými následky, asociace s touto situací bude oslabena. Burrhus Skinner (1904–1990) tyto myšlenky radikalizoval a jeho učení je nazýváno *radikální behaviorismus* (radical behaviorism). Vyvinul metodu nazvanou aplikovaná analýza chování – odvětví psychologie, které se snaží vyvinout jednotný rámec pro zvířecí a lidské chování, který je založen na principech učení. Zkoumal také formování lidského chování pomocí operačního podmiňování.

Robotika se myšlenkou *učení robota*, tj. vybavení robotů možností naučit se novým dovednostem, přizpůsobit se prostředí a postupně se zlepšovat, vážně zabývá a považuje ji za nezbytnou pro vyvinutí skutečně autonomních robotů. V tomto kontextu se RL (posilované učení) jeví velice zajímavě, protože poskytuje rámec pro učení pomocí interakce s prostředím a pro objevování nových a neznámých vztahů mezi různými podněty a reakcemi, přičemž jediným požadavkem je dostupnost vhodného posilování. Takové použití je naštěstí možné, protože teorie posilovaného učení byla matematicky modelována pomocí nástrojů dynamického programování a řadí se k metodám strojového učení v umělé inteligenci (Sutton, Barto, 1998).

V terminologii (posilovaného učení) lze říci, že při každém kroku učení se agent pokusí provést akci a na základě získaných výsledků získá určité *posílení* (reinforcement). Posílení může být ve formě odměny nebo trestu a je výsledkem *hodnocení* (critic). Hodnocení je často modelováno funkcí zvanou *posilovací funkce* (reinforcement function). Cílem učicího procesu je nalezení optimální *strategie* (asociace mezi situacemi a akcemi) pomocí odhadu budoucích posilování. V tomto kontextu je posilovací funkce jediným elementem celého systému, který má informace o úloze, kterou se má systém naučit.

Existuje však rozdíl mezi uvedeným příkladem a typickým problémem posilovaného učení: Odměna (kladné posílení), vyjádřená jako pocit uspokojení, je získána v obou případech, když krysa pozře potravu, a nikoliv v okamžiku, kdy se jí povede zatáhnout za páčku nebo nalézt cestu ven z bludiště. Na tomto příkladu lze ukázat, že charakteristika prostředí určuje, které dovednosti budou naučeny. Skinner tento fenomén jednoznačně identifikoval a použil k důkazu, že většina aspektů chování, které jsou považovány za „vrozené", jsou ve své podstatě naučené operačním podmiňováním pomocí vrozeně posilovaného podnětu. Tento fakt vyvodil z experimentů prováděných s kachními mláďaty: jakmile kachní mládě opustí vejce, začíná následovat svou matku. Mládě kachny můžeme v laboratorních podmínkách snadno přimět k následování libovolného jiného objektu (například krabice od bot), zejména v případě, že se velikost a barva objektu podobá dospělému jedinci kachny. Pokud však mládě kachny vložíme do přístroje, který „obrací směr pohybu" (pohybuje například krabicí pryč, když se kachna blíží, a přibližuje krabici, když se kachna vzdaluje), mládě se snadno naučí odlišnému způsobu pohybu, který mu dovolí udržovat správnou vzdálenost. V tomto případě zřejmě není chování „následovat matku" vrozeným chováním, ale „bytí v blízkosti matky" je důsledkem vrozeného pozitivního posilování.

Existuje mnoho dalších úloh, pro které nejsou v prostředí k dispozici specifické posilovací podněty, ale zvířata se je i přesto naučí. Jak takový proces učení probíhá? Je posilované učení v těchto kontextech stále použitelné? Lidé a zvířata mohou vytvářet nová posilování na základě předešlých zkušeností: pokud je těsně před posilováním opakovaně přítomen neutrální podnět, stane se tento podnět sám o sobě posilováním. Tento nový druh posilování se nazývá *sekundární* nebo *podmíněné* podmiňování a hraje

významnou roli při vývoji dítěte. Malé děti si velmi rychle spojí posilování se zákazy nebo ústupky jiných osob, s láskou či nenávistí nebo se slovy „Správně!" či „Špatně!" (Keller, 1969). Tento fenomén byl již dříve rozpoznán Pavlovem v experimentu s produkcí slin u psů. Hraje významnou roli v procesu, který je znám jako *řetězení*, v němž se postupně vytváří složité posilovací vztahy, a v němž jsou zdánlivě nesouvisející podněty spojeny dohromady, což vede k velmi rozdílným výsledkům.

Jak jsme se již zmínili, psychologové a biologové (zejména behavioristické školy) studovali mnoho konceptů a jevů týkajících se myšlenky posilovaného učení. Ale až na základní elementy nebyla většina z nich nikdy použita ve strojovém učení. V této kapitole se zaměříme na dva koncepty: na vliv prostředí na vývoj různých chování při stejném posilování a na automatickou konstrukci sekundárních posilování. Naším cílem je zahrnout je systematickým a formálním způsobem do rámce posilovaného učení a vytvořit tak robustní metody učení, které jsou méně závislé na externích specifikacích.

## 13.2 Hypotézy a předběžný důkaz

### 13.2.1 Vliv prostředí

V této části navrhneme dvě hypotézy a popíšeme sérii jednoduchých experimentů, kterými je ověříme. Uvedené hypotézy explicitně vyjadřují myšlenky učení dovedností, které je ovlivňováno prostředím a vnitřně vytváří posilovací funkce.

*Hypotéza 1*: Posilované učení lze při pevně stanovené posilovací funkci, která určuje obecnou úlohu, použít k učení různých dovedností pouhou změnou charakteristiky prostředí.

*Hypotéza 2*: Pomocí pevně stanovené posilovací funkce, která platí pro obecnou úlohu a strategii, jež řeší úlohu v určitém prostředí, lze pro určitou dovednost sestavit posilovací funkci.

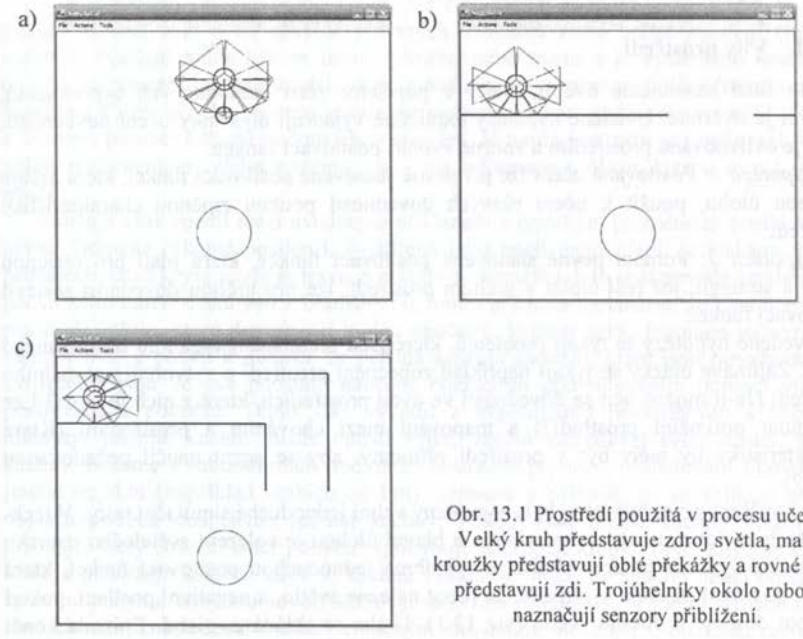
Uvedené hypotézy se týkají problémů, které jsou předmětem výzkumu posilovaného učení. Zajímavé otázky se týkají například zobecnění prostředí a vytvoření optimálního prostředí (Je-li možné učit se dovednosti ve dvou prostředích, které z nich je lepší? Lze navrhnout optimální prostředí?) a mapování mezi chováním a prostředími (Které charakteristiky by měly být v prostředí přítomny, aby se agent naučil požadovanou úlohu?).

Pro ověření první hypotézy byly vytvořeny velmi jednoduché simulační testy. V těchto testech je potrava nahrazena světlem a hlavní úlohou je nalezení světelného paprsku z libovolné výchozí pozice. Úloha je vyjádřena jednoduchou posilovací funkcí, která přiřadí pozitivní posílení v případě, že robot nalezne světlo, a negativní posílení, pokud se robot dostane z dosahu světla (viz 13.1). Úloha se skládá z epizod. Epizoda končí v případě nalezení světla nebo v případě, že se robot nachází mimo dosah světla déle než po dobu 10 cyklů

$$r\,f(s) = \begin{cases} 100{,}0 & \text{když vzdálenost od zdroje světla} \le K_{\min} \\ -1{,}0 & \text{když vzdálenost od zdroje světla} \ge K_{\max} \\ -0{,}1 & \text{v ostatních případech}, \end{cases} \tag{13.1}$$

kde $K_{min}$ a $K_{max}$ jsou prahové hodnoty vzdálenosti od zdroje světla. Proces učení byl spuštěn se shodným algoritmem posilovaného učení a se stejnými parametry ve třech různých prostředích: v prostředí s oblými překážkami, v prostředí se stěnou s otvorem uprostřed a v prostředí s chodbou (viz obr. 13.1). Ve všech pokusech byl použit algoritmus posilovaného učení s názvem Sarsa ($\lambda$) (Sutton a Barto, 1998), který umožňuje rychlejší učení agentů v případě, že je posilování zpožděno (což je tento případ). Robot je schopen měřit vzdálenost a úhel vzhledem k světelnému zdroji (pomocí sedmi různých hodnot pro úhel a osmi různých hodnot pro vzdálenost). Navíc je vybaven devíti dálkoměrnými senzory pro měření hodnot, z nichž lze vypočítat vzdálenost od nejbližšího objektu.

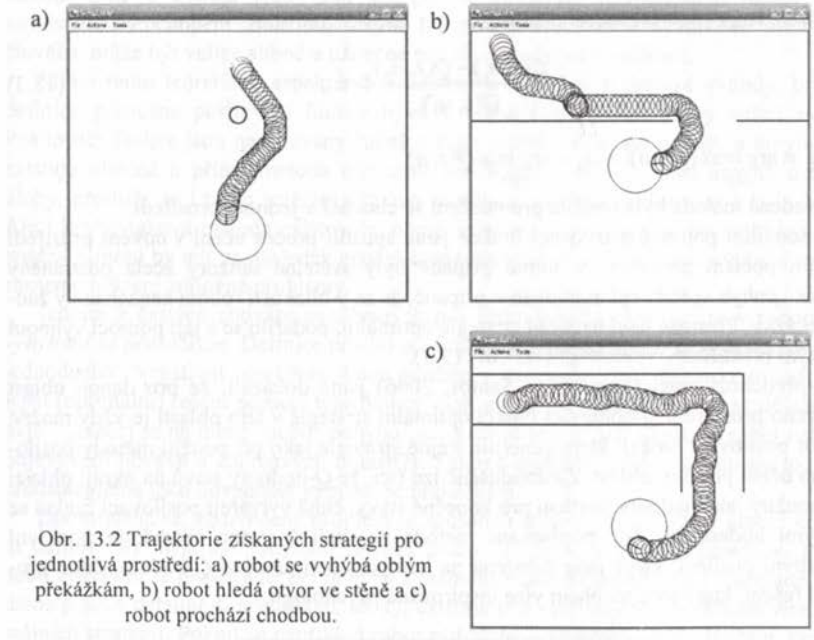
Po skončení procesu učení o 50 000 iteracích jsou chování naučená v jednotlivých prostředích otestována tak, že jsou spuštěna z náhodných výchozích pozic. Ve všech testech se robotu podařilo dosáhnout světelného zdroje a z většiny výchozích pozic pomocí téměř optimální cesty. Na obrázku 13.2 jsou uvedeny ukázky některých trajektorií, které byly získány pro jednotlivá prostředí.




Obr. 13.1 Prostředí použitá v procesu učení. Velký kruh představuje zdroj světla, malé kroužky představují oblé překážky a rovné čáry představují zdi. Trojúhelníky okolo robota naznačují senzory přiblížení.

Podíváme-li se na tyto obrázky ještě před znalostí experimentu, můžeme si myslet, že robot z obr. 13.2(a) zná způsob, jak se vyhýbat překážkám, robot z obr. 13.2(b) umí nalézt otvor ve stěně a projít jím a robot z obr. 13.2(c) umí procházet chodbou. Ale v posilovací funkci se nenachází nic takového, co by tyto dovednosti definovalo. Zde se nabízí první otázka: Odkud pochází ona informace, která je potřebná k naučení se dovednosti? Protože s výjimkou definice prostředí byly všechny tři pokusy nastaveny

zcela stejně, lze učinit následující závěr: *Charakteristika prostředí a vztah mezi posilovací funkcí a prostředím implicitně obsahuje informace, které jsou k naučení dovednosti potřeba.* Pokud souhlasíme s předchozí větou, objevuje se před námi výzva: Bylo by možné tuto informaci izolovat a vyjádřit ji explicitně? Anebo ještě lépe – bylo by možné pro naučenou dovednost vytvořit specifickou posilovací funkci? Jedná se o dosud nevyřešenou otázku, která je v současné době předmětem dalšího výzkumu.




Obr. 13.2 Trajektorie získaných strategií pro jednotlivá prostředí: a) robot se vyhýbá oblým překážkám, b) robot hledá otvor ve stěně a c) robot prochází chodbou.

### 13.2.2 Automatická konstrukce posilovací funkce

Tento odstavec popisuje návrh metody pro automatickou tvorbu posilovací funkce z ohodnocení optimální strategie. Zmíněná metoda byla použita k definici posilovací funkce pro naučení se dovednosti *vyhýbat se překážkám*. Získaná posilovací funkce byla použita v různých prostředích. Ohodnocovací funkce přiřazuje pro každý stav budoucí posilování, které agent získá v případě, že je aktuální strategie využívána od začátku do daného stavu. Posilovací funkci vytvoříme z ohodnocovací funkce nalezením kritických dvojic (stav, akce) podle následujících pravidel:

- Pokud je nejlepší akce pro daný stav mnohem lepší než druhá nejlepší akce, bude s touto dvojicí (stav, akce) spojeno kladné posílení.
- Pokud je nejhorší akce pro daný stav mnohem horší než nejlepší akce, bude s touto dvojicí (stav, akce) spojeno záporné posílení.
- Ostatním dvojicím (stav, akce) bude přiřazeno neutrální posílení.

Formálně vyjádřeno, pozitivní posílení je přiřazeno dvojici $(s, a_1)$ v případě, že

$$\frac{Q(s,a_1)-Q(s,a_2)}{Q(s,a_2)} > k\,, \tag{13.2}$$

kde $a_1 = \arg\max_{a\in A} Q(s,a)$ a $a_2 = \arg\max_{a\in A-\{a_1\}} Q(s,a)$. Negativní posílení je pak přiřazeno dvojici $(s, a_0)$ v případě, že

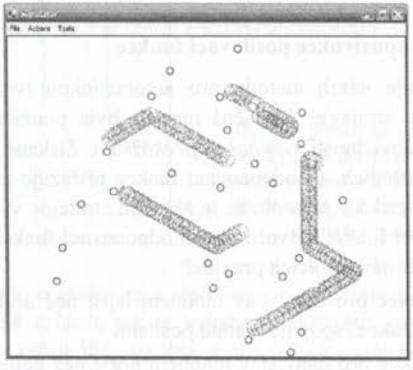
$$\frac{Q(s,a_1)-Q(s,a_0)}{Q(s,a_0)} > k\,, \tag{13.3}$$

kde $a_1 = \arg\max_{a\in A} Q(s,a)$ a $a_0 = \arg\min_{a\in A} Q(s,a)$.

Uvedená metoda byla použita pro naučení se chování v jednom prostředí.

S použitím popsané posilovací funkce jsme spustili proces učení v novém prostředí s větším počtem překážek. V tomto případě byly světelné senzory zcela odstraněny a reflex „pohyb vpřed" byl aktivován v případě, že se v blízkosti robota nenacházely žádné překážky. Přestože není naučená strategie optimální, podařilo se s její pomocí vyhnout některým překážkám v prostředí (viz obr. 13.3).

V předchozí práci (Bendersky, Santos, 2006) jsme dokázali, že pro danou oblast stavového prostoru a $Q$-hodnoticí funkci optimální strategie v této oblasti je vždy možné sestavit posilovací funkci, která generuje stejné strategie jako při použití metody posilovaného učení pro tuto oblast. Zjednodušeně lze říci, že $Q$-hodnoty stavů na okraji oblasti jsou použity jako hodnoty posílení pro konečné stavy, čímž vytvářejí posilovací funkci se spojitými hodnotami. Zde popisovaná metoda na druhou stranu produkuje pozitivní a negativní posílení, která jsou založena na komparativní analýze $Q$-hodnot, čímž produkují řešení, která jsou mnohem více inspirovaná psychologií.



Obr. 13.3 Vykreslené stopy strategie vyhýbání se překážkám, která byla naučena pomocí automaticky sestavené posilovací funkce.

## 13.3 Motivace a diskuse

Pokud je posilované učení použito k učení robotů, je nutný určitý zásah člověka, který specifikuje, které úlohy mají být učeny. Pro každou tuto úlohu musí být určeno, co je úspěchem a co bude považováno za neúspěch. Jinými slovy, člověk-návrhář posilovací funkce musí pro každou úlohu zjistit, které situace a akce by měly být posilovány a jakou měrou. Lidé a zvířata jsou však schopni se některým dovednostem naučit zcela samostatně. Pochopení způsobu, jakým lze použít posilované učení bez intervence člověka, může být velice slibné a užitečné pro různé robotické aplikace.

Mimo tento teoretický aspekt má uvedená metoda jisté technické výhody, protože definice příslušné posilovací funkce bývá v případě netriviální úlohy velmi složitá. Posilovací funkce jsou navrhovány ručně a často laděny metodou pokusu a omylu. Neexistuje obecná a přímá metoda odvození posilovací funkce z vyšší úrovně definice úlohy, přestože se i tímto směrem výzkum zaměřuje, například (Bonarini a kol., 2001). Ale i kdyby taková metoda existovala, popis úlohy ve vyšším jazyce by mohl být víceznačný a mohl by mít za následek neočekávané chování. V následujícím výkladu stručně shrneme některé zmíněné problémy.

Jedním z častých chování používaných pro testování algoritmů učení v robotice je *vyhýbání se překážkám*. Definice posilovací funkce se pro tuto úlohu na první pohled zdá jednoduchá: Negativní posilování bude použito v okamžiku, kdy dojde ke kolizi. Nejlepší (optimální) řešení pomocí této funkce však může být „zůstaň stát, ať se děje, co se děje", „otáčej se na místě" nebo „pohybuj se na místě sem a tam" (proč by robot riskoval objevování nových a zajímavých oblastí?). Takové chování zcela jistě nebude to, co si představujeme jako dovednost vyhýbat se překážkám.

Domníváme se, že uvedený problém je způsoben nekompletní definicí úlohy: korektní definice by měla být *vyhýbání se překážkám za současného prozkoumávání terénu* nebo *vyhýbání se překážkám při pohybu z jednoho místa na druhé*. Ale i v případě, že si dáme práci s detailnější specifikací úlohy, existuje pro vyhýbání se překážkám více optimálních strategií. Pokud se například robot přibližuje k překážce, může ji objet, nebo se může otočit zpět a jet pryč. Obě strategie jsou optimální vzhledem k definované posilovací funkci. Jakého chování chce tedy návrhář dosáhnout? Tento příklad ukazuje, že i popis posilovací funkce v přirozeném jazyce může být zavádějící a může generovat neočekávané chování.

Druhý problém nastane v okamžiku, kdy se funkci pokusíme formalizovat. V případě reálných robotů obsahuje informace ze senzorů rušení, je nejistá, nekompletní a někdy s příliš malou amplitudou. Použít tuto informaci ve vysokoúrovňových konceptech, které slouží k vyjádření posilovací funkce v přirozeném jazyce, není vůbec jednoduché. Některé přístupy se pokouší tento problém řešit parametrizací posilovací funkce, automatickým laděním parametrů v průběhu učení (Santos, 1999) a (Ng a kol., 1999) a formalizací posilovací funkce na základě tzv. konfiguračního prostoru (Bonarini a kol., 2001).

Další potenciální problém je způsoben překladem posilovací funkce z definice na vyšší úrovni do definice, která je založena na senzorech agenta – může existovat více způsobů mapování. Kompletní pozorovatelnost prostředí nebývá možná, a proto nemusí

být agent schopen rozlišit různé situace. S tím souvisí jev nazývaný *falešný vjem* (Matuk, Santos, 2001), který může způsobit následující situace:

1. Stejné akce provedené ve stejné (stejně snímané) situaci mohou dávat různé výsledky.
2. Pokud nelze rozlišit správnou a chybnou akci či situaci od jiné, bude posilovací funkce ovlivňovat více situací a akcí, než bylo zamýšleno.

Navíc v některých případech má robot k dispozici pouze lokální informace. To bývá často případ autonomních robotů, kteří se spoléhají pouze na informace získané vlastními senzory. Úlohy je možné snadněji vyjádřit v pojmech globální informace; definovat posilovací funkci na základě lokálních informací nemusí být vždy snadné, viz například obr. 13.2(b). V tomto prostředí musí robot s infračervenými senzory překonat zeď tak, že projde otvorem. Jak však vyjádřit tuto úlohu pomocí posilovací funkce na základě lokálních senzorů?

Závěrem lze říci, že definovat vhodnou posilovací funkci pro danou úlohu může být velmi složité. Pokud by se robot uměl naučit různým dovednostem pomocí obecné posilovací funkce a pečlivého návrhu prostředí a pokud by uměl vytvářet nové posilovací funkce na základě svých zkušeností, měli bychom k dispozici mocný nástroj pro vývoj autonomních robotů s komplexnějšími schopnostmi.

Na druhé straně i jiní autoři studovali vliv prostředí v procesu učení a získaná chování. Jette Randlov předvedl konvergenci algoritmů posilovací funkce k optimální strategii v případě, že tranzitní funkce (formální reprezentace interakce agenta a prostředí) je upravována spojitě a konverguje k výsledné funkci (Randlov, 2000). Andreas Matt navrhl modifikaci algoritmu posilovaného učení, která umožnila současné učení stejné úlohy v různých prostředích, kdy získal jedinou strategii, která je optimální ve všech těchto prostředích (Matt, Regensbirger, 2003). Sebastian Thrun popisuje metodu *kontinuálního učení*, ve které je dynamika prostředí naučena v době, kdy se agent učí řešit určitou úlohu (Thrun, Mitchell, 1995). Zatímco se agent učí jinou úlohu, je získávaná informace použita k urychlení učení. I přes zmíněné práce však dosud neexistují žádní následovníci studie našich hypotéz a jejich důsledků.

## 13.4 Závěr

V této kapitole byl popsán vliv prostředí na získávání nových dovedností a schopností lidí a zvířat. Ukázali jsme způsob, kterým může prostředí ovlivnit učené dovednosti, a způsob, jakým jsou prováděny. Na druhé straně lidé i zvířata mohou spojovat odměnu s novým podnětem na základě předešlé zkušenosti a na základě spojení předešlých příčin a následků.

Tyto koncepty byly zahrnuty v rámci posilovaného učení jako vliv prostředí na učené dovednosti, které nejsou popsány v posilovací funkci, a jako vliv automatické konstrukce nových funkcí posilování, přičemž byly formulovány dvě pracovní hypotézy.

Naše hypotézy jsme ověřili pomocí řady jednoduchých pokusů s robotickým simulátorem. Při těchto pokusech se robot učil různým dovednostem (vyhýbání se oblým překážkám, nalezení otvoru, průchod otvorem ve stěně a průchod chodbou) pomocí stejného algoritmu učení a stejné posilovací funkce, ale s měnící se charakteristikou prostředí.

Po provedení těchto pokusů jsme na základě informace získané z funkce ohodnocení naučené strategie navrhli metodu konstrukce posilovací funkce pro tyto dovednosti. Tato metoda se skládá z identifikace kritických dvojic (stav, akce) a sestavení posilovací funkce přiřazením pozitivní nebo negativní hodnoty těmto dvojicím. Přestože se jedná o předběžný návrh metody, kterou je nutné vylepšit a formalizovat, získané posilovací funkce lze s uspokojivými výsledky použít k naučení vyhýbání se překážkám v jiném prostředí.

Předběžné výsledky ukazují význam a praktickou užitečnost ověření našich hypotéz pro syntézu chování autonomních robotů. Náš současný výzkum je zaměřen na některé problémy, které jsou úzce spjaty se zde uvedenými hypotézami, například: s mapováním mezi chováními a prostředími, se zobecněním a definicí částečných předpisů nad prostředími a s konstrukcí optimálních prostředí pro učení určité úlohy. V současné době se také pokoušíme definovat formálnější a systematičtější metodu pro extrakci posilovací funkce z funkce ohodnocení optimální strategie.

## Literatura


Bendersky D., Santos J. M.: *Learning from the enviroment with a universal reinfor-cement function.* in Proccedings of the 4th International Conference on Neural Networks and Artificial Intelligence (ICNNAI 2006), Brest, Bělorusko, 2006, s. 182–186.

Bonarini A., Bonacina C., Matteuci M.: *An approach to the design of reincement function in real world agent-based applications.* IEEE Transaction on Systems, Man, and Cybernetics, Part B, vol. 31, no. 3, 2001, s. 288–301.

Brooks R. A.: *Intelligence without reason.* in Proceedings of the 12th international Joint Conference on Artificial Intelligence (IJCAI-91), John Myopoulos and Ray Reiter, eds, Morgan Kaufmann publishers Inc. San Mateo, USA. Sydney, Austrálie, 1991, s. 569–595.

Keller F. S.: *Learning: Reinforcement Theory.* Random House, New York, 1969.

Matt A., Regensburger G.: *Reinforcement Learning for Several Environments: Theory and Applications.* Ph.D. thesis, University of Innsbruck, 2003.

Matuk R., Santos J.: *The clustering aliasing problem in reinforcement learning for robots.* In: Proceedings of the Fifth European Workshop on Reinforcement Learning, Utrecht, Nizozemsko, 2001, s. 33–35.

Ng A., Daishi Harada D., Russell S.: *Policy invariance under reward transformations: theory and application to reward shaping.* in Proceedings of the 16th International Conference on Machine Learning., Morgan Kaufmann, San Francisco, CA, 1999, s. 278–287.

Randlov J.: *Shaping in reinforcement learning by changing the physics of the problem.* In: Proceedings of the Seventeenth International Conference on Machine Learning, 2000.

Santos J. M.,: *Contribution to the study and the design of reinforcement functions.* Ph.D. thesis, Universidad de Buenos Aires, Universite d'Aix-Marseille III, 1999.

Sutton R. S., Barto A. G.: *Reinforcement Learning: An Introduction.* MIT Press, Cambridge, MA, 1998.

Thrun S., Mitchell T.: *Lifelong robot learning.* Robotics and Autonomous Systems, vol. 15, 1995, s. 25–46.

# 14 Jazyk a umělá inteligence: kudy a kam?

## 14.1 Úvod

Jistě není sporu o tom, že k lidské inteligenci neodmyslitelně patří porozumění přirozenému jazyku. Nilssonův (Nilsson, 1980) názor, že umělá inteligence je ta část matematické informatiky, která se zabývá projektováním inteligentních počítačových systémů, tj. systémů, „které vykazují vlastnosti spojované s inteligentním chováním – porozumění jazyku, učení, argumentování, řešení problémů atd.", tedy není nic překvapivého. Je také pochopitelné, že snahy o vytvoření počítačových systémů, které by pracovaly se vstupem (popř. i výstupem) v přirozeném jazyce, a pokusy o jejich vyhodnocování z hlediska jejich „inteligence" se datují od konce první poloviny minulého století. Připomeňme tu jen Weaverovo memorandum Translation z dopisu Weavera Wienerovi z března r. 1947 (Weaver, 1949), Turingovu úvahu o tom, zda stroje mohou myslet (Turing, 1950), nebo Weizenbaumovu Elizu (Weizenbaum, 1960) ukazující, že není příliš složité oklamat Turingův test na inteligenci, pokud je systém vybaven důmyslně uplatněnými (a od lidí „odkoukanými") strategiemi plytké konverzace (v jeho příkladu konverzace psychiatra-počítače s pacientkou).

Rovněž není překvapivé, že uvedené úvahy, ale i relativní úspěch prvních pokusů, především se strojovým překladem, vyvolaly velká očekávání a naděje podporované hlavně tiskem. Ke chvále lingvistů je třeba říci, že ti, kteří se prvních pokusů zúčastnili (i jako autoři), zachovávali postoj spíše rezervovaný. Počítačové porozumění přirozenému jazyku je úkol nesmírně složitý a zdaleka nejsme (ani nemůžeme být) tak daleko, jak by naznačoval film *2001: Vesmírná Odysea*, v němž HAL na příkaz astronauta Davida, aby otevřel jakýsi poklop, říká: „*I'm sorry Dave, I'm afraid I can't do that.*"

Složitost porozumění přirozenému jazyku můžeme však ilustrovat i na mnohem jednodušších příkladech: čteme-li v novinách, že kritika českého poslance Evropského parlamentu byla ostrá, není bez znalosti užšího kontextu zřejmé, zda ostrou kritiku pronesl český poslanec, nebo zda někdo českého poslance ostře kritizoval. Podobně z pronesené věty *Michal se chtěl oženit s Američankou.* nevyrozumíme, zda jde o konkrétní americkou dívku, nebo zda jde o to, že Michalova vyvolená by měla být amerického původu. Někdy je důležité pro vyvození správné informace vzít v úvahu i větnou intonaci: následuje-li za větou *John called Mary a Republican.* věta *Then she insulted him.* s větným přízvukem na zájmenu *him*, vyrozumívá se, že nazvat někoho Republikánem je urážka; pokud je větný přízvuk na slovese *insulted* (někoho urazit), pak o takovou inferenci nejde. V češtině máme k dispozici dva tvary 2. p. zájmena *on*, přízvučný a nepřízvučný, takže bychom v prvním případě větu přeložili *A potom ona urazila jeho*,

s větným přízvukem na dlouhém tvaru zájmena *jeho*, a ve druhém případě *A potom ho ona urazila*, s přízvukem na slovese jako v angličtině, ale s užitím krátkého nepřízvučného tvaru zájmena *ho*.

Je tedy na místě se ptát, kam až teoretická i komputační lingvistika ve svém příspěvku k umělé inteligenci došla a kudy podle našeho mínění i zkušeností vede k tomuto cíli cesta. Pokusíme se toto úsilí ilustrovat na projektech a výsledcích pražské skupiny počítačové lingvistiky na MFF UK, a to ve třech oblastech:

- přípravě jazykových zdrojů (oddíl 14.2.),
- využití těchto zdrojů pro některé aplikace (oddíl 14.3),
- na příkladu jednoho integrovaného projektu týkajícího se rozpoznávání mluveného jazyka a vyhledávání informací ve velmi velkých mluvených souborech (oddíl 14.4).

## 14.2 Vytváření rozsáhlých jazykových zdrojů

Jednou z cest, jak se dovědět o daném jazyce (nebo o jazyce vůbec) co nejvíc a jak této informace efektivně využít pro počítačové zpracování, je využití co nejrozsáhlejších a z obsahového hlediska co nejbohatších jazykových zdrojů, tedy počítačových korpusů textů (ať již jde o texty psané nebo i soubory jazyka mluveného). Nejde ale jen o to shromáždit co nejvíce takových dat, ale také – a domníváme se, že především – o to tato data obohatit jejich anotováním, tj. přiřazováním hodnot gramatických či jiných (např. lexikálně sémantických) kategorií jednotlivým prvkům věty (ať již na úrovni řetězu slov nebo podkladových syntaktických struktur). Tento názor je dnes už doložen existencí či vývojem anotovaných korpusů řady jazyků (např. Penn Treebank s odvozenými korpusy jako je PropBank nebo Penn Discourse Treebank pro angličtinu, Tiger pro němčinu a v neposlední řadě i český anotovaný korpus, viz další výklad). Podmínkou takového zhodnocení ovšem je, že pro anotování je vytvořen dobře promyšlený konzistentní scénář založený na propracované lingvistické teorii; v případě českého anotovaného korpusu jde o *funkční generativní popis* navržený Petrem Sgallem jako alternativní model explicitního popisu jazyka k Chomského transformační gramatice, dále propracovávaný skupinou počítačové lingvistiky na Univerzitě Karlově (Sgall, 1967), (Sgall a kol., 1986).

Pro češtinu existuje tzv. Pražský závislostní korpus (PZK; angl. Prague Dependency Treebank), viz (Hajič, 1998). Je to označkovaný soubor textů (vybraných z Českého národního korpusu[1]), ve kterém jsou věty analyzovány na třech anotačních úrovních (viz zjednodušený příklad na obr. 14.1):

a) morfématické (*m*-rovina, s využitím přibližně 2000 značek pro hodnoty morfématických kategorií, jako jsou slovní druhy, pády, čísla atd.; zápis na této rovině má podobu řetězu),

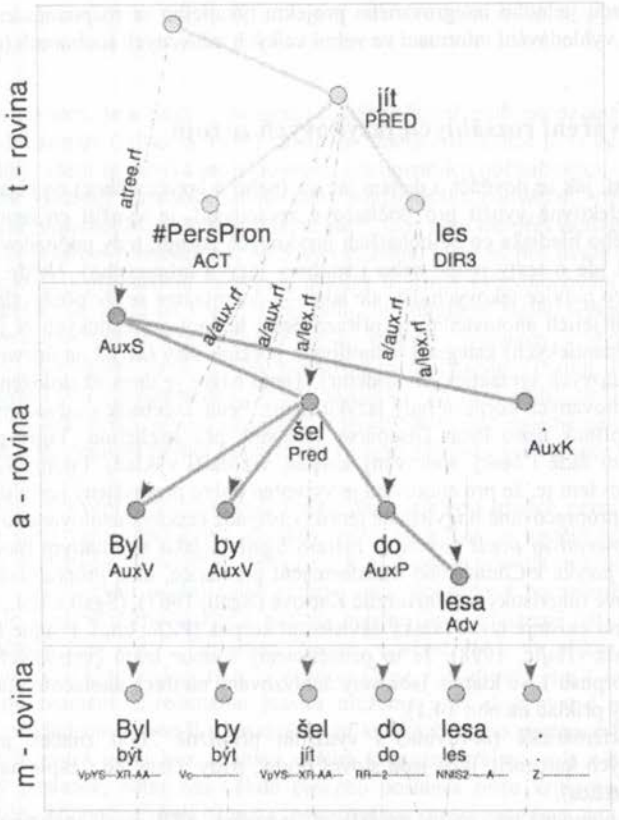
b) na pomocné tzv. rovině analytické (*a*-rovina, ATS, angl. analytical tree structure) s přiřazením závislostních stromových struktur a vyznačením základních závislostních vztahů na povrchové podobě věty,

c) na rovině tektogramatické, tedy podkladové stavby věty (*t*-rovina, TGTS, angl. tectogrammatical tree structure), s přiřazením závislostní struktury věty na podkladové rovině a s přiřazením hodnot podkladových syntaktických závislostních vztahů mezi

[1] http://ucnk.ff.cuni.cz/

autosémantickými jednotkami věty a s vyznačením aktuálního členění věty pomocí hodnot kategorie kontextové zapojenosti. K zachycení povrchové elipsy jsou vypuštěné jednotky věty v tektogramatickém stromě nově doplněny. Kromě toho je v dalším kroku anotace využito TGTS k vyznačení základních koreferenčních vztahů uvnitř věty i mezi větami v souvislém textu.

Obrázek 14.1 zachycuje přiřazení anotací na uvedených rovinách pro větu *Byl by šel do lesa*. Na *m*-rovině odpovídá dané větě řetěz slovníkových tvarů pro jednotlivá slova věty (tedy *jít* pro *šel*, *les* pro *lesa* atd.) a řetěz morfologických značek (např.

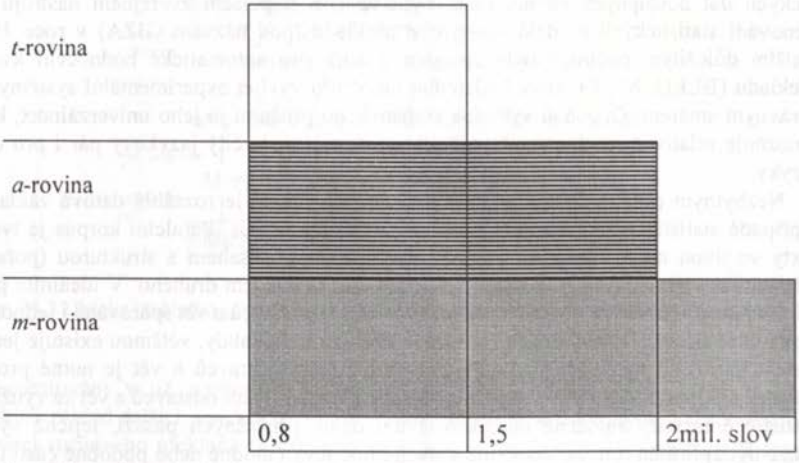



Obr. 14.1 Ukázka anotace věty *Byl by šel do lesa.* na morfematické, analytické a tektogramatické rovině.

NNIS2-----A---- pozitivní podstatné jméno rodu mužského neživotného jednotného čísla v druhém pádě pro *lesa*). Povrchová struktura věty je na *a*-rovině zachycena stromovou strukturou, v níž každému slovu ve větě (včetně interpunkčních znamének) odpovídá jeden uzel; vztahy mezi uzly jsou vyznačeny značkami, jako je například Pred pro

predikát, Adv pro příslovečné určení atd. Na této rovině není možné dodávat uzly, které neodpovídají slovům skutečně v povrchové podobě věty realizovaným, takže například pro náš příklad tu není zachycen (nevyjádřený) podmět věty. Pro zachycení významu věty je však třeba takové uzly doplnit, a tomu odpovídá zápis věty na *t*-rovině: je tu doplněn uzel pro konatele (ACT) se „zástupnou" lexikální hodnotou *PersPron* (osobní zájmeno); na druhé straně spojení *byl by šel* odpovídá jediný uzel, protože jde o složený tvar slovesa *jít*, které je řídicím uzlem celé věty. Také předložka na *t*-rovině nemá svůj vlastní uzel, ale je přiřazena k substantivu, ke kterému patří, je „skryta", ovšem spolu se substantivem tvoří celek (*do lesa*), který je k řídicímu slovesu ve vztahu určení směru (DIR3).

Hlavním cílem autorů koncepce Pražského závislostního korpusu a i jeho následného značkování (anotování) je poskytnout lingvisticky analyzovaný jazykový materiál pro studium nejrůznějších jazykových jevů v češtině, a to jak na úrovni povrchové podoby věty, tak především na úrovní její syntaktické stavby hloubkové. Objem dat druhé verze PZK[2] (Hajič a kol., 2006) anotovaných na jednotlivých rovinách dokresluje obr. 14.2.




Obr. 14.2 Rozložení počtu anotovaných slov v PZK 2.0 na jednotlivých rovinách anotace.

Anotování jazykového korpusu není samoúčelné, ale vedle toho, že poskytuje jedinečnou příležitost jak ověřovat lingvistickou teorii, umožňuje především vypracovat procedury, které by se na základě dat vytvořených experty (lingvisty, ale i za pomoci přívětivého počítačového prostředí) mohly „učit" a „naučit" analyzovat běžný text (neznámý, tedy systémem dosud neviděný) a na tomto základě rozšiřovat procedury pro porozumění textu daného jazyka. Zdá se, že v kombinaci s metodami introspektivními jde v současné době o nejperspektivnější cestu jak pokročit v návrhu systémů od robustnosti k zachycení většího bohatství struktur.

[2] http://ufal.mff.cuni.cz/pdt2.0

## 14.3 Využití počítačových jazykových zdrojů v aplikacích

### 14.3.1 Statistický strojový překlad

Jedním z možných testů, zda počítačový systém „rozumí" textu, který se mu předkládá, je „uložit" systému, aby vstupní text přeložil.

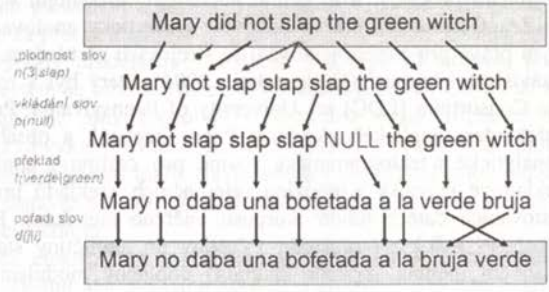
Strojový překlad je ovšem jedním z nejsložitějších a nejkomplexnějších úkolů počítačové lingvistiky. V roce 1965, po kritickém zhodnocení tehdejších výsledků v oblasti strojového překladu americkou poradní komisí ALPAC, došlo ve Spojených státech k víceméně úplnému zastavení výzkumu v této oblasti na téměř třicet let. V Evropě se strojový překlad ubíral cestou ručního vytváření syntaktických pravidel a využívání překladových slovníků obohacených o sémantické informace. V posledním desetiletí se zájem o strojový překlad zvýšil a hledaly se nové cesty jak k rychlému a zároveň pokud možno adekvátnímu překladu dojít. K řešení tohoto problému se začaly využívat metody matematické statistiky. Důvodů rozvoje statistického strojového překladu můžeme najít hned několik. Vedle prvního faktoru, kterým je neustále se rozšiřující množství elektronických dat dostupných na internetu, bylo velkým impulsem zveřejnění nástrojů pro trénování statistických modelů strojového překladu (pod názvem GIZA) v roce 1999. Dalším důležitým počinem bylo zavedení metrik pro automatické hodnocení kvality překladu (BLEU, NIST), které badatelům umožnilo vyvíjet experimentální systémy tím správným směrem. Obecnou výhodou statistického přístupu je jeho univerzálnost, která umožňuje relativně snadno použít metody vyvinuté pro určitý jazykový pár i pro další jazyky.

Nezbytným předpokladem použití statistických metod je rozsáhlá datová základna, v případě statistického strojového překladu paralelní korpus. Paralelní korpus je tvořen texty ve dvou nebo více jazycích, tyto texty si svým obsahem a strukturou (pořadím odstavců a vět) odpovídají. Klasicky je jeden text překladem druhého. V ideálním paralelním korpusu jsou kromě jednoznačného párovaní odstavců a vět spárována i jednotlivá slova či sousloví. Takový korpus je však k dispozici zřídkakdy, většinou existuje jednoznačné párování mezi celými dokumenty a párování odstavců a vět je nutné provést dodatečně, zpravidla automaticky. K automatickému párování odstavců a vět se využívají statistické metody založené na porovnávání délek příslušných pasáží, jejichž výstup může být zpřesněn tím, že vezmeme v úvahu jiné jevy (shodné nebo podobné části textu v obou jazycích, například číslice, jména, či přejatá slova) nebo informace z překladových slovníků. Paralelní korpus s vyznačenými paralelními větami slouží jako trénovací data pro překladový model.

Základní překladový model víceméně odpovídá automatickému sestavení – natrénování – pravděpodobnostního překladového slovníku slov obsažených v paralelním korpusu, tedy slovníku, ve kterém je jednotlivým překladům přiřazena pravděpodobnost, s jakou odpovídají příslušnému heslu. Parametry tohoto modelu (pravděpodobnosti párů heslo – překlad) jsou vypočítány iterativním EM-algoritmem (*estimation-maximization*). Na počátku uvažujeme v rámci každé paralelní věty všechny kombinace párování slov. Všem kombinacím je přiřazena stejná pravděpodobnost. Vytvoříme tabulku částečných součtů společných výskytů dvojic slov v paralelních větách (částečné součty získáme součtem pravděpodobností všech kombinací párování, ve kterých je tato dvojice v páru). V další iteraci algoritmu zvýhodňujeme v rámci každé paralelní věty ty kombinace

párování slov, které obsahují dvojice slov s vyššími částečnými součty, a současně vytváříme novou globální tabulku částečných součtů společných výskytů. Po několika iteracích algoritmus ukončíme a normalizací částečných součtů překladů pro jednotlivá hesla získáme pravděpodobnostní překladový slovník. Vedlejším produktem sestavování tohoto modelu může být překladový slovník, který vytvoříme odfiltrováním méně pravděpodobných překladů jednotlivých hesel.

Princip sestavování základního modelu používáme i při vytváření modelu rozšířeného o další množiny parametrů. Uvažujeme navíc tabulku změn pozic slov ve větě nebo tabulku „plodnosti" slov, která obsahuje pravděpodobnost, že z jednoho slova ve zdrojovém jazyce vznikne *n* slov v jazyce cílovém, popř. že toto slovo nebude mít ve druhém jazyce ekvivalent. Příklad ideálního párování vůči překladovému modelu mezi anglickou a španělskou větou je na obr. 14.3. Je možné stanovit i globální koeficient zkrácení či prodloužení vět při překladu. Tyto modely a metody výpočtu jejich parametrů byly navrženy začátkem devadesátých let vědci v IBM (Brown a kol., 1993), byly




Obr. 14.3 Ukázka ideálního párování překladového modelu v systému statistického strojového překladu.

implementovány v již zmiňovaném nástroji GIZA (Al-Onaizan a kol., 1999) a tvoří obvykle i základ dalších navrhovaných překladových modelů. Úspěšným pokračovatelem je model strojového překladu založeného na frázích implementovaný v nástroji Pharaoh (Koehn, 2004), který tabulku jednoslovných korespondencí nahrazuje překladovým slovníkem slovních spojení neboli frází. V současnosti jsou také podrobně zkoumány možnosti využití syntaktické analýzy a stromových struktur pro strojový překlad.

V systémech statistického strojového překladu uvažujeme kromě překladového modelu i model jazykový, který by měl zaručit, že sestavená věta je v cílovém jazyce přípustná. Takovým jazykovým modelem může být například trigramový model (model založený na tabulce relativní četnosti trojic po sobě následujících slov) sestavený na základě jednojazyčných textů z korpusu cílového jazyka; těch je většinou k dispozici větší množství než textů paralelních. Při sestavování překladového systému je třeba nalézt vhodný poměr mezi překladovým a jazykovým modelem. Lze si představit, že pokud má větší váhu překladový model, jsou výsledné překlady více doslovné vzhledem ke vstupnímu textu, preference jazykového modelu znamená volnější překlad s důrazem na čitelnost v cílovém jazyce. Vlastní překlad věty probíhá tak, že k překládané větě

hledáme větu v cílovém jazyce, která má největší pravděpodobnost vzhledem ke zvolené kombinaci překladového a jazykového modelu. Hledání nejvhodnější cílové věty vzhledem k parametrům modelů – „dekódování" – je NP-úplný problém (Knight, 1999). Proto je třeba při překladu použít aproximační algoritmy, které nejprve navrhnou omezenou množinu překladových hypotéz a optimální překlad hledají iterativně aplikací dílčích změn na této množině hypotéz za současného vyřazování vět s nižším ohodnocením.

Při experimentech je výsledek překladu ihned na výstupu hodnocen některou z automatických metod měření kvality překladu, které jsou založeny na porovnání výstupu s referenčními intelektuálními překlady téhož textu. Například metrika BLEU hodnotí, kolik skupin po sobě jdoucích slov ve výstupním překladu bylo nalezeno v referenčních překladech, delším sekvencím dává větší váhu a penalizuje případné rozdíly v délkách vět.

Strojový překlad má své místo i v české lingvistice. První systémy pro překlad z angličtiny či ruštiny vznikaly již v sedmdesátých letech, velice úspěšný byl i česko-slovenský překladový systém Česílko (Hajič, Kuboň, Hric, 2000), založený na morfematické anotaci. Čeština byla také mezi prvními jazyky, na které byly aplikovány metody statistického strojového překladu, a to přímo na letním workshopu v Baltimore, kde vznikl systém GIZA. Čeština se může pochlubit i syntakticky anotovaným paralelním korpusem určeným právě pro experimentování se strojovým překladem. Je jím Pražský česko-anglický závislostní korpus[3] (Cuřín a kol., 2004), který byl v roce 2004 vydán v Linguistic Data Consortium (LDC) při University of Pennsylvania. Pražský paralelní korpus vznikl překladem anglického korpusu Penn Treebank a obsahuje anotaci na morfematické, analytické a tektogramatické rovině pro češtinu i angličtinu, pravděpodobnostní překladové slovníky a množinu referenčních překladů pro automatickou evaluaci. Na testovacích datech tohoto korpusu, měřeno metrikami BLEU a NIST, dosahuje nejlepších výsledků při překladu z češtiny do angličtiny statistický systém založený na frázovém modelu (systém Pharaoh) doplněný modulem na obohacení českého vstupu informacemi z automatické syntaktické anotace.

### 14.3.2 Styx: elektronická cvičebnice češtiny

Využití anotovaného korpusu jinak než v rámci lingvistického výzkumu či v rámci počítačového zpracování přirozeného jazyka dokládá projekt sestavení elektronické cvičebnice češtiny[4], která čerpá věty přímo z Pražského závislostního korpusu. Pod pojmem elektronické cvičebnice chápeme souhrn vět, na kterých může uživatel procvičovat vybrané jevy s tím, že mu je posléze nabídnuto správné řešení. To vše pochopitelně „na počítači".

Dnešní žáci základních a středních škol běžně počítače používají. Hrají hry, surfují po internetu, chatují s kamarády, píšou si deník nebo malují. Snadno se nabízí otázka, proč by se prostřednictvím počítačů nemohli i vzdělávat, konkrétně v případě prezentované cvičebnice, proč by nemohli určovat morfologické kategorie slov či větný rozbor a označovat větné členy. Pochopitelně nelze očekávat, že by děti byly z této možnosti nadšené tolik jako z těch předchozích jmenovaných. Na druhou stranu však gramatiku procvičovat tak jako tak potřebují a jak doufáme, pro mnohé z nich to bude takto snazší a třeba i zábavnější než při použití tradičních tištěných cvičebnic.

[3] http://ufal.mff.cuni.cz/pcedt
[4] http://ufal.mff.cuni.cz/styx

Řada elektronických učebnic a výukových programů již existuje. Ačkoliv všechny produkty nějakým způsobem umožňují procvičovat tvarosloví a syntax, ani jeden z nich se nepřibližuje našemu cíli – vzít větu a komplexně ji z těchto dvou hledisek rozebrat. Navíc žádný z nich neobsahuje na výběr k procvičení více než několik desítek, možná stovek vět. Především však žádný z nich na úrovni syntaktické nejde dál než k určení některých (popřípadě všech) větných členů ve větě, uživateli nejsou závislosti mezi jednotlivými větnými členy ani zobrazeny (například se správným řešením), natož aby mu bylo umožněno si označování těchto závislostí procvičit.

Vyjmenované vlastnosti odrážejí způsob, jakým programy vznikaly: autoři si věty vymysleli, nebo je odněkud opsali a jednu po druhé zpracovali, určili všechny slovní druhy, jejich mluvnické kategorie, větné členy a závislosti. Bohužel takovýto postup je pro autory nesmírně pracný, náchylný k chybám. Autorům se nejspíše nepodaří dát dohromady bohatší výběr vět než několik desítek (možná stovek), ale především je vysoce pravděpodobné, že zvolené věty nebudou příliš dobře reflektovat skutečné používání jazyka – patrně budou v průměru jednodušší a kratší.

Alternativním přístupem k sestavení elektronické cvičebnice může být přístup automatický, ovšem za předpokladu, že je k dispozici anotovaný korpus. Automatický postup odstraňuje nevýhody předešlého – nejtěžší práce je již hotová, korpus existuje, je anotován a objemem se blíží k tisícům či desítkám tisíc vět. Chyby se v něm sice zajisté rovněž vyskytují, ale pravděpodobně v podstatně nižším rozsahu. Anotování korpusových vět totiž obvykle provádí více lidí najednou a šance, že se anotátoři shodnou na chybném řešení, je relativně malá. Navíc anotace procházejí celou řadou kontrol. Hlavně však korpus je sestaven tak, aby odrážel současné použití jazyka, což se následně přenese i na výslednou cvičebnici.

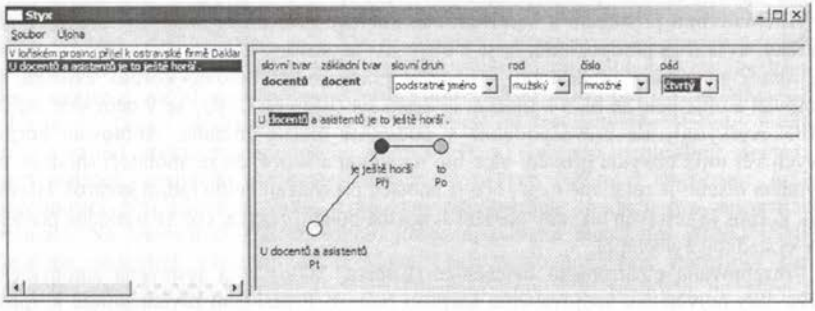
Prezentovaná elektronická cvičebnice (Kučera, 2006) byla sestavena automaticky právě díky největšímu anotovanému korpusu češtiny, Pražskému závislostnímu korpusu (PZK). Jeho využití však nemohlo být úplně přímočaré, bylo potřeba provést řadu úprav. Pražský závislostní korpus totiž obsahuje celou řadu vět nevhodných k procvičování žáků – vět obsahujících takové jevy, na jejichž klasifikacích se různé školské učebnice neshodují, či se o nich dokonce nezmiňují vůbec (ať už pro jejich komplexnost nebo pro jejich okrajovost). Takové věty tedy bylo nutné odstranit, pochopitelně automatickou cestou. Vzniklo tak několik filtrů, z nichž nejdůležitější odstraňoval všechna souvětí, neboť ani na základních, ani na středních školách se neučí jak souvětí syntakticky zpracovávat. Stejně tak byly odstraněny příliš dlouhé věty (delší než dvacet slov), protože i když se jedná technicky o věty jednoduché, mohly by svou stavbou a složitostí žáky zbytečně mást, navíc největší část takovýchto vět tvoří dlouhé řetězce shodných či neshodných přívlastků, které jsou z hlediska procvičování „nezajímavé". Obdobně byly vyřazeny i věty příliš krátké (méně než pět slov), obvykle pouze krátké nadpisy, které opět nemá velký smysl rozebírat. Z celkového počtu 49 442 vět, které do procesu filtrování vstupovaly, jich zbylo 11 705 (24 %), tedy dostatek materiálu k procvičování.

Tyto věty prošly následnou transformací do podoby, se kterou by školáci byli schopni snadno pracovat. Zatímco na úrovni morfématické to nevyžadovalo většího úsilí, bylo již předem jasné, že na úrovni analytické syntaxe bude nutné množství úprav. Analytická rovina PZK (viz *a*-rovina na obr. 14.1) se totiž v nemálo ohledech značně liší od syntaxe vyučované ve školách. V rámci stromové struktury reprezentující větné závislosti každému slovu věty (včetně interpunkce) odpovídá právě jeden uzel, zatímco ve školské repre-

zentaci může obsahovat jediný uzel slov několik. Celému uzlu (nebo chceme-li všem těmto slovům) pak přísluší pouze jeden větný člen.

Z pohledu aplikačního je systém cvičebnice tvořen programy *Charon* a *Styx*. Charon je administračním nástrojem, který slouží k zobrazování a prohlížení všech dostupných vět a k sestavení cvičení z vět. Předpokládá se, že jej bude používat vyučující. Styx je samotnou cvičebnicí, na které si budou žáci prověřovat své znalosti u cvičení vytvořených v programu Charon. Procvičování probíhá interaktivně na obrazovce počítače, jednotlivé hodnoty morfologických kategorií a větných členů jsou vybírány pomocí rozbalovacích seznamů či kontextových nabídek, určování větných rozborů probíhá přímým posouváním uzlů po pracovní ploše metodou *drag&drop* (viz obr. 14.4; uživatel procvičuje druhou větu ze cvičení o dvou větách vybraných programem Charon).

Pokud je nám známo, dosud nebylo takovéto využití anotovaného korpusu realizováno ani v českém, ani v mezinárodním prostředí. Projekt přináší velice zajímavou konfrontaci akademického a školského pohledu na přirozený, v našem případě český jazyk.




Obr. 14.4 Elektronická cvičebnice Styx.

## 14.4 Příklad integrovaného projektu: MALACH

Pojem „porozumění" (*understanding*) je mnohovrstevnatý, a to jak co do složitosti, tak co do rozsahu: od vyhledávání netriviální informace v textech až po úplné porozumění danému textu se schopností vyvozování důsledků. Jako příklad systému integrujícího schopnost „porozumění" mluvené řeči a schopnost vyhledat požadovanou informaci v rozsáhlých vícejazyčných dokumentech zde uvedeme projekt MALACH[5] (akronym pro úplný název *Multilingual Access to Large Spoken Archives*), jehož cílem je umožnit přístup k velkému mnohojazyčnému souboru audiovizuálních dat shromážděných nadací *Shoa Visual History Foundation* založené americkým režisérem Stevenem Spielbergem a obsahující 116 000 hodin digitalizovaných záznamů rozhovorů ve 32 jazycích s 52 000 přeživšími, osvoboditeli a svědky nacistického Holocaustu (Oard, 2004). Pro zajímavost uveďme, že nejvíce nahrávek je z USA – 19 841 a z Izraele – 8 504; Česko přispělo 566 a Slovensko 656 dokumenty. Tato čísla ovšem nic neříkají o jazycích dokumentů, ukazují jen, kde byly záznamy natočeny. V současné době byla Shoa Visual History

[5] http://malach.umiacs.umd.edu

Foundation přetvořena v Shoah Foundation Institute for Visual History při University of South California. Projektu se účastní několik výzkumných pracovišť: vedle zmíněného Institutu je to IBM Thomas J. Watson Research Center v Yorktown Heights, N.Y., dvě americké univerzity (Johns Hopkins University v Baltimore a University of Maryland) a Ústav formální a aplikované lingvistiky na MFF Univerzity Karlovy v Praze a Katedra kybernetiky Fakulty aplikovaných věd Západočeské univerzity v Plzni.

V projektu se řeší hned několik zásadních úkolů z oblasti zpracování přirozeného jazyka (Psutka a kol., 2003), (Psutka a kol., 2006). Především jde o automatické rozpoznávání mluvené řeči (angl. Automatic Speech Recognition, ASR), tj. o problém přepisu mluveného záznamu (audionahrávky separované z nahraného videa) do textové podoby, a to pro mnoho přirozených jazyků (angličtinu, češtinu, slovenštinu, ruštinu, polštinu, maďarštinu a postupně i další). Vzhledem k charakteru nahrávek se ukázalo, že dnes standardní metody rozpoznávání řeči nestačí, a že je třeba jednak připravit specifická tzv. trénovací data a jednak vyvinout i nové přístupy jak ke zpracování digitalizovaného audiosignálu, tak i k tzv. jazykovému modelování, kterého se v těchto postupech využívá. Uplatňují se přitom moderní metody a postupy založené na statistice a strojovém učení. Dále je třeba připravit překlady klíčových slov (tzv. tezauru, opět pokud možno automaticky nebo alespoň poloautomaticky) a nalézt metody, které dokážou tato klíčová slova přiřazovat k úsekům textu automaticky – zdá se, že vědci z historických a příbuzných oborů jsou na tento postup zvyklí. Je třeba rovněž zajistit (metodami strojového překladu), aby jak dotazy uživatelů archivu, tak i výsledné (přepsané) texty výpovědí bylo možné mezi mnoha přirozenými jazyky překládat – nejprve alespoň do angličtiny, později i mezi sebou navzájem. I zde se bude využívat řady statistických metod (viz též oddíl 14.3 o strojovém překladu). A v neposlední řadě je nutné vyvinout metody vyhledávání dokumentů a jejich segmentů v takto přepsaných a připravených textech. I zde se ukazuje, že vzhledem k tomu, že se jedná o původně mluvený záznam, je nutné obvyklé metody vyhledávání přinejmenším adaptovat. Je rovněž třeba vytvořit (manuálně) tzv. testovací anotovaná data, aby bylo možné nové a adaptované metody objektivně hodnotit.

Projekt má přitom i nesmírnou kulturní cenu. Umožňuje uchovat údaje velké historické hodnoty a následně v nich automaticky vyhledávat. Bez možnosti počítačového vyhledávání by nebylo v lidských silách tento cenný materiál zpracovat. Různí mluvčí odkazují často k téže situaci, ale z jiného úhlu pohledu, v různých jazycích i souvislostech atd. Uvádí se, že k postupnému prohlédnutí (vyslechnutí) celého archivu by byl zapotřebí skoro celý lidský život (70 let); je tedy jasné, že bez automatizované podpory je potenciál tohoto archivu nevyužitelný. Naopak, metody automatického rozpoznávání řeči, identifikace tématu, strojového překladu, přiřazování klíčových slov (indexace) a vyhledávání informací v textu po jejich plné implementaci v tomto archivu vytvoří dosud nevídané možnosti pro výzkum daného období pro historiky, učitele škol všech stupňů, psychology, politology, ale i bohatý a jinak nedostupný materiál pro další výzkum v oblasti lingvistiky. Z hlediska výzkumu zpracování přirozeného jazyka, především v oblasti řeči mluvené, sám materiál v souboru poskytuje data výjimečně bohatá co do jejich charakteru. Ne vždy mluvčí užívali svůj mateřský jazyk (po návratu z koncentračního tábora se mnozí nevrátili do své rodné země, někteří z rodné země emigrovali v průběhu poválečného období, jejich znalost nově získaného jazyka ovšem stále nesla stopy po jazyku mateřském, nebo naopak mluvili jazykem mateřským, ale již

s většími nebo menšími stopami po jazyku země, v níž žijí), šlo většinou o starší ročníky mluvčích, jejichž promluvy byly pochopitelně velmi emotivní, prokládali je i německými slovy, která byla v táborech běžná. Videonahrávky umožňují sledovat i vztah mezi gesty a mimikou obličeje a promluvou atd. atd. Projekt je rovněž i velmi dobrým příkladem spojení vědeckého výzkumu s konkrétními úkoly výukovými a výchovnými.

Obrázek 14.5 demonstruje hlavní obrazovku systému, který zpřístupňuje archiv výpovědí s fotografií mluvčího, jeho osobními údaji a s přepisem jeho výpovědí.[6]

Obr. 14.5 Ukázka systému pro přístup k výpovědím v rámci projektu MALACH.

## 14.5 Závěr

Pokusili jsme se v našem příspěvku na příkladech několika projektů řešených v Centru komputační lingvistiky a Ústavu formální a aplikované lingvistiky MFF UK ilustrovat, jaké jsou reálné možnosti aplikace zahrnující jistý stupeň počítačového porozumění přirozenému jazyku. Z hlediska očekávaných aplikací, ale i z hlediska teoretického je třeba dobře propojit výzkum psaného a mluveného jazyka. Ukázali jsme také, že předpokladem pro vývoj systémů počítačového zpracování přirozeného jazyka je mít k dispozici velké množství jazykových dat, která jsou systematicky a na dobrém teoretickém základě opatřena gramatickou informací, tedy anotována. Vývoj v posledních dvou desetiletích naznačuje, že budoucnost bude patřit uváženému a efektivnímu

[6] Všechny tyto údaje jsou v naší ukázce zcela záměrně neuvedeny, popř. zkresleny.

propojení metod spočívajících na uplatnění gramatických pravidel (někdy se jim říká metody pravidlové nebo také introspektivní, protože vycházejí z intuice lingvistů) a modelů statistických (ty dnes už pokročily od bigramů a trigramů, tedy dvou- a tříčlenných kombinací slov, k zohlednění struktury věty, dokonce i struktury hloubkové). Tady se nabízí i využití velkých, neanotovaných dat, která existují ve velkém množství, tedy „levně", a v nichž je struktura svým způsobem skrytá. V každém případě je třeba, aby v dalším výzkumu i aplikacích postupovali nikoliv izolovaně, ale společně lingvisté, informatici, matematici, odborníci v oblasti reprezentace znalostí a v širší oblasti metod a nástrojů umělé inteligence. Tady je stále ještě velký prostor a jedině jeho zaplněním lze očekávat výraznější krok za daným cílem: vybavit počítač schopností porozumět – alespoň do určité míry – přirozenému jazyku.

## Literatura


Al-Onaizan Y., Cuřín J., Jahr M., Knight K., Lafferty J., Melamed D., Och F.-J., Purdy D.,. Smith N. A., Yarowsky D.: *The Statistical Machine Translation.* Technical Report, NLP WS'99, CLSP JHU, Baltimore 1999.

Brown P. F. a kol.: *The Mathematics of Statistical Machine Translation: Parameter Estimation.* In *Computational Linguistics*, 19(2), 1993, s. 263–331.

Cuřín J., Čmejrek M., Havelka J., Hajič J., Kuboň V., Žabokrtský Z.:. " *Prague Czech-English Dependency Treebank, version 1.0*". Linguistic Data Consortium, LDC2004T25, 2004.

Hajič J.: Building a Syntactically Annotated Corpus: The Prague Dependency Treebank. *Issues of Valency and Meaning. Studies in Honour of Jarmila Panevová* (ed. Eva Hajičová). Karolinum, Charles University Press, Praha 1998.

Hajič J., Hajičová E., Panevová J., Sgall P., Pajas P., Štěpánek J., Havelka J., Mikulová M.: *Prague Dependency Treebank 2.0 – CD-ROM.* LDC2006T01, Linguistic Data Consortium, 2006.

Hajič J., Kuboň V., Hric J.: *Česílko - an MT system for closely related languages.* V ACL'2000: Tutorial Abstracts and Demonstration Notes, 2000, s. 7–8.

Koehn P.: „Pharaoh: a beam search decoder for phrase-based statistical machine translation models". In *Proc. of the Sixth Conference of the Association for Machine Translation in the Americas*, 2004, s. 115–124.

Kučera O.: *Pražský závislostní korpus jako cvičebnice jazyka českého.* Diplomová práce, MFF UK, 2006.

Nilsson N. J.: *Principles of Artificial Intelligence.* Palo Alto, Kalifornie 1980.

Oard D. W.: Transforming Access to Spoken Word. In: *Proc. of the International Symposium on Large-Scale Knowledge Resources.* Tokyo Institute of Technology. 2004.

Psutka J., Ircing P., Psutka J. V., Radová V., Byrne W., Hajič J., Mírovský J., Gustman S.: *Large Vocabulary ASR for Spontaneous Czech in the Malach Project.* In: *Proc. Of the European conference on Speech Communication and Technology* (Europspeech 2003), 2003.

Psutka J., Müller L., Matoušek J., Radová V.: *Mluvíme s počítačem česky.* Academia, Praha 2006.

Sgall P.: *Generativní popis jazyka a česká deklinace.* Academia, Praha 1967.

Sgall P., Hajičová E., Panelová J.: *The Meaning of the Sentence in Its Semantic and Pragmatic Aspects.* Reidel, Dordrecht 1986.

Turing A. M.: Computing machinery and intelligence. *Mind*, 59, 1950, s. 433–460.

Weizenbaum J.: Eliza – *A Computer Program for the Study of Natural Language. Communication between Man and Machine.* Communication of the ACM 9, 1960, s. 36–45.

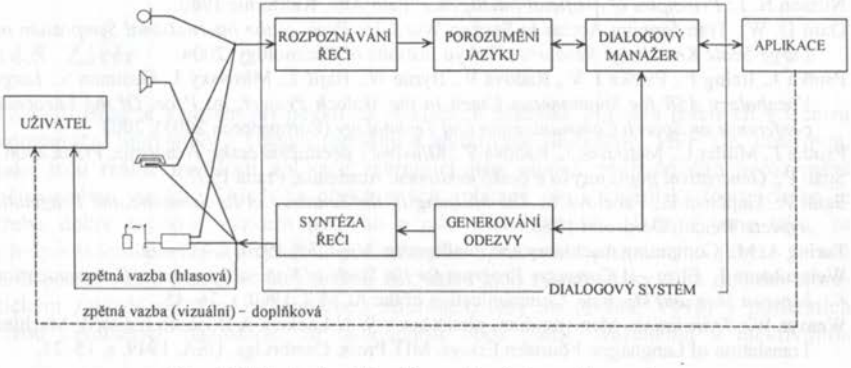
Weaver W.: *Translation.* Memorandum, přetištěno v W.N.Locke a A.D.Booth (editoři), Machine Translation of Languages: Fourteen Essays. MIT Press, Cambridge, USA, 1949, s. 15–23.

# 15 Hlasový dialog s počítačem

## 15.1 Obecná architektura hlasového dialogového systému

Hlasové dialogové systémy dovolují uživatelům komunikovat prostřednictvím hlasu s počítačovými či internetovými aplikacemi, jako jsou databázové a expertní systémy či systémy automatického řízení, ovládání a monitorování. Hlavním účelem hlasových dialogových systémů je tedy vytvořit rozhraní mezi počítačem řízenou aplikací a uživatelem, který komunikuje hlasem. Takto formulované definici vyhovuje celá řada systémů hlasového dialogu, a to od relativně jednoduchých systémů komunikujících pomocí malého souboru izolovaně vyslovovaných povelů až po systémy, které umožňují vést komunikaci s uživatelem souvislou přirozenou řečí. Současný stav vědní disciplíny i současné komerční aplikace hlasových dialogových systémů stále předpokládají, že hlasový dialog je omezen pouze na zvolenou aplikační oblast (např. dialogový systém pro objednávání vstupenek do divadel, rezervační systém autobusových spojení, nákup po telefonu apod.) a nejde tedy zatím o přirozenou a neomezenou konverzaci člověka s počítačem na libovolné téma.

Pro úspěšnou interakci člověka s hlasovým dialogovým systémem je třeba mít k dispozici vedle dobře navrženého modulu, který zajišťuje *řízení dialogu*, též moduly *rozpoznávání* a *syntézy řeči* i modul zajišťující *porozumění mluvenému jazyku* (přesněji modul pro porozumění posloupnosti slov, kterou poskytuje rozpoznávač řeči). Tento posledně jmenovaný modul by měl pracovat tak, aby interpretoval rozpoznané promluvy




Obr. 15.1 Blokové schéma hlasového dialogového systému.

v kontextu s dalšími známými informacemi a dával podklady k uskutečnění odpovídajících akcí. Uvedenému výčtu potřebných funkcí odpovídá i blokové schéma typického hlasového dialogového systému, viz obr. 15.1, které sestává z modulů *rozpoznávání řeči*, *porozumění jazyku*, *generování hlasové odezvy*, včetně ústředního a nejdůležitějšího modulu, jímž je *dialogový manažer*. Dialogový manažer řídí interaktivní strategii a spojuje různé typy informace a znalostí opatřené ostatními moduly dialogového systému tak, aby úspěšně dokončil úlohu „řešenou" dialogem. Pro mnohé aplikace je totiž velmi nepravděpodobné, že by uživatel v interakci se systémem získal požadovanou informaci, popř. zajistil vykonání požadované akce, pouze jediným dotazem (příkazem). Je to proto, že dotaz může být nepřesný, neúplný a někdy i nekonzistentní vzhledem k historii dialogu. Přesto i v případě, že dotaz uživatele je nedvojznačný a úplný, může modul rozpoznávání řeči chybovat a dialogový systém by měl mít proto možnost si v interaktivní komunikaci s uživatelem zajistit objasnění, kompletaci a potvrzení všech potřebných informací k úspěšnému završení úlohy.

Je dobře známo, že lidé dokážou správně interpretovat informaci i v případě ztížených komunikačních podmínek. Je to proto, že sdílejí větší či menší obecnou apriorní znalost o světě a zejména o tématu konverzace. Abychom zajistili podobné vlastnosti i při strojovém zpracování, je nutné provést přesnou analýzu úlohy a navrhnout a zajistit modelování potřebných *zdrojů znalostí* zahrnutých ve všech úrovních procesu interakce (Minker, 2002). Otázkou přitom je jak získat potřebné informace pro konstrukci těchto znalostních zdrojů a jak stanovit odpovídající architekturu systému a navrhnout stěžejní moduly hlasového dialogového systému, které zajistí bezproblémový a uživatelsky příjemný provoz. Při konstrukci těchto modulů může být využit jednak *znalostní přístup* (angl. knowledge-based approach), anebo přístup založený na *statistické indukci*, a to podle toho, jakou apriorní znalost má o úloze návrhář dialogového systému, popř. zda a v jakém rozsahu jsou k dispozici data pro natrénování stěžejních statistik systému z reálných příkladů.

Hlasové dialogové systémy lze klasifikovat též podle přístupu užitého při řízení dialogu, a to na *systémy s konečným počtem stavů*, *systémy založené na rámcích* a konečně na *systémy založené na agentech*. Strategie řízení dialogu má přitom přímou souvislost s tím, jak dialogový systém zpracovává vstupní řečovou informaci od uživatele a jak zachází s případnými chybami rozpoznávání a interpretace.

## 15.2 Modul rozpoznávání řeči

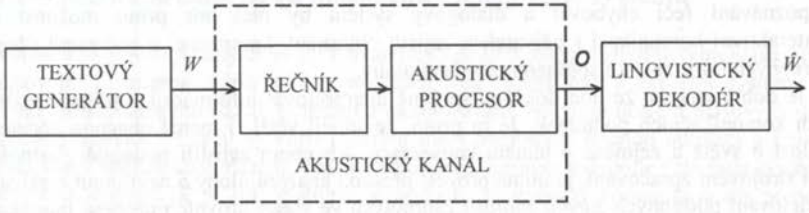
Naprostá většina současných systémů rozpoznávání mluvené řeči pracuje na statistickém principu. Základní schéma statistického přístupu k rozpoznávání mluvené řeči sestává z *akustického procesoru* a *lingvistického dekodéru*, přičemž řečník a akustický procesor jsou spojeni *akustickým kanálem*, obr. 15.2. Akustický procesor transformuje řečové kmity produkované řečníkem na posloupnosti vektorů příznaků a lingvistický dekodér překládá i „zkomolené" řetězce příznaků na řetězce slov. Rozpoznávání je zde formulováno jako problém *dekódování s maximální aposteriorní pravděpodobností*.

Předpokládejme, že $W = \{w_1 w_2 \ldots w_N\}$ je posloupnost $N$ slov a nechť $\boldsymbol{O} = \{\boldsymbol{o}_1 \boldsymbol{o}_2 \ldots \boldsymbol{o}_T\}$ je akustická informace, tj. posloupnost vektorů příznaků odvozená z řečového signálu, z níž se lingvistický dekodér pokouší rozpoznat, která slova byla vyslovena. Cílem je

nalézt posloupnost slov $\hat{W}$, která maximalizuje podmíněnou pravděpodobnost $P(W|\boldsymbol{O})$, tj. nejpravděpodobnější posloupnost slov pro danou akustickou informaci $\boldsymbol{O}$. Využijeme-li Bayesova pravidla, bude

$$\hat{W} = \arg\max_{W} P(W \mid \boldsymbol{O}) = \arg\max_{W} \frac{P(W)P(\boldsymbol{O} \mid W)}{P(\boldsymbol{O})}, \qquad (15.1)$$

kde $P(\boldsymbol{O}|W)$ je pravděpodobnost, že při vyslovení posloupnosti slov $W$ bude produkována posloupnost výstupních vektorů příznaků $\boldsymbol{O}$, $P(W)$ je apriorní pravděpodobnost posloupnosti slov $W$ (tj. pravděpodobnost, že řečník bude číst posloupnost slov $W$) a $P(\boldsymbol{O})$




Obr. 15.2 Blokové schéma systému rozpoznávání řeči založené na statistickém přístupu.

je apriorní pravděpodobnost posloupnosti výstupních vektorů. Protože pravděpodobnost $P(\boldsymbol{O})$ není funkcí $W$, lze ji při hledání maxima v (15.1) ignorovat. Hledanou posloupnost slov $\hat{W}$ lze tedy určit maximalizací sdružené pravděpodobnosti $P(W, \boldsymbol{O})$

$$\hat{W} = \arg\max_{W} P(W \mid \boldsymbol{O}) = \arg\max_{W} P(W)P(\boldsymbol{O} \mid W). \qquad (15.2)$$

Z této rovnice vyplývá, že problém stanovení nejlepší posloupnosti slov k danému akustickému signálu vyjádřenému posloupností pozorovaných vektorů příznaků $\boldsymbol{O} = \{\boldsymbol{o}_1\boldsymbol{o}_2\ldots\boldsymbol{o}_T\}$ lze řešit pomocí dvou oddělených pravděpodobností $P(\boldsymbol{O}|W)$ a $P(W)$, které mohou být modelovány a trénovány nezávisle na sobě. Pravděpodobnostní rozdělení těchto pravděpodobností lze přitom považovat za určité *stochastické zdroje znalostí* o řešené úloze. Podmíněné rozdělení pravděpodobnosti $P(\boldsymbol{O}|W)$ nese informaci o *akustickém modelu* (též modelu řečníka) a apriorní rozdělení pravděpodobnosti $P(W)$ nese informaci o *jazykovém modelu*. Tyto dva zdroje znalostí je třeba určit ještě před vlastním rozpoznáváním, a to nejčastěji na základě trénování z řečových a jazykových dat. Proces rozpoznávání pak spočívá v nalezení takové posloupnosti slov $\hat{W}$, která pro danou posloupnost pozorovaných vektorů příznaků $\boldsymbol{O}$ maximalizuje součin pravděpodobností $P(\boldsymbol{O}|W)$ a $P(W)$ přes všechny možné posloupnosti slov $W$. Výpočetní náročnost takového vyčerpávajícího hledání je však i pro slovníky menšího rozsahu enormní. V praktických aplikacích jsou proto využívány takové důmyslné suboptimální *prohledávací* a *rozhodovací strategie*, které se snaží množství výpočtů účinně redukovat pokud možno s minimálními následky na přesnost rozpoznávání.

Z uvedeného výkladu vyplývá, že úloha automatického rozpoznávání mluvené řeči může být v případě využití statistických rozhodovacích metod řešena dekompozicí na následující subúlohy:

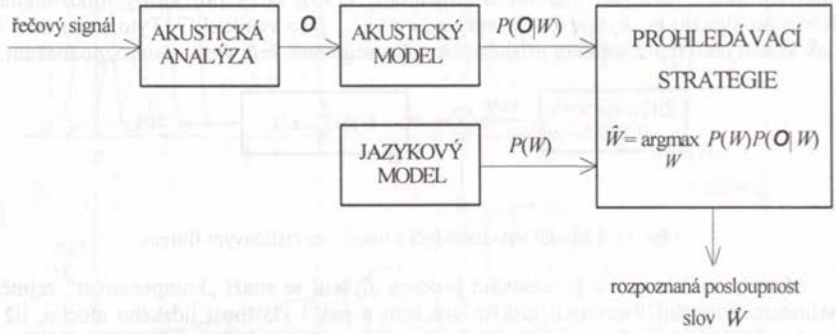
1. Provedení akustické analýzy řečového signálu s cílem určit posloupnost vektorů příznaků $\boldsymbol{O}$.

2. Vytvoření akustického modelu pro ocenění podmíněné pravděpodobnosti $P(\boldsymbol{O}|W)$.

3. Vytvoření jazykového modelu pro ocenění pravděpodobnosti $P(W)$.

4. Nalezení nejpravděpodobnější posloupnosti slov aplikací účinných prohledávacích strategií.

Dekompozici úlohy rozpoznávání souvislé řeči na jednotlivé subúlohy lze ilustrovat obrázkem 15.3.




Obr. 15.3 Systém rozpoznávání řeči dokomponovaný do bloků podle řešených úloh.

### 15.2.1 Analýza akustického signálu

Základem většiny metod analýzy akustického signálu řeči je předpoklad, že se jeho vlastnosti v průběhu času mění pomalu. Tento předpoklad vede na aplikaci tzv. metod krátkodobé analýzy. Metody analýzy řeči pro účely rozpoznávání vycházejí buď z modelu produkce či vytváření řeči, anebo se snaží modelovat proces slyšení.

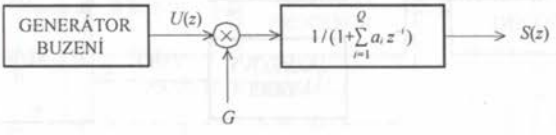
Analýza řeči, která vychází z modelování produkce řečového signálu, je obvykle založena na *lineární prediktivní analýze*. Jde o metodu, která se snaží na krátkodobém základu odhadnout přímo z řečového signálu parametry modelu vytváření řeči. Model vytváření řeči se přitom skládá ze systému s časově proměnným přenosem a generátoru budících funkcí, který budí systém při vytváření znělých zvuků posloupností impulsů a při vytváření neznělých zvuků náhodným šumem. Princip metody LPC je založen na předpokladu, že $k$-tý vzorek signálu $s(k)$ lze popsat lineární kombinací $Q$ předchozích vzorků a buzení $u(k)$, tj.

$$s(k) = -\sum_{i=1}^{Q} a_i s(k-i) + G\,u(k)\,, \tag{15.3}$$

kde $G$ je koeficient zesílení a $Q$ je řád modelu. Přenosovou funkci modelu $H(z)$ lze pak zapsat ve tvaru

$$H(z)=\frac{S(z)}{U(z)}=\frac{G}{A(z)}=\frac{G}{1+\sum_{i=1}^{Q}a_i z^{-i}} \tag{15.4}$$

a proces modelování znázornit obrázkem 15.4. Sledovanými parametry jsou zde koeficienty $a_i$ číslicového filtru a koeficient zesílení $G$. Je-li splněn předpoklad přibližné stacionarity signálu na sledovaném časovém intervalu, lze pro určení $a_i$ a $G$ využít metody nejmenších čtverců. Při neznámém členu $G\,u(k)$ v rovnici (15.3) vzniká chyba predikce $e(k)$ mezi skutečnou hodnotou $s(k)$ a předpovězenou $\hat{s}(k)$. Minimalizace krátkodobé energie této chyby pak vede na soustavu rovnic. K řešení této soustavy lze využít například Durbinův iterativní algoritmus (Makhoul, 1975), který pro každý mikrosegment řečového signálu poskytuje koeficienty $a_i$, $i$ =1, ..., $Q$, a zesílení $G$. Tyto parametry lze pak využít jako reprezentanty příslušných mikrosegmentů řeči v procesu rozpoznávání.




Obr. 15.4 Model vytváření řeči s lineárním číslicovým filtrem.

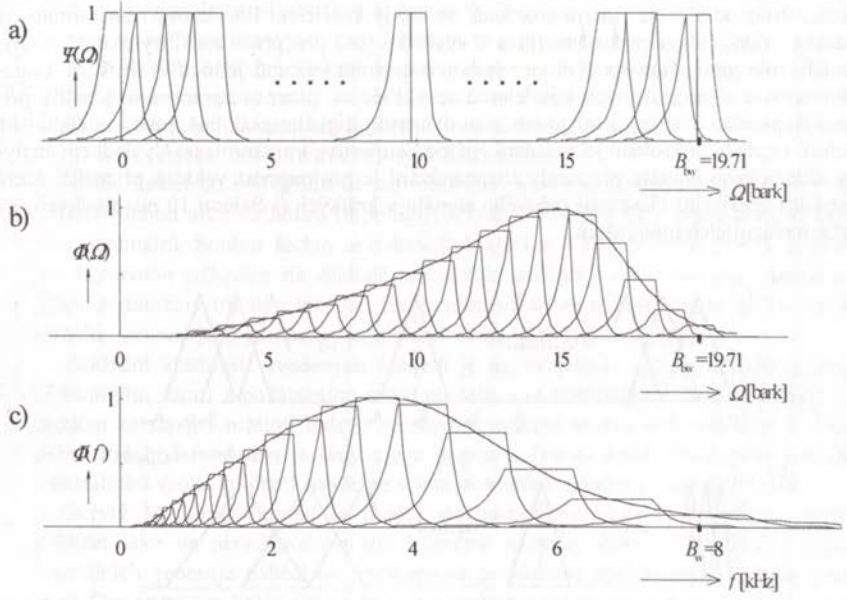
Metody vycházející z modelování procesu slyšení se snaží „kompenzovat" zejména nelineární vnímání frekvencí lidským sluchem a pak i vlastnost lidského sluchu, jíž je maskování zvuků a vliv tzv. kritických pásem. Maskování je jev, kdy vnímání jednoho zvuku je ovlivněno přítomností jiného zvuku. Přesněji řečeno, přítomnost jednoho zvuku způsobuje zvýšení prahu slyšitelnosti pro další zvuk, který buď zní současně, nebo s určitým časovým odstupem. Také přítomnost šumu v určitém frekvenčním pásmu snižuje naši schopnost slyšet tón, který se v tomto pásmu nachází. Přitom na maskování se podílí pouze určité pásmu šumu v okolí kmitočtu maskovaného tónu. Šum mimo toto pásmo už nemá na naši schopnost vnímat daný tón žádný vliv. Tento jev lze vysvětlit tím, že při naslouchání danému tónu člověk používá jakýsi „psychologický filtr", který ignoruje veškerý šum ležící mimo určité *kritické pásmo*. Šířka takového kritického pásma je přitom určena frekvencí poslouchaného tónu. Typickým představitelem metod zpracování řeči, které se snaží modelovat proces slyšení jsou *melovská kepstrální filtrace* a *perceptivní lineární prediktivní analýza*.

V případě perceptivní lineární prediktivní analýzy (angl. Perceptual Linear Predictive, zkr. PLP) je nejprve pro sledovaný mikrosegment řečového signálu vyčísleno Fourierovou transformací krátkodobé výkonové spektrum. Následuje nelineární transformace frekvencí s aplikací kritických pásem spektrální citlivosti slyšení. Modelování těchto jevů je v PLP analýze realizováno jednak nelineární transformací původní osy frekvencí $\omega$[rad/s] na osu frekvencí $\Omega(\omega)$ měřenou v jednotce bark, a to podle vztahu

$$\Omega(\omega)=6\ln\left\{\frac{\omega}{1200\pi}+\sqrt{\left(\frac{\omega}{1200\pi}\right)^2+1}\right\}, \tag{15.5}$$

kde $\omega = 2\pi f$ [rad/s] a $\Omega(\omega)$[bark], a dále konstrukcí maskujících křivek (v podstatě jde o pásmové propusti), které simulují kritická pásma slyšení. Protože člověk vnímá

intenzitu zvuku „nelineárně" v závislosti na frekvenci, dojde v dalším kroku zpracování k přizpůsobení kritických pásmových filtrů křivkám stejné hlasitosti. Na obr. 15.5, části a), je znázorněno rozložení banky kritických pásmových filtrů, a to pro případ zpracování signálu s šířkou pásma $B_w = 19{,}71$ bark, jež odpovídá šířce pásma $B_w = 8$kHz při frekvenci vzorkování $F_w = 16$ kHz. Části b), a c) obr. 15 znázorňují stejnou banku filtrů korigovanou křivkami stejné hlasitosti, a to ve frekvenční škále s jednotkou bark, popř. s jednotkou Hz. V dalších krocích je provedena vážená spektrální sumarizace vzorků výkonového spektra a je uplatněn vztah vyjadřující závislost mezi intenzitou zvuku a vnímanou hlasitostí. Parametry PLP analýzy jsou finálně aproximovány celopólovým modelem (Psutka a kol., 2006).

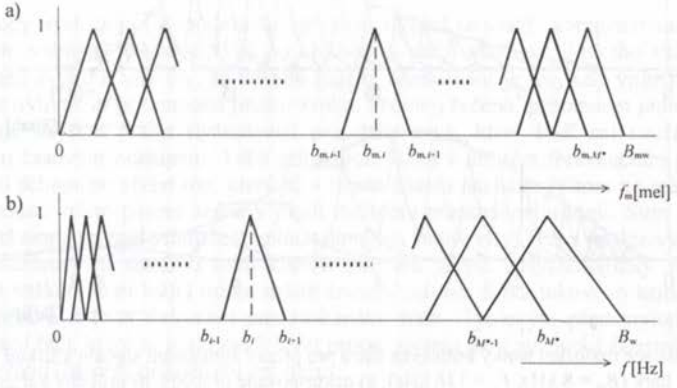



Obr. 15.5 Ilustrace rozložení banky kritických filtrů pro případ zpracování signálu s šířkou pásma $B_{bw} = 19{,}71$ bark ($B_w = 8$ kHz, $F_v = 116$ kHz); a) nekorigované průběhy, b) průběhy korigované křivkami stejné hlasitosti ve frekvenční škále s jednotkou bark, c) průběhy korigované křivkami stejné hlasitosti ve frekvenční škále s jednotkou Hz.

Podobně jako perceptivní lineární analýza je i zpracování pomocí melovských kepstrálních koeficientů navrženo tak, aby do jisté míry respektovalo nelineární vlastnosti vnímání zvuků lidským uchem. Melovské kepstrální koeficienty se snaží „kompenzovat" zejména nelineární vnímání frekvencí, a to využitím banky trojúhelníkových pásmových filtrů s lineárním rozložením frekvencí v tzv. *melovské frekvenční škále*, jež je definována vztahem

$$f_m = 2595 \log_{10}\left(1 + \frac{f}{700}\right), \qquad (15.6)$$

kde $f$ [Hz] je frekvence v lineární škále a $f_m$ [mel] je odpovídající frekvence v nelineární melovské škále. Klíčovou částí celého procesu zpracování je melovská filtrace. Výpočetní algoritmus této filtrace je realizován bankou trojúhelníkových pásmových filtrů s rovnoměrným rozložením středních frekvencí jednotlivých trojúhelníkových filtrů podél frekvenční osy s měřítkem v melovské škále. Počet pásem této banky filtrů je rozumné volit v závislosti na počtu a umístění kritických pásem, a to při respektování velikosti vzorkovací frekvence $F_v$ [Hz] a celkové šířky přenášeného pásma $B_w$ [Hz], resp. $B_{mw}$ [mel]. Trojúhelníkové filtry jsou standardně rozloženy přes celé frekvenční pásmo od nuly až do Nyquistovy frekvence. Obrázek 15.6 ukazuje příklad takové banky filtrů, v níž každý filtr má trojúhelníkovou pásmovou frekvenční odezvu a vzdálenost filtrů a rovněž jejich šířka pásma jsou určeny konstantním intervalem v melovské frekvenční škále. Průchod signálu filtrem znamená, že každý koeficient Fourierovy transformace je násoben odpovídajícím ziskem filtru a výsledky jsou pro příslušné filtry akumulovány. Další krok zpracování spočívá ve výpočtu logaritmu výstupů jednotlivých filtrů. Logaritmováním akumulovaných koeficientů se vkládá do procesu zpracování důležitý příznak kepstrální analýzy, jímž je omezení dynamiky signálu (podobně, jako to dělá lidské ucho). Posledním krokem je zavedení zpětné Fourierovy transformace. Výsledkem analýzy akustického signálu pro účely rozpoznávání je posloupnost vektorů příznaků, které popisují spektrální vlastnosti řečového signálu v krátkých (většinou 10 ms dlouhých) na sebe navazujících intervalech.




Obr. 15.6 Banka trojúhelníkových filtrů a) v melovské frekvenční škále, b) v původní netransformované škále v [Hz].

### 15.2.2 Akustické modelování

Úkolem akustického modelu je poskytnout co nejpřesnější a nejrychlejší odhad podmíněné pravděpodobnosti $P(\boldsymbol{O}|W)$ pro libovolnou pozorovanou posloupnost vektorů příznaků $\boldsymbol{O}$ a každou uvažovanou posloupnost slov. Jako velmi efektivní způsob řešení této náročné úlohy se ukázalo být využití tzv. *skrytých Markovových modelů* (angl. Hidden Markov Model, zkr. HMM). Princip metody modelování řeči Markovovými modely vychází z představy o vytváření řeči. Při generování řeči člověkem si lze představit, že hlasové ústrojí je během krátkého časového intervalu (např. mikrosegmentu)

v jednom z konečného počtu stavů artikulačních konfigurací (např. je nastaveno pro vyslovení určité hlásky). V uvažovaném mikrosegmentu je pak hlasovým ústrojím produkován krátký signál, který závisí na stavu artikulačního ústrojí a může být popsán určitými spektrálními charakteristikami, jež jsou reprezentovány vhodnými příznaky (vektorem příznaků). Z představy o vytváření řeči vychází i konstrukce klasifikátoru založená na modelování řečového signálu pomocí Markovova procesu. Při tomto procesu jsou generovány dvě vzájemně svázané časové posloupnosti náhodných proměnných, a to *podpůrný Markovův řetězec*, který je posloupností konečného počtu stavů, a *řetězec vektorů příznaků*, jež reprezentují spektrální charakter krátkých úseků (mikrosegmentů) řečového signálu. Pro tyto spektrální charakteristiky jsou vytvořeny „náhodné funkce", které pravděpodobnostně ohodnocují vztah charakteristik ke všem stavům. Předpokládá se, že v diskrétních časových okamžicích je proces v jediném stavu a lze jej pozorovat prostřednictvím „náhodné funkce" korespondující s tímto stavem. Podpůrný Markovův řetězec pak mění stavy podle odpovídajících pravděpodobností přechodu. Pozorovatel „vidí" jen výstup „náhodných funkcí" a nemůže přímo pozorovat stavy podpůrného Markovova řetězce, odtud termín skrytý Markovův model.

Úloha nalézt co nejlepšího odhad rozdělení pravděpodobnosti $P(\boldsymbol{O}|W)$ je v našem případě úlohou určit strukturu (topologii) skrytého Markovova modelu a typ a hodnoty jeho parametrů. Soubor těchto neznámých údajů lze v zásadě určit dvojím způsobem, a to *expertním odhadem* na základě apriorních znalostí a dále *metodou statistické indukce* z množiny trénovacích dat. Expertním odhadem se standardně určuje struktura modelu, zatímco jeho parametry jsou stanovovány statistickou indukcí.

Základní vlastností uvedených modelů je to, že proces začíná příchodem prvního spektrálního vzoru z počátečního stavu modelu a se vzrůstajícím časem dochází k přechodům ze stavů s nižšími indexy do stavů s vyššími indexy nebo setrvání ve stejném stavu. Průchod modelem je tedy zleva doprava. Proces končí příchodem posledního spektrálního vzoru, přičemž model se v tom okamžiku nachází v koncovém stavu.

Skrytý Markovův model je model stochastického procesu, na něhož je možné pohlížet jako na pravděpodobnostní konečný automat, který v diskrétních časových okamžicích generuje náhodnou posloupnost pozorování (posloupnost vektorů pozorování) $\boldsymbol{O} = \{\boldsymbol{o}_1\boldsymbol{o}_2...\boldsymbol{o}_T\}$. V každém časovém kroku změní model svůj stav $s_i$ podle souboru předem daných pravděpodobností přechodu $a_{ij}$. Stav $s_j$, do kterého model přejde, generuje příznakový vektor (vektor pozorování) $\boldsymbol{o}_t$, a to podle rozdělení výstupní pravděpodobnosti $b_j(\boldsymbol{o}_t)$ příslušné k tomuto stavu. Pravděpodobnosti přechodu $a_{ij}$ jsou podmíněné a určují, s jakou pravděpodobností přechází model ze stavu $s_j$, v kterémkoli čase $t$, do stavu $s_j$ v čase $t + 1$. Platí tedy

$$a_{ij} = P(s(t+1) = s_j \mid s(t) = s_i), \tag{15.7}$$

kde $s(t)$ je stav modelu v čase $t$. Dále se předpokládá, že pravděpodobnosti přechodů $a_{ij}$ jsou v čase konstantní, přičemž pro všechny stavy $s_i$, $i = 1, 2,..., N$, je splněna podmínka

$$\sum_{i=1}^{N} a_{ij} = 1. \tag{15.8}$$

Funkce rozdělení výstupní pravděpodobnosti $b_j(\boldsymbol{o}_t)$ popisují rozdělení pravděpodobnosti pozorování $\boldsymbol{o}_t$ produkovaného ve stavu $s_j$ v čase $t$. Výstupní pravděpodobnosti jsou nejčastěji uvažovány jako spojité náhodné veličiny (spojitého náhodného vektoru) a funkce $b_j(\boldsymbol{o}_t)$ tak představuje hustotou pravděpodobnosti jevu, že stav $s_j$ v čase $t$ generuje pozorování $\boldsymbol{o}_t$. Pro funkci $b_j(\boldsymbol{o}_t)$ platí

$$b_j(\boldsymbol{o}_t) = P(\boldsymbol{o}_t \mid s(t) = s_j). \tag{15.9}$$

Pro všechny emitující stavy (tj. pro stavy, které jsou schopny generovat výstupní vektor pozorování) pak platí, že

$$\int_{o} b_j(\boldsymbol{o})\, \mathrm{d}\boldsymbol{o}_t = 1. \tag{15.10}$$

Rozdělení výstupní pravděpodobnosti musí být dostatečně specifické, aby od sebe oddělilo různé zvuky, a zároveň dostatečně robustní, aby zahrnulo značnou variabilitu řečového signálu. Z tohoto hlediska nejlépe vyhovuje modelovat výstupní pravděpodobnosti spojitým rozdělením se směsí normálních hustotních funkcí

$$b_j(\boldsymbol{o}_t) = \sum_{m=1}^{M} c_{jm} N(\boldsymbol{o}_t; \mu_{jm}, \boldsymbol{C}_{jm}), \tag{15.11}$$

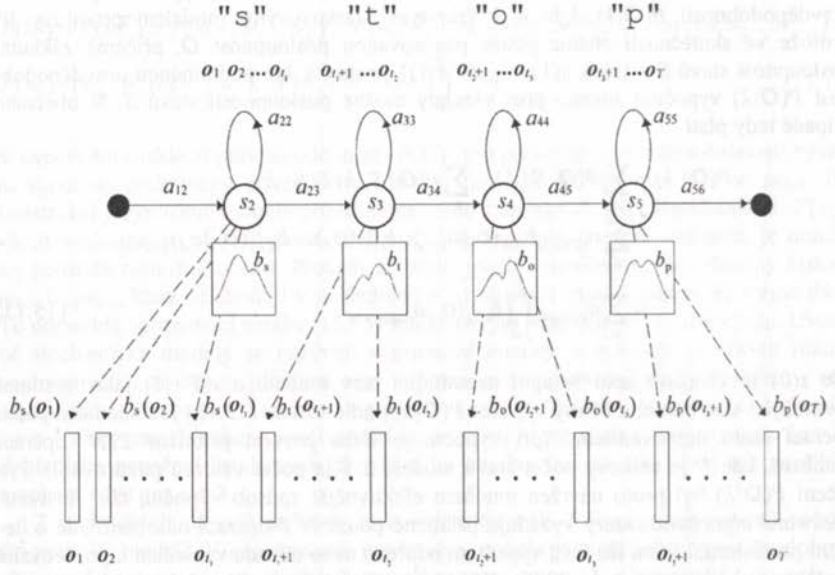
kde $M$ značí počet složek hustotní směsi, $c_{jm}$ vyjadřuje váhu $m$-té složky $j$-tého stavu a $N(\boldsymbol{o}_t; \mu_{jm}, \boldsymbol{C}_{jm})$ je vícerozměrné (vícedimenzionální) normální rozdělení se střední hodnotou $\mu_{jm}$ a kovariančí maticí $\boldsymbol{C}_{jm}$, tj.

$$N(\boldsymbol{o}_t; \mu_{jm}, \boldsymbol{C}_{jm}) = \frac{1}{\sqrt{(2\pi)^n \mid \boldsymbol{C}_{jm} \mid}} \exp\left( -\frac{1}{2} (\boldsymbol{o}_t - \mu_{jm})^T \boldsymbol{C}_{jm}^{-1} (\boldsymbol{o}_t - \mu_{jm}) \right), \tag{15.12}$$

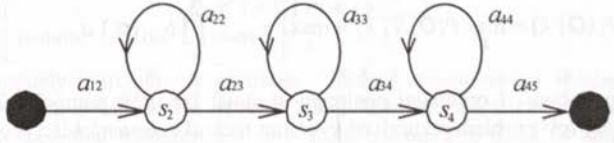
kde $n$ je dimenze vektoru pozorování $\boldsymbol{o}_t$.

Při modelování mluvené řeči se využívají zejména tzv. *levo-pravé Markovovy modely* (angl. left-to-right models), které jsou zvláště vhodné pro modelování procesů, jejichž vývoj je spojen s postupujícím časem. Podle typu úlohy a velikosti rozpoznávaného slovníku lze konstruovat modely celých slov anebo modely menších subslovních jednotek, například fonémů nebo kontextově závislých fonémů, tzv. trifónů. Modely celých slov, viz například Markovův model slova „stop" znázorněný na obr. 15.7, je vhodné konstruovat v případě, je-li slovník méně rozsáhlý. Důvodem je potřeba mít k dispozici pro trénování parametrů modelu poměrně velké množství trénovacích dat, tj. promluv daných, do slovníku zařazených, slov. Při práci s rozměrnějšími slovníky (tisíce či desetitisíce slov) a potřebě alespoň několika trénovacích promluv pro každé slovo vyvstává u modelů celých slov problém s těžkopádným a zdlouhavým trénováním klasifikátoru. Je zřejmé, že pro zjednodušení a urychlení procesu trénování parametrů skrytých Markovových modelů slov by bylo vhodné odvodit tento proces od menších jednotek řeči, než jsou slova. Takovými jednotkami mohou být například fonémy či kontextově závislé trifóny. Příkladem modelu fonému s velmi jednoduchou strukturou

může být třístavový model znázorněný na obr. 15.8. První a třetí stav modelu jsou tzv. *neemitující stavy*, tj. jsou to stavy, které negenerují žádná pozorování a nemají tedy žádné k nim příslušné rozdělení výstupní pravděpodobnosti. Druhý stav je klasický *emitující stav*.




Obr. 15.7 Markovův model slova *stop* s počtem stavů odvozených od počtu odpovídajících hlásek.




Obr. 15.8 Příklad topologie modelu fonému s třemi emitujícími stavy.

Slova a celé promluvy lze vytvářet z modelů fonémů na základě jejich spojování (zřetězování), a to tak, že neemitující koncový stav modelu fonému a neemitující počáteční stav následujícího fonému se ztotožní. V případě, že do koncového stavu modelu vede jen jeden přechod a z počátečního neemitujícího stavu následujícího modelu vychází též jen jeden přechod, lze tyto neemitující stavy vypustit. Dekompozice slov na posloupnosti fonémů odstraňuje závislost velikosti trénovacích dat na rozměru slovníku. Pro natrénování systému potřebujeme v tomto případě pouze soubor vzorků každého fonému a ne soubor vzorků každého slova. Zavedením modelů fonémů zvětšíme pružnost systému, tj. klasifikovaný slovník lze rozšiřovat či měnit, aniž musíme provádět pracné trénování nových slov.

Úkolem akustického modelu je pro libovolnou pozorovanou posloupnost vektorů příznaků $\boldsymbol{O}$ najít co nejpravděpodobnější posloupnost slov (resp. nejpravděpodobnější slovo, jde-li o klasifikaci izolovaných slov). Předpokládejme, že každé slovo $w$, resp. posloupnost slov $W$, jsou modelovány odpovídajícím skrytým Markovovým modelem $\lambda$. Určení podmíněné pravděpodobnosti $P(\boldsymbol{O}|W)$ lze pak nahradit výpočtem podmíněné pravděpodobnosti $P(\boldsymbol{O}|\lambda)$, kde $\lambda$ je (skrytým Markovovým) modelem promluvy $W$. Protože ve skutečnosti známe pouze pozorovanou posloupnost $\boldsymbol{O}$, přičemž základní posloupnost stavů $S = \{s(0), s(1), ..., s(T+1)\}$ je skrytá, lze podmíněnou pravděpodobnost $P(\boldsymbol{O}|\lambda)$ vypočítat sumací přes všechny možné posloupnosti stavů $S$. V obecném případě tedy platí

$$\begin{aligned} P(\boldsymbol{O} \mid \lambda) &= \sum_S P(\boldsymbol{O}, S \mid \lambda) = \sum_S P(\boldsymbol{O} \mid S, \lambda) P(S \mid \lambda) \\ &= \sum_S a_{s(0)s(1)} b_{s(1)}(\boldsymbol{o}_1) a_{s(1)s(2)} b_{s(2)}(\boldsymbol{o}_2) \ldots b_{s(T)}(\boldsymbol{o}_T) a_{s(T)s(T+1)} \\ &= \sum_S a_{s(0)s(1)} \prod_{t=1}^{T} b_{s(t)}(\boldsymbol{o}_t) a_{s(t)s(t+1)} \,, \end{aligned} \tag{15.13}$$

kde $s(0)$ je chápáno jako vstupní neemitující stav modelu a $s(T+1)$ jako výstupní neemitující stav modelu. Přímý výpočet $P(\boldsymbol{O}|\lambda)$ podle vztahu (15.13) je z hlediska počtu operací často neproveditelný (při výpočtu je třeba provést přibližně $2TN^T$ operací násobení, kde $N$ je celkový počet stavů modelu a $T$ je počet vektorů pozorování). Pro určení $P(\boldsymbol{O}|\lambda)$ byl proto navržen mnohem efektivnější způsob výpočtu, tzv. *forward-backward algoritmus*, který vyžaduje přibližně pouze $N^2T$ operací násobení. Jde o iterační proceduru, kterou lze řešit výpočtem odpředu nebo odzadu vzhledem k pozorované posloupnosti (Psutka a kol., 2006). Výpočet podmíněné pravděpodobnosti lze alternativně aproximovat pravděpodobností $P_S(\boldsymbol{O}|\lambda)$, jako nejpravděpodobnější posloupností stavů, kterou projde posloupnost $\boldsymbol{O}$ modelem $\lambda$

$$P_S(\boldsymbol{O} \mid \lambda) = \max_S P(\boldsymbol{O}, S \mid \lambda) = \max_S \left[ a_{s(0)s(1)} \prod_{t=1}^{T} b_{s(t)}(\boldsymbol{o}_t)\, a_{s(t)s(t+1)} \right]. \tag{15.14}$$

Tuto pravděpodobnost i optimální posloupnost stavů lze určit pomocí tzv. *Viterbiova algoritmu*, který řeší problém rekurzivně využitím *techniky dynamického programování*.

Jestliže volba struktury (topologie) skrytého Markovova modelu je spíše úlohou expertního návrhu, pak stanovení hodnot parametrů modelu je uskutečňováno statistickou indukcí nebo-li trénováním (odhadem, estimací) parametrů na základě přesně „popsaných" (anotovaných) trénovacích akustických dat. Úkolem je přitom stanovit hodnoty parametrů modelu. Jako metoda odhadu se nejčastěji využívá metoda maximální věrohodnosti (angl. Maximum Likelihood Estimation, zkr. MLE). V úloze modelování řeči skrytými Markovovými modely existuje velmi efektivní metoda pro trénování parametrů modelu, jež má základ v kritériu maximální věrohodnosti. Pro maximalizaci věrohodnostní funkce se zde využívá iterativní procedura, tzv. Baumův–Welchův algoritmus, který je speciálním případem EM algoritmu (zkr. Expectation–Maximization algorithm) (Psutka a kol., 2006).

### 15.2.3 Jazykové modelování

Úkolem konstrukce stochastického jazykového modelu je stanovit pro každou posloupnost slov $W$ apriorní pravděpodobnost $P(W)$ této posloupnosti. Pravděpodobnost posloupnosti $W$, která čítá $K$ slov, je možné určit podle následujícího vtahu

$$P(W) = P(w_1^K) = P(w_1 w_2 w_3 \ldots w_K) = P(w_1)P(w_2|\, w_1)P(w_3|\, w_1 w_2) \ldots P(w_K|\, w_1 w_2 \ldots w_{K-1}) =$$

$$P(w_1)P(w_2 \mid w_1^1)P(w_3 \mid w_1^2)\ldots P(w_K \mid w_1^{K-1}) = \prod_{i=1}^{K} P(w_i \mid w_1^{i-1}). \qquad (15.15)$$

V uvedeném rozkladu pravděpodobnosti $P(W)$ jsou podmíněné pravděpodobnosti výskytu slova $w_i$ podmíněny pouze svou historií, tj. posloupností slov $w_1 \ldots w_{i-2} w_{i-1}$. Pro konstrukci jazykového modelu požadujeme znalost apriorních pravděpodobností $P(w_1^K)$ všech posloupností slov libovolné délky $K$. Všechny tyto pravděpodobnosti je obtížné a v podstatě nemožné ocenit. Provádí se proto jejich aproximace, kdy všechny historie $w_1 \ldots w_{i-2} w_{i-1}$, které se shodují v posledních $n-1$ slovech, jsou zařazeny do stejné třídy. To odpovídá aproximaci vztahu (15.15) Markovovým modelem $(n-1)$-vého řádu. Uvedené stochastické modely se nazývají *n-gramové modely*. *n*-gramem se přitom rozumí posloupnost $n$ za sebou jdoucích slov v pozorování jejich náhodného výběru – tedy například v trénovacím korpusu. *n*-gramům s $n = 0$ říkáme *zerogramy* a *n*-gramům s $n = 1$ *unigramy*. Mnohem používanější jsou ovšem *bigramy* ($n = 2$) a *trigramy* ($n = 3$). Z hlediska postižení jazykových závislostí by vhodný řád modelu mohl být například mnohem větší než $n = 3$.

Z praktických důvodů se však za $n$ nejčastěji volí 2 (bigramový model) nebo 3 (trigramový model). V *n*-gramovém jazykovém modelu je podmíněná pravděpodobnost $P(w_k \mid w_1^{k-1})$ slova $w_k$ na pozici $k$ závislá pouze na $n-1$ předchozích slovech, tj. aproximuje se vztahem $P(w_k \mid w_1^{k-1}) \approx P(w_k \mid w_{k-n+1}^{k-1})$, takže pro $P(w_1^{k-1})$ platí

$$P(w_1^k) = \prod_{i=1}^{k} P(w_i \mid w_{i-n+1}^{i-1}). \qquad (15.16)$$

*n*-gramové modely mají několik předností. Především jsou velmi vhodné pro jazyky s relativně pevným pořadím slov ve větě, neboť zde existují silné statistické závislosti mezi výskyty za sebou následujících slov. Další výhodou je poměrně snadný (alespoň teoreticky) způsob jejich odhadu. Estimace jednotlivých pravděpodobností *n*-gramových modelů je v podstatě založena na zjišťování relativní četnosti výskytu slov, resp. posloupností slov, v trénovacích datech, tj. v rozsáhlých textových korpusech daného jazyka $\mathscr{L}$. Například pro odhad $\overline{P}(w_k \mid w_{k-2} w_{k-1})$ pravděpodobnosti $P(w_k \mid w_{k-2} w_{k-1})$ trigramového modelu platí

$$\overline{P}(w_k \mid w_{k-2} w_{k-1}) = \frac{N(w_{k-2}, w_{k-1}, w_k)}{N(w_{k-2}, w_{k-1})}, \qquad (15.17)$$

kde $N(w_{k-2}, w_{k-1}, w_k)$ je počet, kolikrát se trigram $w_{k-2} w_{k-1} w_k$ objevil v trénovacích datech a $N(w_{k-2}, w_{k-1})$ je počet, kolikrát se v trénovacích datech objevil bigram $w_{k-2} w_{k-1}$.

Uvážíme-li nyní slovník o velikosti například 50 000 slov, což je poměrně běžný rozsah slovníku v systémech rozpoznávání souvislé řeči, budeme mít $50\,000^3 = 1{,}25.10^{14}$ potenciálních trigramů. Takové množství trénovacích dat, i když v textové a nikoliv akustické formě, aby pokryla dostatečně robustně všechny možné trigramy, je téměř nemožné pořídit. Proto nedostatek trénovacích dat je hlavním problémem $n$-gramových modelů. V praxi je proto potřeba provádět odhady i těch slovních $n$-gramů, které nebyly pozorovány v trénovacích datech. Pokud bychom totiž takovým $n$-gramům přiřadili nulové pravděpodobnosti výskytu, nebylo by možné taková slova, popř. slovní kombinace, vůbec rozpoznat. Pro řešení tohoto problému jsou využívány postupy založené metodě maximální věrohodnosti doplněné dalšími technikami, např. metodou odhadu s postupným vynecháním jednoho jevu, ústupovými (angl. backing-off) a interpolačními schématy vyhlazování apod. (Ney a kol., 1997) a (Psutka a kol., 2006).

### 15.2.4 Dekódování

Nechť $P(\boldsymbol{O}|W)$ je podmíněná pravděpodobnost pozorování posloupnosti $\boldsymbol{O}$ za podmínky, že byla vyslovena posloupnost slov $W$ a nechť $P(W)$ je apriorní pravděpodobnost vyslovení posloupnosti slov $W$. Úlohou dekódování je najít k dané časové posloupnosti pozorování $\boldsymbol{O} = \{\boldsymbol{o}_1\boldsymbol{o}_2...\boldsymbol{o}_T\}$, kde $\boldsymbol{o}_t$ je pro $t = 1, 2, ..., T$ vektor příznaků odpovídající $t$-tému mikrosegmentu promluvy, takovou posloupnost slov $\hat{W} = w_1 w_2 \ldots w_M$, pro kterou dle kritéria maximální aposteriorní pravděpodobnosti platí

$$\hat{W} = \arg\max_{W} P(W)\,P(\boldsymbol{O} \mid W)\,. \tag{15.18}$$

Dekódovací algoritmus, který by provedl úplné prohledání všech posloupností slov $W$ v úloze se slovníkem o $V$ slovech a průměrné délce posloupnosti $M$ slov, by musel vyčíslit až $V^M$ posloupností slov, tj. například $10^{50}$ vyhodnocení při volbě $V = 100\,000$ a $M = 10$. Pro každou takovouto posloupnost by dále musel spočítat hodnotu součinu $P(\boldsymbol{O}|W)$ a $P(W)$. Již určení samotné $P(\boldsymbol{O}|W)$ pro jednu posloupnost slov však představuje výpočetně velmi náročný problém. Proto se z praktických důvodů při rozpoznávání neuvažují všechny možné promluvy vytvořitelné jazykovým modelem, ale jejich množství se redukuje použitím rozpoznávací sítě. Rozpoznávací síť je tvořena orientovaným grafem a obsahuje inventář slov, která jsou v síti umístěna do jejích uzlů, a možná spojení mezi nimi, která jsou reprezentována orientovanými hranami grafu. Každému slovu je ve slovníku přiřazena posloupnost fonémů, jejichž zřetězením se vytvoří dané slovo. V pravděpodobnostech hran mohou již být vzata v úvahu omezení plynoucí z jazykového modelu. Modely jsou přitom nejčastěji založeny na složených skrytých Markovových modelech fonémů, které jsou zřetězovány do složitých modelů celých promluv. Každá promluva je pak reprezentována cestou v grafu. Při rozpoznávání je třeba zjistit, jakou cestou v rozpoznávací síti se posloupnost pozorování $\boldsymbol{O}$ nejpravděpodobněji vygenerovala. Tomuto procesu, který najde k rozpoznávané posloupnosti pozorování $\boldsymbol{O}$ příslušnou posloupnost stavů složeného skrytého Markovova modelu, říkáme dekódování. Najde-li systém nejpravděpodobnější posloupnost stavů složeného skrytého Markovova modelu, najde tím i posloupnost slov v rozpoznávací síti, a tedy i posloupnost slov $\boldsymbol{W}$, kterou pak prohlásí za rozpoznanou promluvu.

Z důvodu obrovské kombinatorické exploze větvení prohledávací sítě jsou při dekódování aplikovány techniky prohledávání grafů známé z umělé inteligence. Nejčastěji jsou aplikovány postupy založené na Viterbiově algoritmu, který využívá techniku prohledávání typu beam (v každém kroku expanze rozpoznávací sítě se provede prořezávání málo perspektivních cest) anebo se využívá prohledávání typu A star, často podpořené vhodně navrženou heuristickou funkcí (Müller, 2002).

## 15.3 Porozumění mluvenému jazyku

Na rozdíl od systémů rozpoznávání řeči, které pracují často s velkými slovníky a jejichž cílem je „pouze" transformovat řečový signál do řetězce rozpoznaných slov, jsou hlasové dialogové systémy nuceny zajistit navíc jisté porozumění či interpretaci rozpoznaného řetězce slov a na základě toho provést požadovanou akci. Pro vyřešení tohoto úkolu se nabízí využít výsledků dlouholetého výzkumu lingvistů. Lingvistické metody, resp. metody výpočtové lingvistiky, se nicméně vyvíjely dlouhou dobu nezávisle na metodách zpracování mluveného jazyka. Lingvisté pracovali obvykle s předpokladem, že vstupem jejich analýzy je gramaticky korektní a většinou i sémanticky smysluplný text. Takové předpoklady však většinou nelze při zpracování vstupních promluv v hlasových dialogových systémech dodržet. Vstupem do modulu porozumění mluvenému jazyku je totiž obvykle řetězec slov, který je výsledkem ne zcela správně rozpoznané promluvy. Nepřesnosti rozpoznávání jsou způsobeny zejména faktem, že rozpoznáváme spontánní řeč běžných nezaškolených uživatelů dialogového systému, která obsahuje mnoho „neřečových událostí" a gramatických nekorektností. Vedle toho může promluva obsahovat mnoho dvojznačností a ani sémantická a pragmatická informace obsažená v promluvě nemusí být konzistentní s modelem dialogu.

Modul porozumění mluvenému jazyku v hlasových dialogových systémech zahrnuje obvykle tři složky, které se uplatňují v různé míře na odvození významu ze vstupní promluvy. Je to jednak formální *syntaktická analýza* (angl. parsing), *reprezentace znalostí* (týkající se úlohy, kterou má dialogový systém řešit) a konečně *interpretace významu* rozpoznané promluvy.

Mezi formální nástroje pro porozumění patří jednak *gramatika*, která obvykle pracuje se syntaktickými (gramatickými) kategoriemi (jako jsou například podstatná jména, přídavná jména, slovesa, slovesné fráze apod.) a stanovuje, jak mohou být tyto kategorie vzájemně spojovány či řazeny, a dále *sémantická reprezentace*, která vyjadřuje vlastnosti „jevů", které jsou zastupovány slovy, včetně jejich vztahu k realitě. Tyto dvě reprezentace mohou být zkoumány zcela odděleně, a to zejména kvůli obecnosti, nicméně z důvodů účinnosti jsou velmi často kombinovány. Jsou známy i přístupy, které sémantickou reprezentaci odvozují přímo z řetězce slov.

Nejužívanější formalismy reprezentace znalostí jsou v dialogových systémech založeny na síťových modelech. Základní idea tohoto přístupu je založena na předpokladu, že se problémová oblast zkoumá jako soubor objektů (jevů, činností apod.) a vzájemných vztahů mezi nimi. Nejznámějšími síťovými modely jsou *sémantické sítě* a *sémantické rámce*. Rozdíl mezi sémantickými sítěmi a rámci je především v hloubce strukturovanosti souboru znalostí a v zaměření na procedurální stránku reprezentace u rámců. Za-

tímco sémantické sítě jsou organizovány kolem významových jednotek (objektů, jevů) často až v průběhu řešení úlohy, vychází koncepce rámců a ještě obecnější koncepce *scénářů* při organizaci znalostí z vyšších struktur, ve kterých jsou k sobě asociovány celé bloky informací týkající se daného objektu, stereotypní situace nebo děje.

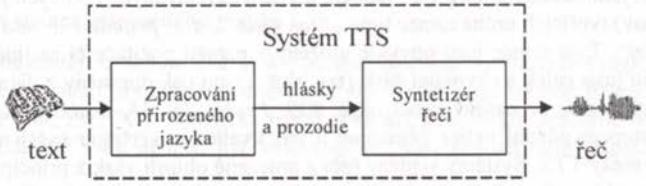
Problematika porozumění přirozenému jazyku je velmi aktuální téma, které je zkoumáno na mnoha pracovištích po celém světě. Zaměření výzkumu bylo na těchto pracovištích dlouhá léta soustředěno zejména na problémy porozumění přirozenému jazyku z textových dat (tím je myšleno z gramaticky a syntakticky korektních textů a nikoli z řetězců slov, které poskytuje systém rozpoznávání řeči). Většina zde vyvinutých technik přitom typicky generuje na základě vstupního řetězce slov tzv. parsovací (derivační) strom (Jurfansky, Martin, 2000), přičemž význam věty je následně získán na základě interpretace (sémantického) významu slov. Uveďme, že tento způsob zpracování se příliš nehodí pro syntaktickou a sémantickou analýzu spontánně pronesené promluvy. Hlavním důvodem, který hraje, vedle chybovosti vlastního rozpoznávače řeči, největší roli, jsou nespojitosti spontánní řeči, negramatická stavba promluvy a chybějící jasně vymezené hranice věty. Dosažení úplné syntaktické analýzy rozpoznané promluvy je proto velmi obtížné a často spíše nemožné. Z tohoto důvodu systémy porozumění mluvené řeči spoléhají méně na úplnou analýzu syntaktické struktury promluvy a omezují svůj „jazyk" na pokrytí pouze omezených částí promluvy vztažené k řešené aplikaci. Pro parsování spontánní promluvy je proto většinou snaha využít vhodných, spíše přímo pro danou úlohu navržených, *robustních parserů*. Hlavní myšlenkou robustního parsingu je analýza pouze „užitečných" segmentů promluvy, přičemž ostatní část promluvy se považuje pouze za „šum" nebo „výplň" (Huang a kol., 2001).

Takto navržené speciální „malé" gramatiky jsou pak využity pro prohlížení slovních mřížek rozpoznané promluvy, ve kterých se hledají takové řetězce slov, které odpovídají navržené gramatice. Alternativně lze pro tento účel využít i postupy založené na hledání klíčových slov, resp. frází.

Při parsování promluv chceme obvykle analyzovat i jejich význam či sémantický obsah. Je proto užitečné zavést pro skupiny klíčových termínů určitého významu tzv. „sémantické" třídy a navrhnout vhodnou gramatiku, která respektuje i tyto sémantické kategorie. V hlasovém dialogovém systému, který poskytuje informace o odjezdech autobusů mohou být takovými třídami například třída „čas", jejíž hodnoty se objevují v segmentech promluvy vypovídajících o časovém určení (např. ... „*ve čtvrt na jednu*", ... „*odpoledne po páté*", ... „*dneska večer*"... apod.), nebo třída „cílová stanice" (s hodnotami např. ... „*jet do Tymákova*" ... , ... „*(chci) být v Tymákově*"... apod.). Je-li pak rozpoznána taková gramatická konstrukce, je společně s rozpoznanými slovy získána i sémantická specifikace, která následně způsobí vyvolání funkcí, které provedou požadovanou akci. Uveďme, že takto definované robustní gramatiky se obvykle navrhují „ručně" (angl. handcrafted grammar), a to na základě expertního posouzení a analýzy skutečných promluv v mnoha dialozích dané tematické úlohy. Získání gramatických pravidel pro extrakci sémantické znalosti je poměrně nákladný proces, který vyžaduje účast lidského experta – návrháře dialogového systému. Přitom, jak se zvětšuje oblast pokrytí dialogu, stává se výsledná struktura složitější a těžkopádnější, což způsobuje problémy s kompatibilitou a udržováním systému. Zároveň je pro takto navrhované systémy patrná malá pružnost, je-li dialogový systém přenášen na jinou aplikační oblast.

## 15.4 Modul syntézy řeči

Nezbytnou součástí každého dialogového systému je *modul syntézy řeči*. Jeho úkolem je zajistit hlasový výstup, tj. informovat uživatele dialogového systému mluvenou řečí. V obecném dialogovém systému, kdy slovník není omezen na žádnou specifickou oblast a systém tak musí být schopen generovat libovolnou řeč, se k vytváření řeči používá technologie syntézy řeči, resp. syntézy řeči z textu.




Obr. 15.9 Zjednodušené schéma systému konverze textu na řeč.

Syntéza řeči představuje důležitou oblast problematiky zpracování řečového signálu a je již po dlouhá léta předmětem intenzivního výzkumu. Jde o proces, při němž se uměle vytváří řeč. V dnešní době se řeč téměř výhradně vytváří softwarově s využitím počítače[1]. Umělé vytváření řeči počítačem si klade za cíl „zpřirozenit" komunikaci člověka s počítačem a stát se tak rovnocenným partnerem tradiční vizuální komunikaci. Zařízení, které proces vytváření řeči provádí, nazýváme *syntetizér řeči*. Syntetizér je jádrem každého systému syntézy řeči z textu. Konečným cílem syntézy řeči je vytvářet řeč v takové formě a kvalitě, aby nebyla rozpoznatelná od řeči člověka. Syntetická řeč by tedy neměla působit jednotvárně, měla by znít přirozeně a její poslech by neměl unavovat ani vyžadovat zvýšenou pozornost. Na syntetizér řeči se můžeme dívat jako na systém, který na základě vstupní informace vytváří řeč. Onou vstupní informací zpravidla bývá společná fonetická a prozodická informace o promluvě, která se má generovat. Fonetická informace je přitom reprezentována posloupností fonémů (či hlásek) a popisuje, jaká řeč se má vytvořit. Prozodická informace definuje průběhy základních prozodických charakteristik (tj. melodie, časování a intenzity) v rámci generované promluvy a popisuje, jak se má výsledná řeč vytvářet. V obecném případě syntézy řeči z textu (nebo též konverze textu na řeč – z angl. text-to-speech, zkr. TTS) je vstupní informací systému (v tomto případě ho nazýváme *systémem konverze textu na řeč* nebo systémem TTS) „obyčejný"psaný text. Aby systém mohl na základě tohoto prostého textu vytvořit odpovídající řeč (tj. „přečíst" daný text), musí nejprve text analyzovat a odhadnout fonetické i prozodické vlastnosti vytvářené řeči. To je úkolem submodulu *zpracování přirozeného jazyka* (z angl. natural language processing, zkr. NLP).Vstupní informace může být na víc také obohacena o určité direktivy či tagy, které mohou zvýšit kvalitu vytvářené řeči[2].

[1] Před nástupem počítačů a výpočetní techniky se k vytváření umělé řeči používaly mechanická zařízení nebo později elektronické obvody (Psutka a kol., 2006).

[2] Tyto direktivy mohou vyjadřovat jisté sémantické (např. mohou označovat slovo, na které je položen důraz) nebo prozodické (např. „nestandardní" změnu melodie, zrychlenou nebo hlasitou řeč apod.) informace o větě, která se má syntetizovat. Tyto informace je obecně velmi složité a mnohdy i nemožné odvozovat z prostého

V jednoduchém dialogovém systému, kde máme zaručeno, že systém bude odpovídat s využitím menšího počtu vět, frází, či slov z předem známé specifické oblasti (např. vlakové nádraží, zastávky metra či tramvají, předpověď počasí, telefonní automat atd.), není nutné (a většinou ani žádoucí) využívat obecné technologie syntézy řeči z textu. Zde postačí využít jednodušší techniky *syntézy řeči z omezené oblasti* (nebo též syntézy s omezeným slovníkem, angl. limited-domain synthesis nebo closed-domain synthesis). V krajním případě by se chtělo říci, že nejde o syntézu v pravém slova smyslu, ale spíše o mnohem jednodušší „přehrávač" větších, předem definovaných řečových jednotek (vět, frází či slov) tvořících určité rámce typu *„Vlak číslo ... z ... přijede v ... hodin a ... minut na ... kolej"*. Tyto rámce jsou obvykle uloženy v paměti počítače či na jiném záznamovém médiu jako celek a chybějící části (tzv. sloty) jsou pak doplněny z databáze předem namluvených slov (v našem rámci např. *839*, *Prahy*, *18*, *15*, *druhou*). Řeč vytvářená tímto přístupem působí velice přirozeně a její kvalita je většinou vyšší než v případě obecné syntézy TTS. Systémy syntézy řeči z omezené oblasti však z principu nedovedou vygenerovat libovolnou řeč – je pro ně nemožné, aby například upozornily pana Nováka na to, že se má dostavit k východu z nádraží, protože tam na něj čeká manželka apod. Ve speciálních případech, kdy se stále předpokládá nasazení systému v jisté předem dané oblasti s definovaným okruhem frází, vět či slov, nicméně slovník není uzavřený a čas od času může nastat potřeba „vyslovení" nového slova. Řešením by mohlo být doplnit „přehrávač" definovaných frekventovaných frází o technologii TTS, která by našla uplatnění v případech, kdy by bylo zapotřebí syntetizovat nové slovo či slovní spojení (tj. slova mimo definovanou oblast). Jako příklad uvedeme „mluvicí bankomat", který vás pozdraví vaším jménem. Dále pak jen a pouze povídá zaběhlé fráze typu *Váš zůstatek činí dva tisíce padesát šest korun českých* nebo *Rozhodl jste se vybrat dvě stě korun českých* atd. Na tomto příkladu je zřejmé, že systém by mohl přehrávat celé fráze typu *váš zůstatek činí*, *rozhodl jste se vybrat*, *korun českých*, slova *tisíce*, *dvě* atd. Pro syntézu některých číslovek, ale hlavně jmen, by bylo vhodné, aby systém použil technologii TTS (kdokoliv si může chtít vybrat peníze – systém by měl umět vygenerovat libovolné jméno).

V obecném dialogovém systému nepředpokládáme preferenci nějakých předem definovaných frází ani neočekáváme, že by se některé věty či fráze objevovaly na výstupu systému výrazně častěji než ostatní. V takovém případě je nutné využít obecné technologie syntézy řeči z textu, nejobecnější a také nejkomplikovanější úlohu počítačové syntézy řeči. Jde o úlohu, jejímž řešením by měla být schopnost systému automaticky převést libovolný psaný text (např. knihy, noviny, úřední korespondenci, e-mail, SMS zprávy apod. – nejen tedy omezený počet předem definovaných frází) na mluvenou řeč, a to nejlépe v takové formě a kvalitě, jakou by stejný text přečetl člověk s dobrým přednesem. Je zřejmé, že jde o mnohem složitější problém než v případě jednoduchého dialogového systému. Podívejme se nyní podrobněji na jednotlivé komponenty systému TTS. Detailní popis problematiky syntézy řeči z textu lze nalézt například v pracích (Psutka a kol., 2006) nebo (Dutoit, 1992).

Na obrázku 15.9 je znázorněno zjednodušené schéma obecného systému TTS. Skládá se ze dvou hlavních modulů: zpracování přirozeného jazyka (NLP) a vlastního synteti-

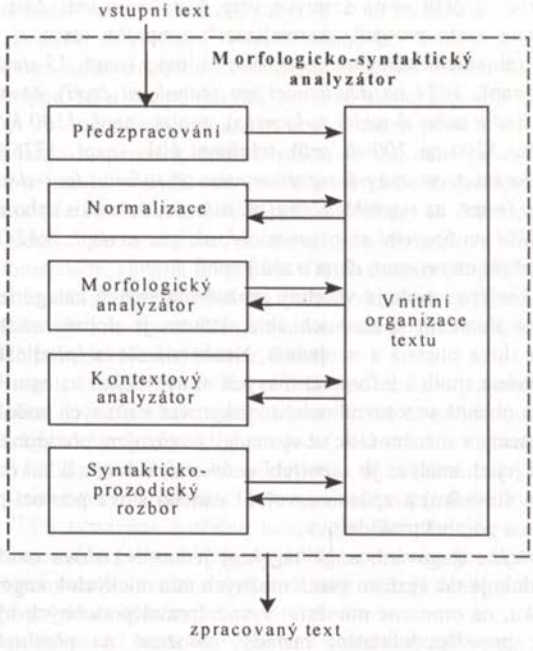
---

textu. Syntéza řeči z „obohaceného" textu je známa pod názvem *syntéza řeči z konceptu* (angl. concept-to-speech).

zéru řeči. Oba moduly jsou na sobě relativně nezávislé a oba mají svou důležitost. Modul NLP má za úkol zpracovat text na vstupu systému, analyzovat ho a získat pokud možno co nejvíce informací o tom, co se má syntetizovat (někdy se tato činnost označuje jako syntéza vyšší úrovně, angl. high-level synthesis). Tento modul provádí tzv. těsnou fonetickou transkripci textu. Na výstupu se potom často objevuje posloupnost hlásek příslušného jazyka doplněná o prozodické značky. Pak přichází na řadu syntetizér řeči, který na základě této informace generuje řeč – provádí vlastní syntézu řeči na signálové úrovni.

### 15.4.1 Zpracování přirozeného jazyka

Modul NLP nejčastěji obsahuje dva bloky: *modul fonetické transkripce* (angl. letter-to-sound, zkr. LTS, nebo grapheme-to-phoneme, zkr. GTP) a *generátor prozodie* (angl. prosody generator, zkr. PG). Fonetická transkripce ani generování prozodie na základě slov textu odděleně však nemusí být jednoznačné. Jako příklad nejednoznačnosti při




Obr. 15.10 Schéma morfologicko-syntaktického analyzátoru (literatura).

fonetické transkripci uvedeme například slovo panice, které se může vyslovovat „tvrdě" s [n] (pokud jde o slovo *panika*) anebo „měkce" s [ň] ve slově *panic*. Abychom podobná slova přepsali správně foneticky, musíme znát dodatečné morfologické, kontextové nebo syntaktické, popř. i sémantické, informace. Proto je vhodné do modulu NLP doplnit další blok, *morfologicko-syntaktický analyzátor* (angl. morpho-syntactic analyzer, zkr. MSA), který provádí morfologickou a syntaktickou analýzu vstupního textu.

Při morfologicko-syntaktické analýze textu se používají metody výpočetní lingvistiky (Manning, Schütze, 1999), (Hajičová a kol., 2002). Schéma typického morfologicko-syntaktického analyzátoru je zobrazeno na obr. 15.10. V současných systémech TTS nejsou jednotlivé komponenty analyzátoru řazeny sekvenčně, ale spíše paralelně. Každá z komponent totiž může do zpracovávaného textu v průběhu zpracování přidat novou informaci, ostatním komponentám do té doby nepřístupnou. Tato nová informace pak může způsobit opětovné zavolání příslušné komponenty, která na jejím základě například zpřesní odhad některých vlastností zpracovávaného textu. Morfologicko-syntaktický analyzátor se obvykle skládá z komponent, které jsou popsány v následujícím výkladu (Dutoit, 1997).

P ř e d z p r a c o v á n í t e x t u většinou funguje jako interface mezi vstupním textem a vnitřní organizací textu v MSA. Provádí obvykle unifikaci textu, tj. detekuje typ vstupního textu (prostý text, HTML, XML, e-mail atd.), filtruje znaky textu (odstraňuje například některé „nadbytečné" znaky formátování, bílé znaky apod.). Dále může detekovat strukturu textu: rozdělit jej na odstavce, věty, či dokonce větší části jako kapitoly.

Blok *normalizace textu* provádí „normalizaci" vstupního textu, tj. do plné slovní formy (ve správné mluvnické kategorii) přepisuje číslovky (např. *15 studentů* na *patnáct studentů*), datum (např. *1974* na *devatenáct set sedmdesát čtyři*), časové údaje (např. *12:00* na *dvanáct hodin* nebo *dvanáct nula nula*), peníze (např. *1500 Kč* na *tisíc pět set korun českých* nebo *$200* na *200 dolarů*), telefonní čísla (např. *377632530* na *tři sta sedmdesát sedm šest set třicet dva pět set třicet* nebo *tři sedm sedm šedesát tři dvacet pět třicet* atd.), zkratky (*např.* na *například*, *Ing.* na *inženýr*, *inženýra* nebo *inženýru* apod.), akronymy (např. *USA* na *Spojené státy americké*, ale pozor např. *NATO* zůstává *NATO* atd.), symboly (např. *%* na *procent*, *&* na *a* atd.) apod.

*Morfologická analýza* navrhuje všechny možné mluvnické kategorie každého slova věty izolovaně bez kontextu sousedních slov. Přitom je dobré rozdělit slova podle slovních druhů na slova ohebná a neohebná. Neohebná slova (předložky, spojky atd.) mohou být uchovávána spolu s informací o svých mluvnických kategoriích ve slovníku. Naproti tomu slova ohebná se v textu mohou vyskytovat v různých podobách (v různých pádech, rodech, časech, v různém čísle) a vyznačují se různými předponami či příponami a koncovkami. Při jejich analýze je zapotřebí detekovat slovní základ (kmen – většinou pomocí rozsáhlého slovníku) a způsob utvoření daného slova pomocí předpon, přípon a koncovek (většinou pomocí pravidel).

*Kontextová analýza* (tagování, angl. tagging) jednotlivá slova uvažuje v kontextu okolních slov a redukuje tak seznam všech možných mluvnických kategorií slov, získaný v předchozím kroku, na omezené množství vysoce pravděpodobných hypotéz. Používá dva typy metod: pravděpodobnostní metody, založené na přechodových pravděpodobnostech mezi sousedními mluvnickými kategoriemi dvou slov, a deterministické (nepravděpodobnostní) metody, které využívají pravidla typu ano/ne, pomocí nichž přijmou nebo odmítnou určitou kombinaci syntaktických kategorií (jsou založeny na klasifikačních a regresních stromech, angl. classification and regression tree, zkr. CART).

Úkolem *syntakticko-prozodického rozboru* (angl. parsing, zkr. SPP) je ve zpracovávané větě nalézt větné úseky (tzv. syntakticko-prozodické fráze či jen fráze), které úzce souvisejí s očekávanou prozodickou realizací věty. Rozbor je možné provádět

pomocí ručně odvozených heuristik, kdy se hranice mezi frázemi určují na základě pozic čárek, středníků, pomlček, závorek apod., gramatik vzniklých rozšířením klasických bezkontextových gramatik, nebo s využitím korpusově orientovaných metod (např. klasifikačních a regresních stromů nebo markovovských řetězců), které využívají rozsáhlého textového korpusu s předem označkovanými hranicemi mezi frázemi k automatickému odvození pravidel pro určení hranic mezi frázemi.

Blok *fonetické transkripce* zpracovává text na výstupu morfologicko-syntaktického analyzátoru a jeho úkolem je popsat výslovnostní (fonetickou) podobu zpracovávaného textu, tj. odhadnout reprezentaci textu pomocí posloupnosti fonémů (resp. hlásek) daného jazyka. Při automatickém vytváření řeči v dialogovém systému se fonetická transkripce provádí bez přímé účasti člověka-experta. Pro analytické jazyky (např. angličtinu), ve kterých neexistuje mnoho tvarů odvozených od stejného slova (např. pomocí koncovek) se transkripce většinou provádí na základě fonetického slovníku, který obsahuje ortografickou (textovou) a fonetickou podobu jednotlivých slov. Pro flexivní jazyky (např. češtinu a ostatní slovanské jazyky), v nichž existuje velké množství tvarů odvozených od stejného slova (např. vlivem skloňování nebo použitím mnoha koncovek), z důvodu obrovského množství slov nebývá efektivní využití fonetického slovníku. Fonetická transkripce se u těchto jazyků provádí pomocí fonetických transkripčních pravidel, která je možné navrhnout analyticky, často s pomocí člověka-experta (jde vlastně potom o jakýsi expertní systém) (Psutka, 1995), nebo je možné využít metody strojového učení a pravidla hledat automaticky na základě dostatečně reprezentativní trénovací množiny (Zelinka, Müller, 2004). Oba přístupy se často používají současně: v češtině se pro domácí slova většinou používají pravidla a slova cizího původu se pak uchovávají ve fonetickém slovníku výjimek. Další možností využití obou přístupů je rozdělení slov na kmen, předponu, příponu a koncovku; kmeny je pak možné uchovávat spolu s jejich výslovností ve fonetickém slovníku a při zpětném složení slova potom lze využít fonetických pravidel (Kanis, Müller, 2004).

Blok *generování prozodie* je dalším důležitým modulem zpracování přirozeného jazyka. Jeho úkolem je na základě zpracovávaného textu odhadnout a poté generovat průběhy prozodických charakteristik (tj. melodie, časování a intenzity) vytvářené řeči. Správné generování prozodických charakteristik do značné míry přispívá k vyšší přirozenosti syntetické řeči. Automatickému generování prozodie by tedy měla být při vývoji systému TTS věnována patřičná pozornost. Bohužel jde o extrémně složitou úlohu, neboť různorodost akustického vyjádření prozodie je částečně nezávislá na textové reprezentaci promluvy, což do značné míry ztěžuje úlohu odhadu prozodického vyjádření promluvy v úloze syntézy řeči z textu.

*Melodie* (resp. *intonace*) *promluvy* souvisí s průběhem frekvence základního hlasivkového tónu ($F_0$), resp. výškou hlasu. Generování intonace v úloze syntézy řeči z textu je možné obecně popsat ve dvou krocích. První krok zahrnuje vytvoření symbolického popisu intonace (resp. kompletní prozodie) promluvy jako doplněk k ortografické a fonetické transkripci promluvy a v druhém kroku pak zvolený intonační model na základě tohoto popisu generuje výslednou intonační konturu (konturu $F_0$). Použitý symbolický popis prozodie přitom zpravidla předurčuje intonační modely vhodné pro generování intonace. Všechny symbolické popisy prozodie jsou charakteristické snahou co nejvěrněji (v rámci definovaných symbolů) popsat průběh prozodie (a zejména

intonace) dané promluvy. Nejčastěji se přitom omezují na popis „význačných" událostí v průběhu základního hlasivkového tónu (tj. hlavně na přízvuky a koncové tóny). Popis přitom téměř výlučně probíhá v akustické oblasti (zkoumá se přímo průběh základního hlasivkového tónu). Často se intonační průběhy také popisují pomocí lingvisticky motivovaných pravidel (Romportl, 2004). O generování výsledného průběhu základního hlasivkového tónu se starají intonační modely. V současných systémech TTS se nejčastěji používají akustické modely intonace, které vycházejí z akustické reprezentace intonace pomocí průběhu základního hlasivkového tónu v čase. Z hlediska samotného generování kontury $F_0$ je můžeme dále rozdělit na *modely generující intonaci podle pravidel*, které intonační průběh promluvy generují podle lingvisticky motivovaných pravidel, *parametrické modely intonace*, které průběh základního hlasivkového tónu a zejména důležité události v tomto průběhu (přízvuk, hranice syntakticko-prozodických frází) reprezentují parametricky pomocí vhodně zvoleného parametrického modelu, např. Fujisakiho model (Fujisaki, 1992), model Tilt (Taylor, 1998) nebo parametrické modely vycházející ze symbolického popisu ToBI (Haung, 2001), a *modely korpusově orientovaného generování intonace*, jež využívají korpusy přirozených řečových nahrávek, na jejichž základě se automaticky nastavují parametry generátoru $F_0$, např. pomocí neuronových sítí (Tučková, Matoušek, 2003), skrytých Markovových modelů, klasifikačních a regresních stromů, a výsledná intonace promluvy se vytváří řetězením dílčích intonačních kontur natrénovaných či přímo se vyskytujících v korpusu (Romportl, 2006). Uvedené rozdělení intonačních modelů není disjunktní, v praxi často dochází k překrývání jednotlivých typů modelů. Parametrické i korpusově orientované modely mohou při generování výsledného intonačního průběhu využívat zvláštních heuristik, tj. různě navržených pravidel. Též je možné například vytvořit korpusově orientovaný parametrický model generování intonace atd.

Základními jednotkami pro generování intenzity jsou především hlásky (popř. fonémy). Jejich intenzita se obvykle určuje pomocí statistik na základě rozsáhlých řečových korpusů, často ve spojení se specifickými informacemi o pozici hlásky ve slově, větném úseku či ve větě, přízvučnosti, popř. nepřízvučnosti, apod. (např. pomocí techniky CART). Často se používají pravidla (Romportl, 2004). Možné je též použít části reálných kontur intenzity analogických ke konturám $F_0$. Poznamenejme, že v české řeči bývá intenzita považována za nejméně významnou prozodickou charakteristiku. Proto ji lze v českých systémech TTS v porovnání s $F_0$ mnohdy zanedbat.

Problematika generování časování v systémech konverze textu na řeč zahrnuje odhad všech aspektů spojených s časováním řeči, tj. s trváním (rychlostí) segmentů řeči, umísťováním pauz, rytmickým členěním promluv pomocí přízvuku atd. Základními jednotkami *modulace trvání* jsou v současných systémech TTS opět především segmenty na úrovni fonémů. Modely trvání těchto segmentů využívají znalostí o artikulačních a fonologických aspektech (které opět zahrnují například pozici hlásky ve slově, větném úseku, či ve větě, přízvučnost, popř. nepřízvučnost, apod.) těchto segmentů ve formě pravidel nebo statistik spočtených z rozsáhlých řečových korpusů (např. pomocí klasifikačních a regresních stromů nebo neuronových sítí). Pro generování pauz se využívá informace o syntakticko-prozodické struktuře věty, neboť pauzy se umísťují na hranice mezi některými frázemi. Mohou být různého trvání: mezi dvěma úseky se slabší textovou koherencí se umísťuje delší pauza a naopak mezi dva úseky se

silnější textovou koherencí se vkládá krátká pauza. K tomuto účelu je možné využít například syntakticko-prozodický parser natrénovaný pomocí transkripčního systému ToBI. Generování přízvuku se obecně realizuje změnou všech tří základních prozodických charakteristik – frekvence základního hlasivkového tónu, trvání i intenzity. V systémech TTS se zatím většinou pracuje pouze se slovním přízvukem (pro detekci větného přízvuku by bylo totiž zapotřebí pochopit význam věty). Hlavním úkolem je tedy nalézt pro každé slovo přízvučnou slabiku. U jazyků s pevným přízvukem (mezi něž patří i čeština, v níž se přízvuk nachází většinou na první slabice slova) jde o relativně jednoduchou úlohu. Některá krátká slova však přízvuk neobsahují a při generování přízvuku je tedy třeba s takovými slovy počítat (a uchovávat je například ve speciálních slovnících).

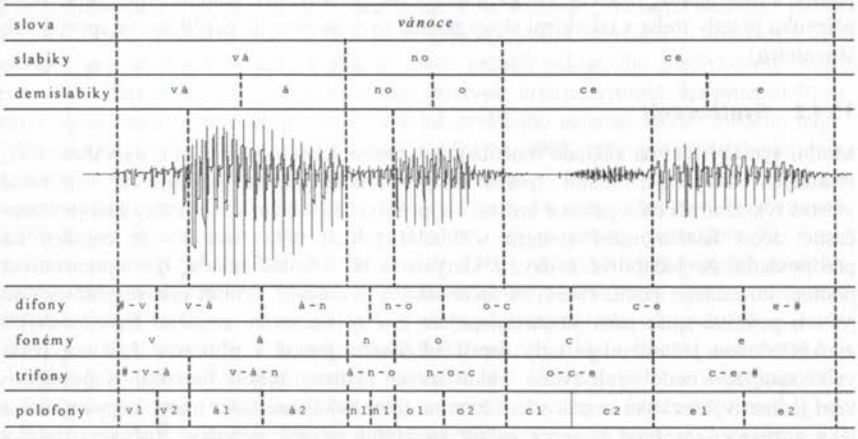
### 15.4.2 Syntéza řeči

Modul syntézy řeči na základě fonetických a prozodických informací o vytvářené řeči, získaných na výstupu modulu zpracování přirozeného jazyka, generuje řeč – provádí vlastní syntézu řeči na signálové úrovni. Nejpoužívanější technikou syntézy řeči je v současné době *konkatenační syntéza*. Základní princip této techniky je založen na předpokladu, že jednotlivé zvuky, z kterých se řeč obecně skládá, lze reprezentovat pomocí konečného počtu *řečových (akustických) jednotek*. Termín řečová jednotka se přitom používá spíše jako abstraktní pojem pro pojmenování stejného typu řečových zvuků(řečovou jednotkou je tedy například foném, jemuž v přirozené řeči odpovídá velké množství podobných zvuků – akustických realizací tohoto fonému). K pojmenování jednotlivých zvuků se pak užívá termínu (akustická) *realizace* (resp. *segment*) nebo také *zástupce*, *instance* či *reprezentant* konkrétní řečové jednotky. Řečové jednotky (resp. jejich realizace) jsou uloženy v inventáři řečových (akustických) jednotek. Tímto inventářem obvykle bývá označkovaný(segmentovaný) řečový korpus, který vedle vlastních řečových dat také obsahuje hranice jednotlivých (realizací) řečových jednotek, nejčastěji fonémů. Vhodně předem vybrané zástupce jednotek lze uchovávat v databázi řečových segmentů, odkud jsou během syntézy vybírány a vhodným způsobem řetězeny. Výsledná syntetická řeč tedy vzniká řetězením (konkatenací) řečových segmentů.

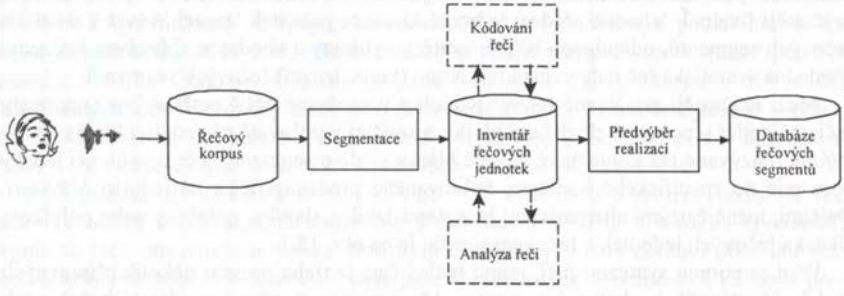
Mezi nejčastěji používané řečové jednotky v současné době patří *difony* (segmenty řeči začínající v polovině předchozí hlásky a končící v polovině následující hlásky) nebo *trifony* (nazývané též kontextově závislé hlásky – jde o segmenty řeči o velikosti hlásky posazené do specifického kontextu definovaného předcházející a následující hláskou). Dalšími, méně častými alternativami jsou demislabiky, slabiky, polofony nebo polyfony. Ukázka řečových jednotek v řečovém signálu je na obr. 15.11.

Před samotnou syntézou řeči, mimo reálný čas, je třeba provést několik přípravných kroků. V případě konkatenační syntézy jde zejména o přípravu vlastních řečových jednotek – tj. přípravu inventáře řečových jednotek. Fáze přípravy inventáře řečových jednotek je schematicky znázorněna na obr. 15.12. Inventář se vytváří na základě řečového korpusu, souboru řečových nahrávek speciálně vybraných vět (Matoušek a kol., 2001). Velikost korpusu se pohybuje podle charakteru úlohy od jedné až po desítky hodin řeči. Poté je třeba v tomto korpusu nalézt instance jednotlivých řečových jednotek. Tomuto kroku se říká segmentace řečového korpusu – vymezení hranic řečových jednotek v korpusu. Nejpoužívanější technikou automatické segmentace řeči je v sou-

časné době aplikace skrytých Markovových modelů, techniky známé z oblasti automatického rozpoznávání řeči (Donovanr, Woodlandp, 1999), (Matoušek a kol., 2003). Před samotnou syntézou řeči lze dále pro každou jednotku z inventáře řečových jednotek „předvybrat" zástupce této jednotky (jednoho, několik nebo třeba všechny realizace) a uložit je do databáze řečových segmentů. V případě parametrické konkatenační syntézy se navíc provádí parametrizace řečových segmentů: jednotlivé segmenty jsou popisovány pomocí vektorů parametrů a v této formě ukládány do databáze parametrických segmentů. Tato fáze se nazývá analýza řeči. Volitelně je také




Obr. 15.11 Ukázka segmentace řečového signálu slova *vánoce* na základní řečové jednotky. Symbol # reprezentuje pauzu mezi slovy.

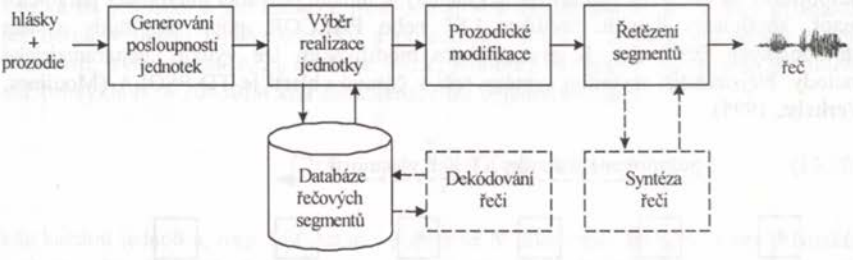



Obr. 15.12 Blokové schéma procesu vytváření inventáře řečových jednotek a databáze řečových segmentů. Volitelné bloky jsou označeny čárkovaně.

možné provést kódování řečových segmentů. V časové oblasti se používají různé techniky typu DPCM, ADPCM, $\mu$-law, $A$-law apod. Ve frekvenční oblasti je možné využít například vokodérů.

Po provedení popsaných přípravných kroků již můžeme přistoupit k samotnému generování syntetické řeči, které samozřejmě probíhá v reálném čase. Obecné schéma

generování řeči konkatenační technikou syntézy řeči je zobrazeno na obr. 15.13. Předpokládá se, že je k dispozici podrobná informace o promluvě, kterou chceme syntetizovat. Touto informací bývá nejčastěji posloupnost hlásek či fonémů doplněná o prozodické značky. Z této informace si syntetizér vyrobí sekvenci řečových jednotek, s kterými umí pracovat, a z celkové prozodické informace o syntetizované promluvě odvodí prozodické vlastnosti jednotlivých jednotek. Pro každou řečovou jednotku pak z databáze segmentů vybere jeden segment, který ji během syntézy dané konkrétní promluvy reprezentuje. Obsahuje-li jich více, je vhodné vybrat takový segment, který nejlépe postihuje aktuální (syntetický) fonetický i prozodický kontext. V případě, že řečové segmenty byly dříve zakódovány, je třeba vybrané segmenty řečových jednotek dekódovat pomocí dekodéru řeči. Protože segmenty uložené v databázi vykazují obecně různé prozodické vlastnosti často odlišné od požadovaných, bývá obvykle dalším krokem nastavení prozodie. V bloku „prozodické modifikace" se prozodie databázových

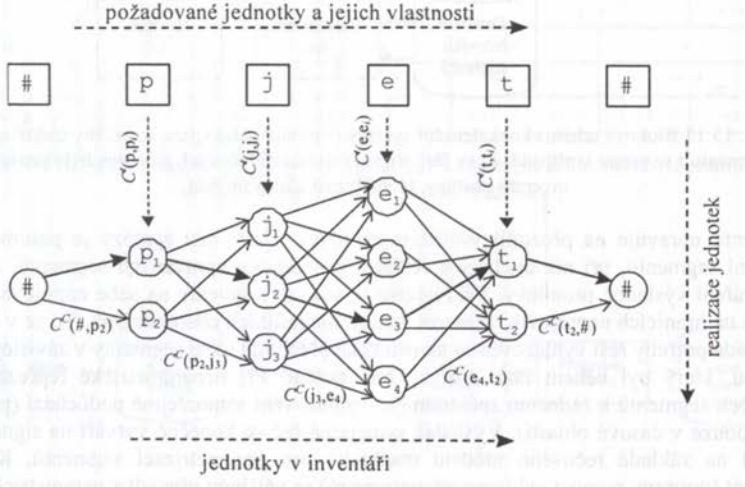



Obr. 15.13 Blokové schéma konkatenační syntézy. Volitelné bloky jsou označeny čárkovaně. Implementují se pouze v případě, že ve fázi přípravy databáze řečových jednotek byly provedeny inverzní postupy, tj. analýza a kódování řeči.

segmentů upravuje na prozodii výsledné promluvy. Další fází syntézy je potom fáze řetězení segmentů, při níž dochází k řetězení prozodicky upravených segmentů, a tím k vytváření výsledné promluvy. Při řetězení je snahou segmenty na sebe napojovat tak, aby se na hranicích neprojevily skokové změny spektrálních charakteristik. To se v praxi v případě potřeby řeší vyhlazováním akustických přechodů mezi segmenty v závislosti na modelu, který byl během fáze analýzy řeči použit. Při neparametrické reprezentaci řečových segmentů k žádnému spektrálnímu vyhlazování samozřejmě nedochází (pracujeme pouze v časové oblasti). Výsledná syntetická řeč se konečně vytváří na signálové úrovni na základě řečového modelu použitého pro parametrizaci segmentů. Každý segment (popsaný pomocí vyhlazených parametrů) se většinou převádí z parametrické do časové oblasti a řetězí se do výstupního proudu syntetické řeči. Modul signálové syntézy řeči je tak přesným protějškem modulu analýzy řeči, použitého pro parametrizaci. V případě neparametrické syntézy proces signálové syntézy opět odpadá a provádí se pouze samotné řetězení.

Na základě uvedeného obecného schématu konkatenační syntézy řeči se v současné době nejčastěji konstruují dva odlišné typy systémů syntézy řeči: systém s jednou instancí každé řečové jednotky a systém s více instancemi řečových jednotek (Matoušek a kol., 2004). Systémy s jednou instancí každé řečové jednotky obvykle využívají menší řečové korpusy (řádově hodiny řeči) a samozřejmě také menší databáze řečových seg-

mentů. Tím je možné je využívat i v méně výkonných zařízení, kromě starších počítačů i například různých kapesních počítačů nebo mobilních telefonů. Vyznačují se „průměrnou" kvalitou výsledné syntetické řeči. Během vytváření inventáře řečových jednotek se pro každou řečovou jednotku mimo reálný čas vybírá jeden „dostatečně průměrný" reprezentant, který se potom používá při vlastní syntéze řeči. Snahou je vybrat takového reprezentanta, jehož prozodické i spektrální vlastnosti je možné během syntézy v rozumné míře modifikovat a naplnit tak požadavky dané syntetizovaným kontextem. K prozodickým modifikacím se nejčastěji používají parametrické metody, např. lineární prediktivní syntéza (resp. RELP syntéza využívající informaci o buzení hlasového traktu uchovanou v reziduálním signálu lineární predikce) (Dutoit, 1997), sinusoidální syntéza (např. harmonický a šumový model vytváření řeči, HNM (Stylianou, 2001)) nebo parametrické techniky PSOLA (Moulines, Charpentier, 1990) (např. RELP-PSOLA nebo FD-PSOLA). Modifikace spektrálních vlastností řeči (například vyhlazování spektrálních nespojitostí na hranicích řetězených segmentů) je možné provádět interpolací parametrů (např. koeficienty lineární predikce LSF nebo PARCOR, popř. amplitudy a fáze harmonických frekvencí). K prozodickým modifikacím lze využít i neparametrické metody. Nejznámější metodou syntézy řeči v časové oblasti je TD-PSOLA (Moulines, Verhelst, 1995).




Obr. 15.14 Ilustruje hledání optimální posloupnosti řečových jednotek pro slovo pět. Symbol # značí pauzu. Zvýrazněné čáry naznačují optimální cestu sítí realizací řečových jednotek a vybranou posloupnost jednotek.

Systémy s více instancemi řečových jednotek používají rozsáhlé řečové korpusy (řádově desítky hodin řeči) a databáze řečových segmentů obsahují mnoho realizací každé řečové jednotky (často i všechny instance z korpusu). Během syntézy řeči je cílem pro danou posloupnost řečových jednotek a dané prozodické charakteristiky popisující vytvářenou promluvu nalézt optimální posloupnost instancí odpovídajících řečových

jednotek v databázi řečových segmentů. Tato technika je známa jako *technika výběru jednotek* (z angl. unit selection) nebo obecně *korpusově orientovaná syntéza řeči* (angl. corpus-based speech synthesis). Při hledání se berou v úvahu dvě hodnoticí funkce (viz obr. 15.14) (Hunt, Black, 1996). *Cena cíle*, označíme ji $C^t(t_i, u_i)$ (z angl. target cost), popisuje, jak se databázová jednotka $u_i$ liší od cílové požadované jednotky té, kterou má během syntézy reprezentovat ($i = 1, ..., N$, kde $N$ je počet jednotek v syntetizované promluvě). *Cena konkatenace* $C^c(u_{i-1}, u_i)$ (z angl. concatenation cost) na druhou stranu vyjadřuje, jak dobře se sousední jednotky $u_{i-1}$ a $u_i$ řetězí. Klíčem k úspěchu bývá správná definice obou hodnoticích funkcí a vyvážený odhad jejich parametrů. Pro cenu cíle platí

$$C^t(t_i, u_i) = \sum_{j=1}^{N^t} w_j^t C_j^t(t_i, u_i), \tag{15.19}$$

kde každá jednotka a její instance jsou popsány $N^t$-dimenzionálními vektory příznaků, jež berou v úvahu jejich fonetický a prozodický kontext, např. typ jednotky a její vlastnosti, trvání instance, $F_0$ instance, pozici jednotky ve slabice, slově nebo promluvě atd. (obvykle $N^t$ = 20–30). Cenu konkatenace lze vyjádřit ve tvaru

$$C^c(u_{i-1}, u_i) = \sum_{j=1}^{N^c} w_j^c C_j^c(u_{i-1}, u_i), \tag{15.20}$$

kde každou jednotku, resp. její instanci popíšeme $N^c$-dimenzionálními vektory příznaků, v tomto případě je vhodné volit příznaky vyjadřující spektrální charakteristiky (např. koeficienty MFCC), energii a $F_0$ počítané přímo z řečového signálu v místě řetězení (v našem případě $N^c$ =3). Je možné použít i diference těchto hodnot. Za zmínku stojí speciální případ, kdy $u_{i-1}$ a $u_i$ jsou sousední jednotky v řečovém korpusu (tj. „fyzicky" spolu sousedí v určitém slově). Konkatenace takových jednotek je samozřejmě přirozená, a proto cena konkatenace je rovna nule. Tímto způsobem popisovaná metoda řeči podporuje používání tzv. neuniformních řečových jednotek. V jistém případě totiž, pokud bude v databázi uložena například celá věta, je možné tuto větu použít jako celek (navenek jako jednu jednotku) při syntéze řeči (samozřejmě za předpokladu, že se cílová specifikace jednotek, ze kterých se skládá, neliší od vlastností jednotek uložených v databázi). Celková cena výběru $N$ jednotek je pak dána součtem cen cíle a konkatenace

$$C(t_1^N, u_1^N) = \sum_{i=1}^{N} C^t(t_i, u_i) + \sum_{i=1}^{N} C^c(u_{i-1}, u_i) + C^c(\#, u_i) + C^c(u_N, \#), \tag{15.21}$$

kde # označuje pauzu a $C^c(\#, u_1)$ a $C^c(u_N, \#)$ definuje počáteční a koncové podmínky za předpokladu, že daná promluva začíná a končí pauzou. Dosadíme-li do vztahu (15.21) rovnice pro výpočet jednotlivých cen (15.19) a (15.20), dostaneme výsledný vztah pro výpočet celkové hodnoticí funkce

$$C(t_1^N, u_1^N) = \sum_{i=1}^{N} \sum_{j=1}^{N^t} w_j^t C_j^t(t_i, u_i) + \sum_{i=2}^{N} \sum_{j=1}^{N^c} w_j^c C_j^c(u_{i-1}, u_i) + C^c(\#, u_1) + C^c(u_N, \#). \tag{15.22}$$

Úkolem metody syntézy výběru jednotek je potom najít takovou posloupnost jednotek $\overline{u_1^N}$ v databázi, pro niž platí, že celková cena je minimální, tj.

$$\overline{u_1^N} = \underset{u_1,\ldots,u_N}{\arg\min} C(t_1^N, u_1^N). \tag{15.23}$$

K nalezení optimální posloupnosti jednotek se téměř výhradně používá Viterbiův algoritmus (vyčerpávající prohledávání celého stavového prostoru v reálném čase je z výpočetního hlediska nerealizovatelné). Z praktického hlediska (a zejména v případě rozsáhlých databází) je i u Viterbiova algoritmu prohledávání obvykle vhodné použít prořezávání stavového prostoru (Hunt, 1996). Po nalezení optimální sekvence instancí jednotek se jednotlivé segmenty řetězí stejným způsobem jako v případě systému s jednou instancí každé řečové jednotky. V případě potřeby je opět možné modifikovat prozodické i spektrální vlastnosti řetězených segmentů některou z metod zmíněných dříve. Díky schématu výběru jednotek však modifikace probíhají v mnohem menší míře nebo dokonce vůbec.

Dosud popsané metody výběru jednotek předpokládají, že je před vlastním výběrem jednotek v čase syntézy vygenerována kontura $F_0$ a pro všechny jednotky odhadnuto trvání a intenzita. Prozodické parametry jsou tak určovány explicitně nějakou z metod popsaných v odst. 15.4.1. Poslední výzkumy se zabývají i implicitním generováním prozodie přímo v rámci techniky výběru jednotek. Volbou vhodných obecně lingvistických příznaků (jako je například pozice hlásky v prozodicko-syntaktické frázi, přítomnost přízvuku apod.) a vhodně nastaveného algoritmu výběru je možné vybírat takové realizace řečových jednotek, které kopírují prozodické vlastnosti požadované v syntetické řeči, aniž by bylo nutné tyto prozodické vlastnosti explicitně generovat (Tihelka, Matoušek, 2006). Nepracujeme tedy přímo s hodnotami prozodických parametrů, ale spíše s příznaky a symboly, které je „skrytě" popisují. Hovoříme o *symbolickém popisu prozodie.*

Na závěr našeho výkladu o korpusově orientované syntéze uveďme, že jde v současné době o nejpopulárnější a také o nejprogresivnější techniku syntézy řeči. Ukázalo se, že rozsáhlé korpusy přirozené lidské řeči přispívají ke zvýšení kvality syntetické řeči. Jisté nebezpečí, kterému se při použití konečného řečového korpusu nikdy úplně nevyhneme, představují tzv. „řídké řečové kontexty", tj. takové segmenty řeči, které se ve zdrojovém řečovém korpusu vyskytují jen velmi zřídka nebo dokonce vůbec. Takové segmenty lze na principu korpusově orientované syntézy řeči jen velmi těžko syntetizovat, většinou nezbývá než sáhnout po podobném (i když teoreticky i velmi odlišném) řečovém segmentu, což může velmi negativně ovlivnit kvalitu výsledné vytvářené řeči. Protože jde o řídké jevy (i v syntetické řeči se objeví velmi zřídka), je jim zatím věnována jen omezená pozornost.

## 15.5 Dialogový manažer

Funkce hlasového dialogového systému může být definována jako rozhodovací proces, který probíhá v prostoru stavů, akcí a strategií dialogu. *Stav dialogu* je veškerá znalost,

kterou systém získal vzájemnou interakcí jednotlivých komponent dialogového systému s uživatelem a aplikací, včetně reprezentace stavů i úlohy řešené dialogem. Stav dialogu se mění, když je uskutečněna *akce dialogu*, přičemž souborem akcí jsou všechny možné akční zásahy (např. nápověda uživateli, obdržení informace od uživatele, přístup k databázi apod.), které může dialogový systém provést (Cox a kol., 2000). *Strategie dialogu* specifikuje příští akci, která je vybírána pro každý stav, který byl dosažen v průběhu dialogu. Připomeňme též, že strategie dialogu by měla být navržena tak, aby zajistila *cílové chování*, tj. uspokojení požadavků uživatele. Za získání požadované informace (znalosti) o stavu dialogové úlohy, za výběr vhodné strategie a odpovídajících akcí dialogového systému, které zajišťují cílové chování, je odpovědný *dialogový manažer*. Úlohou dialogového manažeru je tedy jednak organizovat vzájemné vazby mezi jednotlivými komponentami dialogového systému, dále využívat efektivně dostupné zdroje znalostí a současně zajistit komunikaci dialogového systému s uživatelem a externími aplikacemi (databázemi, řízenými systémy apod.), a to tak, aby bylo zajištěno cílové chování.

### 15.5.1 Vedení dialogu

Způsob vedení dialogu lze zkoumat z různých hledisek. Asi nejpřirozenějším způsobem se zdá být pohled zaměřený na to, jakým způsobem dialogový manažer zachází s promluvou uživatele, jak reaguje na případné nejasnosti v promluvě způsobené dialogovým systémem (např. chybně rozpoznaná promluva, nesprávná interpretace promluvy apod.), uživatelem (promluva syntakticky, sémanticky nebo pragmaticky nekonzistentní s probíhajícím dialogem) či komunikačním kanálem a prostředím, ve kterém je dialog uskutečněn (např. zkreslení telefonního kanálu, okolní hluk apod.). Dalším hlediskem může být například to, jakým způsobem dialogový manažer ověřuje, že promluva rozpoznaná a interpretovaná dialogovým systémem se shoduje s tím, co bylo zamýšleno uživatelem (McTear, 2002).

Pro podporu činnosti dialogového manažeru jsou standardně využívány různé *zdroje znalostí*, které přispívají k dosažení správné interpretace promluvy uživatele a v případě nejasností se podílejí na zformulování správného dotazu, ve kterém dialogový systém žádá nové nebo doplňující informace, popř. potvrzení správnosti toho, co již bylo interpretováno.

#### *15.5.1.1 Modelování zdrojů znalostí pro vedení dialogu*

Znalostní modely zahrnují obvykle dvě části, které závisí více či méně na aplikační oblasti. Jde o statickou část modelů, která je navržena při přípravě dialogového systému, a pak o dynamickou část, která je postupně vytvářena a modifikována po celou dobu trvání dialogových výměn mezi uživatelem a systémem (Minker, Néel, 2002).

**Model prostředí.** Modelování prostředí je užitečné v případech, jsou-li dialogová spojení uskutečňována například různými přenosovými kanály (ISDN telefonními linkami, mobilními telefony, mobilními pojítky apod.) anebo v prostředích s různou mírou a kvalitou aditivního šumu. Budou-li například očekávána telefonní spojení z jedoucího

automobilu či vlaku, popř. bude-li dialogový systém konstruován pro vojenské účely, bude muset být dialogový systém vybaven mechanismy pro potlačování případných rušivých projevů a v případě, že rušení překročí mez, při které hrozí soustavně chybná interpretace promluv uživatele, bude muset dialogový manažer dialog raději pozastavit.

**Model interakce systému**. V průběhu vzájemných výměn mezi uživatelem a systémem by měl mít dialogový manažer dostatečnou informaci o tom, který modul dialogového systému je právě aktivní a která akce je právě prováděna. Modelování interakcí dialogového systému předpokládá důkladnou znalost funkce a různých omezení všech komponent systému. Systém by měl být schopen pracovat s jistou důvěrou ve svoje „okamžité" schopnosti vyřešit aktuální stav či celou dialogovou úlohu a podle toho by měl měnit případnou strategii řízení dialogu, viz odst. 15.5.2.

**Model uživatele**. Informace o uživateli, která je známa před startem dialogu, anebo byla získána či odvozena v průběhu dialogu, je často považována za model uživatele. Model uživatele může být využit ke zlepšení efektivity vedení dialogu, protože dovoluje návrháři přizpůsobit dialogový systém individuálním požadavkům uživatele. Obecně lze říci, že modelování uživatele může pokrývat velký rozsah individuálních stránek uživatelovy osobnosti od fyzických až ke kognitivním. Při *modelování fyzických vlastností* je někdy potřeba respektovat pohybová či smyslová (zrak, sluch) omezení tělesně handicapovaných uživatelů, častěji však jde o identifikaci a případnou adaptaci na různé defekty výslovnosti (šišlání, ráčkování, koktání, dialekt, dlouhé hlasité „přemýšlení" apod.). *Modelování kognitivních vlastností* uživatele zahrnuje zejména zachycení jeho schopností se přizpůsobit dané dialogové úloze. Je například známo, že uživatelé, kteří pravidelně používají dialogový systém, nepotřebují různé nápovědy a snaží se ukončit dialog co nejrychleji. K tomu jim může pomoci implementovaná vlastnost dialogového systému, která umožní uživateli „provalit se" svojí promluvou do nápovědy systému, čímž ji předčasně ukončí a urychlí tak dialog. Podobně je třeba předpokládat, že k dialogu přistoupí i uživatelé, kteří mají malé anebo žádné zkušenosti s danou aplikací, anebo nemají žádnou představu o tom, jaké jsou možnosti současných počítačů. Pro takovéto typy uživatelů se doporučuje odhadnout a v průběhu dialogu postupně upřesňovat pravděpodobný cíl komunikace nezkušeného uživatele se systémem a tomuto cíli přizpůsobit i vedení dialogu včetně formulace nápověd.

**Model dialogu**. Model dialogu je většinou založen na implicitně předpokládaném kooperativním chování jak uživatele, tak systému. Grice (Grice, 1975) formuloval soubor *kooperativních principů*, který sestává z tzv. konverzačních maxim, jež by měl uživatel dialogového systému ve svém projevu dodržovat. Jsou jimi:

- kvantita, tj. řečník se snaží podat tolik informace, kolik je třeba;
- kvalita, tj. řečník se snaží být pravdivý a nepodává informaci, která není podpořena fakty;
- závažnost, tj. řečník se snaží sdělovat informace, které jsou vhodné a relevantní pro danou diskusi;
- způsob, tj. řečník se snaží být ve své výpovědi co možná nejstručnější, jasně formuluje své cíle a vyhýbá se nesrozumitelným a mnohoznačným výpovědím.

Odchylky v nedodržování konverzačních maxim jsou přitom pozorovány zejména v případech, kdy účastníci dialogu (uživatel a systém) sdílí neúplnou anebo nestejnou znalost o doméně diskuse. V takovém případě je obvykle třeba uskutečnit několik výměn mezi uživatelem a systémem, ve kterých se aktualizují společné znalosti.

Obecně lze konstatovat, že dialog není proud výměn, které mezi sebou nemají žádný vztah. Promluvy, které tvoří dialog, mohou být interpretovány podle Austina (Austin, 1962) jako jisté akce (činy), které jsou prováděny řečníkem. Austin nazval tento druh akcí *řečové akty*. Řečové nebo též někdy nazývané lingvistické akty mají tři základní funkce:

- promluvovou (angl. locutionary) odpovídající formulaci promluvy,
- nepromluvovou (angl. illocutionary) kategorizující promluvu z pohledu řečníka (ve funkci řečníka může být i počítačový systém), např. vyjadřuje, že jde o dotaz, slib, žádost apod.,
- mimopromluvovou (angl. perlocutionary) specifikující požadované nebo zamýšlené změny vnitřního stavu posluchače.

Například odpověď automatického systému zprostředkujícího vstupenky do divadel „*Prezidentská lóže je obsazena!*" je prohlášení složené ze 4 slov (funkce locutionary), má formu oznámení s akcentem na posledním slovu (funkce illocutionary) a posluchač po jejím vyslechnutí může z důvodu očekávaného nepohodlí pocítit nechuť divadelní představení vůbec navštívit (funkce perlocutionary).

V hlasovém dialogovém systému, kdy spolu konverzuje člověk s počítačem a musí být tedy sledováno několik paralelně se rozvíjejících souvislostí, je užitečné rozšířit pojem řečového aktu na *dialogový akt*. Dialogové akty zahrnují přinejmenším tři těsně svázaná hlediska (Minker, Néel, 2002) – informační obsah (týká se všech faktických informací odkazujících se na úlohu), řízení komunikace (týká se promluv, které zajišťují hladký průběh dialogu, např. promluvy „*Slyšíš mě?*" nebo „*Prosím, mluvte na telefon!*") a rozdělení rolí (týká se formy dialogu odkazující na rozdělení rolí, které hrají účastníci v dialogu). Dialogový akt reprezentuje význam nebo smysl promluvy v úrovni nepromluvové (illocutionary) funkce promluvy. Na dialogové akty proto můžeme pohlížet jako na soubor tagů, které klasifikují promluvy podle kombinace syntaktických, sémantických a pragmatických měřítek. Pro aplikace dialogových systémů jsou obvykle definovány kategorie dialogových aktů, které jsou relevantní pro danou úlohu, nicméně snahou je navrhnout obecně použitelný systém tagů, který je nezávislý na doméně.

Velmi propracovaný soubor typů dialogových aktů byl navržen v souvislosti se zpracováním korpusu telefonních konverzací známým pod označením Switchboard (Stolcke a kol., 2000). Soubor sestává ze 42 typů dialogových aktů, jimiž byly konverzace anotovány. Mezi nejčastěji se objevujícími typy dialogových aktů se objevily například: OZNÁMENÍ (např. „*Jsem na letišti v Ruzyni*"), (krátké) POTVRZENÍ (v angličtině nazývané buď backchannel/acknowledgement, nebo continuer, což představuje krátkou promluvu, která potvrzuje nějakým způsobem předchozí hovor s pobídkou, aby pokračoval; např. „*Hmm*"), MÍNĚNÍ (např. „*Váš návrh není dobrý*"), NEINTERPRETOVATELNÝ výraz (např. „*Nu*", „*Tak*"), SOUHLAS (např. „ *Je to v pořádku*"), POCHOPENÍ (např. „*Neumím si představit*"), OTÁZKA ano/ne (např. „*Potřebujete pomoci?*"), MIMOVERBÁLNÍ výraz (např. smích, vzlykání, hlasité přemýšlení apod.), ODPOVĚĎ ano („*Ano*"), KONVENČNÍ ZAKONČENÍ (např. „*Děkuji za informace a nashledanou*"), ODPOVĚĎ ne („*Ne*") atd. Podobný soubor typů dialogových aktů byl též navržen v (Jurčíček a kol., 2005) pro specifickou úlohu zpracování dialogů telefonní služby informující o vlakových spojeních.

Ukažme si nyní fragment dialogu mezi prodavačem obchodu „nákup po telefonu" a zákazníkem a následně i způsob anotace tohoto rozhovoru dialogovými akty:

Prodavač: *Takže ten kartáč nechcete?*
Zákazník: *Ne. Objednal jsem dvě šoupátka s nástavcem a jsem velmi spokojen.*
Prodavač: *Budete tedy dobře vybaven?*
Zákazník: *Ano. Promiňte, ale musím končit, nahledanou.*
Prodavač: *Trh-*, <mimoverbální zvuk>.

Pro anotaci promluv, resp. dialogových výměn, dialogovými akty využijeme notace zavedené v článku (Stocke a kol., 2000):

| | | |
|---|---|---|
| Prodavač: | OTÁZKA ano/ne | *Takže ten kartáč nechcete?* |
| Zákazník: | ODPOVĚĎ ne | *Ne.* |
| Zákazník: | OZNÁMENÍ | *Objednal jsem dvě šoupátka s nástavcem.* |
| Zákazník: | MÍNĚNÍ | *Jsem velmi spokojen.* |
| Prodavač: | OTÁZKA ano/ne | *Budete tedy dobře vybaven?* |
| Zákazník: | ODPOVĚĎ ano | *Ano.* |
| Zákazník: | OMLUVA | *Promiňte.* |
| Zákazník: | KONVENČNÍ ZAKONČENÍ | *Musím končit, nashledanou.* |
| Prodavač: | NEINTERPRETOVATELNÝ | *Trh-* |
| Prodavač: | MIMOVERBÁLNÍ | <mimoverbální zvuk>. |

Je zřejmé, že anotace dialogových promluv (výměn) dialogovými akty, přináší další informace a znalosti vyšší úrovně, které mohou být využity při dekódování promluv a zvýšit tak přesnost modulů rozpoznávání řeči a porozumění jazyku.

### *15.5.1.2 Ověřování správnosti interpretace promluvy uživatele*

Využití různých znalostních zdrojů umožňuje dialogovému systému kompenzovat do jisté míry některé nedostatky spojené s rozpoznáním a interpretací promluvy uživatele, aniž by bylo třeba požadovat opakování promluvy. Na druhou stranu, má-li systém provést požadovanou akci (najít informaci v databázi, objednat lístek do divadla, koupit pizzu apod.), je často nutné provést ověření, zda systémem interpretovaný záměr se shoduje se záměrem uživatele. Existují v podstatě dva hlavní přístupy k verifikaci a potvrzení toho, že záměr byl systémem správně interpretován – explicitní a implicitní ověření.

**Explicitní ověření**. Vyžaduje formu otázky, která přímo žádá potvrzení obsahu dané promluvy často formou odpovědi *ano/ne*. Například:

Uživatel: *Přivezte mi mongolskou pizzu, tmavé pivo a meloun.*
Systém: *Chcete přivést mongolskou pizzu?*
Uživatel: *Ano.*
Systém: *Chcete přivést také tmavé pivo?*
Uživatel: *Ano.*
Systém: *Chcete přivést také amaroun?*
Uživatel: *Ne.*

Problémem je, že když jsou potvrzovány jednotlivé hodnoty dané promluvy postupně, dialog to neúměrně prodlužuje. Naopak, chceme-li ověřit celou výpověď najednou, může to, v případě chybně uvedené jedné nebo více položek, způsobit komplikace dalšího vedení dialogu.

**Implicitní ověřování.** Při něm je verifikace rozpoznaných údajů včleněna vždy do následující otázky systému. Uživatel tak může stále opravovat opakovanou hodnotu. Jakmile uživatel odpoví, aniž by opravil některou z uvedených hodnot, bude to považováno za stav, že hodnoty byly správně rozpoznány a interpretovány a budou tak implicitně potvrzeny.

Uživatel: *Přivezte mi mongolskou pizzu, tmavé pivo a meloun.*
Systém: *Na jakou adresu Vám máme přivést mongolskou pizzu, tmavé pivo a amaroun?*
Uživatel: *Nechci amaroun, ale meloun na adresu ....*

Implicitní způsob ověřování je rychlejší a přirozenější, je přijímán vstřícněji uživateli dialogu, klade však mnohem vyšší nároky na funkci modulů rozpoznávání řeči a porozumění mluvenému jazyku.

### 15.5.2 Strategie řízení dialogu

Jak již bylo uvedeno, strategie dialogového systému specifikuje akci, která bude provedena pro daný stav, v němž se dialogový systém právě nachází, tak, aby bylo zajištěno cílové chování. Pro každý stav úlohy jsou proto navrhovány subdialogy, které se snaží vyřešit dílčí cíle dialogové úlohy. Dílčí cíle, které jsou těmito subdialogy uživatelem řešeny, se většinou zaměřují na některou z následujících úloh – **potvrzení** (užíváno k zjištění správnosti rozpoznané promluvy), **zotavení z chyby** (náprava chyby poté, co systém špatně porozuměl), **opětovnou pobídkou** (postup poté, co systém očekával vstupní informaci, ale nic neobdržel), **dokončení** (zjištění chybějící informace od uživatele), **omezení** (redukce rozsahu požadavku, pro který bude získáno rozumné množství informace), **uvolnění** (zvětšení rozsahu požadavku, když nebyla získána žádná informace), **zjednoznačnění** (řešení nekonzistentního vstupu od uživatele), **pozdrav/zakončení** (zajištění společenského protokolu na začátku a na konci interakce).

Strategie řízení dialogu souvisí úzce s tím, který z účastníků dialogu řídí vzájemné působení, tedy kdo má v dialogu iniciativu. Existují v podstatě tři základní typy dialogových strategií – *s iniciativou systému*, *se smíšenou iniciativou* a *s iniciativou uživatele*.

#### *15.5.2.1 Dialog s iniciativou systému*

V dialogu s iniciativou systému (angl. System-Initiative Dialogue) zajišťuje systém (počítač) řízení celého dialogu, tj. systém vybírá hlavní cíl i obsah a posloupnost všech podcílů, které musí být pro úspěšné ukončení dialogu splněny. Příkladem může být následující konzultace uživatele s automatickou službou poskytování informací o odjezdech vlaků:

Systém: *Dobrý den. Dovolali jste se na automatickou službu poskytování informací o odjezdech vlaků. Prosím řekněte místo, do kterého chcete cestovat.*

Uživatel: *Beroun.*
Systém: *Řekněte stanici, ve které chcete nastoupit.*
Uživatel: *Rokycany.*
Systém: *Který den chcete cestovat z Rokycan do Berouna?*
Uživatel: *Ve čtvrtek.*
Systém: *V kolik hodin chcete vyjet z Rokycan?*
Uživatel: *Po desáté.*
Systém: *Vlak do Berouna odjíždí zítra z Rokycan v 10:25 a v Berouně je v 11:06.*
Uživatel: *Děkuji a nashledanou.*
Systém: *Nashledanou.*

Při tomto způsobu dialogu je obvykle uživatel vyzván systémem, aby reagoval na požadavek anebo odpověděl na položenou otázku promluvou, která je vybírána z omezeného slovníku předem definovaných odpovědí. Často jsou pro tyto případy nabídnuty uživateli možné odpovědi formou nápovědy.

### *15.5.2.2 Dialog se smíšenou iniciativou*

V dialogu se smíšenou iniciativou (angl. Mixed-Initiative Dialogue) zahajuje obvykle dialog systém, který může podle potřeb a vývoje dialogových výměn přenechat iniciativu uživateli. Jednotlivé dílčí cíle dialogu pak mohou být modifikovány podle odpovědí uživatele. Systém však většinou dohlíží nad tím, aby posloupnost dílčích cílů sledovala dosažení celkového cíle a v případě potřeby přebírá zpět řízení. Jako příklad dialogu se smíšenou iniciativou je opět použita konzultace uživatele s automatickou službou poskytování informací o odjezdech vlaků s tím, že na rozdíl od minulého případu uvedeného v odst. 15.5.2.1 je zde ponechána uživateli volnost ve formování jeho odpovědí:

Systém: *Dobrý den. Dovolali jste se na automatickou službu poskytování informací o dopravních spojeních. Kam si přejete cestovat?*
Uživatel: *Potřebuji se dostat z Rokycan do Berouna.*
Systém: *Kdy chcete cestovat z Rokycan do Berouna?*
Uživatel: *Chci jet zítra někdy po desáté.*
Systém: *Zítra odjíždí vlak z Rokycan v 10:25 hodin, příjezd do Berouna v 11:06.*
Uživatel: *Děkuji a nashledanou.*
Systém: *Děkujeme, že jste využili naší službu a nashledanou.*

Hlasové dialogové systémy se smíšenou iniciativou jsou v současné době využívány v různých aplikacích zcela prioritně, i když jejich funkcionalita je stále omezena na relativně úzkou komunikační oblast. Obecně lze konstatovat, že pravděpodobnost úspěšného dosažení celkového cíle je u této iniciativy mírně nižší než u dialogů se strategií založenou na iniciativě systému. Je to způsobeno jednak tím, že v dialozích, ve kterých nejsou na odpovědi uživatele kladena taková omezení, jako je tomu u dialogů s iniciativou systému, dosahuje funkce bloku rozpoznávání a porozumění řeči často nižší úspěšnost. Dalším důvodem je i to, že přistupují-li k dialogu noví a nezkušení uživatelé, mohou mít větší problémy s interpretací nápověd s „otevřenějším" koncem. Na druhou stranu zkušení uživatelé, kteří jsou obeznámeni s komunikací s obdobnými systémy, hodnotí dialog se smíšenou iniciativou jako dialog, který lze dovést k cíli mnohem rychleji v mnohem přirozenějším rozhovoru. V článku (Litman, Pan, 1999) je uvedeno, že

dovolíme-li novým a nezkušeným uživatelům dialogového systému přepnout z dialogu se smíšenou iniciativou na dialog s iniciativou systému, končí to obvykle vyšší spokojeností uživatele a též vyšší pravděpodobností dosažení požadovaného cíle. Je proto velmi užitečné detekovat automaticky v průběhu dialogu případné potíže uživatele s vedením dialogu a včas přepnout řízení dialogu na mnohem podpůrnější strategii s iniciativou systému, která končívá v tomto případě úspěšnějšími interakcemi. Zároveň je prokázáno, že zvyšování robustnosti modulu rozpoznávání řeči a zlepšování funkce modulu porozumění promluvám vede též k prokazatelnému zvyšování spokojenosti uživatelů s funkcí dialogového systému.

#### *15.5.2.3 Dialog s iniciativou uživatele*

V dialogu s iniciativou uživatele (angl. User-Initiative Dialogue) se každý příkaz uživatele interpretuje bez nějakých předběžných omezení či stanoveného plánu. Jednoduchým příkladem dialogu tohoto typu je zde opět komunikace uživatele s automatickou službou poskytující dopravní a cestovní informace:

| | |
|---|---|
| Systém: | *Dobrý den. Vítá Vás automatická služba poskytování dopravních informací a zprostředkování cestovních služeb.* |
| Uživatel: | *Dobrý den. Potřeboval bych se dostat z Kozolup do Řitky.* |
| Systém: | *Z Kozolup do Řitky lze cestovat přes Plzeň a Prahu v kombinaci vlakové a autobusové dopravy.* |
| Uživatel: | *Chci cestovat přes Příbram. Je to možné?* |
| Systém: | *Ano, je to možné.* |
| Uživatel: | *Z Kozolup chci vyjet zítra kolem poledne.* |
| Systém: | *Z Kozolup odjíždí vlak v 11:38 do Plzně, kde přestoupíte na rychlík ....., v Řitce budete v 16:51.* |
| Uživatel: | *Děkuji a nashledanou.* |
| Systém: | *Děkujeme, že jste využili naší službu a nashledanou.* |

Při vedení obdobných dialogů je obvyklé, že zde může docházet k „přepínání" jednotlivých témat hovoru. To klade zvýšené nároky na funkce jednotlivých komponent dialogového systému a nutí konstruovat dialogový systém s mnohem větší podporou různých znalostních zdrojů.

### 15.5.3 Typy dialogových systémů

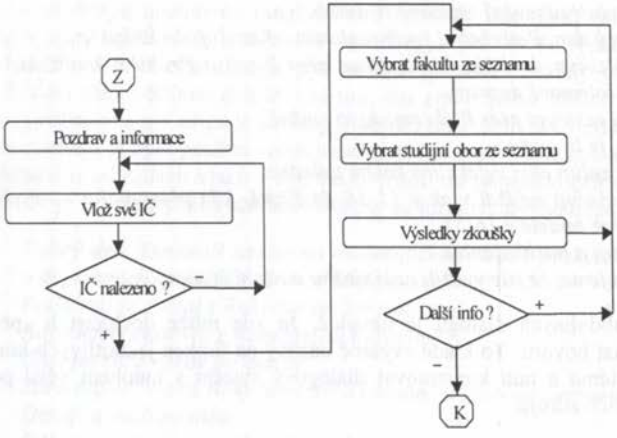
Podle toho, jakým způsobem dialogový systém zpracovává promluvy uživatele, lze dialogové systémy klasifikovat do tří hlavních skupin na
- dialogové systémy s konečným počtem stavů,
- dialogové systémy využívající strukturu rámců,
- dialogové systémy založené na agentech.

#### *15.5.3.1 Dialogový systém s konečným počtem stavů*

U dialogových systémů s konečným počtem stavů je struktura dialogu reprezentována ve tvaru stavově přechodové sítě. Jednotlivé uzly sítě přitom reprezentují stavy dialogu, ve kterých dialogový systém zjišťuje nějakou informaci, popř. je tato informace uživatelem potvrzována. Přechody mezi uzly (stavy) sítě pak definují přípustné cesty sítí a určují tak všechny povolené dialogy. Získání či potvrzení určité informace může být pro daný stav

dialogu uskutečněno prostřednictví dílčích, předem navržených a s daným stavem svázaných subdialogů. V dialogových systémech s konečným počtem stavů je většinou využívána strategie řízení s iniciativou systému. Pro tyto systémy jsou typické jednoduché nápovědy, které uživatele v každém stavu dialogu „navigují", z jakých slov či krátkých frází má vybírat svoji odpověď, aby úspěšně vyřešil dílčí úlohu svázanou s daným stavem dialogu. Na obrázku 15.15 je znázorněn stavový diagram jednoduchého dialogu, který zpřístupňuje uchazečům o studium na Západočeské univerzitě v Plzni v hlasové komunikaci s informačním systémem vyhledat výsledky jejich přijímací zkoušky, popř. závěry přijímací komise[3]. Takový dialog může například proběhnout následujícím způsobem:

Systém: *Dobrý den. Zdraví Vás automat informující o výsledcích přijímacího řízení na Západočeské univerzitě v Plzni. Zadejte, prosím, své univerzitní číslo.*
Uživatel: *Jedna pět šest osm dva osm.*
Systém: *Bylo zadáno jedna pět šest osm dva osm. Pokud údaje souhlasí, řekněte ano. Pro novou registraci řekněte registrovat.*




Obr. 15.15 Stavový diagram hlasového dialogového systému umožňujícího uchazečům o studium na univerzitě se v telefonickém dialogu s univerzitním informačním systémem dozvědět o výsledku své přijímací zkoušky (pozn.: pro přihlášení se uživatele k systému je nutné vyslovit nebo zadat pomocí tónové volby osobní číslo studenta).

[3] Tento telefonní dialogový systém funguje na Západočeské univerzitě v Plzni již od r. 2001. Technologiemi rozpoznávání a syntézy řeči zpřístupňuje uchazečům o studium na univerzitě výsledky jejich přijímacího řízení, které jsou vystaveny na webových stránkách univerzity (Šmídl a kol., 2005), (Zahradil, Šmídl, 2005) (v létech 2001 až 2005 více než 15 tisíc uživatelů). Uplatněná koncepce je založena na standardu Voice XML. Vstupem systému je hlasová promluva uživatele či DTMF (tónová) volba, výstupem pak informace vyjádřená v podobě automaticky syntetizované odpovědi. Množina rozpoznávaných frází je v každém stavu dialogu definována bezkontextovou gramatikou, jež může být generována on-line (dynamicky generovaným skriptem). Dialogový systém je vybaven blokem zamítání nespolehlivě rozpoznaných slov, detektorem ticho/řeč, napojením na SQL databázi, rozhraním TAPI či CAPI, GSM bránou atd.

Uživatel: *Ano.*
Systém: *Vyberte jednu z následujících fakult: řekněte Fakulta aplikovaných věd nebo řekněte Fakulta elektrotechnická.*
Uživatel: *Fakulta aplikovaných věd.*
Systém: *Byly nalezeny dva záznamy o přijímacím řízení na Fakultě aplikovaných věd. Chcete-li výsledky pro studijní program Aplikované vědy a informatika obor mechanika řekněte ano.*
Uživatel: *- / Ne.*
Systém: *Chcete-li výsledky pro studijní program Aplikované vědy a informatika, obor kybernetika a řídicí technika řekněte ano.*
Uživatel: *Ano.*
Systém: *Výsledky přijímacího řízení na Fakultu aplikovaných věd, studijní program Aplikované vědy a informatika, obor kybernetika a řídicí technika. Uchazeč získal celkem 107 bodů z přijímacího testu. Uchazeč byl doporučen a přijat na základě přijímací zkoušky. Vyberte z následující nabídky: Pro zopakování řekněte opakovat, pro konec řekněte konec.*
Uživatel: *Konec.*
Systém: *Děkujeme za použití našeho systému. Na slyšenou.*

Výhodou dialogových systémů s konečným počten stavů je jednoduchost dialogu. Protože spektrum odpovědí uživatele je v každém stavu dialogu většinou dáno omezeným souborem promluv, často nabízených uživateli formou nápověd, má tento typ dialogového systému i nižší nároky na výkon modulu rozpoznávání a též i porozumění řeči. Lze totiž často vystačit s rozpoznáváním malého počtu izolovaných slov, jejichž slovník je pro každý stav předem definován.

Nevýhodou dialogových systémů s konečným počtem stavů je jejich malá pružnost. Protože celá přechodová síť je definována předem, je velmi obtížné nebo spíše nemožné uskutečnit nějakou dialogovou výměnu mimo rámec této sítě. Problémy například vznikají, když potřebuje uživatel opravit již vloženou položku a nebo zavést nějakou informaci, která nebyla předvídána v čase návrhu dialogu. Uživatel obvykle nemůže vložit více položek najednou, a tak urychlit dosažení cíle dialogu. Dialogový systém s touto strukturou je proto vhodný zejména pro úlohy, které jsou dobře strukturované a mají v každém uzlu přechodové sítě malý počet variant dalšího pokračování. Obvykle se tyto systémy mohou využívat při provádění jednoduchých bankovních operacích (po telefonu), v systémech poskytujících informace o počasí, o vlakových či leteckých spojeních apod.

### *15.5.3.2 Dialogový systém využívající strukturu rámců*

V dialogových systémech pracujících se strukturou rámců je dialog konstruován tak, že je na základě analýzy řešené úlohy navržen rámec či struktura rámců, jejichž sloty obvykle reprezentují jednotlivá hlediska řešené úlohy. Tyto sloty musí být částečně anebo úplně vyplněny, aby mohla být vykonána požadovaná akce (např. učiněn dotaz na databázi, proveden požadovaný akční zásah nebo operace apod.). V praxi to vypadá tak, že dialogový systém klade uživateli vhodně formulované otázky, aby na základě obdržených odpovědí mohly být vyplněny požadované sloty. Dialog je přitom řízen

dialogovým manažerem obvykle tak, že uživatel může vkládat informace v různém pořadí a často lze vložit i jednou promluvou (v jedné dialogové výměně) více informací najednou, tj. vyplnit více slotů. Tím se tento typ dialogového systému zásadně liší od systému s konečným počtem stavů.

Funkci dialogového systému, který využívá strukturu rámců si lze ukázat na úloze poskytování informací o vlakových spojeních. Úlohu lze jednoduše navrhnout s jedním rámcem a souborem slotů v následujícím tvaru:

```
{<INFORMACE O VLAKOVÉM SPOJENÍ>
[CÍLOVÁ STANICE]      →      ( - )
[DEN PŘÍJEZDU]        →      ( - )
[ČAS PŘÍJEZDU]        →      ( - )
[VÝCHOZÍ STANICE]     →      ( - )
[DEN ODJEZDU]         →      ( - )
[ČAS ODJEZDU]         →      ( - )}
```

K tomu, aby mohl dialogový systém zformulovat dotaz na databázový systém, kde jsou uloženy informace o vlakových spojeních, musí mít k dispozici údaje o cílové a výchozí stanici, a pak buď den a předpokládaný čas příjezdu do cílové stanice, nebo alternativně den a čas odjezdu z výchozí stanice. Pokud nebudou uvedené údaje k dispozici, nelze vygenerovat dotaz na databázi a úspěšně zakončit dialogovou úlohu.

Připomeňme, že vedle faktických informací (znalostí), které získává dialogový systém v dialogu s uživatelem a jimiž „plní" připravené sloty, lze s každým slotem též spojit procedurální znalost, která může mít v případě dialogových systémů podobu předem připravených otázek, jež je třeba položit uživateli, aby byla získána zatím nedostupná informace (znalost). Je-li například v průběhu dialogu nedostupná informace o výchozí stanici, může být uživateli položena vhodná otázka svázaná s procedurální znalostí slotu `[VÝCHOZÍ STANICE]`:

- *Jak Vám můžeme pomoci?* (není-li vyplněn žádný slot)
- *Kdy a odkud chcete cestovat do* `[CÍLOVÁ STANICE]`? (je-li již známa cílová stanice)
- *Odkud chcete* `[DEN VÝJEZDU]` *cestovat do* `[CÍLOVÁ STANICE]`? (je-li již známa cílová stanice a den výjezdu).

Vhodně formulovanou otázkou, která obsahuje údaje o již získaných relevantních informacích, lze implicitně ověřit pravdivost těchto údajů a učinit tak dialog mnohem přirozenější a rychlejší, což je obecně přijímáno uživatelem velmi pozitivně. Podívejme se nyní, jak by mohl proběhnout celý dialog informačního systému s uživatelem:

| | |
|---|---|
| Systém: | *Dobrý den. Dovolali jste se na automatický systém poskytující informace o vlakových spojeních. Jak Vám můžeme pomoci?* |
| Uživatel: | *Ale, potřebuju se dostat do Merklína.* |
| Systém: | *Kdy a kde chcete nastoupit při Vaší cestě do Merklína?* |
| Uživatel: | *Dnes v Domažlicích.* |

Systém: *V kolik hodin chcete dnes vyjet z Domažlic?*
Uživatel: *V Merklíně musím být kolem poledne.*

Po této poslední výměně byly získány všechny údaje pro to, aby mohly být vyplněny relevantní sloty rámcové struktury potřebné pro zformulování dotazu na databázi. Po doplnění příslušných slotů bude mít rámcová struktura tvar:

```
{<INFORMACE O VLAKOVÉM SPOJENÍ>
[CÍLOVÁ STANICE]      →    ( Merklín )
[DEN PŘÍJEZDU]        →    ( dnes )
[ČAS PŘÍJEZDU]        →    ( 12:00 )
[VÝCHOZÍ STANICE]     →    ( Domažlice )
[DEN ODJEZDU]         →    ( - )
[ČAS ODJEZDU]         →    ( - )}
```

Dialogový manažer pak vygeneruje dotaz na databázi informačního systému a po obdržení konkrétních údajů o vlakových spojeních musí blok generování odpovědi převést obdržené údaje do formy srozumitelné promluvy a předat je uživateli.

Je zřejmé, že dialogový systém s rámcovou strukturou neomezuje uživatele na vyslovení pouze izolovaných slov nebo předem připravených frází. Je-li modul porozumění navíc vybaven schopností interpretovat promluvy typu „*brzo ráno*", „ *pozdě odpoledne*" apod. může být dialog často ještě více urychlen. Uvažme například první promluvu uživatele zmiňovaného dialogu vyslovenou jako

Uživatel: *Ale, potřebuju se dostat rychle do Merklína.*

Takto formulovaný požadavek v sobě obsahuje jednak název cílové stanice a též implicitně uvedený údaj o dnu a hodině odjezdu (pozn.: promluvu „*dostat se rychle*" interpretuje dialogový systém jako „*vyjet dnes a hned*", tj. je znám den i aktuální čas). Protože jsou tedy známy položky `[CÍLOVÁ STANICE]`, `[DEN VÝJEZDU]` a `[ČAS VÝJEZDU]`, musí se systém uživatele dotázat na poslední chybějící údaj, kterým je `[VÝCHOZÍ STANICE]`, a to dříve, než bude schopen zformulovat dotaz na databázi o vlakových spojeních. Otázka na uživatele bude vytvořena tak, aby se implicitně ověřily všechny nově získané položky (údaje) získané v minulé výměně a byl navíc získán poslední požadovaný údaj

Systém: *Odkud chcete dnes cestovat nejbližším vlakem do Merklína?*
Uživatel: *Jsem v Domažlicích.*

Po doplnění posledního údaje, tj. `[VÝCHOZÍ STANICE]`, do rámcové struktury, může dialogový systém vyžádat hledané vlakové spojení a zakončit úlohu.

Z tohoto příkladu jsou dobře vidět výhodné vlastnosti dialogového systému s rámcovou strukturou, neboť úspěšné dokončení dialogu lze často završit mnohem menším počtem výměn, než by tomu bylo v případě dialogu s konečným počtem stavů. Uveďme, že zde prezentované příklady dialogů jsou částečně zjednodušeny. V praxi je potřeba často řešit problémy s nejednoznačnostmi (např. z požadavku „*chci být v Liberci před*

*desátou*" není zřejmé, zda jde o dopolední nebo večerní hodinu), problémy s opravami již akceptovaných údajů, problémy s klasifikací promluv, které modul rozpoznávání není schopen opakovaně zpracovat s dostatečnou spolehlivostí apod. V některých případech je proto užitečné, aby byl dialogový systém schopen včas rozpoznat potíže se získáváním a interpretací relevantních údajů od uživatele a přepnul strategii dialogu se smíšenou iniciativou na dialog s iniciativou systému. Následně, v režimu otázek, které vyžadují pouze jednoslovné odpovědi a „přepnutím" modulu rozpoznávání řeči do režimu rozpoznávání povelů, lze obvykle dovést dialogovou úlohu spolehlivěji k cíli.

Složitější dialogové úlohy někdy vyžadují využití systému několika rámců, a to zejména v případech, konstruujeme-li dialogový systém, který má poskytnout uživateli služby ve více oblastech, např. zajištění komplexních cestovních služeb (tj. rezervaci letenek, ubytování, výpůjčky automobilu apod.). V některých případech je pak vhodné využít i znalostní strukturu scénáře, která je schopna reprezentovat i stereotypní posloupnosti událostí.

Dialogový systém s rámcovou strukturou poskytuje větší pružnost vedení dialogu než systém s konečným počtem stavů. Kontext dialogu, který přispívá ke stanovení příští akce systému, však stále dialog omezuje, neboť je závislý na výsledku analýzy předchozí promluvy uživatele a koresponduje s potřebou vyplnění relevantních slotů. Složitější vyjednávání mezi dialogovým systémem a uživatelem není proto v tomto případě stále možné modelovat, poněvadž různí uživatelé se mohou odlišovat v úrovni apriorních znalostí, které přinášejí do úlohy, a hrozilo by tak neúměrné rozšíření množství všech možných vstupů, které by musel systém rozpoznat a interpretovat. Při vývoji složitějších dialogových systémů založených na rámcích narážíme též na nevýhody velkého množství pravidel a kontextů. Někdy je potřeba provést mnoho experimentů, abychom například zjistili, zda dialogový systém negeneruje při nějakých okolnostech, které jsme při návrhu nepředvídali, nevhodné otázky či nápovědy.

### *15.5.3.3 Dialogový systém založený na agentech*

Dialogové systémy s konečným počtem stavů i s rámcovou strukturou jsou v současné době úspěšně provozovány v mnoha aplikacích navržených obvykle pro jednu specifickou oblast. Pokud bychom chtěli provést návrh a realizaci systému při současné možnosti komunikovat v mnoha předmětných oblastech, narazíme na problémy spojené se značným růstem složitosti systému. Rozšíříme-li hlasový dialogový systém na novou oblast, bude nevyhnutelné rozšířit slovník, natrénovat nové jazykové modely, rozšířit znalostní domény dialogového manažeru, připravit nové scénáře pro generování promluv (otázek, nápověd apod.). Podobná situace se stane i s parsovacími pravidly, které budeme muset též přizpůsobit pro novou oblast. Dalším problémem těchto tzv. multidoménových dialogových systémů je též jejich obtížné udržování.

Základní ideou vzniku dialogových systémů založených na agentech byla představa současné a nezávislé komunikace uživatele s více informačními zdroji. Analogii takové komunikace můžeme vidět při práci s grafickým rozhraním na počítači, kdy se uživatel podle potřeby spojí současně s několika informačními servery (např.: internetovou službou objednávání letenek, objednávání ubytování, půjčování aut, předpovědi počasí, zprostředkování „asistentek" pro speciální práce apod.) a mezi nimi volně přechází v závislosti na postupném řešením svých požadavků. Chceme-li využít podobnou strategii i v prostředí hlasového dialogu, narazíme na obtíže spojené s přirozenou nutností

zadávat a odebírat všechny informace sekvenčně, což přináší obtíže jak uživateli, tak systému, neboť je pro ně nutností udržovat dokonalý přehled o stavu dialogu s tou kterou oblastí.

Efektivním řešením obdobných úloh může být využití moderní architektury se spolupracujícími agenty.V této architektuře mohou být jednotliví agenti vyvíjeni zcela nezávisle a při řešení úlohy spolu spolupracovat tak, aby dosáhli cíle, který požaduje uživatel (tj. například vzhledem k vysokým teplotám, které panují v cílovém městě, bude uživatel požadovat pokoj i auto s klimatizací apod.). Úspěšná a účinná komunikace mezi agenty, která vede nejen k výměně dat, ale i znalostí, vyžaduje, aby agenti byli schopni sdílet znalosti na třech úrovních: syntaktické, sémantické a pragmatické.

Technologie konstrukce inteligentních agentů a multiagentních systémů je velmi intenzivně rozvíjena v rámci vědní disciplíny umělá inteligence a je detailně popsána v mnoha renomovaných publikacích (Mařík a kol., 2001). Proto se zde nebudeme zabývat obecnými problémy této technologie, ale zmíníme se stručně jen o některých zásadách jejího uplatnění při využití v hlasových dialogových systémech.

Při konstrukci multidoménového dialogového systému lze využít buď *centralizovaný model*, anebo *distribuovaný model*. V informačním dialogovém systému s centralizovaným modelem, který poskytuje cestovní služby, lze navrhnout soubor agentů, kteří vyhledávají potřebné informace, popř. zajišťují potřebné služby nad příslušnými informačními zdroji – informačními databázemi. V případě plánování cest lze tak navrhnout agenta zajišťujícího letové služby, agenta zajišťujícího ubytovací služby, půjčení automobilu apod. Další agent pak může pracovat ve funkci zprostředkovatele, tj. přijme a porozumí požadavku uživatele, pošle dotazy databázovým agentům, dostane od nich odpovědi a konečně odpoví uživateli. Funkce systému s tímto modelem je efektivní jen pro dialogy s menším počtem výměn. Při větším počtu výměn je obvykle dialog plynule přepínán mezi různými oblastmi a vzhledem k tomu, že stav dialogu a znalost všech odlišných oblastí musí být stále uchovávány v agentu s funkcí zprostředkovatele, stává se tento model pro takové úlohy těžkopádný.

V informačním dialogovém systému s distribuovaným modelem je architektura založena na modelu klient–agent–server (Hsu, 1999). V rámci této architektury je navržen agent uživatelského rozhraní (angl. User Interface Agent, zkr. UIA), který je ve funkci klienta napojen na další agenty systému. Agenti mluveného dialogu (angl. Spoken Dialog Agents, zkr. SDA) pak přistupují k příslušným databázovým serverům, aby zprostředkovali danou službu, tj. vyhledali vhodný let a objednali letenku, objednali ubytování apod. V tomto modelu si každý agent mluveného dialogu zajišťuje sám svoji oblast dané domény, a to jak práci se znalostmi, tak řízení dialogu. Stav dialogu i všechna data závislá na stavu a historii dialogu jsou uchovávány vně agentů tak, aby mohly být informace (znalosti) sdíleny všemi spolupracujícími agenty. Tyto systémy pracují obvykle se smíšenou strategií, což znamená, že uživatel může dle potřeby převzít řízení dialogu a zavést například nové téma hovoru. Z těchto důvodů forma promluvy uživatele dialogového systému nemůže být určena předem ve tvaru souboru slov, frází či konceptů. Proto je vhodné vybavit tyto systémy dobře fungujícím modulem porozumění mluvenému jazyku.

Využití architektury spolupracujících agentů při návrhu hlasových dialogových systémů je užitečné zejména v takových úlohách, kdy požadujeme, aby agent vyřešil

danou úlohu komplexně a uživatel tedy nebyl nucen v mnoha subdialozích řešit postupně její dílčí hlediska. Příkladem mohou být systémy, kde v hlasovém dialogu lze plánovat různé akce (schůzky, služební cesty apod.), zadávat „složitější" úkoly domácím zařízením (vyhledání oblíbeného televizního programu a jeho záznam), zajištění komplexních služeb operacemi na internetu (Lin a kol., 1999), (Sadek, De Mori, 1998), (Sidner, 2002) apod. Zajímavou aplikací je využití inteligentního telefonního rozhraní multiagentního systému pro podporu rozhodování při plánování výroby (Bečvář a kol., 2005).

## 15.6 Závěrečné poznámky

Na závěr shrňme, že dialogové systémy s konečným počtem stavů jsou vhodné pro nejjednodušší dobře strukturované aplikace, kdy lze z každého stavu dialogu pokračovat jen velmi omezeným počtem přechodů. Přechod dialogu do dalšího stavu se obvykle uskuteční na základě vyslovení slova (krátkého slovního spojení), které je vybíráno z omezeného slovníku, popř. je výběr přípustných promluv nabídnut uživateli dialogového systému v každém stavu dialogu formou nápovědy. V takto navrženém dialogu nejsou obvykle kladeny na modul rozpoznávání řeči a porozumění mluvenému jazyku vysoké požadavky, neboť je zde často možné vystačit pouze s rozpoznáváním izolovaných slov a s modulem porozumění, který rozpoznaná slova nebo kombinace slov celkem bez větších problémů interpretuje na odpovídající sémantické významy, jež vyvolávají požadované akce. Přirozenější dialog je obvykle konstruován s využitím struktury sémantických rámců a speciálně navržených gramatik. Protože syntaktická analýza spontánní promluvy je velmi obtížná či spíše nemožná, využívají se pro parsování spontánních promluv speciálně navržené robustní parsery. Hlavní myšlenkou robustního parsingu je analýza pouze „užitečných" segmentů promluvy. Pro tyto účely se proto nabízí využít detektoru klíčových slov (klíčových frází), který hledá v proudu řeči, resp. v rozpoznaných fonetických mřížkách, relevantní slova nebo slovní spojení, která vyhovují dané gramatice a mohou tak podpořit proces vyplňování slotů připravených rámců.

Dialogové systémy navrhované expertně (často se mluví o „manuálním" návrhu) s využitím pravidel mají základní nedostatek v pracnosti a nákladnosti přípravy systému nejen pro každou novou doménu, ale často i pro novou implementaci dialogového systému v pouze částečně pozměněném prostředí. Tento nedostatek se snaží odstranit zcela nově se rozvíjející technika statistického modelování, jejíž výklad lze nalézt například v (Young, 2002) nebo (Psutka a kol., 2006). Vedle již standardně využívaného statistického přístupu při vytváření akustických a jazykových modelů je zde snahou statisticky modelovat i sémantický obsah promluv a vlastní řízení dialogu. Dialog je tak modelován jako Markovův rozhodovací proces, který má cílové chování a který lze ve stavovém prostoru sledovat prostřednictvím trajektorie určené akcemi systému a odpověďmi uživatele. Výhodou tohoto přístupu je jistá standardizace a i zřejmá možnost celý proces návrhu dialogu automatizovat. To jednak zmírňuje pracnost ve srovnání s „expertní" přípravou pravidel a jednak zjednodušuje i přenos dialogového systému na nová prostředí. Nevýhodou je zde nutnost sebrat dostatečné množství dialogových dat

z dané aplikační oblasti, což může někdy znemožnit či přinejmenším znesnadnit a prodloužit vlastní implementaci dialogového systému.

## Literatura


Austin J. L.: *How to Do Things with Words*. Oxford University Press, Oxford, Velká Británie, 1962.

Bečvář P., Pěchouček M., Psutka J., Šmídl L.:*Multi-agentDecisionSupport Systems with Remote Multimedia Access*. - In: Proceedings of the 31st IEEE Industrial Electronics Society Conf. IECON'2005, Releigh, USA, 2005, s. 2204–2209.

Cox R. V., Kamm C. A., Rabiner L. R., Schroeter J., Wilpon J. G.: *Speech and Language Processing for Next-Millennium Communications Services*. Proceedings of the IEEE, **88**, č. 8, 2000, s. 1314–1337.

Donovanr E., Woodlandp C.: *A Hidden Markov-Model-Based Trainable Speech Synthe-sizer*. Computer Speech and Language, **13**, č. 3, 1999, s. 223–241.

Dutoit T.: *An Introduction to Text-To-Speech Processing*. Kluwer Academic Publishers, Dordrecht 1997.

Fujisaki H.: *The Role of Quantitative Modeling in the Study of Intonation*. In: Proc. the Int. Symp. on Japanese Prosody, 1992, s. 163–174.

Grice P.: *Logic and conversation*. In: Cole P., Morgan J. L. (eds): Syntax and semantics. Vol. 3: Speech Acts. Academic Press, New York 1975, s. 41–58.

Hajičová E., Panevová J., Sgall P.: *Úvod do teoretické a počítačové lingvistiky. I. svazek – Teoretická lingvistika*. Karolinum, Praha 2002.

Hsu J. Y.: *A Multi-Agent Framework for Intranet Service Integration*. Lecture Notes in AI 1599, Berlin, Heidelberg, Springer-Verlag, 1999, s. 148–161.

Huang X., Acero A., Hon H. W.: *Spoken Language Processing. A Guide to Theory, Algorithm, and System Development*. Prentice Hall PTR, New Jersey 2001, 980 s.

Hunt A. J., Black A. W.: *Unit Selection in Concatenative Speech Synthesis System Using a Large Speech Database*. In: Proc. ICASSP, Atlanta, USA, 1996, s. 373–376.

Jurafsky D., Martin J. H.: *Speech and Language Processing*. Prentice Hall, Upper Saddle River, New Jersey 2000, 934 s.

Jurčíček F., Zahradil J., Jelínek L.: *A Human-Human Train Timetable Dialogue Corpus*. In: Proceedings of the 9th Eurospeech Conference on Speech Communication and Technology, Lisabon, Portugalsko, 2005, s. 1525–1528 (na CD ROM).

Kanis J., Müller L.: *Using the Lemmatization Technique for Phonetic Transcription in Text-to-Speech System*. TSD 2004. Lecture Notes in Artificial Intelligence 3206, Springer-Verlag, Berlín, Heidelberg 2004, s. 355–361.

Lin B. S., Wang H. M., Lee L.S.: *A Distributed Architecture for Cooperative Spoken Dialogue Agents with Coherent Dialogue State and History*. In: Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop ASRU'99, Keystone, USA, 1999.

Litman D. J., Pan S.: *Empirically evaluation an adaptable spoken dialogue system*. In: Proceedings of the 7th International Conference User Modeling, Banff, Kanada, 1999, s. 55–64.

Makhoul L.: *Linear Prediction: A Tutorial Review*. Proceedings of the IEEE, **63**, č. 4, 1975, s. 561–580.

Manning C. D., Schütze H.: *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts, Londýn 1999.

Mařík V., Štěpánková O., Lažanský J.: *Umělá inteligence 1*. Academia, Praha 1993, 264 s.

Mařík V., Štěpánková O., Lažanský J.: *Umělá inteligence 3*. Academia, Praha 2001, 328 s.

Matoušek J., Psutka J., Krůta J.: *Design of Speech Corpus for Text-to-Speech Synthesis*. In: Proc. Eurospeech, **3**, Aalborg, Dánsko, 2001, s. 2047–2050.

Matoušek J., Tihelka D., Psutka J.: *Automatic Segmentation for Czech Concatenative Synthesis Using Statistical Approach with Boundary-Specific Correction*. In: Proc. Eurospeech, Ženeva, Švýcarsko, s. 301–304.

Matoušek J., Romportl J., Tihelka D., Tychtl Z.: *Recent Improvements on ARTIC: Czech Text-to-Speech System*. In: Proc. ICSLP, Jeju Island, Korea, 2004, s. 1933–1936.

McTear M. F.: *Spoken Dialogue Technology: Enabling the Conversational User Interface*. ACM Computing Surveys, **34**, č. 1, 2002, s. 90–169.

Minker W., Néel F.: *Handling Knowledge Sources in Human-Machine Interaction*. International Journal of Speech Technology, **5**, č. 2, 2002, s. 171–188.

Moulines E., Charpentier F. J.: *Pitch-Synchronous Waveform Processing Technique for Text-to-Speech Synthesis Using Diphones*. Speech Communication, **9**, 1990, s. 453–467.

Moulines E., Verhelst W.: *Time-Domain and Frequency-Domain Techniques for Prosodic Modification of Speech*. In: Kleijn W. B., Paliwal K. J. (eds): Speech Codingand Synthesis, Elsevier, 1995, s. 519–555.

Müller L.: Dekódovací techniky rozpoznávání souvislé řeči. Habilitační práce. ZČU FAV, Plzeň 2002, 140 s.

Ney H., Martin S., Wessel F.: *Statistical Language Modeling Using Leaving–One–Out*. –In: Young, S., and Bloothooft G. (ed.) Corpus-Based Statistical Methods in Speech and Language Processing, Kluwer Academic Publishers, Dordrecht, Boston, Londýn, 1997, s. 174–207.

Psutka J.: *Komunikace s počítačem mluvenou řečí*. Academia, Praha 1995. 283 s.

Psutka J., Müller L., Matoušek J., Radová V.: *Mluvíme s počítačem česky*. Academia, Praha 2006, 752 s.

Romportl J.: *Generování prozodie z textu pro účely syntézy řeči*. Diplomová práce, ZČU FAV, Plzeň 2004.

Romportl J.: *Structural Data-Driven Prosody Model for TTS Synthesis*. In: Proc. of Speech Prosody, TUDpress, Drážďany, s. 549–552.

Sadek M. D., De Mori R.: *Dialogue systems*. In: Spoken Dialogues with Computers (ed. De Mori). Academic Press, London, U.K. s. 523–561.

Sidner C. L.: *Building Spoken Language Collaborative Interface Agents*. TR2002-38, Mitsubishi Electric Research Laboratories, Inc., Cambridge, USA, 2002.

Stolcke A., Ries K., Coccaro N., Shriberg E., Bates R., Jurafsky D., Taylor P., Martin R., Ess-Dykema C. V., Meteer M.: *Dialogue act modeling for automatic tagging and recognition of conversational speech*. Computational Linguistics, **26**, č. 3, 2000, s. 339–373.

Stylianou Y.: *Applying the Harmonic Plus Noise Model in Concatenative Speech Synthesis*. IEEE Trans. on Speech and Audio Processing, **9**, č. 1, 2001, s. 21–29.

Šmídl L., Müller L., Psutka J.: *VoiceXML Based Telephone Dialog System Providing Access to Entrance Examination Results Stored in the University Database*. In: Proceedings of the 6th World Multiconference on Systemics, Cybernetics SCI'2002, Orlando, USA, 2002, s. 276–280.

Taylor P. A.: *The Tilt Intonation Model*. In: Proc. ICSLP, **IV**, Sydney, Austrálie, 1998, s. 1383–1386.

Tihelka D., Matoušek J.: *Unit Selection and its Relation to Symbolic Prosody: a New Approach*. In: Proc. ICSLP (Interspeech) 2006, Pittsburgh, USA, 2006, s. 2042–2045.

Tučková J., Matoušek J.: *Czech Language Features Selection and Prosody Modelling for Text-to-Speech Synthesis*. In: Proc. Int. Workshop on Electronics, Control, Measurment and Signals, ECMS 2003, Liberec, 2003, **1**, s. 98–102.

Young S.: *The Statistical Approach to the Design of Spoken Dialogue Systems*. Tech. ReportCUED/F-INFENG/TR. 433. Cambridge University Engineering Department, 2002.

Zelinka J., Müller L.: *Automatic General Letter-to-Sound Rules Generation for German Text-to-Speech System*. TSD 2004. Lecture Notes in Artificial Intelligence 3206, Springer-Verlag, Berlín, Heidelberg 2004, s. 537–543.

Zahradil J., Šmídl L.: *Czech Dialog System for Examination Registration at UWB*. In: Proceedings of the 10th International Conference Speech and Computer SPECOM'2005, Patras, Řecko, 2005, s. 417–420.

# 16 Geometrie v počítačovém vidění – principy a aplikace rekonstrukce prostorové scény z obrazů

Geometrie vždy hrála v technice velmi důležitou úlohu. Po dva tisíce let byla základem formální vědy a dala základ dnešní axiomatické metodě práce v matematice a technice. Geometrie se těší vzácnému privilegiu, že z několika málo axiomů je možné čistou dedukcí odvodit netriviální a neintuitivní výsledky, které její formalismus přesahují a mají významné aplikace v technice. V této kapitole ukážeme, jak důležitou roli hraje geometrie v počítačovém vidění, a představíme základní principy, na kterých stojí dvě zajímavé aplikace, které lze použít v běžném životě.

## 16.1 Geometrie v počítačovém vidění

Počítačové vidění (Hlaváč, Šonka, 1992), (Šonka a kol, 1998) je disciplínou technických věd. Jeho hlavním tématem je porozumět principům tvorby dvojdimenzionálního obrazu trojdimenzionálního světa a zpětná „rekonstrukce" tvaru a vzhledu světa a podmínek, za kterých obraz vznikl. Geometrie zde hraje zásadní úlohu. Dovoluje přesně modelovat vznik obrazu a s použitím algebry formalizovat techniku řešení problémů.

Geometrije se vždy inspirovala praktickými a technickými problémy. Základní pojmy a metody se konstruovaly tak, aby se jimi technické problémy daly efektivně řešit. V devatenáctém století, kdy se bouřlivě rozvinula projektivní a neeukleidovská geometrie, byla intuitivní metoda nahrazena axiomatickou metodou řešení matematických problémů, která výsledek odvozuje konečnou posloupností dedukcí z počátečních axiomů (Hilbert, 1900). Hledání definitivní podoby základů geometrie ve tvaru malého úplného počtu axiomů završila na konci devatenáctého století práce Davida Hilberta (Hilbert, 1899), ve které Hilbert formuloval 21 axiomů, z nichž je možné odvodit eukleidovskou geometrii, zkonstruovat lineární prostor, algebru i reálná čísla a tím základ velké části matematiky (Artin, 1957). To dalo pevné základy i řešení mnohých technických problémů.

Problémy počítačového vidění nejsou čistě geometrické. Cílem bývá vyřešit praktický problém, jako je například konstrukce panoramatického obrazu, rekonstrukce tvaru prostorového objektu z jeho fotografií, sledování pohybu kamery ze sekvence obrazů, které snímala, segmentace scény na statické pozadí a pohybující se objekty, náhrada jednoho pozadí za jiné, skládání videosekvencí sejmutých na různých místech a v různých časech do jednoho „filmu", který vypadá jakoby pořízen jednou kamerou na jednom místě. Řešení takových problémů vyžaduje kombinovat modelování a analýzu založenou na

geometrizaci problému, statistiku, která se vyrovná s nepřesnostmi vznikajícími při pořízení reálného digitálního obrazu, a optimalizaci, v jejímž rámci se problém obvykle formuluje. Úloha geometrie je klíčová, neboť bez přesného popisu situace není možné úlohu smysluplně formulovat a dosáhnout užitečných výsledků.

Pionýrské práce v počítačovém vidění (Horn, 1987) na geometrii stavěly, ale skutečný rozvoj geometrických technik v počítačovém vidění nastal až po uveřejnění práce „A computer algorithm for reconstruction a scene from two projections" Longuet-Higginse (Longuet-Higgins, 1981), která napojila zpracování digitálního obrazu a umělou inteligenci na klasické výsledky v projektivní geometrii a fotogrammetrii.

Základy projektivní geometrie, které položil J.-V. Poncelet v roce 1822, byly dále rozpracovány ve von Staudtově práci „Geometrie der Lage" publikované v roce 1847 (Buchanan, 1993). Na ně navazuje řada klasických monografií, kterou završuje kniha „Algebraic Projective Geometry" autorů Sempleho a Kneebona z roku 1952 (Semple a Kneebone, 1998). Ta se stala základní literaturou o projektivní geometrii pro počítačové vidění a v jejím rámci byl do značné míry vyřešen jeden ze základních problémů počítačového vidění – rekonstrukce tvaru scény z obrazů.

Problém rekonstrukce scény z jejích obrazů byl poprvé řešen fotogrammetry v devatenáctém století (Buchanan, 1993). Pojmy epipolární přímka a projektivní rekonstrukce se objevily v přednášce S. Finsterwaldera, kterou přednesl před Německou matematickou společností v roce 1897. Známý „osmibodový algoritmus", který explicitně konstruuje smysluplné trojdimenzionální souřadnice bodů z jejich projekcí do dvou obrazů a který do počítačového vidění přinesl zmíněný Longuet-Higginsův článek, poprvé formuloval M. Chasles v roce 1853. Poprvé jej v roce 1863 vyřešil O. Hesse, roku 1869 řešení vylepšil R. Sturm a moderní efektivní algoritmus publikoval v roce 2004 D. Nistér (Nistér, 2004).

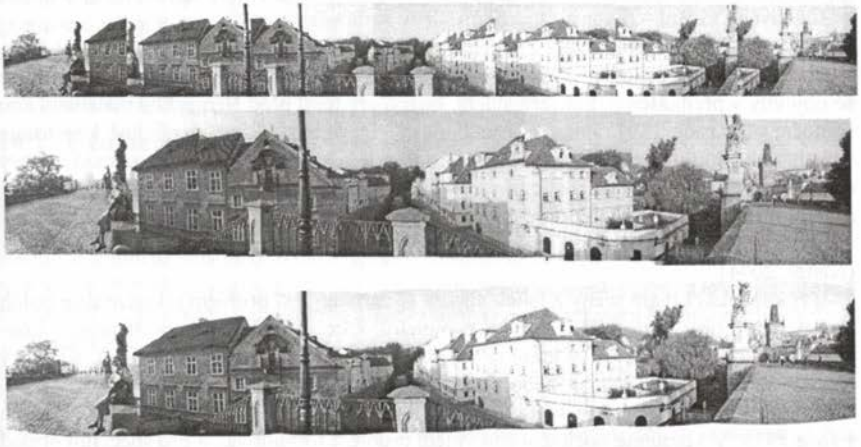
Při konstrukci tvaru scény z jejích obrazů se zároveň řeší problém rekonstrukce poloh kamer, které obrazy pořídily, a jejich parametrů, jako je ohnisková vzdálenost, a konstrukce souřadné soustavy, ve které se v obrazu měří polohy obrazových bodů. Klasický problém autokalibrace kamer, neboli jak identifikovat parametry a polohy kamery z několika obrazů téže neznámé scény, poprvé formuloval a částečně i vyřešil E. Kruppa v roce 1913. V komunitě počítačového vidění problém formuloval, a pro speciální případ volného rovnoběžného promítání vyřešil, S. Ullmann v roce 1977 (Ullman, 1979). V plné obecnosti byl problém dořešen S. Maybankem v roce 1992 (Maybank, Faugeras, 1992).

Jeden z hlavních přínosů počítačového vidění spočívá ve spojení teoretických výsledků z geometrie a fotogrammetrie se statistickými a optimalizačními metodami pro zpracování nepřesných a i zcela chybných měření. Typickou úlohou ve fotogrammetrii je změřit rozměry prostorového tělesa pomocí speciální přesně kalibrované fotogrammetrické kamery. Přitom se běžně předpokládá, že měření prováděná v obrazech jsou dána zvenku, například uživatelem, a jsou přesná nebo zatížená nepřesností odečtu, kterou lze dobře modelovat. Počítačové vidění řeší obecnější úlohu, ve které se konstruují měření v obrazech tak, aby z nich bylo možné rekonstruovat tvar a vzhled scény a scénu interpretovat. To přináší nové problémy, v nichž se kombinují klasické geometrické výsledky s výpočetními technikami ze statistiky a optimalizace.

## 16.2 Konstrukce panoramatického obrazu – klasický problém počítačového vidění

Nastiňme si, jak vypadá řešení klasického problému počítačového vidění – automatické konstrukce panoramatického obrazu z několika fotografií, které se částečně překrývají. Snad každý, kdo vyzkoušel spojit několik částečně se překrývajících fotografií do jedné panoramatické fotografie, si musel položit otázku: „Jak to funguje?" Nejprve se podívejme, čeho lze v tomto konkrétním problému dosáhnout, když použijeme korektní geometrický model, a jak se výsledek liší oproti výsledku, který dostaneme s „naivním" přístupem.

Na obrázku 16.1 (též barev. příl.) nahoře vidíme sedm fotografií zachycujících panoráma Kampy. Fotografie byly pořízeny „z ruky", ale tak, aby se částečně překrývaly a fotoaparát se otáčel na místě.



Obr. 16.1 Sedm obrazů zachycujících panoráma Kampy. „Naivně" složený panoramatický snímek se zřetelnými švy. Panoramatický snímek bez zřetelných švů složený s použitím korektního modelu kamery.

Uprostřed vidíme panoramatický obraz zkonstruovaný „naivní" technikou bez využití přesného geometrického modelu perspektivního promítání. Fotografie byly ručně umístěny a překryty tak, aby na sebe navazovaly objekty, které byly blízko fotoaparátu. Na panoramatickém obrazu jsou zřetelné „švy", které jsou tím zřetelnější, čím jsou překrývající se objekty od fotoaparátu dále. Dole vidíme panoramatický obraz zkonstruovaný automaticky na základě přesného modelu perspektivního promítání. Obraz je bez švů a navíc, jak si dále vysvětlíme, přesný model umožňuje fotografie složit automaticky.

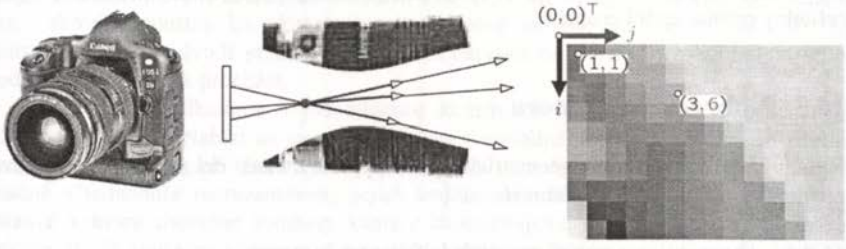
Jak to lze udělat? Stačí vyřešit tři podproblémy. Podproblém první spočívá v tom jak nalézt lícovací transformaci sousedních obrazů, která obrazy přes sebe překryje tak, aby na sebe objekty na hranici obrazů plynule navazovaly nezávisle na jejich vzdálenosti od fotoaparátu. To je možné, pokud správně modelujeme perspektivní projekci a zajistíme,

aby se fotoaparát otáčel okolo svého středu promítání. V takovém případě je lícovací transformace, v projektivní geometrii známá jako rovinná kolineace nebo homografie, racionální funkcí souřadnic obrazových bodů a nezávisí na vzdálenosti bodů scény od fotoaparátu.

Obvykle nevíme, kde přesně se nachází střed projekce fotoaparátu, ale zkušenost ukazuje, že otáčením fotoaparátu „na místě" okolo středu objektivu dostáváme dobrý výsledek pro všechny body ve scéně od vzdálenosti, jež je obvykle menší než minimální vzdálenost, od které je obraz ostrý. Není tedy těžké zajistit takové podmínky snímání, aby se dal použít předpoklad, že se fotoaparát otáčí kolem středu promítání.

Mnohem obtížnější je zajistit, aby otočení mezi každými dvěma snímky bylo stejné, a tak byly stejné i lícovací transformace, které jsou otočením určeny. Při fotografování „z ruky" je obecně každá lícovací transformace jiná a neznámá. Je třeba ji získat z obrazů. Projektivní geometrie nás učí, že každá rovinná kolineace $H$ je určena vzory a obrazy čtyř bodů, z nichž žádné tři neleží na jedné přímce. Nabízí se tedy nalézt čtyři takové páry korespondujících bodů – korespondencí – a z nich lícovací transformaci získat.

Druhý podproblém tedy spočívá v automatickém nalezení korespondujících bodů a lícovacích transformací. Zde se jedná o problém detekce, rozpoznání a verifikace.




a) b) c)

Obr. 16.2 a) Digitální fotoaparát. b) Geometrický model dírkové kamery. c) Souřadná soustava v digitálním obrazu.

Předpokládáme, že se obrazy částečně překrývají, a tedy alespoň částečně zachycují stejnou scénu. Dále předpokládejme, že ve scéně jsou objekty, jejichž projekce lze v obrazech detekovat, k nim jednoznačně přiřadit jejich lokální souřadné soustavy a v nich zkonstruovat popis, který objekty vzájemně odlišuje a identifikuje. Korespondující obrazy objektů pak lze předběžně spárovat na základě rozpoznání podobnosti jejich popisů. Nelze v žádném případě doufat, že tak vždy získáme správné korespondence, ale velmi často získáme směs správných a špatných korespondencí, ve které je správných korespondencí dostatek k určení lícovacích transformací. Je však třeba oddělit korespondence správné od těch špatných.

Víme, že správné korespondence jsou vázány stejnou lícovací transformací. Zároveň se ukazuje, že „špatnosti" špatných korespondencí jsou obecně nezávislé, a tak pravděpodobnost, že velké množství špatných korespondencí bude vázáno stejnou lícovací transformací, je velmi malá. Má tedy smysl hledat největší podmnožinu množiny předběžných korespondencí, jejíž všechny prvky lze lícovat stejnou transformací. To je

v principu obtížný kombinatorický optimalizační problém, který v úplné obecnosti vyžaduje testovat všech $2^N$ podmnožin $N$-prvkové množiny předběžných korespondencí. Naštěstí v tomto případě lze řešení získat jednodušeji díky silnému geometrickému modelu, který je k dispozici. Trik spočívá v tom, že lícovací transformace jsou jednoznačně určeny pouhými čtyřmi korespondencemi. Přitom každá čtveřice správných korespondencí v ideálním případě generuje stejnou lícovací transformaci. To znamená, že stačí projít „jen" $N(N-1)(N-2)(N-3)/24$ čtveřic korespondencí, pro každou z nich vypočíst lícovací transformaci a vybrat tu, která lícuje největší počet korespondencí. V praktických situacích navíc obvykle stačí testovat mnohem menší počet náhodně vybraných čtveřic, protože pravděpodobnost, že při náhodném generování nebude ani jednou vybrána čtveřice správných korespondencí, rychle klesá s rostoucím poměrem počtu správných korespondencí ku počtu korespondencí špatných.

Třetí podproblém spočívá v konstrukci barev v pixelech výsledného panoramatického obrazu. Ty konstruujeme tak, že pro každý pixel $x$ výsledného obrazu najdeme jeho předobrazy $\mathrm{inv}(H_i)(x)$ ve vstupních obrazech, interpolací získáme barvy v bodech $y_i$ a z nich, například jako vážený průměr, zkonstruujeme barvu v pixelu $x$.

Díky přesnému geometrickému modelu formování obrazu jsme schopni obrazy nejen přesně slícovat, ale i naformulovat automatické nalezení lícovacích transformací jako řešitelný optimalizační problém.

## 16.3 Z obrazu do prostoru

Nyní vysvětlíme základní geometrické principy, díky nimž dokážeme rekonstruovat prostorový tvar a z plošných obrazů.

### 16.3.1 Perspektiva a model promítání dírkovou kamerou

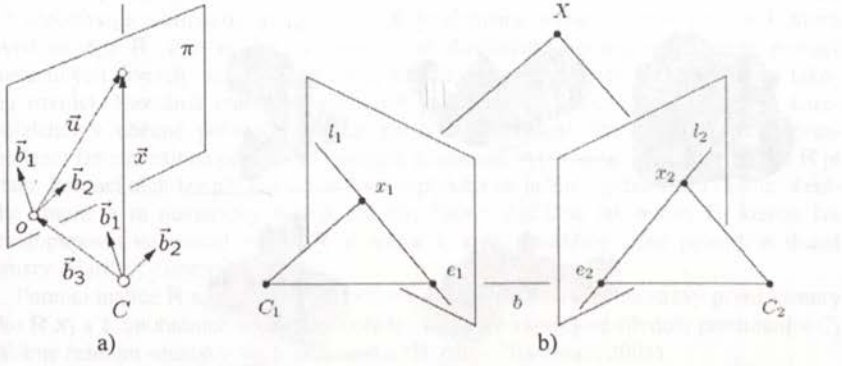
Nejzákladnějším principem je model perspektivního promítání, které vzniká při promítání dírkovou kamerou na rovinnou průmětnu. Tajemství modelu perspektivního promítání bylo odhaleno v patnáctém století. V roce 1426 namaloval Massacio podle pravidel, která formuloval Alberti, první korektní perspektivní obraz Trinita con la Vergine a San Giovanni (Kemp, 1992). Že Massacio správně použil Albertiho teorii, je do značné míry vidět prostým okem, ale, že to udělal velmi přesně, ukazuje Criminisiho práce (Criminisi, 2001), který s modelem perspektivního promítání zrekonstruoval prostorový model „Trinity". Model perspektivního promítání velmi dobře popisuje, jak vzniká obraz, takže dominuje v umění, fotografii, fotogrammetrii, počítačové grafice a počítačovém vidění (Wade, 1998). Většina technik počítačového vidění, díky nimž jsme dnes schopni rekonstruovat tvar trojdimenzionální scény z jejích obrazů (Hartley, Zisserman, 2003), (Faugeras a kol., 2001), na tomto modelu staví. Existují i situace, kdy je třeba použít modely obecnější. Například obraz získaný z plochého skeneru nelze modelovat dírkovou kamerou, protože paprsky, kterými vzniká, se neprotínají v jednom bodě. I pro takové situace lze vybudovat teorii a algoritmy pro rekonstrukci scény z obrazů (Pajdla, 2002), (Mičušík, Pajdla, 2006), ale zde se soustředíme na základní principy rekonstrukce z perspektivních obrazů.

Nyní model popíšeme. Složitá optická soustava typického digitálního fotoaparátu, obrázek 16.2(a), též barev. příl., se nahradí jednoduchým geometrickým modelem, který odpovídá dírkové kameře. Body z prostoru jsou promítány podél přímek, paprsků, protínajících se ve středu promítání, který je na obrázku 16.2(b) (též barev. příl.) znázorněn jako bod uvnitř objektu, do rovinné průmětny, kterou představuje obdélníček. Obrázek 16.3(a) zachycuje schematický model perspektivního promítání z prostoru do rovinné průmětny $\pi$ přes střed promítání $C$. Abychom mohli situaci přehledně znázornit, průmětnu kreslíme před střed promítání.

V průmětně je obraz digitalizován. V drtivé většině digitálních fotoaparátů jsou pixely uspořádány do řádků a sloupců pravoúhlé mřížky, obrázek 16.2(c), též barev. příl., takže lze řádkový a sloupcový index pixelu interpretovat jako souřadnice vektoru, který střed pixelu zaměřuje v nějaké souřadné soustavě obrazu. Na obrázku 16.3(a) je souřadná soustava obrazu zakreslena s počátkem v bodě $o$ a bázovými vektory $\boldsymbol{b}_1$, $\boldsymbol{b}_2$. V ní je obrazový bod s řádkovým indexem $i$ a sloupcovým indexem $j$ zaměřen vektorem $\boldsymbol{u}$ se souřadnicemi $(i, j)$.

Každému bodu v průmětně odpovídá paprsek procházející středem promítání. Tím je paprsek jednoznačně určen. Přiřaďme mu smysluplné souřadnice. K tomu je třeba nějakým způsobem zvolit souřadnou soustavu kamery, k níž se budou souřadnice vztahovat. Takových soustav lze zkonstruovat nekonečně mnoho, ale nás zajímají jen ty konstrukce, které odvodí předpis konkrétního paprsku od obrazových souřadnic toho bodu, kterým paprsek prochází.

Ukažme jednu šikovnou a jednoduchou konstrukci. Všechny paprsky procházejí středem promítání. Nabízí se tedy volit počátek souřadné soustavy kamery ve středu promítání $C$, protože při této volbě budou paprsky v souřadné soustavě kamery jednoznačně a jednoduše reprezentovány jejich trojdimenzionálními směrovými vektory $\boldsymbol{x}$. Bázové vektory souřadné soustavy kamery zkonstruujeme tak, že použijeme bázové vektory $\boldsymbol{b}_1$, $\boldsymbol{b}_2$ souřadné soustavy obrazu a doplníme je o třetí bázový vektor $\boldsymbol{b}_3$, který je přiřazen dvojici bodů $C$, $o$. V takové bázi bude mít směrový vektor $\boldsymbol{x}$ paprsku, který prochází bodem $(i, j)$, souřadnice $(i, j, 1)$.

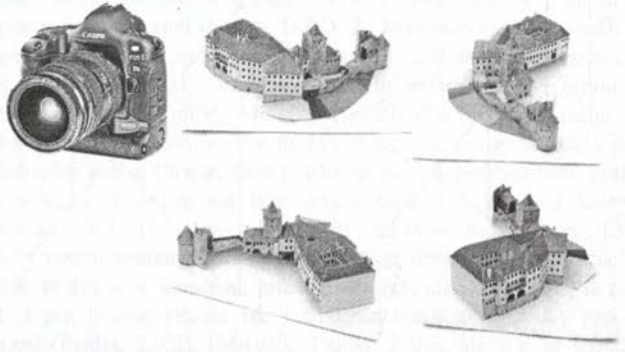



Obr. 16.3 a) Geometrický model dírkové kamery. b) Epipolární geometrie. Bod $X$ se promítá do obrazů $x_1$, $x_2$. Body $X$, $C_1$, $C_2$ tvoří epipolární rovinu.

Získali jsme tak velmi jednoduchý předpis jak z obrazových souřadnic konstruovat smysluplné souřadnice paprsků. Ke všem obrazovým souřadnicím pouze připojíme jedničku. Všimněme si ale, že souřadnice směrového vektoru paprsku nezávisí na poloze bodů $C$ a $o$ ani na úhlech mezi bázovými vektory $\boldsymbol{b}_1$, $\boldsymbol{b}_2$, $\boldsymbol{b}_3$. Nemůžeme tedy přímo měřit úhly mezi paprsky ze souřadnic jejich směrových vektorů tak, že bychom použili standardní vzorec pro výpočet skalárního součinu vzhledem k ortonormální bázi, protože výsledek závisí na volbě bázových vektorů. Dostáváme tak model tzv. nekalibrované kamery. Model nekalibrované kamery je velmi důležitý, protože umožňuje smysluplně pracovat i s obrazy, u nichž ani nevíme, jak přesně byly pořízeny.

Pokud bychom dokázali zvolit souřadnou soustavu obrazu tak, aby vektory $\boldsymbol{b}_1$, $\boldsymbol{b}_2$, $\boldsymbol{b}_3$ tvořily ortogonální bázi, tedy aby bod $o$ byl patou kolmice spuštěné z bodu $C$ na rovinu promítání, a $\boldsymbol{b}_1$, $\boldsymbol{b}_2$, $\boldsymbol{b}_3$ byly stejně dlouhé a vzájemně na sebe kolmé, pak by úhel mezi dvěma paprsky byl přímo dán vzorcem standardního skalárního součinu jejich směrových vektorů. V tomto případě bychom získali model tzv. kalibrované kamery.

Jedním z přínosů fotogrammetrie a počítačového vidění je postup jak z obrazů známé scény určit polohu kamery ve scéně a jak zkonstruovat vhodnou souřadnou soustavu kamery a tím ji „zkalibrovat" (Hartley, Zisserman, 2003). Je překvapivé, že polohu pohybující se kamery a její kalibraci lze zjistit, a tak ji tzv. „autokalibrovat", i když scéna, kterou kamera pozoruje, je neznámá. Stačí mít k dispozici tři obrazy téže scény sejmuté z různých míst tou samou kamerou a v nich šest trojic korespondujících bodů (Hartley, Zisserman, 2003). Řešení autokalibrace vede na systém tří algebraických rovnic čtvrtého a jedné rovnice třetího stupně ve třech proměnných. Takový systém rovnic lze v principu řešit metodami algebraické geometrie (Cox, Little, O'Shea, 2005), avšak obvykle není snadné získat užitečné výsledky, protože chyby v odečtu poloh bodů v obrazech velmi silně ovlivňují kořeny systému. Praktickým přístupem k řešení je využít toho, že některé parametry kamery jsou v praxi přibližně známy, zafixovat je na přibližných hodnotách, tím problém zjednodušit, dopočíst odhady ostatních parametrů a následně všechny parametry „doladit" gradientní optimalizací (Hartley, Zisserman, 2003). My se zde problémem autokalibrace dále hlouběji zabývat nebudeme a postoupíme ke konstrukci smysluplných souřadnic bodů v prostoru z jejich projekcí do obrazů.



Obr. 16.4. Digitální fotoaparát a čtyři obrazy části Langweilova modelu Prahy.

### 16.3.2 Konstrukce smysluplných trojdimenzionálních souřadnic z korespondencí v obrazech

Na obrázku 16.3(b) vidíme bod $X$ v prostoru, který se promítá dvěma kamerami do dvou korespondujících obrazových bodů. Obrazové body reprezentujeme trojdimenzionálními směrovými vektory $\boldsymbol{x}_1$, $\boldsymbol{x}_2$ jejich paprsků, které jsme zkonstruovali metodou popsanou v předchozím výkladu. Naším cílem je bod $X$ zpětně zrekonstruovat z vektorů $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, o kterých předpokládáme, že jsou naměřeny zcela přesně. Získat bod $X$ se zdá být snadné. Stačí paprsky generované vektory $\boldsymbol{x}_1$, $\boldsymbol{x}_2$ vyjádřit ve stejné souřadné soustavě a vypočíst jejich průsečík. Jak ale vyjádřit oba paprsky ve stejné souřadné soustavě, když nevíme, odkud kamery scénu snímaly?

Předpokládejme, že kamery jsou kalibrovány, tedy že souřadné soustavy kamer mají počátek v bodech $C_1$, $C_2$ a ortogonální báze se stejně dlouhými bázovými vektory. Souřadné soustavy kamer jsou plnohodnotnými prostorovými souřadnými soustavami, takže lze jednu z nich zvolit a vzhledem k ní vše vyjadřovat. Zvolíme souřadnou soustavu kamery se středem v $C_1$. Paprsek $\boldsymbol{x}_1$ je v této souřadné soustavě určen vektorem $\boldsymbol{x}_1$, který je nám dán, a počátkem $C_1$ o souřadnicích (0,0,0). Paprsek $\boldsymbol{x}_2$ je obdobně určen v souřadné soustavě druhé kamery vektorem $\boldsymbol{x}_2$ a bodem $C_2$. Bohužel neznáme transformaci souřadnic ze souřadné soustavy druhé kamery do souřadné soustavy první kamery, protože ta závisí na relativní poloze kamer, která není známa. Proto nemůžeme paprsek $\boldsymbol{x}_2$ přímo přepočítat do souřadné soustavy první kamery. Jak tedy dále pokročit?

Významným výsledkem geometrie, který byl dále rozvinut ve fotogrammetrii a počítačovém vidění (Hartley, Zisserman, 2003) je technika jak určit neznámou transformaci, a tím i relativní polohu kamer, ze znalosti pěti korespondencí ve dvou obrazech. Jejím základem je tzv. epipolární omezení vyplývající z toho, že všechny korespondující paprsky se v prostoru protínají. Geometricky to znamená, že vektor $\boldsymbol{t}$, který zaměřuje střed druhé kamery $C_2$ v souřadné soustavě první kamery, vektor $\boldsymbol{x}_1$ a vektor $\boldsymbol{x}_2$ jsou lineárně závislé. To lze algebraicky zapsat jako identitu $(\boldsymbol{x}_1 \times \boldsymbol{t}) \,.\, (\mathbf{R}\,\boldsymbol{x}_2) = 0$, kde „$\times$“ představuje vektorový a „$.$“ skalární součin. Identita znamená, že vektor $(\boldsymbol{x}_1 \times \boldsymbol{t})$, který je kolmý na $\boldsymbol{x}_1$ i na $\boldsymbol{t}$, je také kolmý na vektor $\boldsymbol{x}_2$ přepočtený do souřadné soustavy první kamery rotací $\mathbf{R}$. Identitu lze přepsat do maticového tvaru $\boldsymbol{x}_1' \,\mathbf{E}\, \boldsymbol{x}_2 = 0$, kde čárka „$'$“ představuje maticovou transpozici a $\mathbf{E}$ je neznámá matice o velikosti $3 \times 3$, která závisí na $\boldsymbol{t}$ a $\mathbf{R}$. Pro známé korespondence dostáváme homogenní lineární rovnici v neznámých prvcích matice $\mathbf{E}$ a vidíme, že každá korespondence poskytne jednu takovou rovnici. Neznámá matice $\mathbf{E}$ má devět prvků, takže bychom potřebovali 8 korespondencí v obecné poloze k získání netriviálního řešení. Počet potřebných korespondencí lze zmenšit na pět, pokud využijeme omezení vyplývající z toho, že matice $\mathbf{R}$ je rotací. Pomocí nich lze pět lineárních rovnic převést na jednu algebraickou rovnici desátého stupně a tu numericky vyřešit (Nistér, 2004). Získáme tak matici $\mathbf{E}$, kterou lze dekomponovat na matici rotace $\mathbf{R}$ a vektor $\boldsymbol{t}$, který zaměřuje střed promítání druhé kamery (Hartley, Zisserman, 2003).

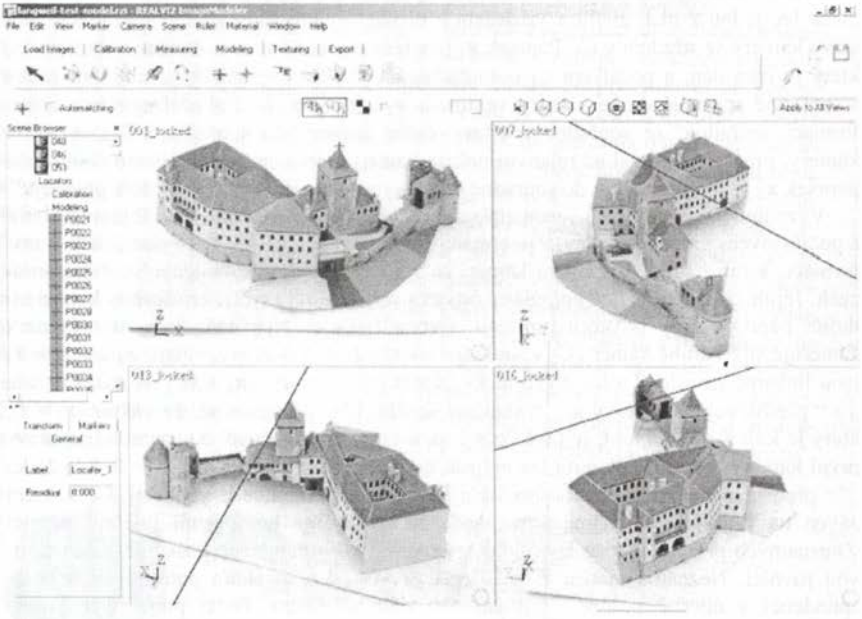
Pomocí matice $\mathbf{R}$ a vektoru $\boldsymbol{t}$ přepočteme $\boldsymbol{x}_2$ a $C_2$ do souřadné soustavy první kamery jako $\mathbf{R}\,\boldsymbol{x}_2$ a $\boldsymbol{t}$. Souřadnice bodu $X$ v souřadné soustavě kamery se středem promítání v $C_1$ získáme řešením soustavy lineárních rovnic (Hartley, Zisserman, 2003).

Tím jsme završili výklad základního principu, na kterém stojí rekonstrukce trojdimenzionální scény z jejích obrazů. Přitom jsme pominuli, že měření v obrazech nejsou

absolutně přesná, a tedy epipolární omezení na korespondující paprsky neplatí zcela přesně. V praxi tedy místo přímých výpočtů založených na protínání musíme řešit optimalizační úlohy, při kterých minimalizujeme odchylky mezi body naměřenými v obrazech a projekcemi bodů rekonstruovaných (Hartley, Zisserman, 2003).

### 16.3.3 ImageModeler – interaktivní prostorový modelář

Popsané principy rekonstrukce se vyvíjely v akademickém prostředí zhruba od roku 1985 do roku 2000, kdy dosáhly takové vyspělosti, že je bylo možné aplikovat na praktické problémy. Po experimentálních nástrojích, které implementovaly a testovaly různé rekonstrukční algoritmy, jako byl například rekonstrukční toolbox RECX pro Matlab (Buriánek a Pajdla, 1998), se objevily obdobné profesionální programy. Příkladem může být program REALVIZ ImageModeler [1].



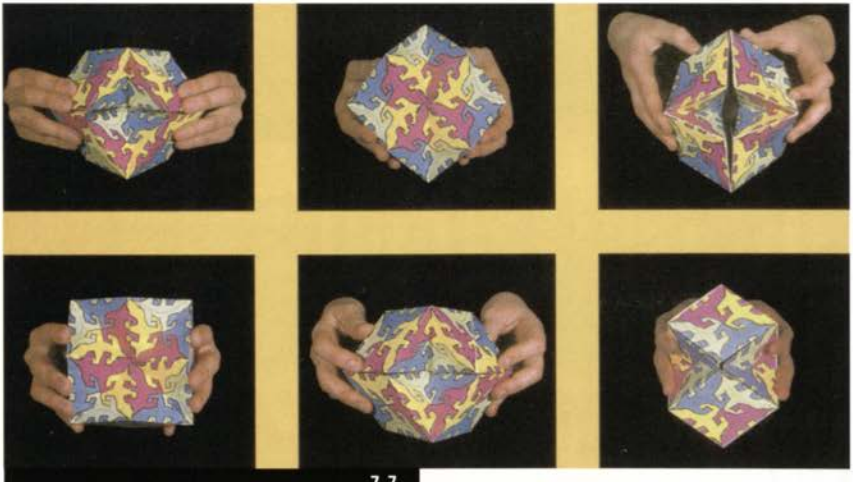
Obr. 16.5 Interaktivní modelář REALVIZ ImageModeler.

ImageModeler je interaktivní modelář, který umožňuje ručně zadat korespondující body, z nich vypočíst kalibrace a relativní polohy kamer, zrekonstruovat body v prostoru, definovat polygony s vrcholy v rekonstruovaných bodech a ty pak pokrýt texturou získanou z obrazů. Rekonstrukce bodů využívá vysvětlených principů, které jsou doplněny o optimalizace tam, kde je třeba se vyrovnat s nepřesnostmi měření v obrazech.

Ukažme si na příkladu, čeho lze s takovým modelářem dosáhnout. Na obrázku 16.4 vidíme digitální fotoaparát Canon EOS 1Ds a čtyři z patnácti snímků části kopie Langweilova modelu Pražského hradu, který jím byl vyfotografován. Snímky byly poří-

5.8

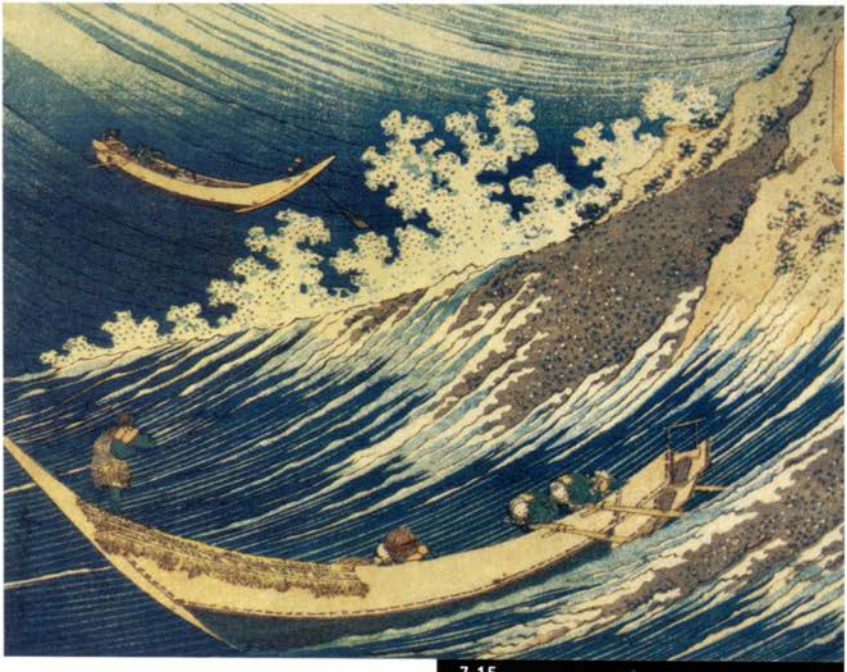
5.10



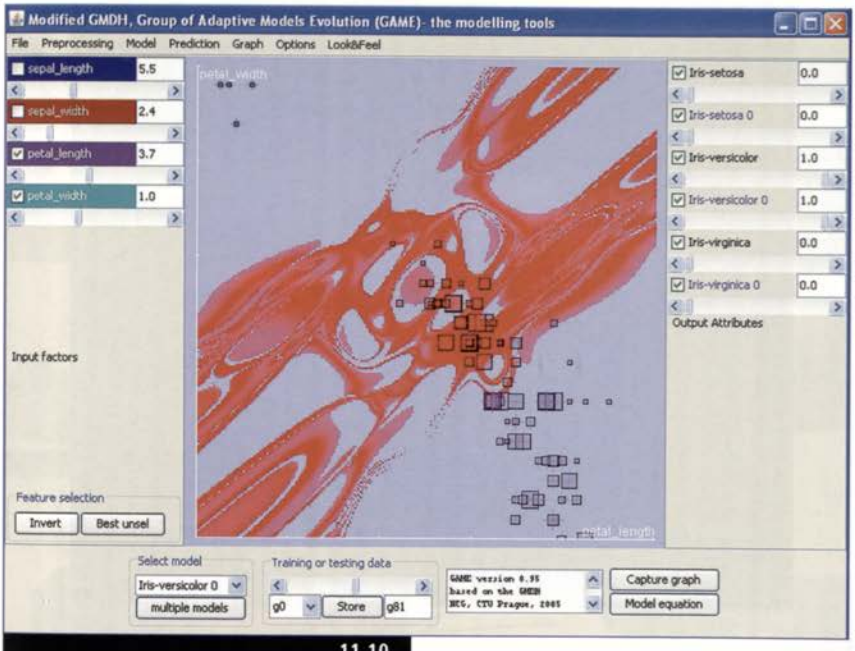
7.7

7.11

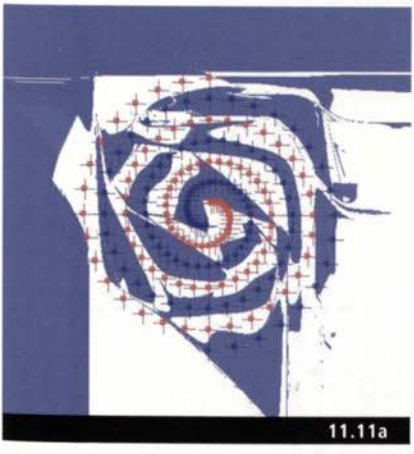
7.14



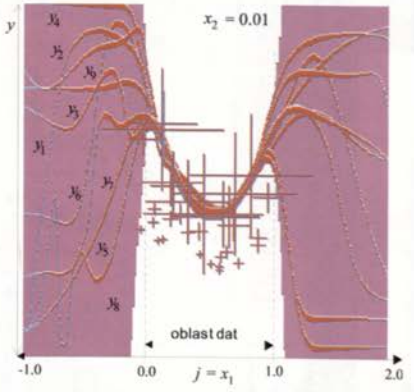
7.15



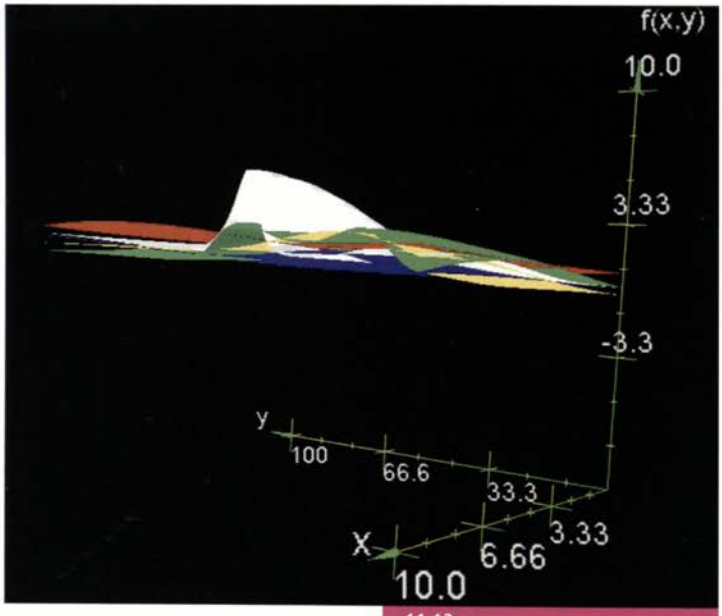

11.10



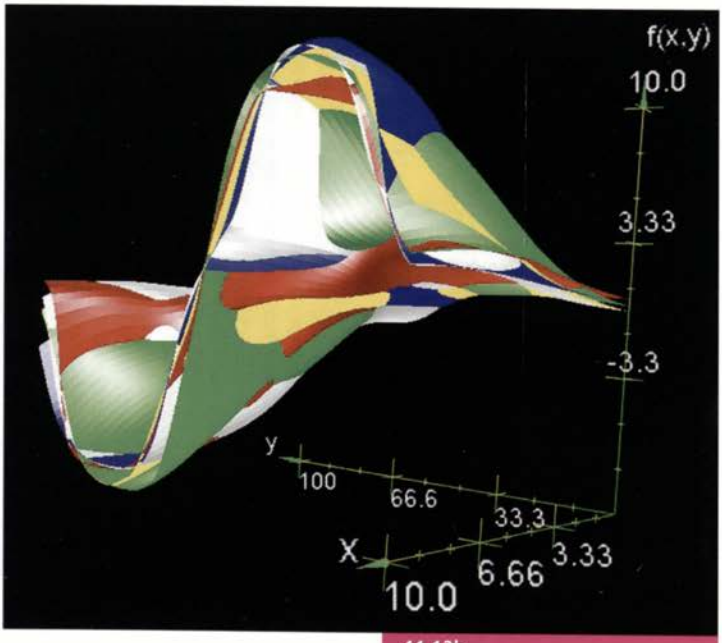
11.11a




11.11b

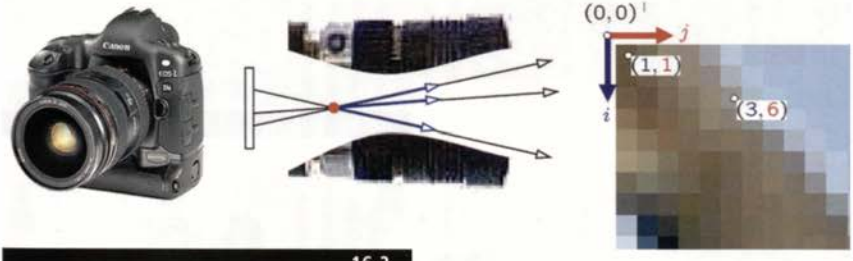

f(x,y)
10.0
3.33
-3.3
y
100
66.6
33.3
3.33
X
6.66
10.0

11.12a


f(x,y)
10.0
3.33
-3.3
y
100
66.6
33.3
3.33
X
6.66
10.0

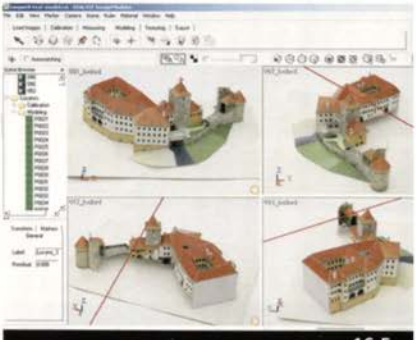
11.12b

16.1

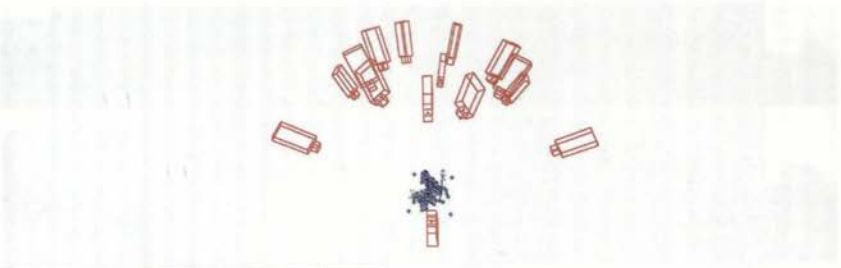


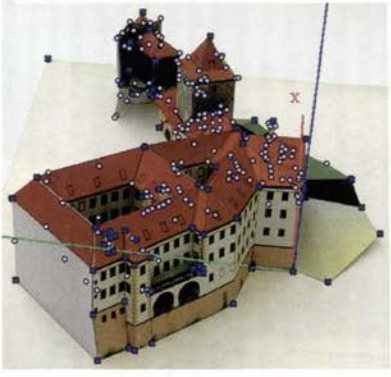

16.2

16.4



16.5

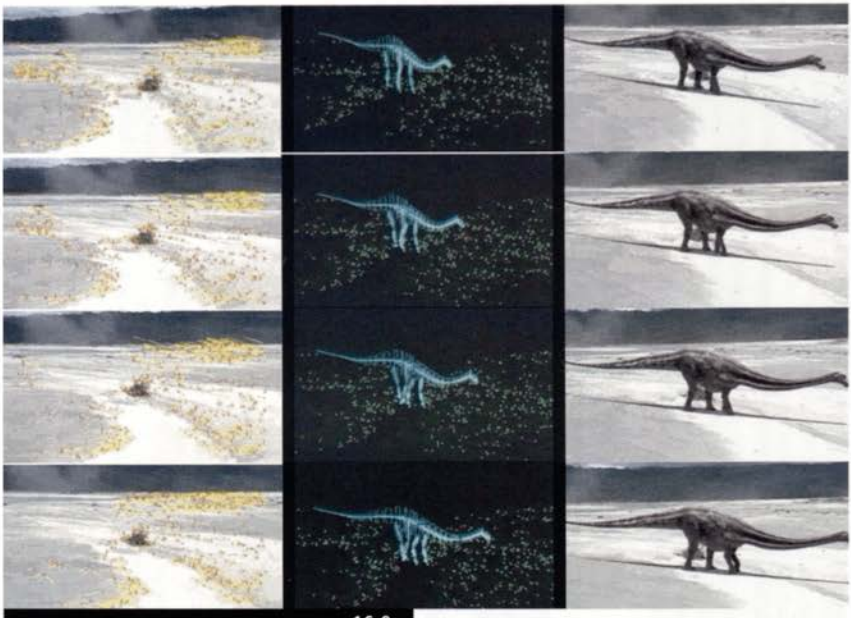




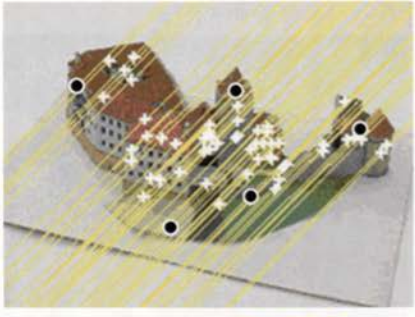
16.6

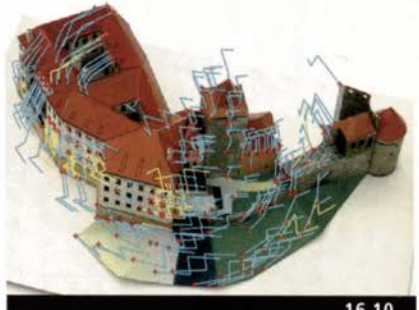
16.7

16.8

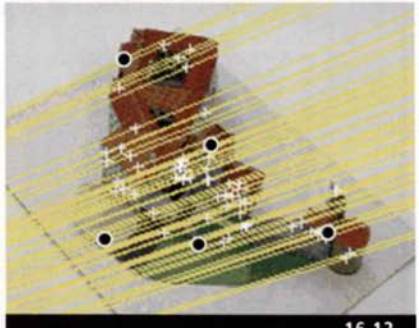


16.9

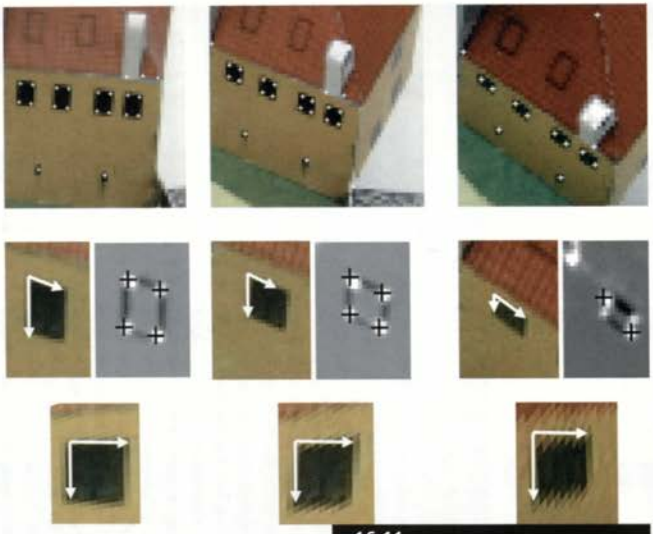




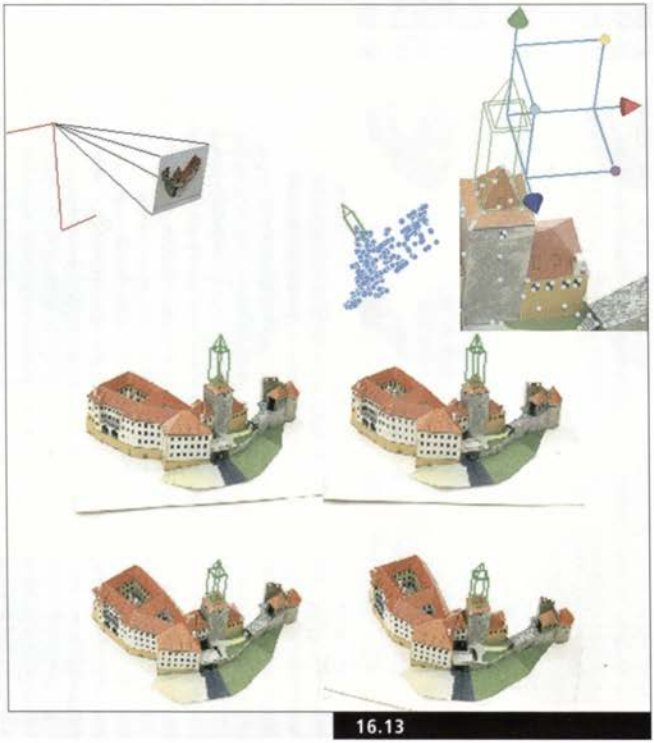
16.10



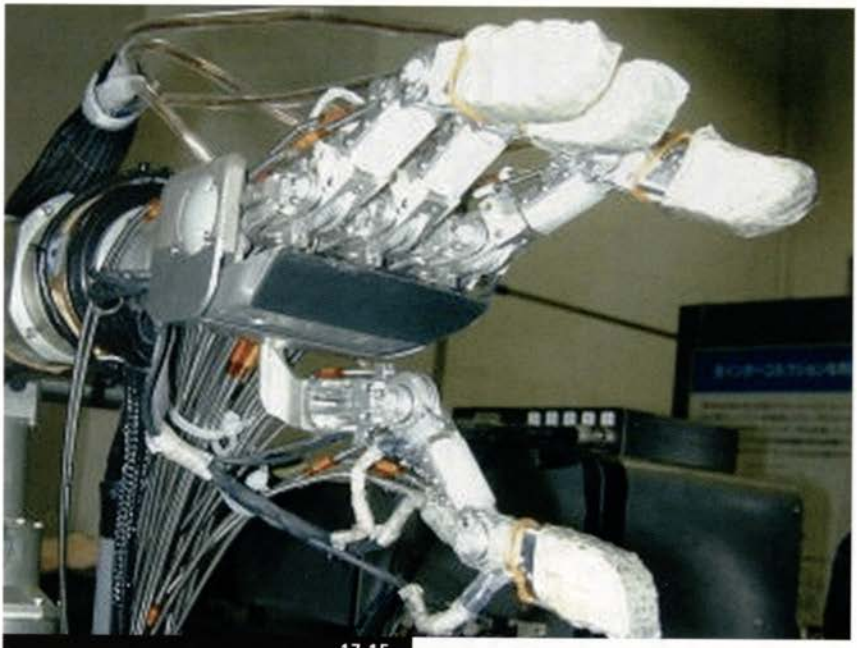
16.12



16.11



16.13

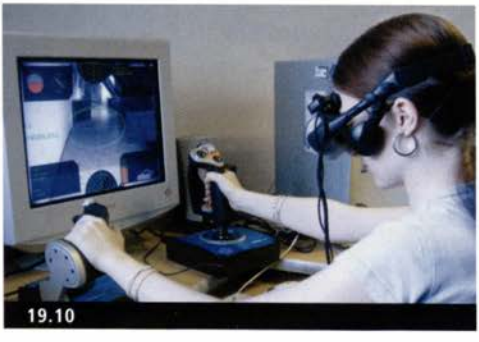


17.15

17.16

19.3

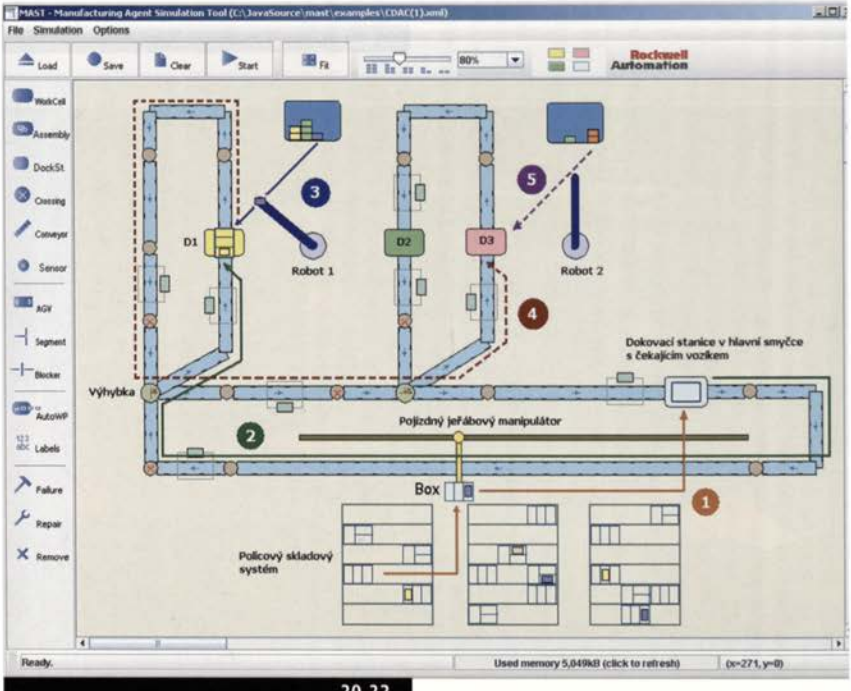
19.8



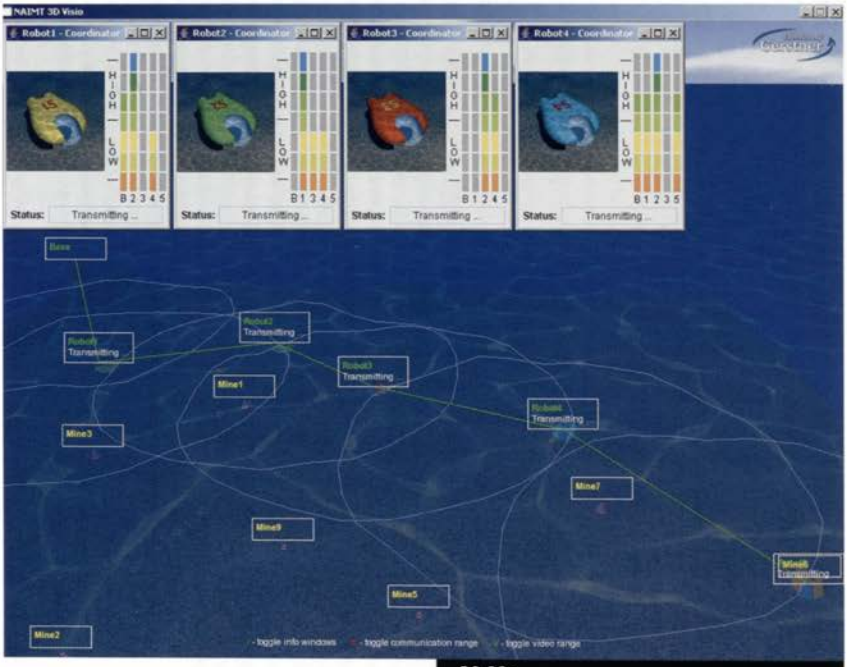
19.10

SICK
SICK

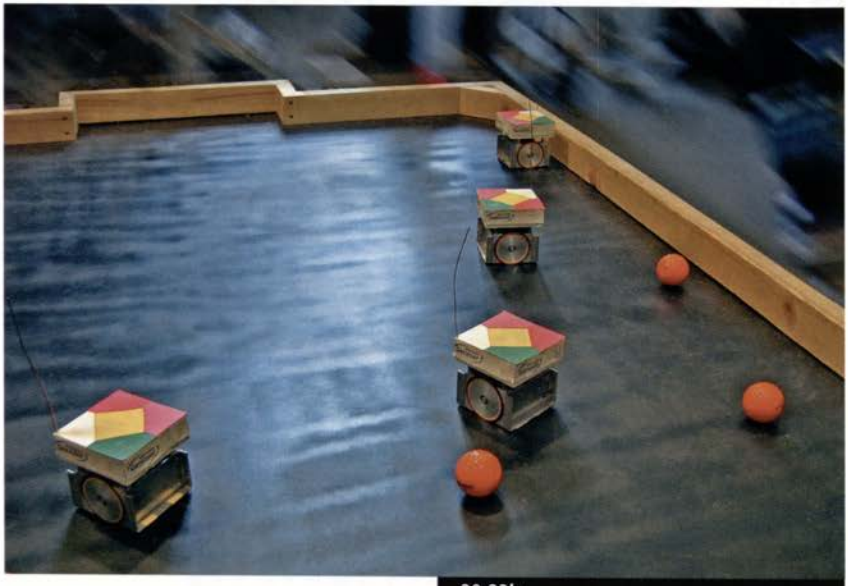
20.13


MAST - Manufacturing Agent Simulation Tool (C:\JavaSource\mast\examples\CDAC(1).xml)
File Simulation Options
Load
Save
Clear
Start
Fit
80%
Rockwell Automation
WorkCell
Assembly
DockSt
Crossing
Conveyor
Sensor
AGV
Segment
Blocker
AutoWP
Labels
Failure
Repair
Remove
D1
D2
D3
Robot 1
Robot 2
3
5
4
Dokovací stanice v hlavní smyčce s čekajícím vozíkem
Výhybka
Pojízdný jeřábový manipulátor
2
Box
1
Policový skladový systém
Ready.
Used memory 5,049kB (click to refresh)
(x=271, y=0)

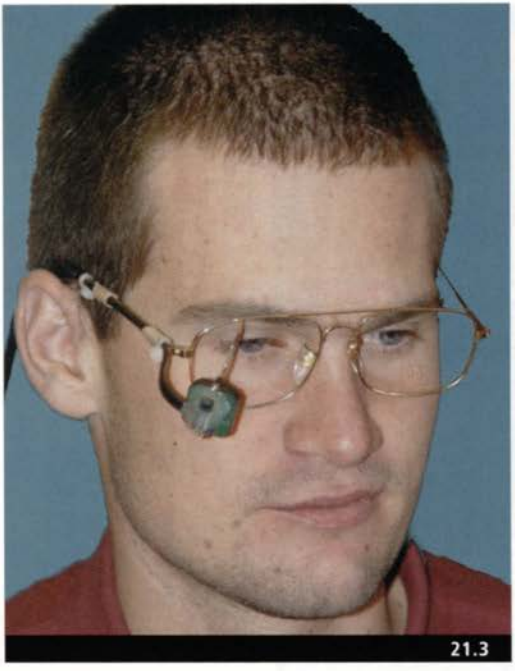
20.22


NAIMT 3D Visio
Robot1 - Coordinator
Robot2 - Coordinator
Robot3 - Coordinator
Robot4 - Coordinator
HIGH
LOW
B1 2 3 4 5
Status: Transmitting ...
Base
Robot2
Transmitting
Robot3
Transmitting
Mine1
Mine3
Mine7
Mine9
Mine5
Mine2
- toggle info windows
- toggle communication range
- toggle video range

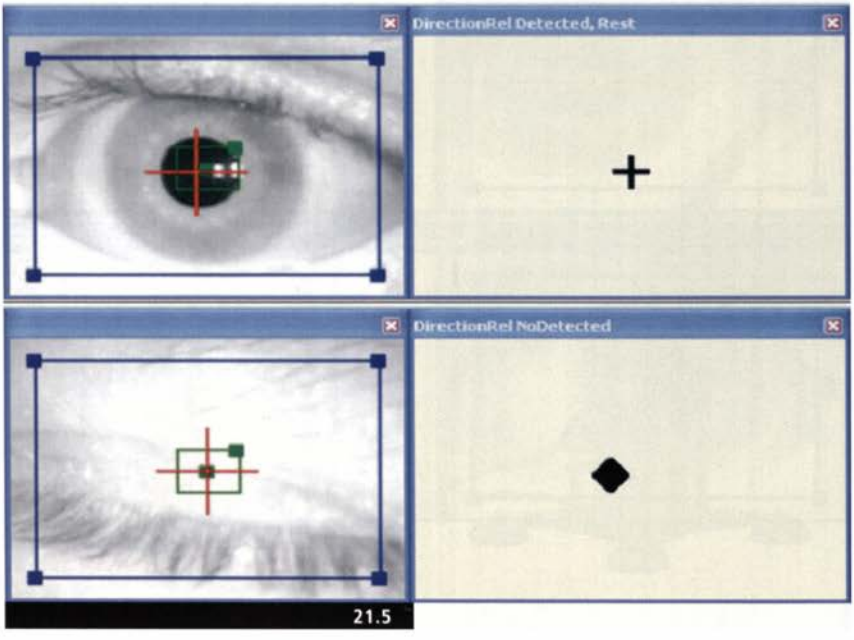
20.23a



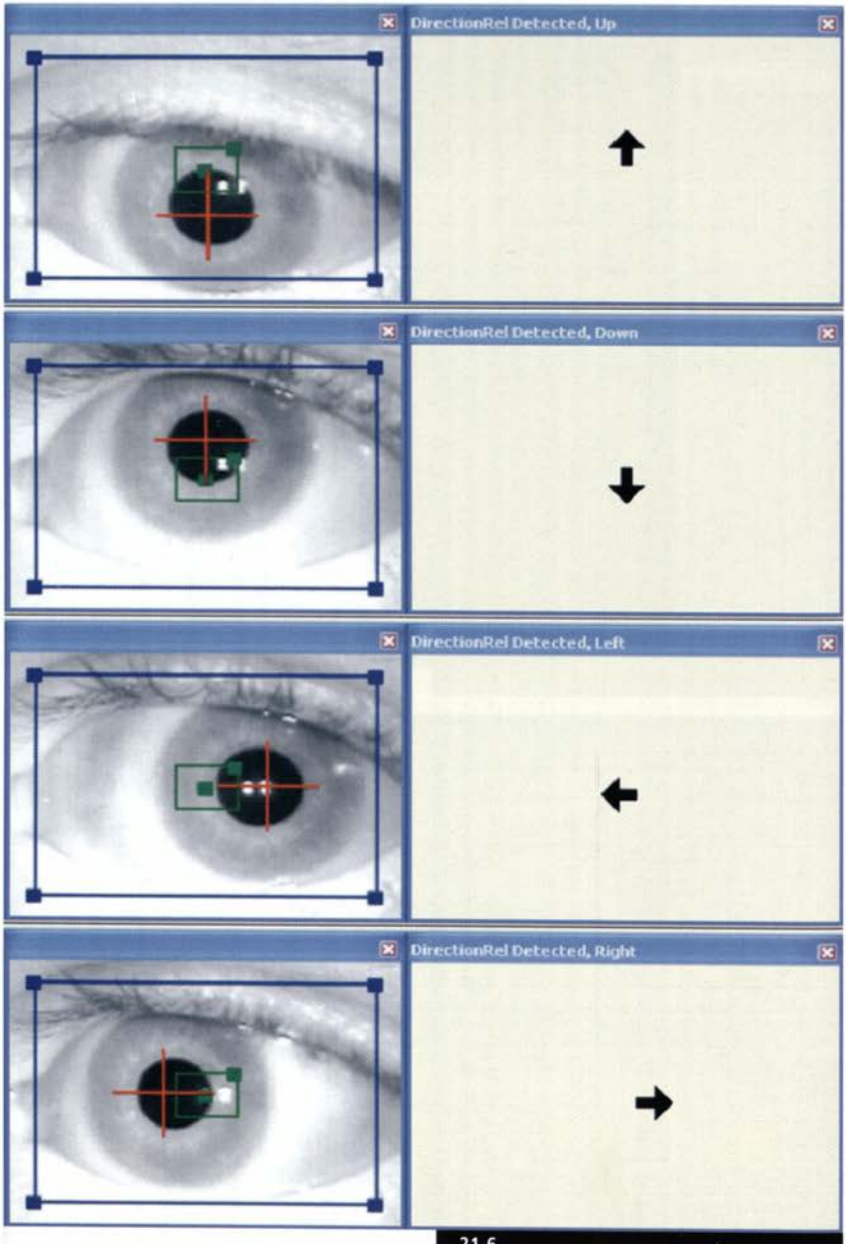
20.23b



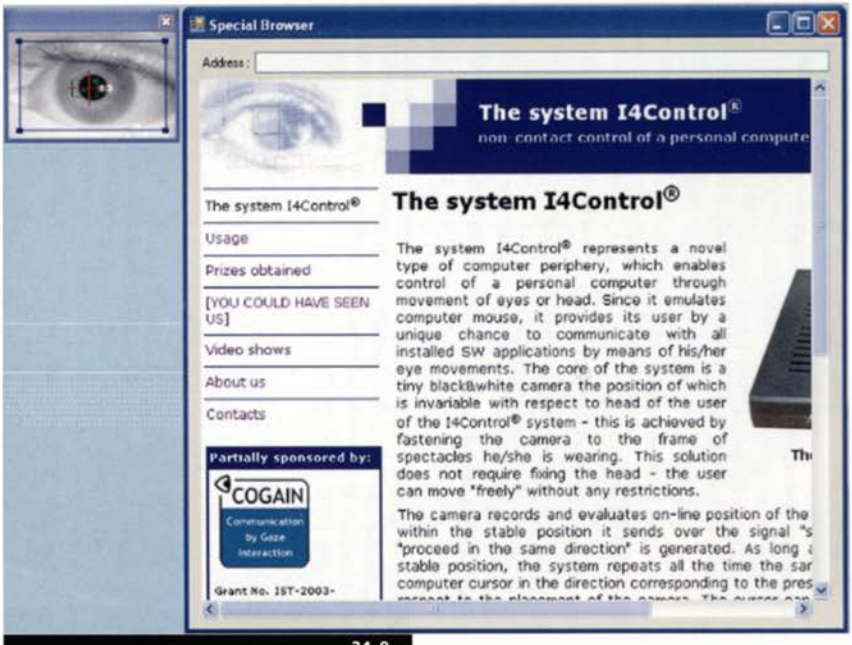
21.3




21.5


DirectionRel Detected, Up
DirectionRel Detected, Down
DirectionRel Detected, Left
DirectionRel Detected, Right

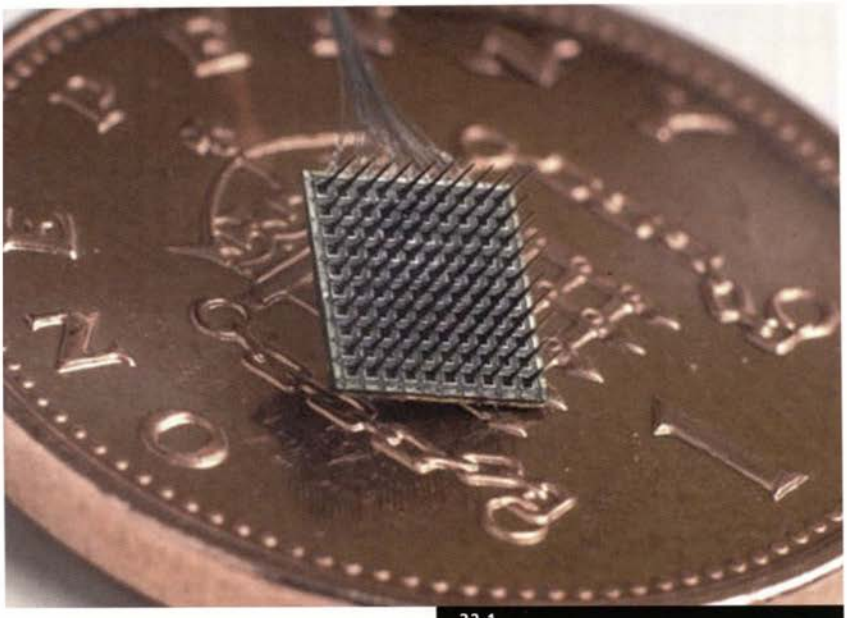
21.6



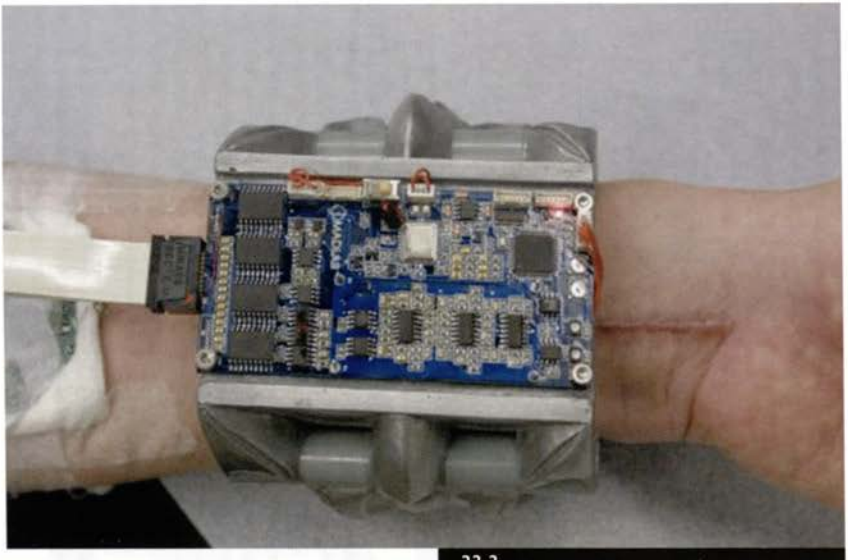

21.8

21.10

21.11



22.1

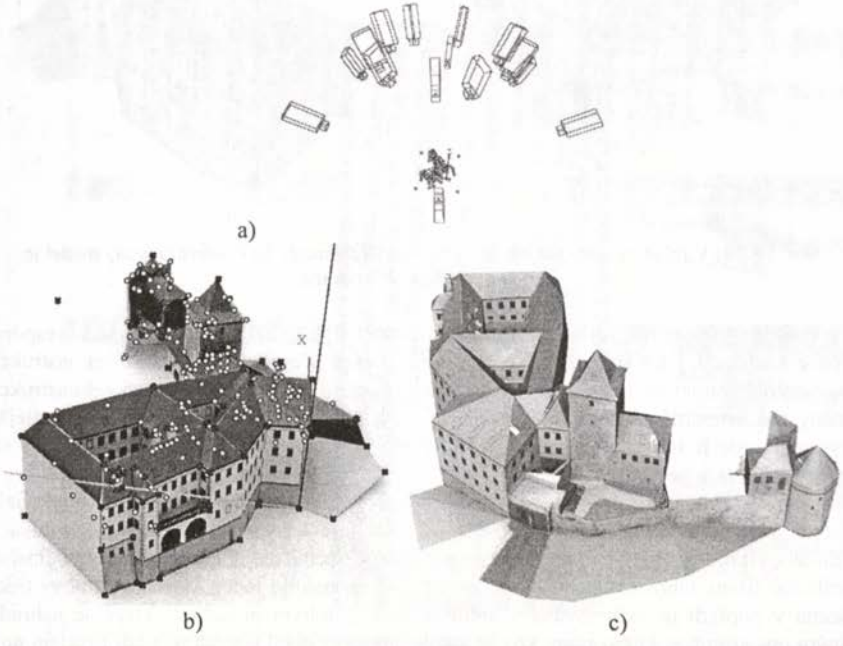


22.2

zeny ve velikosti 1100 × 850 pixelů tak, aby byl pokud možno každý bod modelu vidět alespoň ve dvou snímcích.

Na obrázku 16.5 (též barev. příl.) vidíme stejné čtyři obrazy v interaktivním prostředí ImageModeleru, které umožňuje ručně korespondence zadat. ImageModeler je zde ukázán již ve stavu, kdy byla provedena kalibrace kamer, a to autokalibrační metodou, která je zobecněním zmíněné autokalibrační metody pro více než dva snímky. V levém horním okně je křížkem označena špička Černé věže. Paprsek, který křížkem prochází, se promítá do ostatních obrazů jako tzv. epipolární přímka. Polohy epipolárních přímek závisí na relativní poloze kamer, a proto se obraz od obrazu liší, ale všechny epipolární přímky procházejí špičkou Černé věže. Přesně změřené korespondující body musí na epipolárních přímkách ležet. V ImageModeleru není toto omezení používáno a uživatel má volnost umístit body kamkoliv, ale vzdálenost mezi bodem a epipolární přímkou je užitečným indikátorem kvality korespondence. Korespondence je tím lepší, čím blíže jsou obrazové body ke svým epipolárním přímkám.

Obrázek 16.6(a) ukazuje rekonstruované kamery. Na obrázcích 16.6(b) a 16.7(a), též v barev. příl., vidíme jako kolečka projekce přibližně 300 rekonstruovaných bodů a vzdálenosti změřené mezi třemi dvojicemi vybraných bodů. Zde je třeba poznamenat, že pokud chceme měřit absolutní vzdálenosti rekonstruovaných bodů, musíme definovat měřítko rekonstrukce například tak, že zadáme vzdálenost mezi dvěma vybranými body. Měřítko nelze určit automaticky, protože velká scéna pozorovaná z dálky se jeví v obra-

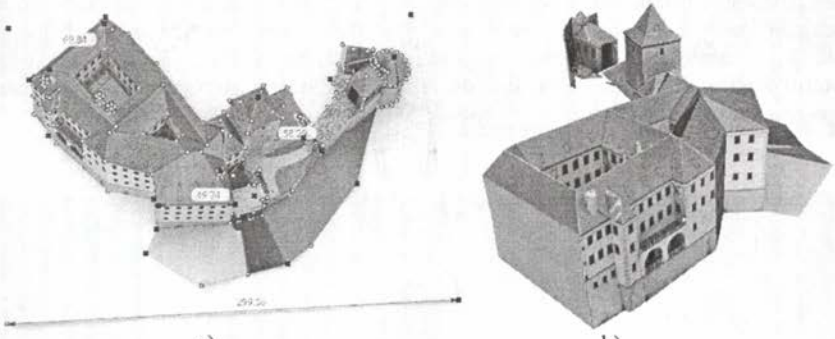



Obr. 16.6 a) Kamery. b) Body. c) Texturovaný model.

zu stejně jako malá scéna pozorována z blízka. Obrázky 16.6(c) a 16.7(b), též barev. příl., ukazují výsledný polygonální model. Polygony byly zadány ručně jako sekvence předtím zrekonstruovaných prostorových bodů. Jejich textura byla získána vždy z toho obrazu, ve kterém má jejich projekce největší plochu. Texturovaný WRML model scény najdete na Internetu [2].

### 16.3.4 Automatická rekonstrukce pohybu kamery v prostoru ze sekvence obrazů

Ruční označování korespondencí je velmi pracné a často i nepřesné, protože přesnost umístění korespondujících bodů je omezena lidskými schopnostmi. Pokud však lze scénu modelovat malým počtem polygonů, má ruční označování velkou výhodu v tom, že člověk ví, kam korespondující body umístit, aby získal právě tu rekonstrukci, kterou potřebuje. Například při rekonstrukci architektury je třeba korespondence obvykle umisťovat do rohů a koutů. To není snadné udělat automaticky, protože to vyžaduje scéně rozumět, poznat, co je kout a co roh a podobně. Toho ještě dnešní systémy umělé inteligence nejsou schopny.



a) b)

Obr. 16.7 a) V rekonstruovaném modelu lze měřit vzdálenosti. b) Rekonstruovaný model je „obarven" texturou z obrazů.

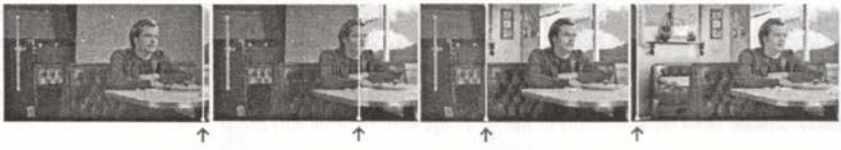
Existuje však velmi zajímavá úloha, v níž ani tak nezáleží na tom, kde se korespondence nacházejí, jako spíše že jich je dostatek a jsou přesně určeny. Jde o rekonstrukci polohy pohybující se kamery z obrazů, které kamera snímá. Zatímco při rekonstrukci scény nás primárně zajímá, jak vypadá scéna, a polohy kamer se získají jako vedlejší výsledek, zde je tomu opačně. Primárním cílem je určit polohu kamery. Které body se zrekonstruovaly, je sekundární.

K čemu je rekonstrukce kamery tak potřebná? Každý, kdo se dívá na film a televizi, chápe, že velmi často nevidí skutečné události, ale triky, které vznikají kombinací reálného filmu a umělých modelů generovaných počítačovou grafikou nebo i jiného reálného filmu. Obrázek 16.8 (též barev. příl.) demonstruje jeden klasický filmový trik. Scéna v popředí je nafilmována v ateliéru před „modrým pozadím", které se nahradí jiným pozadím tzv. klíčováním, kdy se každý „modrý" pixel obrazu nahradí pixelem nového pozadí. Vznikne tak dojem, že se děj odehrává někde jinde, aniž by bylo skutečně

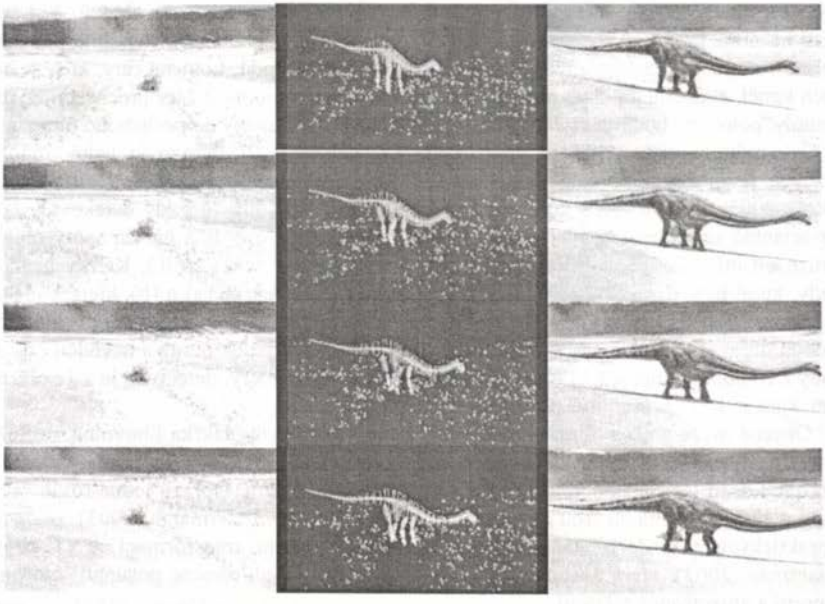
nutné tam „někde jinde" děj natáčet. To významně snižuje náklady a urychluje produkci televizních pořadů. Každý den sledujeme výsledky této techniky, když se díváme na zpravodajství o počasí.

Nejjednodušší trik s modrým pozadím předpokládá, že kamera, do které se klíčuje, se nepohybuje. Statické pozadí se tak promítá do stále stejných pixelů v obrazu. Což, kdyby se kamera pohybovala? Což, kdyby nebylo k dispozici „modré pozadí" ze studia, ale jen běžný film, který bychom chtěli kombinovat s pohybujícím se umělým objektem? To by úloha byla těžší. Umělý objekt by bylo třeba promítat v souladu se změnou pohledu kamery, aby se vyvolal dojem skutečného pohybu. K tomu je třeba vědět, jak se v prostoru pohybuje kamera, jak se pohybuje objekt a jak se má v prostoru objekt se scénou zakrývat.

V úplné obecnosti vede tato úloha opět na rekonstrukci prostorové scény a pohybu kamery v ní. Je zde ale velký prostor pro zjednodušení úlohy.



Obr. 16.8. Obraz scény se vklíčuje namísto modrého pozadí.



Obr. 16.9. Pohyb kamery a terén se zrekonstruují z korespondencí a virtuální pohyblivý objekt se umístí do scény tak, jakoby již s ní byl nafilmován (převzato z 2D3 Boujou Showreel).

Levý sloupec obrázku 16.9 (též barev. příl.) ukazuje čtyři obrazy z filmu spolu s automaticky nalezenými korespondencemi. Prostřední sloupec ukazuje prostorovou rekonstrukci bodů a počítačem generovaný skelet dinosaura. Pohyb dinosaura je generován relativně k rekonstruovaným bodům scény, a tak se dinosaurus správně ve scéně pohybuje. Pravý sloupec ukazuje obrazy výsledného filmu. V tomto klasickém příkladu z trikového filmu se nerekonstruuje úplný trojdimenzionální model scény. Není to třeba a bylo by to zbytečně složité a nákladné. Stačí spočítat pohyb kamery a zrekonstruovat přibližný tvar terénu, po kterém bude dinosaurus kráčet. Zákryty se řeší triviálně, protože dinosaurus je vždy v popředí a vždy zakrývá obraz, do kterého se vkládá.

Pro rekonstrukci pohybu kalibrované kamery ze sekvence obrazů filmu metodou popsanou v předešlém výkladu teoreticky stačí, aby v každých třech následujících snímcích existovalo pět korespondencí v obecné poloze, které obrazy spojují. Přitom je teoreticky jedno, kde se ty korespondence nacházejí. Obvykle se však snažíme použít korespondencí více, protože čím více dobrých korespondencí máme, tím přesnější je rekonstrukce, i když neměříme zcela přesně. Čím více dobrých korespondencí máme, tím je také větší šance, že nebudou blízko degenerované polohy a rekonstrukce bude dobře podmíněná. Podobně se snažíme korespondence v sekvenci „sledovat" co nejdéle a spojovat je do dlouhých řetězců, protože čím delší řetězce jsou, tím pevněji drží celá rekonstrukce pohromadě a tím méně se projevuje hromadění chyb.

### 16.3.5 Automatické nalezení korespondencí v obrazech

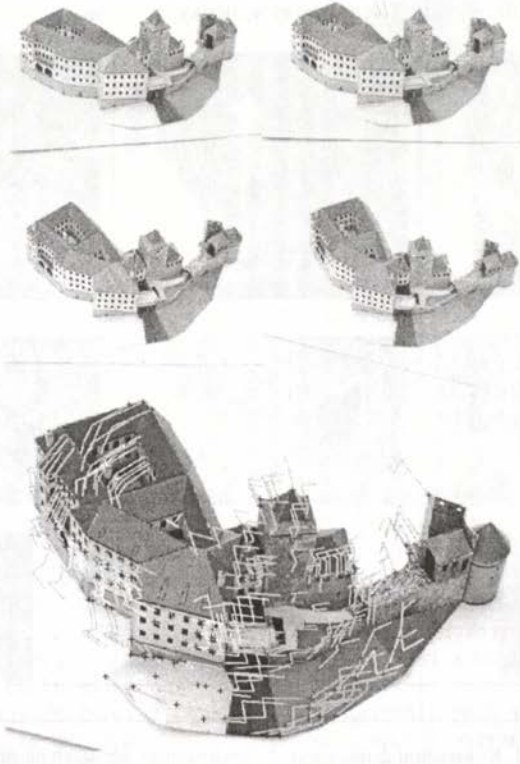
Vysvětleme si, jak lze automaticky nalézt korespondence v sekvenci obrazů. V horní části na obrázku 16.10 (též barev. příl.) je sekvence čtyř obrazů. Dole vidíme poslední obraz, v němž jsou zakresleny projekce rekonstruovaných bodů. Lomené čáry, které v bodech končí, znázorňují pohyb projekcí v sekvenci. Všechny lomené čáry procházejí čtyřmi vrcholy, polohami bodů ve čtyřech obrazech, které jsou zakresleny do posledního obrazu.

Jak byly vybrány projekce bodů, které se sledovaly? V obrazu je velmi mnoho různých jevů, které lze detekovat (Lowe, 2004), (Matas a kol., 2004), (Mikolajczyk a kol., 2005). Pro nalezení korespondencí je ale důležité, aby se body detekovaly tzv. kovariantně se změnou pohledu. Představme si, že se kamera dívá na ten samý objekt z různých míst, jak to ukazuje obrázek 16.11(a, b, c) (též barev. příl.). Křížky ukazují body, které byly detekovány. Vidíme, že na obrázcích v částech (a) a (b), které budeme nadále nazývat *I1* a *I2*, jsou všechny rohy oken označeny. Detektor $d$ nalezl rohy oken v obou obrazech. I když měl detektor k dispozici vždy jen jeden obraz a nevěděl, kde se body z *I1* objeví v obrazu *I2* v důsledku změny pohledu kamery, detekoval je v *I2* přesně tam, kam se z *I1* „přesunuly" po změně pohledu.

Obecně může změna obrazu vyvolaná změnou pohledu být takřka libovolná, protože závisí na tvaru scény a ten není v principu omezen. V praxi se ale ukazuje, že velké množství scén lze lokálně velmi dobře aproximovat rovinami. Obrazy jedné roviny sejmuté z různých pohledů jsou vázány homografií (Hartley, Zissermanm, 2003), přičemž v praktických případech stačí homografii nahradit afinní transformací $A$ (Hartley, Zisserman, 2003), která zachovává rovnoběžnost, ale dovolí obecné posunutí, otočení, zkosení a anizotropní zvětšení.

Od detektoru požadujeme stejný výsledek, ať body $x = d(I1)$ nejprve detekujeme v *I1* a pak je transformací $A$ přeneseme do $y = A(x)$ v *I2*, anebo nejprve transformujeme obraz

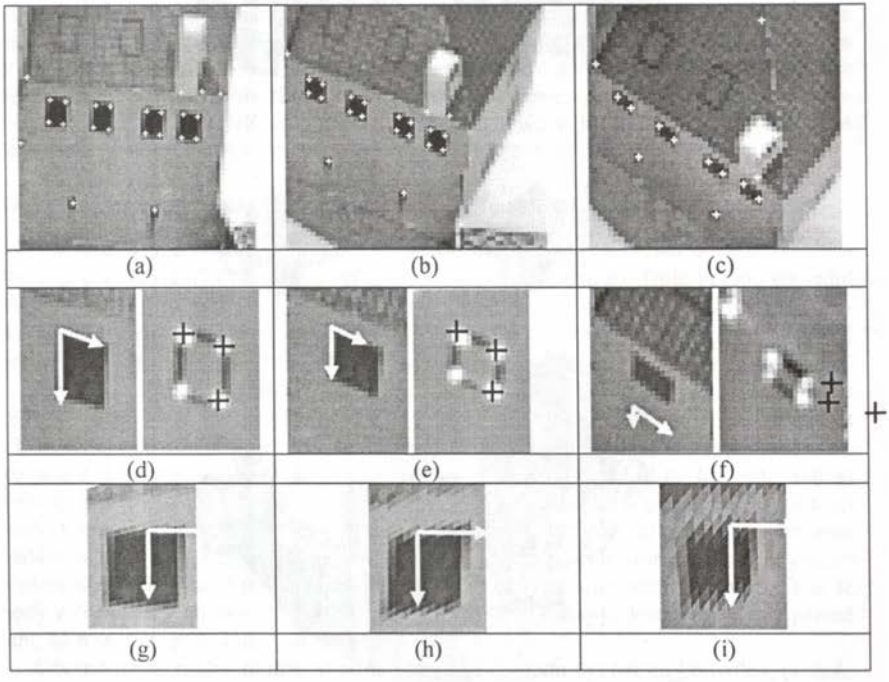
*I1* transformací $A$ do obrazu $I2 = A(I2)$ a tam pak detekujeme body $y = d(I2)$. Dobrý detektor musí komutovat s lokální transformací obrazu indukovanou změnou pohledu kamery, neboli musí platit $A(d(I1)) = d(A(I1)) = d(I2)$. V praxi málokdy dostaneme přesně kovariantní detektor, ale snažíme se mu co nejvíce přiblížit. Jen tak totiž máme šanci detekovat projekce téhož prostorového jevu do různých obrazů, které vypadají různě. Funkčnost každého detektoru je ovlivněna mnoha dalšími faktory. Například v obrazu 16.10 (c), též barev. příl., se v oknech detekovaly jen dva rohy, protože obraz je již příliš zdeformován a předpoklad o afinní transformaci neplatí. Okno je příliš malé vzhledem k velikosti pixelů.



Obr. 16.10. Sekvence čtyř obrazů a pohyb korespondencí.

Kovariantních, nebo takřka kovariantních, detektorů byla navržena celá řada (Mikolajczyk, a kol. 2005). Klasickým příkladem je známý Harrisův operátor (Harris, Stephen, 1988). Obrázky 16.11 (d, e, f), též barev. příl., zachycující tři různé pohledy na totéž okno spolu s odezvou Harrisova operátoru. Operátor reaguje na tvar jasové funkce. Silnou odezvu poskytuje v bodě $x$, kde se obraz hodně mění, a to v tom smyslu, že jasová

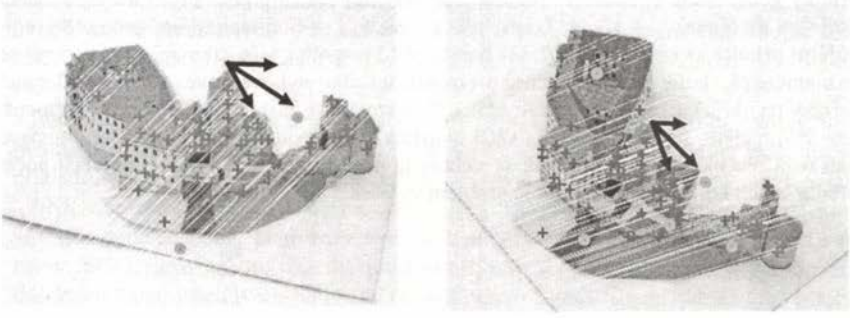
funkce v malém okolí se středem v $x$ se hodně liší od jasových funkcí v okolí bodů okolo $x$. Jasová funkce v okolí rohu okna je velmi odlišná od jasových funkcí v okolních okolí. Odezva v rohu je tedy velká a kladná, což je na obrázcích 16.11(d, e, f), též barev. příl., znázorněno bíle. Oproti tomu jasová funkce v okolí bodu, který leží uprostřed hrany okna, je podobná jasové funkci v okolí sousedního bodu na té hraně. Proto je tam odezva sice velká, ale záporná (černá). Uvnitř černého okna, kde je jasová funkce konstantní, je odezva nulová (šedá), protože jasové funkce ve všech blízkých okolí jsou téměř identické. Vidíme, že Harrisův detektor se pro obrazy 16.11 (a, b), též barev. příl., chová kovariantně. Pro obrazy 16.11(a, c), též barev. příl., však už stoprocentně kovariantní není, protože ze čtyř rohů oken detekoval v obrazu 16.11(c), též barev. příl., jen rohy dva. Pro nalezení korespondencí mezi obrazy ve videosekvencích, kde se obrazy jdoucí po sobě příliš neliší, však dává velmi dobré výsledky.




Obr. 16.11. Kovariantní detekce jevů ve třech různých obrazech na stejné scény.

Kovariantní detektor poskytne jevy, pro které lze smysluplně hledat korespondence na základě podobnosti příznaků, kterými se detekované jevy popíší. V ideální situaci, kdy lze na detekovaném jevu kovariantně zkonstruovat lokální souřadnou soustavu, lze okolí jevu transformovat do nominální souřadné soustavy (Tuytelaars, van Gool, 1999), (Matas a kol., 2002), a pak přímo porovnávat transformované jasové funkce. Jasové funkce skutečně korespondujících bodů mají být stejné. Obrázky 16.11 (d, e, f), též

barev. příl., ilustrují konstrukci lokálních souřadných soustav. Křížky označují lokální maxima odezvy Harrisova operátoru a vektory ukazují souřadné soustavy, které z nich lze zkonstruovat. Na obrázcích 16.11 (g, h, i), též barev. příl., jsou obrazy transformovány tak, aby se lokální souřadné soustavy dostaly do nominální polohy. Vidíme, že po této transformaci se obrazy shodují až na nepřesnosti měření a rozdíly ve vzorkování. Ne vždy je možné nebo praktické konstruovat celé souřadné soustavy. Pak se místo transformovaných jasových funkcí používají jiné příznaky, které jsou stejné pro jevy korespondující a pokud možno odlišné pro jevy nekorespondující (Lowe, 2004), (Matas a kol., 2004).



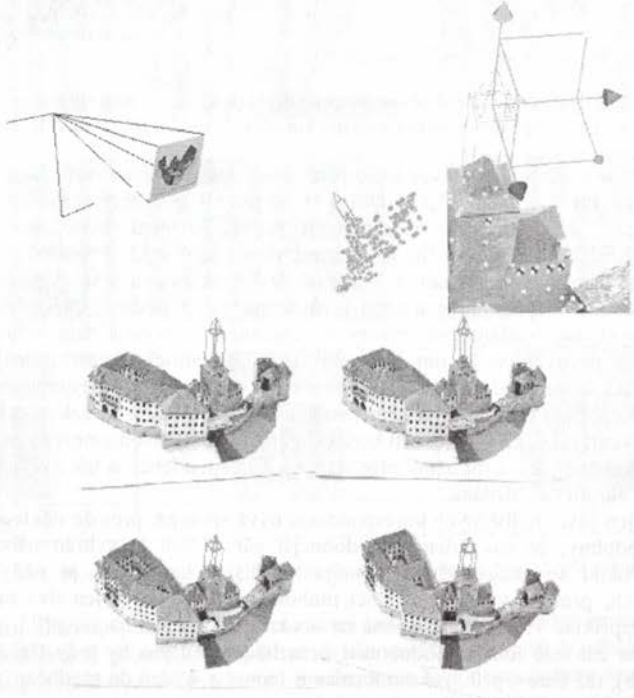
Obr. 16.12. Náhodně zvolená pětice korespondencí (kolečka), z nich spočítaná epipolární geometrie (epipolární přímky) a ostatní korespondence, které jí vyhovují (křížky).

Předběžné korespondence vzniknou párováním detekovaných jevů, často tak, že k nějakému jevu, který označíme E1 v obrazu I1, se přiřadí jev, který označíme E2 v obrazu I2, aby platilo, že popis E1 je nejpodobnější popisu E2 mezi všemi jevy v I1, a symetricky, aby E2 byl nejpodobnější k E1 mezi všemi jevy v I2. Případně lze místo jevu nejpodobnějšího vytvořit $N$ párů s $N$ nejpodobnějšími jevy, aby se vyzkoušely všechny hodně podobné jevy, protože mnoho jevů si může být podobných, zvláště pokud se v obrazu vyskytují opakující se objekty. S párováním je spojena řada technických triků, zejména jak porovnat velké množství objektů s logaritmicko-lineární složitostí (Lowe, 2004). Často se též využívá dodatečná informace o tom, které korespondence přicházejí v úvahu. Například pro nalezení korespondencí mezi obrazy z obrázku 16.10, též barev. příl., lze využít toho, že vzdálenosti korespondujících bodů jsou omezeny, a tím odstranit některé nemožné páry z množiny předběžných korespondencí, a tak zvětšit spolehlivost nalezení a zkrátit čas hledání.

Často jen část předběžných korespondencí bývá správná, protože některé jevy jsou si natolik podobny, že ten jeden nejpodobnější pár z nich je vybrán náhodně chybou měření. Pokud se zkouší všech $N$ nejpodobnějších kandidátů, je vždy $N-1$ párů nesprávných, protože v korespondenci mohou být vždy právě jen dva body ze dvou obrazů. Například všechna čtyři okna na obrázku 16.11, též barev. příl., jsou skutečně stejná a na základě lokální podobnosti nerozlišitelná. Bylo by tedy třeba pro obrazy 16.11 (a, b), též barev. příl., zahrnout celkem $16 = 4 \times 4$ párů do předběžných korespondencí. Správný je však jen jeden pár.

Pro výpočet polohy kamery potřebujeme vybrat jen správné korespondence, které odpovídají projekcím prostorových bodů do obrazů, To formulujeme jako optimalizační

úlohu, jejíž řešení lze efektivně nalézt randomizovaným algoritmem RANSAC (Hartley, Zisserman, 2003). Obdobně, jako při hledání lícovacích transformací pro konstrukci panoramatického obrazu, i zde hledáme největší podmnožinu předběžných korespondencí, která vyhovuje nějaké epipolární geometrii. Epipolární geometrii kalibrované kamery lze vypočítat zmíněným postupem z pěti korespondencí. Algoritmus RANSAC nejprve zafixuje toleranční pásmo okolo epipolárních přímek, od kterého se odvíjí podmínka pro korespondenci dvou bodů. Obrazový bod *P1* v obrazu *I1* koresponduje obrazovému bodu *P2* v obrazu *I2*, pokud se *P1* nachází v tolerančním pásmu epipolární přímky generované bodem *P2* a zároveň se *P2* nachází v tolerančním pásmu epipolární přímky generované bodem *P1*. Náhodně se vzorkují pětice předběžných korespondencí (kolečka na obrázku 16.12, též barev. příl.), z nich se počítají epipolární geometrie (epipolární přímky na obrázku 16.12, též barev. příl.) a spolu s vypočtenými geometriemi se zaznamenává, kolik předběžných korespondencí jim vyhovělo ve zvolené toleranci (křížky na obrázku 16.12, též barev. příl.). Vzorkování se zastaví, když už je malá (menší než 5%) naděje, že bude nalezena větší množina korespondencí než ta největší doposud nalezená. Po ukončení vzorkování se vybere ta geometrie, které vyhověl největší počet předběžných korespondencí. Tím je problém vyřešen.



Obr. 16.13. Rekonstruovaná scéna (body), trajektorie kamery (lomená čára) a „drátěná" věžička, dodatečně přidaný objekt.

### 16.3.6 Boujou – nástroj pro rekonstrukci pohybu kamery z videa

Rekonstrukce pohybu kamery a současné hledání korespondencí je klasický problém počítačového vidění. Nejprve se jej podařilo vyřešit pro videosekvenci, kdy změna mezi následnými obrazy není příliš veliká, Harrisův operátor se chová kovariantně a počet předběžných korespondencí lze omezit malým pohybem korespondujících bodů. Následně se podařilo vyřešit i složitější situace, kdy obrazy nejsou uspořádány do sekvence a kdy je pohyb velký (Martinec, Pajdla, 2006), (Matas a kol., 2004), (Kostková, Šára, 2003), (Šára, Bajcsy, 1998). Příklady rekonstrukcí, včetně automatické rekonstrukce části Langweilova modelu Prahy, lze nalézt na [2].

Efekty na obrázku 16.9 (též barev. příl.) byly vytvořeny programem Boujou firmy 2D3, viz Internet [3]. Boujou umožňuje sledovat pohyb kamery z automaticky nalezených korespondencí v sekvenci obrazů. Z korespondencí se rekonstruují prostorové body a jimi se definuje souřadná soustava scény a vůči ní pak statický nebo pohyblivý objekt pomocí dalších animačních programů. Ukažme si, jak Boujou funguje na jednoduchém příkladu, ve kterém zrekonstruujeme pohyb kamery a do scény „postavíme" nový objekt.

Na obrázku 16.10 (též barev. příl.) jsou čtyři obrazy sekvence a trajektorie korespondujících bodů, které byly v Boujou automaticky získány. Obrázek 16.13 (též barev. příl.) vlevo nahoře ukazuje prostorové body zrekonstruované z korespondencí, trajektorii kamery se čtyřmi polohami (lomená čára) a nový objekt, „drátěnou" věžičku, která byla do scény dodatečně vložena. Ve třetí poloze je zobrazena kamera a její obraz. Vpravo nahoře vidíme souřadnou soustavu, která byla odvozena od tří ručně zvolených bodů na střeše „skutečné věže" scény. Tato souřadná soustava je pevně spojena se skutečnou věží a vzhledem k ní je nová věžička umístěna do scény. Čtyři obrazy dole ukazují novou sekvenci obrazů, ve které doplněná věžička vypadá, jakoby na skutečné věži stála.

## 16.4 Závěr

V této kapitole jsme nastínili některé geometrické principy, na kterých stojí počítačové vidění, a jejich souvislost s aplikacemi v počítačové grafice a filmu. Více o geometrii, algebře a jejich využití v počítačovém vidění a robotice, stejně tak jako o aplikacích počítačového vidění v průmyslu, medicíně, dopravě a obchodu, lze nalézt na stránce Katedry kybernetiky FEL ČVUT, http://cyber.felk.cvut.cz/.

### Literatura

Artin E.: *Geometric Algebra*. Interscience, New York 1957.

Buchanan T.: *Photogrammetry and Projective Geometry – a Historical Survey*. Integrating Photogrammetric Techniques with Scene Analysis and Machine Vision, svazek 1944, 1993, s. 82–91.

Buriánek J., Pajdla T.: *Graficke uživatelské rozhraní pro vytváření korespondencí ve 3D vidění*. V P. Byron and HUMUSOFT co., ed., Matlab, Simulink and Stateflow, HUMUSOFT, 1998, s. 1–10.

Cox D., Little J., O'Shea D.: *Using Algebraic Geometry*. Springer-Verlag, 2005.

Criminisi A.: *Accurate Visual Metrology from Single and Multiple Uncalibrated Images*. Springer-Verlag Ltd., Londýn 2001.

Faugeras O., Luong Q.-T. a Papadopoulo T.: *The Geometry of Multiple Images: The Laws That Govern the Formation of Multiple Images of a Scene and Some of Their Applications*. MIT Press, 2001.

Harris C., Stephen M.: *A Combined Corner and Edge Detection*. 4th ALVEY Vision Conference, University of Manchaster, Anglie, 1988, s. 147–151.

Hartley R., Zisserman A.: *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2. vydání, 2003.

Hilbert D.: *Grundlagen der Geometrie*. Teubner, Lipsko 1899.

Hilbert D.: *Mathematical Problems*. Přednáška na Mezinárodním kongresu matematiků v Paříži v roce 1900.

Hlaváč V., Šonka M.: *Počítačové vidění*. Grada, 1992.

Horn B. K. P.: *Robot Vision*. MIT Press, 1986.

Kemp M.: *The Science of Art: Optical Themes in Western Art from Brunelleschi to Seurat*. Yale University Press, 1992.

Kostková J., Šára R.: *Stratified Dense Matching for Stereopsis in Complex Scenes*. BMVC 2003: 14th British Machine Vision Conference, svazek 1, 2003, s. 339–348.

Longuet-Higgins H. C.: *A Computer Algorithm for Reconstruction a Scene from Two Projections*. Nature, 293(10), 1981, s. 133–135.

Lowe D.: *Distinctive Image Features from Scale-Invariant Keypoints*. International Journal of Computer Vision, 60(2), 2004, s. 91–110.

Martinec D., Pajdla T.: *3D Reconstruction by Gluing Pair-Wise Euclidean Recon-structions, or „How to Achieve a Good Reconstruction from Bad Images"*. Third International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT), s. 8, IEEE Computer Society, 2006.

Matas J., Chum O., Urban M., Pajdla T.: *Robust Wide-Baseline Stereo from Maximally Stable Extremal Regions*. Image and Vision Computing, 22(10), 2004, s. 761–767.

Matas J., Obdržálek S., Chum O.: *Local Affine Frames for Wide-Baseline Stereo*. International Conference on Pattern Recognition, svazek 4, IEEE Computer Society, 2002, s. 363–366.

Maybank S. J., Faugeras O. D.: *A Theory of Self Calibration of a Moving Camera*. International Journal of Computer Vision, 8(2): 1992, s. 123–151.

Mičušík B., Pajdla T.: *Structure from Motion with Wide Circular Field of View Cameras*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 7, 2006, s. 1135–1149.

Mikolajczyk K., Tuytelaars T., Schmid C., Zisserman A., Matas J., Schaffalitzky F., Kadir T., van Gool L.: *A Comparison of Affine Region Detectors*. International Journal of Computer Vision, 65(7), 2005, s. 43–72.

Nistér D.: *An Efficient Solution to the Five-Point Relative Pose Problem*. IEEE Trans. Pattern Analysis and Machine Intelligence, 26(6), 2004, s. 756–770.

Pajdla T.: *Stereo with Oblique Cameras*. IJCV, 47(1), Kluwer May 2002, s. 161–170.

Semple G., Kneebone G. T.: *Algebraic Projective Geometry*. Oxford Classic Texts in the Physical Sciences. Clarendon Press, 4. vydání, 1998.

Šára R., Bajcsy R.: *Fish-Scales: Representing Fuzzy Manifolds*. Proc. IEEE Conf. ICCV '98, Bombay, Indie, January 1998, s. 811–817.

Šonka M., Hlaváč V., Boyle R. D.: *Image Processing, Analysis and Machine Vision*. PWS, 2. vydání, 1998.

Tuytelaars T. a Gool L. J. V.: *Content-Based Image Retrieval Based on Local Affinely Invariant Regions*. Visual Information and Information Systems, 1999, s. 493–500.

Ullman S.: *The Interpretation of Visual Motion*. MIT Press, 1979.

Wade N. J.: *A Natural History of Vision*. MIT Press, 1998.

[1] http://imagemodeler.realviz.com/.

[2] http://cmp.felk.cvut.cz/~pajdla/ui5.

[3] http://www.2d3.com/.

# 17 Taktilní informace v systémech s umělou inteligencí

## 17.1 Úvod

Začněme otázkou. Proč taktilní informace? Příroda odpovídá za nás. Hmat je jedním z pěti lidských smyslů. Můžete namítnout, že nejvíce informací získáváme zrakem. Ano, to je pravda, ale co slepí lidé? A jsme blízko podstatě věci. Hmat je jedním z nejdůležitějších a nenahraditelných smyslů. Někdo může namítnout, že většina lidí přece vidí. Ano, ale do té doby, než zhasneme. Když se ocitneme v úplné tmě, co pak? Jak se budeme orientovat? Jedině hmatem! Každý z nás má s podobnou situací určitě své zkušenosti. A co teprve robot při uchopení předmětů, které mají různou hmotnost, tuhost i prokluz. Zde je taktilní informace opět nenahraditelná. A co lidé, kteří z nějakých důvodů přišli o ruku? Umělý hmat (a opět taktilní informace) jim může značně zlepšit kvalitu života. Může existovat ještě více důvodů, proč se zabývat taktilní informací, potažmo taktilními senzory a snímači?

## 17.2 Obecné pojmy

Taktilní senzory a snímače umožňují získat specifickou informaci, kterou nelze získat jiným způsobem a která je nutná při interakci mezi různými objekty a předměty. Umožňují ochranu robotu, měkké uchopení předmětu, pohyb při kinematických omezeních, při požadovaném působení silou či momentem v daném směru, při přesném měření prostorových souřadnic apod. Představují širokou paletu senzorů: od mikrospínačů, indikujících dotyk, přes senzory měřící sílu uchopení nebo prokluzování uchopeného předmětu až po složité snímače, zabezpečující náhradu hmatu.

Pro lepší orientaci v problematice uveďme definici a specifikaci taktilních senzorů. *Taktilní senzor* je prvek schopný snímat informaci o dotyku s elementem vnějšího prostředí a převádět ji na elektrický signál. *Taktilním snímačem* je obvykle míněno maticové nebo jiné geometrické uspořádání taktilních senzorů.

Nejvíce se dnes rozvíjejí systémy pro získání a zpracování vizuální informace. Taktilní systémy však mají četné výhody:

- taktilní senzory a snímače jsou malá a lehká zařízení, která mohou být integrována v úchopné hlavici robota,
- možnost změny pracovního prostoru vzhledem k jejich umístění v úchopné hlavici je značná,
- lze využívat paralelní registrace údajů a jejich paralelní rychlý přenos,

- taktilní senzory jsou do značné míry nezávislé na vlivu okolního prostředí v porovnání s vizuálními systémy, kde dopadající stíny mohou vést k závažným chybám,
- snadno lze získat výstupní signál v číslicové formě,
- nízké jsou náklady na realizaci,
- lze užít při řízení robotu v reálném čase.

K nevýhodám taktilních systémů ve srovnání s vizuálními patří:

- omezená rozlišovací schopnost, způsobená mechanickými vlivy konstrukce,
- nutná existence minimální úchopné síly,
- nutnost sledovat opotřebení.

Z hlediska účelu můžeme taktilní snímače rozdělit do čtyř kategorií:

1. Taktilní snímače sloužící k rozpoznání tvaru a rozměru předmětů.

a) Taktilní snímač je vhodně umístěn na aktivní úchopné hlavici robotu. Taktilní informace se pak získává z primárních nebo sekundárních taktilních senzorů taktilního snímače při uchopení (sevření) předmětu úchopnou hlavicí. Primární taktilní senzory jsou senzory bezprostředně vcházející ve fyzický kontakt s objektem. Sekundární senzory nepřicházejí do fyzického kontaktu s uchopeným objektem, ale měří taktilní informaci nepřímo, např. jako úhel mezi dvěma členy úchopné hlavice po uchopení. Tímto získáme informaci o globálních vlastnostech objektu.

b) Dále může být taktilní snímač umístěn na zařízení, které vykonává postupné aktivní ohmatávání předmětu tím, že se posouvá po povrchu rozpoznávaného předmětu. Získává tedy taktilní informaci o lokálních vlastnostech povrchu objektu v relativně malé oblasti dotyku taktilního snímače.

2. Taktilní senzory nebo jednoduché taktilní snímače, umístěné rovněž na aktivní úchopné hlavici, sloužící však jen k identifikaci (verifikaci) správného uchopení objektu. Generují pak jen dotykovou informaci. Lze je použít i k vyhodnocení síly stisku čelistí úchopné hlavice.

3. Taktilní senzory, vhodně umístěné na exponovaných místech ramena robotu, sloužící k bezpečnostním účelům. Například vydávají signál k zastavení pohybu ramena, jestliže toto narazí na neočekávaný objekt.

4. Taktilní senzory a snímače generující informaci, která se využívá k řízení pohybu robotů, či jiných automatických mechanismů a k usnadnění jejich orientace v prostoru.

## 17.3 Trocha anatomie

Lidský hmat je složitým a komplexním smyslem umožňujícím hmatové (taktilní) podněty nejen snímat, ale i rozpoznat jejich intenzitu a průběh. Hmat je realizován kožními mechanoreceptory, které transformují podněty z povrchu těla na elektrický signál. Podnětem bývá zpravidla deformace kůže, nicméně podráždění může vyvolat i ohnutí vlasu či chlupu. Aktivita všech současně podrážděných čidel nám v CNS vyskládá komplexní vjem, díky čemuž dokážeme identifikovat některé vlastnosti objektu, které vyvolaly toto podráždění.

Taktilní receptory lze rozdělit podle několika kritérií, jimiž jsou:

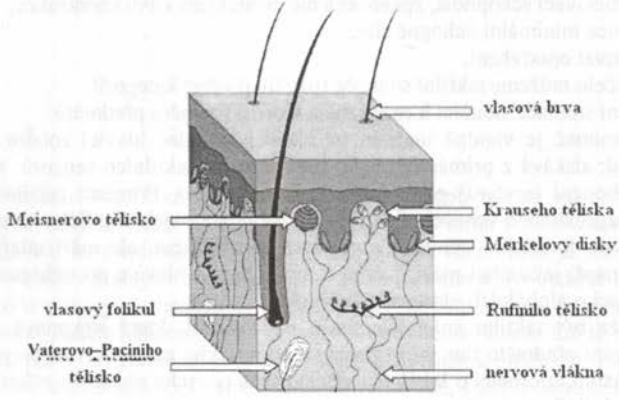
a) morfologická kritéria

- volná zakončení nervových vláken $A_\beta$,
- opouzdřená nemyelizovaná zakončení vláken typu $A_\beta$;

b) rychlost adaptace

- pomalu se adaptující taktilní senzory – produkují signál po celou dobu působení podnětu,
- rychle se adaptující taktilní senzory – reagují pouze na začátek a konec podnětu.

Umístění receptorů v lidské kůži zobrazuje obr. 17.1.




Obr. 17.1 Řez lidskou kůží.

*Merkelovy disky* jsou uloženy v nejvrchnější části kůže (= epidermis). Patří mezi pomalu se adaptující a zpracovávají lehký dotyk, či tlak, působící na kůži, detekují kontury, umístěny jsou na bříškách prstů. V ochlupené kůži se Merkelovy disky sdružují do tzv. *hmatových destiček* o průměru kolem 0,2 až 0,5 mm. Ohnutí vlasu zachycují také Merkelovy disky v součinnosti s rychle se adaptujícími nervovými zakončeními oplétajícími vlasový folikul. Další druhy – *Meisnerova tělíska* – jsou uloženy v coriu (pod epidermis). Patří mezi rychle se adaptující a zpracovávají podnět – začátek vibrací do mezní frekvence 80 Hz, detekují strukturu ohmataného objektu. Nalezneme je v bříškách prstů. *Ruffiniho tělíska* jsou uložena v coriu hlouběji než Meisnerova tělíska, ale na rozdíl od nich patří mezi pomalu se adaptující senzory. Reagují na napínání kůže, zvláště když je způsobeno pohybem prstů či končetiny. Nejznámější jsou *Vaterova–Paciniho tělíska* uložená v tela subcutanea (pod corium). Velmi rychle se adaptují a detekují vibrace, nejlépe v rozmezí frekvencí 100 až 300 Hz. Hustota tělísek a zakončení není na celém těle stejná. Nejvíce jich najdeme na špičce jazyka, bříškách prstů a rtech. Naopak nejméně na zádech či chodidlech.

S hustotou výskytu tělísek úzce souvisí i citlivost příslušného místa. K tomu jsou používána následující kritéria:

- **prahový tlak**, což je minimální hodnota velikosti tlaku mechanického podnětu k vyvolání taktilního počitku. Hodnota prahového tlaku závisí na ploše dotyku. Nejvyšší hodnoty jsou na zádech a chodidlech, naopak nejnižší na čele, rtech a bříškách prstů.
- **prostorový práh**, jenž značí nejmenší vzdálenost mezi dvěma podněty, které jsme schopni od sebe odlišit. Nejnižší hodnoty jsou například na špičce jazyka (1,1 mm), bříšku 3. prstu ruky (2,3 mm), nejvyšší na zádech (7 cm).

- **autotopognosie** představující schopnost přesně lokalizovat místo působení podnětu při zakrytých očích.
- **stereognosie** vyjadřující schopnost identifikovat předmět hmatem při vyloučení ostatních smyslů.

Dále se v kůži vyskytují termoreceptory, které slouží k identifikaci teploty ohmatávaného objektu. Lze je zase rozdělit na:

- **chladové receptory** – pro aktivaci musí být teplota zkoumaného tělesa nižší než teplota tělesná, maximum aktivace nastává při teplotách 10 °C až 30 °C, paradoxně může dojít k podráždění i při teplotě přes 45 °C.
- **tepelné receptory** – pro aktivaci musí být teplota vyšší než teplota tělesná, optimální kolem 40 °C až 45 °C.

Co je důležité, receptory nereagují na teplotu tělesa přímo, ale prostřednictvím změn teploty kůže v místě styku.

Při technické realizaci taktilního snímače se obvykle hledá shoda a podobnost s lidským hmatem. Jeho realizace v plném rozsahu je velmi obtížná a dosud se nepodařila uspokojivě vyřešit. Výsledky našich veškerých snah se více nebo méně (ale spíše méně) blíží funkcionalitě lidského hmatu. Pro roboty se jeho realizace vcelku daří, problémy však vyvstávají při realizaci náhrady lidské ruky, kde příroda má zatím nad naší realizací značně navrch.

## 17.4 Zpracování taktilní informace

Termínem taktilní informace, který je v odborné literatuře v této souvislosti běžně používán, je míněna informace získaná taktilním snímačem. Ze způsobu získávání této informace též plyne i podstatný rozdíl metod, které zpracovávají taktilní otisk nebo vizuální snímek a vytvářejí taktilní nebo vizuální obraz. Signál z kamery obsahuje velké množství redundantních dat, kdežto signál z taktilních snímačů obsahuje podstatně menší množství informace o uchopeném předmětu. Proto u metod zpracovávajících snímek dochází k *redukci* dat, kdežto u metod zpracovávajících otisk jde přesně o opačný problém, kdy se musí data d o p l n i t . Z toho vyplývá podstatný rozdíl mezi oběma skupinami metod.

Rozpoznávání okolních objektů hmatem můžeme rozdělit do dvou skupin:

1. osaháváním předmětu, kdy snímačem postupně osaháváme jeho povrch, nebo se po něm definovaným způsobem pohybujeme,
2. uchopením, kdy předmět poznáme jeho uchopením.

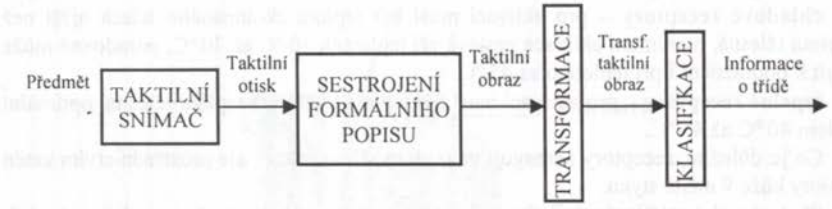
Člověk používá obou způsobů dle okolního prostředí, způsobu orientace a zájmu o objekt. Vzpomeňme na naše chování při pohybu v neznámém tmavém prostředí, při zkoumání povrchu předmětu apod.

### 17.4.1 Průběh zpracování taktilní informace

Proces zpracování taktilní informace můžeme jednoduše znázornit blokovým schématem podle obr. 17.2.

V prvním bloku dochází ke snímání taktilní informace. Výsledkem je taktilní otisk, který je neformálním popisem, a proto činnost prvního bloku nelze formálně popsat.

Obraz předmětu získáme tak, že na taktilním otisku vybereme jistou množinu elementárních vlastností, jejichž způsob popisu apriorně zvolíme. Z těchto elementárních popisů se vhodným uspořádáním získá obraz předmětu. Tento obraz může být podroben určité transformaci, kterou se vytváří vhodná reprezentace předmětu. Informaci o třídě získáme na výstupu klasifikátoru.




Obr. 17.2 Průběh zpracování taktilní informace.

### 17.4.2 Metody tvorby taktilních obrazů

Úkolem těchto metod je nalézt vhodný popis taktilního otisku, takto vytvořit taktilní obraz a v něm specifikovat ty jeho části, které nesou o rozpoznávaném předmětu největší informaci. Těchto metod je velké množství a jejich popis přesahuje rámec této publikace, zájemce je např. najde v (Benhadj-djilali, 2003), (Berger, Khosla, 1991), (Duda, Hart, 1976), (Fischler, Wolf, 1993), (Fu, 1976), (Gaston, Perez, 1984), (Gen-Ichiro a kol., 1975), (Chien, Aggarwal, 1989), (Kanajev, Šnějděr, 1974), (Karason a kol., 1999), (Kinoshita a kol., 1973), (Kolektiv, 1976), (Kučera, 1981), (Lindenbaum, Bruckstein, 1994), (Ljapunov a kol., 1960), (Mařík, 1980), (Mařík, Zdráhal, 1982), (Mařík, 1976), (Mařík, 1978), (Mařík, Zdráhal, 1985), (Mařík a kol., 1993), (Mehdian, Thomas, 1995), (Nicholls, Lee, 1989), (Niemann, 1981), (Pliska, 1986), (Reynaerts, Brussel, 1993), (Schneiter, Sheridan, 1990), (Teh, Chin, 1989), (Tsang, Billings, 1994), (Volf, 1987), (Volf, 1989), (Volf, 1990), (Volf, 1991), (Volf, 1992a), (Volf, 1992b), (Volf, 1993), (Volf, 1994), (Volf, 1996a), (Volf, 1996b), (Volf, 2000), (Volf, 2001), (Yan-Bin, 2001). Zaměřme se alespoň na ty nejdůležitější.

#### *17.4.2.1 Metoda A. A. Charkeviče*

Závěry této metody jsou velmi důležité pro výběr oblasti taktilního otisku, které jsou podstatné pro jeho klasifikaci. Metoda vychází z klasické teorie informace. Množství informace je určováno jako míra neurčitosti dané situace.

A. A. Charkevič (Ljapunov a kol., 1960) se zamýšlel nad problémem, jak se zvětší pravděpodobnost dosažení cíle, v našem případě správného rozpoznání objektu, při různé apriorní informaci a při dalším získávání informace o rozpoznaném objektu během procesu klasifikace. Přínos informace ke konečnému dosažení cíle se nazývá hodnotou informace ve smyslu dále uvedené definice.

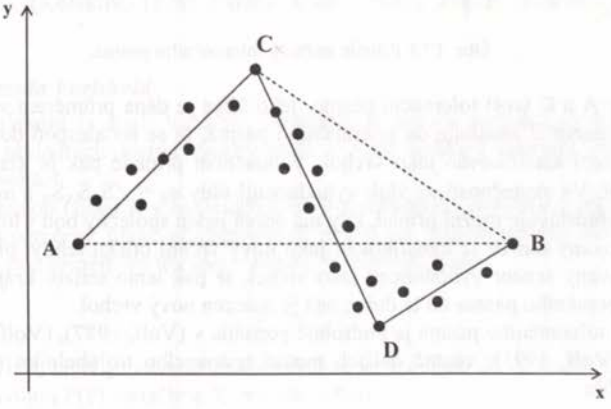
Jestliže před získáním informace byla pravděpodobnost dosažení cíle $p_0$ a po získání informace byla $p_1$, pak hodnotu získané informace lze definovat takto

$$I = \log_2 p_1 - \log_2 p_0 = \log \frac{p_1}{p_0} . \qquad (17.1)$$

Charkevič dospěl k závěru, že body s největší hodnotou informace jsou body na kulové ploše, poté vrcholy a hrany. Nejmenší informaci o objektu nesou rovinné plochy. Tyto výsledky dávají ideový základ všem metodám pro zpracování taktilní informace, které v taktilním otisku vyhledávají hrany a vrcholy, neboť právě ty maximální měrou přispívají k jeho rozpoznání. Rovněž tyto závěry mohou urychlit zpracování taktilní informace a nároky na paměť, neboť stačí uchovávat souřadnice jen těch čidel, které aproximují právě hrany nebo vrcholy.

### *17.4.2.2 Iterativní metoda určení vrcholů*

Předpokládejme, že je dána množina bodů tvořících obrys obrazce. Úkolem metody je nalézt vrcholy a jimi pak proložit úsečky, které obrys otisku aproximují (Duda a Hart, 1976). Princip metody je patrný z obr. 17.3.



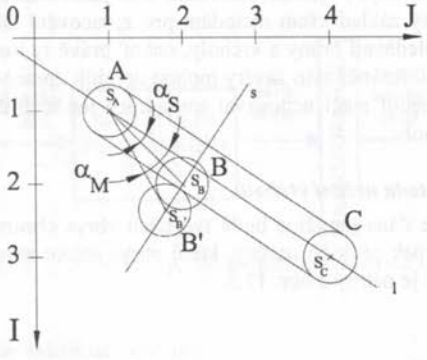

Obr. 17.3 Iterativní výběr vrcholů.

Vybereme jako první dva vrcholy například body, které mají nejmenší a největší souřadnici $x$. Na obr. 17.3 to jsou body $A$ a $B$. Těmito body veďme přímku. Vypočítáme vzdálenosti všech bodů od této přímky. Jestliže jsou všechny vzdálenosti menší než dříve stanovený práh, je proces ukončen. V opačném případě nalezneme nejvíce vzdálený bod od přímky $AB$. Na obr. 17.3 to je bod $C$. Výchozí přímku $AB$ nahradíme novými přímkami $AC$ a $CB$. Tento proces se pak opakuje odděleně pro obě nové přímky.

Nevýhoda metody spočívá v tom, že výsledek je silně ovlivněn jednotlivými body. Jediný vzdálený bod tak může ovlivnit konečný výsledek. To lze odstranit například procesem předběžného vyrovnávání nebo použitím metody na případy, kdy původní otisk neobsahuje poruchy.

### *17.4.2.3 Metoda tolerančního pásma*

Tato metoda uvažuje taktilní senzor jako senzor s konečnou nenulovou plochou. Princip metody je ukázán na obr. 17.4.




Obr. 17.4 Princip metody tolerančního pásma.

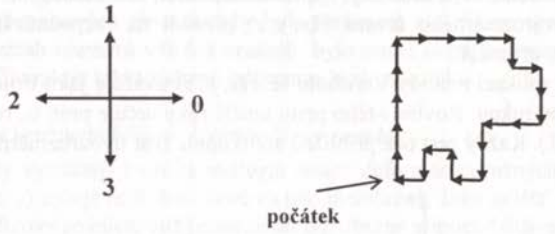
Senzory A a C tvoří toleranční pásmo, jeho šířka je dána průměrem senzorů. Testujeme, zda senzor B zasahuje do tolerančního pásma, či se ho alespoň dotýká. V těchto případech není klasifikován jako vrchol. V opačném případě pak je klasifikován jako nový vrchol. Ve skutečnosti se však vyhodnocují úhly $\alpha_S = \angle S_C S_A S_B$ a $\alpha_M = \angle S_C S_A S_{B'}$. Senzor B′ představuje mezní případ, kdy má právě jeden společný bod s tolerančním pásmem. Testovaný senzor je klasifikován jako nový vrchol otisku tehdy, platí-li $\alpha_S > \alpha_M$. Byl-li testovaný senzor vyhodnocen jako vrchol, je pak tento senzor krajním senzorem nového tolerančního pásma do té doby, než je nalezen nový vrchol.

Metoda tolerančního pásma je podrobně popsána v (Volf, 1987), (Volf, 1989), (Volf, 1992a) a (Volf, 1993), včetně dalších metod testovacího trojúhelníku a metody proměnného úhlu.

### *17.4.2.4 Metody řetězových kódů*

K reprezentaci hranic lze s výhodou použít řetězových kódů. Řetězové kódy se skládají z úseček jednotkové délky, které mají pevnou orientaci. Hranice je definována počátečním bodem a ostatní body hranice jsou popsány přírůstky. Hranice je potom popsána řetězem čísel, např. 1111100003033223321232, viz obr. 17.5.

Řetězové kódy mohou být nezávislé na umístění. V takovém případě ignorujeme počáteční bod a provedeme tzv. derivaci (ve skutečnosti se jedná o výpočet diferencí sousedících hodnot) řetězového kódu. Ta je výhodná proto, že je nezávislá na otočení. Derivace (tedy 1. diference kódu mod 4 nebo mod 8) je opět posloupnost čísel, která ukazuje relativní směr části řetězového kódu, tedy počet nutných otočení vlevo o 90° nebo 45°, která jsou potřebná k dosažení směru další části kódu. Nejčastěji používaným kódem je osmisměrný Freemannův kód, který byl použit pro klasifikaci oblých a hranatých taktilních otisků.




Obr. 17.5 Řetězový kód.

### 17.4.3 Metody klasifikace taktilních otisků

Tyto metody můžeme rozdělit do tří základních skupin: příznakových, strukturálních a hybridních. Obecné teorie jsou popsány v (Duda, Hart, 1976), (Fu, 1976), (Kinoshita a kol., 1973), (Kolektiv, 1976), (Mařík a kol., 1993). Zde si uveďme pár vybraných metod.

#### *17.4.3.1 Metoda Varishold*

Metoda Varishold (z angl. Variable Thresholt.D) nebo-li metoda „rovinných řezů" je jednou ze základních metod používaných v řadě aplikací snímačů, nebo se z její myšlenky vychází.

Mějme danou oblast $H$ v eukleidovském trojrozměrném prostoru a nechť $V(x, y, z)$ je obecně trojrozměrný útvar představovaný například trojrozměrným rozložením tlaků vyvolaných trojrozměrným tělesem v oblasti $H$, přičemž platí

$$V \subset H \subset E^3. \tag{17.2}$$

Každou rovinu, která je rovnoběžná s rovinou $xy$ v oblasti $H$, nazveme prahem $\theta$. Řez $V(x, y, z)$ rovinou $P(\theta)$ označíme $X_i$ (viz obr. 17.6)

$$X_i = V(x, y, z) \cap P(\theta_i),\ i = 1, \ldots, n, \tag{17.3}$$

kde $\theta_i$ volíme takové, aby řez $X$ nebyl prázdnou množinou.

$\tilde{P} = P(\theta_i = 1)$ nazýváme základní rovinou pro $V(x, y, z)$. Předpokládáme-li, že

$$\tilde{P}(\theta_i) = \theta_i,\ \Delta\theta_i = \theta_i - \theta_{i+1},$$

pak $V(x, y, z)$ představuje

$$V(x, y, z) = V_1 \oplus V_2 \oplus \ldots \oplus V_n, \tag{17.4}$$

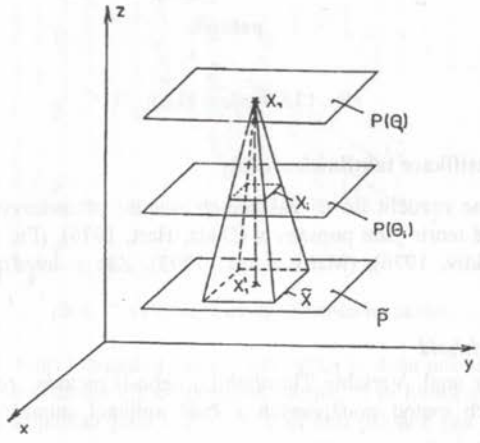
kde

$$V_i = X_i \otimes \Delta\theta_i \tag{17.5}$$

a kde $\oplus$ a $\otimes$ jsou symboly pro přímý součet a součin.

$V_i$ je vlastně hranicí $X_i$ a může být například řezem $X_i$ ohodnoceným váhou. Takto se rozpoznávání trojrozměrného útvaru $V(x, y, z)$ převádí na rozpoznávání posloupnosti dvourozměrných útvarů $X_i$.

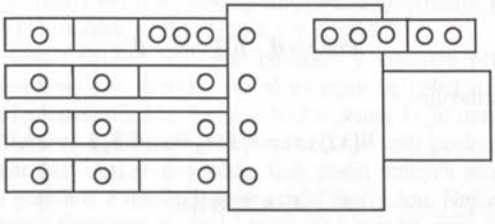
Při globální aplikaci metody Varishold se $V(x, y, z)$ uvažuje jako trojrozměrné těleso uchopené umělou rukou. Rovina $i$-tého prstu umělé ruky určuje práh $\theta_i$, rovinu pak označujeme jako $P(\theta_i)$. Každý prst pak přibližně aproximuje tvar dvourozměrného útvaru $X_i$.




Obr. 17.6 Metoda rovinných řezů.

Při lokální aplikaci metody rovinných řezů pracujeme s rozložením tlaků. Dvě dimenze trojrozměrného útvaru $V(x, y, z)$ jsou určeny rozmístěním taktilních čidel, třetí dimenze je určena tlakem. Ten je reprezentován údajem z proporcionálního čidla, tj. velikost výstupního signálu je úměrná působícímu tlaku, nebo vzniká sepnutím diskrétního čidla, které má konstantní hodnotu prahu $\theta_i$.

Globální použití metody Varishold představila skupina Japonců pod vedením prof. Kinoshity (Duda, Hart, 1976), (Kinoshita a kol., 1973). Vytvořili taktilní snímač z 22 dvoupolohovými mikrospínači, které byly vhodně rozmístěny na povrchu umělé pětiprsté ruky, viz obr. 17.7.



Obr. 17.7 Umístění mikrospínačů na povrchu umělé ruky.

Údaje z mikrospínačů se dále zpracovávaly adaptivním klasifikátorem tvořeným jednoduchou sítí prahových obvodů. Aby byla správnost klasifikace větší než 50 % již pro omezený rozsah rozměrů válců a hranolů, bylo nutné uchopit předmět 6 předepsanými způsoby. Charakter takto získané informace je stochastický.

### *17.4.3.2 Metody využívající řešení Helmholtzovy rovnice*

Uvedené metody vycházejí z určité analogie řešení deformace pružných membrán, kde jistá funkce $\Phi(x, y)$ určuje rozložení tlaků na této membráně. Jako zvlášť výhodné se jeví použití Helmholtzovy rovnice, což je speciální typ obecné eliptické diferenciální rovnice. Funkce $\Phi$ představuje tlakovou sílu působící na jednotlivé senzory snímače. Bude-li velikost působící tlakové síly (hodnota funkce $\Phi$) větší než jistá hodnota nastaveného prahu, dostáváme otisky ohodnocené velikostí této síly. Problém vytvoření taktilního obrazu se tak převádí na řešení Helmholtzovy rovnice. Úloha se řeší za těchto předpokladů: na hranici C otisku se položí hodnota funkce $\Phi = 0$ a ve vnitřní oblasti R otisku je $\Phi \neq 0$.

Pro funkci $\Phi$, za uvedených okrajových podmínek, pak vede úloha na řešení Helmhotzovy rovnice ve tvaru

$$\Delta\Phi + \lambda\Phi = 0 \text{ v R}, \tag{17.6}$$

$$\Phi = 0 \text{ na C},$$

kde je $\lambda$ vlastní číslo spojitého otisku a $\Phi$ představuje tlakovou funkci působící na rovinu snímače, přičemž $\Phi \neq 0$.

Vzhledem k diskretizaci otisku snímačem se rovnice (17.6) řeší v diskrétním tvaru

$$\left[-\frac{1}{h^2}A + \lambda_h E\right]\boldsymbol{\Phi} = 0, \tag{17.7}$$

kterou lze zapsat ve tvaru

$$\det(\Lambda E - A) = 0, \tag{17.8}$$

kde $h$ přestavuje vzdálenost středů senzorů, $\lambda_h$ je vlastní číslo diskrétního otisku, $\Lambda$ je vlastní číslo matice $A$, $A$ značí matici otisku typu $N \times N$, $\boldsymbol{\Phi}$ je vektor dimenze $N$ představující hodnotu funkce $\Phi$.

Profesor Aida (Duda, Hart, 1976) řešil tento problém pro dvě třídy otisků (kruhy a čtverce) pro vlastní číslo $\lambda$ spojitého otisku a pro diskrétní otisk, resp. vlastní číslo matice vytvořené pětibodovým Laplaceovým operátorem z celého otisku.

V pracích (Volf, 1990), (Volf, 1991), (Volf, 1994) autor navrhl úplně nové způsoby řešení pro pět tříd otisků (kruhy, čtverce, obdélníky rovnoramenné trojúhelníky, pravoúhlé trojúhelníky).

Na základě řešení Helmholtzovy rovnice a tvorby matice $A$ pěti- i devítibodovým Laplaceovým operátorem byly vytvořeny tři skupiny metod vycházejících z:

1. výpočtu vlastních čísel matice $A$

   a) matice $A$ byla vytvořena pro celý otisk pěti- nebo devítibodovým Laplaceovým operátorem,

b) matice $A$ byla vytvořena z hraničních bodů po odfiltrování vnitřních bodů pomocí pěti- nebo devítibodového Laplaceova operátoru;

2. ze skalární charakteristiky rozložení prvků matice $A$;

3. z geometrických vlastností matice $A$.

Výpočet vlastních čísel je velmi náročný na výpočetní čas. Byl vytvořen příznakový vektor, skládající se například z hodnoty minimálního vlastního čísla $A$, počtu senzorů otisku, popř. hranice otisku apod., a poté provedena klasifikace. Pravděpodobnost správného rozpoznání se pohybovala okolo 85 % pro tři třídy obrazů, pro pět tříd okolo 60 %, u nichž byla matice $A$ tvořena z celého otisku. Lepších výsledků bylo dosaženo pro obrazy, které byly vytvořeny jen z hraničních senzorů otisku. Bylo dosaženo úspěšnosti správné klasifikace až 80 % pro všech 5 tříd, pro 4 třídy 100 %.

V metodě využívající skalární charakteristiky matice $A$ je definován normovaný koeficient rozpoznatelnosti otisku rovnicí

$$\bar{I} = \frac{1}{N}\sum_{i=1}^{N}\left(\sum_{j=1}^{N} a(i,j)\right)^2, \tag{17.9}$$

kde $a(i,j)$ představuje prvek matice $A$, $N$ je počet senzorů otisku.

Pro tuto skupinu bylo dosaženo správné klasifikace okolo 75 % pro všech pět tříd obrazů a až 92 % pro tři třídy.

V metodě využívající geometrických vlastností matice $A$ bylo vypočítáno těžiště dolní trojúhelníkové matice $A$, neboť celá matice $A$ je symetrická. Souřadnice těžiště $T(i_T, j_T)$ byly počítány z následujících vztahů:

$$i_T = \frac{1}{N_0}\sum_{i=1}^{N}\sum_{j=1}^{i-1}\frac{i-1}{N_0-1}a(i,j), \tag{17.10}$$

$$j_T = \frac{1}{N_0}\sum_{i=1}^{N}\sum_{j=1}^{i-1}\frac{j-1}{N_0-1}a(i,j),$$

$$N_0 = \sum_{i=1}^{N}\sum_{j=1}^{i-1} a(i,j),$$

kde $N$ představuje počet senzorů otisku, $N_0$ značí počet prvků dolní trojúhelníkové matice.

Zde bylo dosaženo úspěšnosti okolo 70 % při klasifikaci do všech pěti tříd a až 97 % při klasifikaci do čtyř tříd. Přičemž u obou metod výpočet probíhal v reálném čase.

### *17.4.3.3 Metody využívající diskriminační analýzy*

Tato metoda je typickou příznakovou metodou založenou na diskriminační analýze. Popis této metody zájemce najde v (Volf, 1996a), (Volf, 1996b), (Volf, 2001), (Volf, 2000). Výpočet probíhá přes jednotlivé třídy obrazů a jako diskriminační funkce je zde použita Ivanovičova odchylka. Její matematický popis a definice je popsána v (Ivanovic, 1963) a použití ve stavebnictví v (Volf, 1982).

Metoda byla použita pro $n = 8$ navržených tříd $T_i$: kruhy, pravoúhlé, rovnostranné a rovnoramenné trojúhelníky, pětiúhelníky, šestiúhelníky, čtverce a obdélníky. Bylo

zvoleno rovněž $m = 8$ příznaků $X_i$: pravoúhlost, čtvercovitost, kruhovitost, počet vrcholů, symetrie podle osy $x$ a $y$, nekompatibilita 1 a 2. Pravoúhlost, čtvercovitost a kruhovitost jsou definovány jako poměr ploch taktilního otisku a opsaného obdélníku, čtverce, popř. kruhu. Nekompatibilita je definována jako poměr délky hranice taktilního otisku ku jeho ploše.

Pro vlastní klasifikaci taktilních obrazů nechť $\boldsymbol{T}_i = (\overline{x}_{i1}, \overline{x}_{i2}, \ldots, \overline{x}_{im})$ přestavuje příznakový vektor středních hodnot příznaků pro třídu $T_i$ a $\boldsymbol{NEZN} = (z_1, \ldots, z_j, \ldots, z_m)$ příznakový vektor neznámého obrazu. Vztahem (17.11) je definována diference mezi $j$-tým příznakem $i$-té třídy a $j$-tým příznakem neznámého obrazu

$$\left| d_{ij} \right| = \left| \overline{x}_{ij} - z_j \right|, \; i = 1, \ldots, n, \; j = 1, \ldots, m. \tag{17.11}$$

Při pozorovaných $m$ příznacích existuje mezi třídou $T_i$ a neznámým obrazem $\boldsymbol{NEZN}$ odchylka $D$ jako číselně vyjádřená hodnota množiny diskriminačních efektů pro všech $m$ příznaků. Tyto odchylky musí obecně splňovat několik podmínek. Z mnoha statistických odchylek pro náš případ rozpoznávání nejlépe vyhovuje Ivanovičova odchylka, definovaná vztahem

$$D_i = \sum_{j=1}^{m} \frac{\left| d_{ij} \right|}{\sigma_j} \prod_{k=1}^{j-1} (1 - r_{kj}), \; i = 1, \ldots, n, j = 1, \ldots, m, \tag{17.12}$$

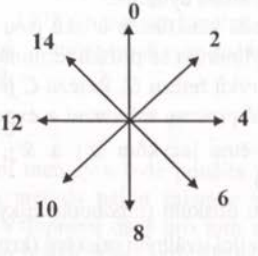
kde $d_{ij}$ značí diferenci definovanou vztahem (17.11), $\sigma_j$ představuje standardní odchylku příznaků $X_j$ a $r_{kj}$ je koeficient korelace mezi příznaky $X_k$ a $X_j$.

Systém byl naučen na simulovaných datech a otestován na 360 000 taktilních otiscích. Správnost klasifikace se pohybovala v rozmezí od 89,2 % do 100 % pro všechna možná pořadí příznaků v příznakovém vektoru. Při vhodném pořadí příznaků správnost klasifikace dosáhla 99,8 % až 100 %.

### *17.4.3.4 Použití lingvistické metody s využitím Freemanova kódu*

Pro popis otisku byl použit Freemanův kód, viz obr. 17.8.

V pracích (Mařík, 1980), (Mařík, Zdráhal, 1982), (Mařík, 1978), (Mařík, Zdráhal, 1985) a (Kučera, 1981) je popsána metoda využívající k vytvoření taktilního obrazu




Obr. 17.8 Freemanův kód.

a vlastní klasifikaci lingvistických strukturálních metod. Tyto metody jsou teoreticky podrobně popsány v (Mařík a kol., 1993).

Předpokládejme, že obrys otisku je po úsecích lineárně aproximován. Jednotlivé vrcholy tohoto obrysu označme $A_1, A_2, ..., A_n$. Obrys otisku popišme nejdříve směrovými vektory $\boldsymbol{a}_i$, $i = 1, ..., n-1$, které spojují dva sousední vrcholy $A_i$, $A_{i+1}$, $i = 1, ..., n-1$. Získáme tak řetěz $A = a_1, a_2, ..., a_{n-1}$ numerických symbolů, které odpovídají směrovým vektorů ve smyslu Freemanova kódu.

Řetěz $A$ však popisuje značné změny zakřivení otisku, které jsou způsobeny jeho diskretizací senzory snímače. K omezení vlivu této diskretizace, představující vlastně šumovou složku při jeho klasifikaci, se provádí digitální filtrace řetězu $A$. Tuto filtraci je vhodné provést pro $n$ sousedních bodů digitálním filtrem s přenosem

$$F_n(z) = \frac{1}{n}\sum_{j=1}^{n} z^{-j} . \tag{17.13}$$

Takto je řetěz $A$ převeden na řetěz $B$, který stále ještě vyjadřuje absolutní zakřivení obrysu.

Pro účely klasifikace jsou však potřebná relativní zakřivení. Proto se řetěz $B$ transformuje na řetěz $C$ vztahy:

$$\begin{aligned} C_{2i-1} &= (b_{i+1} - b_i)_{\mathrm{mod}16} , \\ C_{2i} &= (b_{i+2} - b_i)_{\mathrm{mod}16} . \end{aligned} \tag{17.14}$$

Liché členy řetězu $C$ jsou rovny první diferenci mezi $i$-tým a $(i+1)$-ním prvkem řetězu $B$, sudé členy jsou pak rovny první diferenci mezi $(i+2)$-hým a $i$-tým prvkem.

V řetězu $C$ se vyhledají jednotlivá primitiva a označí se terminálními symboly $d_i \in V_T$. Za primitiva jsou uvažovány cylindry délky 2. Cylindr je zde chápán jako pojem z teorie informace takto:

Nechť $Y$ je konečná množina. Označme $Y_T$ množinou všech oboustranně nekonečných posloupností $(..., y_{-1}, y_0, y_1, ...)$, kde $y_i \in Y$. Nechť $n$ je přirozené číslo, $v_0, v_1, ..., v_{n-1} \in Y$ a $t$ je celé číslo. Potom množinu

$$W(n, t, v_0, ..., v_{n-1}) = \{(..., y_{-1}, y_0, y_1, ...) \in Y^T ;\ y_{t+k} = v_k, k = 0, ..., n-1\} \tag{17.15}$$

nazýváme cylindrem a číslo $n$ délkou cylindru.

V tomto uvažovaném případě klasifikace otisku jsou tedy vybrány cylindry, pro něž $n = 2$ a $t$ je liché. Vybraným cylindrům se přiřadí terminální symboly, dané písmeny $a$ až $r$, podle vzájemného spojení prvků řetězu $C$. Řetězu $C$ je tak přiřazen řetěz terminálních symbolů, nazvaný slovem $D$. V procesu trénování a experimentů bylo zjištěno, že slova trénovacích otisků přísluší dvěma jazykům $\mathscr{L}_1$ a $\mathscr{L}_2$, viz teorie strukturálního rozpoznávání (Mařík a kol., 1993).

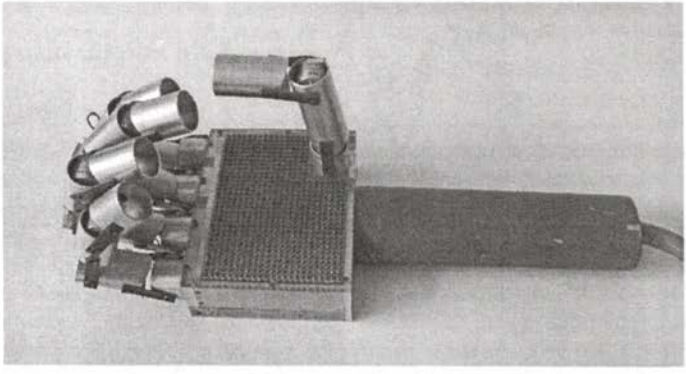
Slova, příslušející hranatým otiskům (mnohoúhelníky – troj- až šestiúhelníky), tvoří jeden jazyk $\mathscr{L}_1$ a slova, příslušející oválným otiskům (kružnice, elipsa), tvoří druhý jazyk $\mathscr{L}_2$. Pro oba jazyky se naleznou odpovídající gramatiky.

Syntaktickou analýzou slova $D$ je otisk klasifikován do jedné ze dvou základních tříd: do třídy $M_H$ hranatých otisků nebo do třídy $M_O$ oblých otisků.

Sémantickou analýzou lze získat detailnější informaci o tvaru, poloze a velikosti klasifikovaného otisku. Jedná se zde o výpočty hlavních numerických parametrů otisků v rámci každé třídy. Výsledkem sémantické analýzy je klasifikace otisku do jedné z celkem deseti tříd, jimiž jsou: obecný trojúhelník, pravoúhlý trojúhelník, rovnoramenný a rovnoramenný pravoúhlý trojúhelník, rovnostranný trojúhelník, čtverec, obdélník, kosočtverec, kosodélník, obecný čtyřúhelník, pravidelný pěti- a šestiúhelník, kružnice a elipsa. Úspěšnost klasifikace se pohybovala mezi 80 až 100 %.

### *17.4.3.5 Lingvistická metoda použitá při klasifikaci uchopeného předmětu antropomorfní hlavicí*

Na ČVUT FEL byl navržen a realizován taktilní snímač tvořený potenciometry v kloubech umělé ruky (Mařík, 1985). Tato ruka je analogií lidské pětiprsté ruky. Údaje z potenciometru se dále zpracovávaly sítí adaptivních prahových obvodů s automatickou volbou struktury. Při rozpoznávání válců, $n$-bokých hranolů, kuželů, koulí a jehlanů při jediném uchopení bylo dosaženo správnosti rozpoznání v rozmezí od 86 % do 100 % pro



Obr. 17.9 Antropomorfní úchopná hlavice.

velmi široký rozsah rozměrů těles. Ruka byla poté doplněna taktilním snímačem se 659 senzory v dlani. V práci (Mařík, 1978) je popsán lingvistický systém, při jehož realizaci se klasifikace zlepšila na 96 %. Zmíněná ruka je znázorněna na dnes již historické fotografii zobrazené na obr. 17.9.

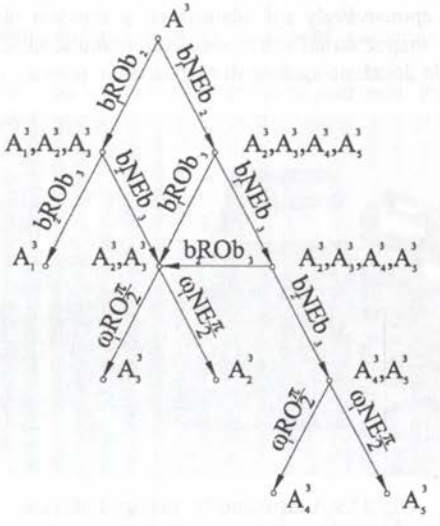
### *17.4.3.6 Metoda VORO*

Tato metoda patří mezi hybridní metody a byla použita ve spojení s metodou tolerančního pásma. Kompletní popis metody najde zájemce v (Volf, 1987), (Volf, 1989), (Volf, 1992a) a (Volf, 1993). Vstupními daty pro tuto metodu jsou příznaky obrazu, získané některou z metod popisu otisku, které provádějí po úsecích lineární aproximaci jeho obrysu. Jako příznaky byly zvoleny: počet vrcholů, délky jednotlivých aproximu-

jících úseček, jejich směrnice a úhly, které tyto úsečky svírají. Z těchto příznaků byl vytvořen příznakový vektor, který charakterizuje každý taktilní obraz. Vlastní metoda VORO je založena na teorii relací a grafů. Jsou zavedeny unární a binární relace mezi jednotlivými příznaky, jimiž jsou ohodnoceny uzly a hrany orientovaného relačního diagramu, který je stromem. Postup po něm je pak řízen právě splněním jednotlivých relací. Celkem bylo zvoleno 8 základních tříd obrazů: bod, hrana, trojúhelníky, čtyřúhelníky, pěti- až sedmiúhelníky a kruhy. Ty byly dále děleny do podtříd, celkem byla klasifikace provedena do 20 tříd obrazů.

Klasifikace do základních tříd se uskutečňuje podle počtu vrcholů $n$ na základě orientovaného relačního diagramu.

Uvnitř základních tříd se klasifikuje na základě relačních vztahů mezi délkami hran $b_j$ a jejich vzájemných úhlů $\omega_i$. Obrázek 17.10 ukazuje například relační diagram pro třídění trojúhelníků.




Obr. 17.10 Třídění ve třídě trojúhelníků.

Třída trojúhelníků $A^3$ je složena z následujících podskupin $A^3_j$:

- $A^3_1$ třída rovnostranných trojúhelníků,
- $A^3_2$ třída rovnoramenných trojúhelníků,
- $A^3_3$ třída rovnoramenných pravoúhlých trojúhelníků,
- $A^3_4$ třída pravoúhlých trojúhelníků,
- $A^3_5$ třída obecných trojúhelníků,

dále $\omega_i$ představuje vnitřní úhly mezi hranami $b_i$ a $b_j$, NE, RO relace nerovno a rovno.

Obdobně probíhá třídění ve čtyřúhelnících. Pěti- až šestiúhelníky jsou pak děleny na pravidelné a nepravidelné. Na testovaných otiscích bylo dosaženo správnosti klasifikace 100 %. Klasifikace proběhla v reálném čase.

#### *17.4.3.7 Aktivní ohmatávání*

Dalším způsobem, užívaným pro taktilní rozpoznávání, bylo aktivní ohmatávání. Tuto konstrukci, použitou prof. Aidou (Kinoshita a kol., 1973), představoval umělý palec se čtyřmi mikrospínači, který ohmatával povrch tělesa. Při ztrátě kontaktu s povrchem, byl aktivován příslušný pohonný podsystém, jehož úkolem je nalézt opět ztracený kontakt s tělesem.

Vzhledem k tomu, že taktilní podsystém sám o sobě není dostatečně efektivní, bývá často užíván ve spojení s kamerou.

Systémy byly postupně zdokonalovány, aby vyhledávaly řízeně příznaky, které nesou největší informaci o osahávaném objektu ve shodě s teorií informace (Ljapunov a kol., 1960) – jednalo se o příznaky: kulová plocha, vrcholy, hrany a rovinná plocha s téměř nulovou informací. Jedním z nich byl systém SATR (Duda, Hart, 1976), (Ljapunov a kol., 1960), který určoval lokální tvar předmětu v místě dotyku. Podle katalogu rozpoznávaných předmětů se taktilní snímač přemístil do míst, kde očekával výskyt zvoleného příznaku. V případě nenalezení příznaku byl zmenšen počet možných předmětů a vygenerována nová hypotéza o jeho tvaru. Správnost rozpoznání se pohybovala v rozmezí 90–100 %.

## 17.5 Využití taktilní informace

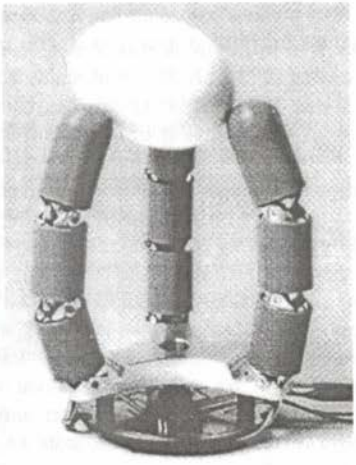
### 17.5.1 Využití v robotice

V průmyslovém použití je nejstarší a nejčastější využití taktilní informace právě v robotice. Zpočátku byly taktilní snímače užívány jen pro indikaci uchopení předmětu v úchopné hlavici, později pro řízení úchopu, měření prokluzu, popř. pro přesné polohování objektů.

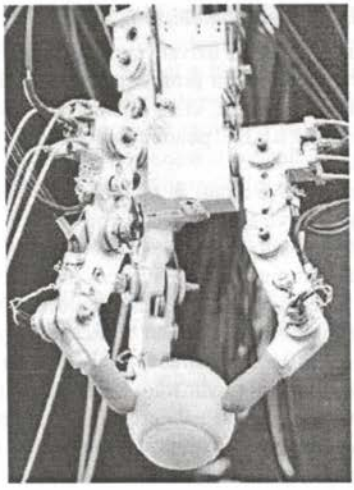
V blízké budoucnosti se očekává výrazné rozšíření servisních robotů, které mají poskytovat všestrannou pomoc a podporu lidem v nejrůznějších oblastech, např. v nebezpečných prostorech (chemicky, radioaktivně zamořených prostředích nebo jinak životně nebezpečných), v krizových a různých havarijních situacích, dále při péči o nemocné a handicapované osoby, při rehabilitacích, popř. jako pomocníci v domácnosti. Roboty musí být vybaveni dostatečně výkonným řídicím systémem s využitím umělé inteligence, aby se mohly pohybovat v přirozeném prostředí, vytvořeném pro člověka. V řadě případů to vyžaduje univerzální antropomorfní úchopnou hlavici. Ty mohou být tříprsté až pětiprsté a vybavené různými typy senzorů.

Tříprstou ruku, vhodnou při průmyslovém nasazení, popisuje Liu (Liu a kol., 1998) a (Liu a kol., 2004). Ruka je vybavena senzory typu DLR a zobrazena na obr. 17.11. Jiná tříprstá ruka je popsána v [9], viz. obr. 17.12. Tato hlavice je osazena dvouvrstvými senzory znázorněnými na obr. 17.13. Všechny jsou vybaveny inteligentní zpětnou vazbou řízení úchopu.

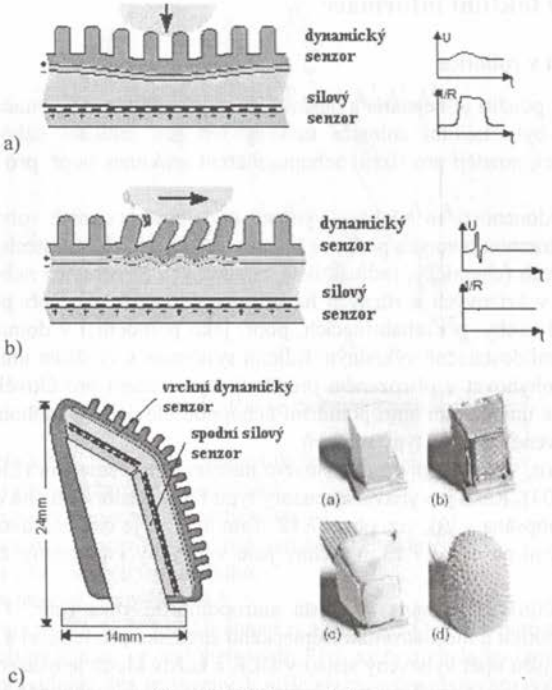
V (KAB, 2006) je popsána čtyřprstá antropomorfní ruka (obr. 17.14) vyvinutá v Ústavu pro robotiku a mechatroniku Německého střediska pro letectví a kosmonautiku. Poslední články jsou opět vybaveny senzory DLR a každý kloub je osazen momentovým senzorem. Do ruky jsou kromě senzorů kompletně integrovány pohonné jednotky, výko-



Obr. 17.11 Tříprstá ruka s DLR senzorem.



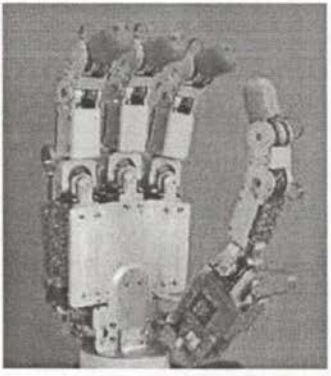
Obr. 17.12 Tříprstá ruka.



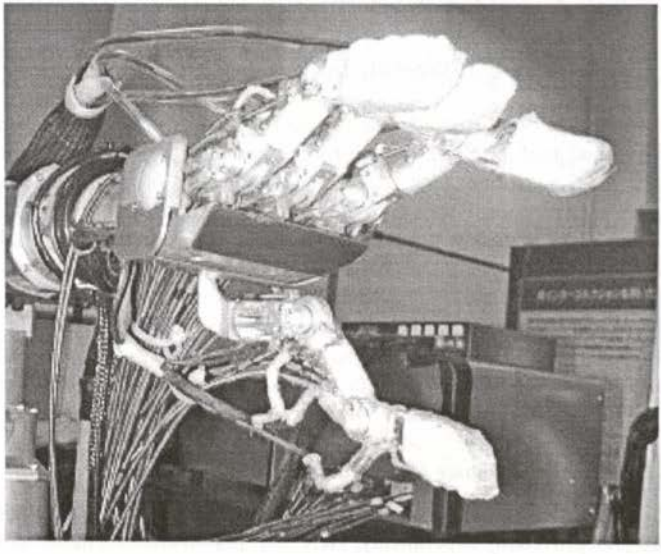

Obr. 17.13 Dvouvrstvý senzor.

nová a komunikační elektronika. Tím je umožněna aktivní ohebnost ruky a operativnost v neznámém prostředí a při navazování kontaktu s lidmi. Při praktických zkouškách se osvědčila jako robustní, stabilní a rychlá inteligentní úchopná hlavice, která dokáže přenést nejen šálek kávy nebo čaje, ale spolehlivě chytit i hozený míček.

Další čtyřprstá antropomorfní hlavice, vyvinutá na Tokijské univerzitě, je znázorněna na obr. 17.15 (též barev. příl.). Tato ruka je také vybavena obdobou dvouvrstvých taktilních senzorů dle obr. 17.13.

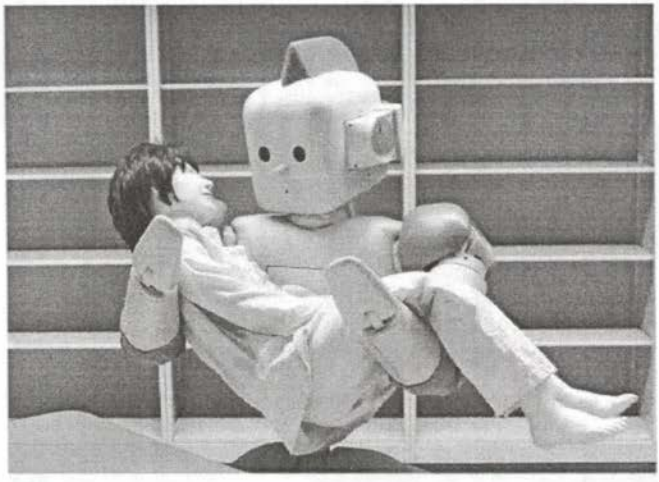


Obr. 17.14 Antropomorfní čtyřprstá ruka se DLR senzory.



Obr. 17.15 Antropomorfní čtyřprstá ruka.

Vrchol užití umělé inteligence v robotice představuje měkký interaktivní lidský robot RI-MAN, vyvinutý v RIKEN, Bio-Mimetic Control Research Centrum v Japonsku. Robot má schopnost uvědomit si lidskou péči a sociální úkoly. RI-MAN se může stát neocenitelným partnerským robotem. Celkový pohled na robota ukazuje obr. 17.16 (též barev. příl.). Robot komunikuje lidským hlasem, dvěma kamerami snímá okolní scénu, v níž se dokáže orientovat v reálném čase, reaguje na povely, jejichž význam si uvědomuje a analyzuje, a jim pak přizpůsobuje svou činnost. Robot je opatřen sluchem a čichem.



Obr. 17.16 Robot RI-MAN s pacientkou v náručí.

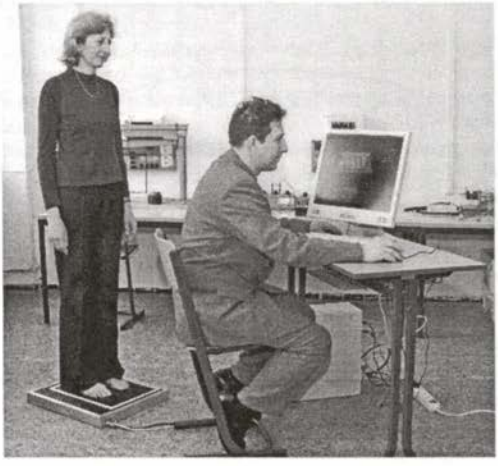
„Uši" má umístěny na obou stranách hlavy a jsou tvarovány tak, aby byly schopny rozpoznat směr, odkud zvuk přichází. Povrch robotu je měkký, aby byl příjemný na dotyk. Je určen pro práci v nemocnicích, sociálních zařízeních, při manipulaci s pacienty a jejich ošetřování a obsluze. Robot je rovněž opatřen taktilními senzory pro biologickou zpětnou vazbu. Jsou umístěny v předloktí, nadloktí a hrudi tak, aby mohl fyzicky komunikovat s lidmi, např. při držení, dotyku apod. Robot RI-MAN má celkem 6 kloubů, z nichž každý je poháněn šesti motory ovládanými v párech. Tím je umožněna kombinace ohybu a kroutivých pohybů. Tento mechanismus je potřebný, aby byla umožněna dostatečná síla ramenu a loketnímu kloubu při zdvihání a držení člověka při relativně malých rozměrech mechanických členů. Robot je opatřen distribuovaným řídicím systémem, např. každý kloub je opatřen vlastním řídicím systémem.

### 17.5.2 Využití v biomechanice a biomedicíně

V biomechanice a biomedicíně je snaha využít taktilní informaci při náhradě lidského hmatu. Proto jsou prováděny experimenty, které mají popsat chování lidského hmatu.

Taktilní informace nalezla také uplatnění ve sportovním lékařství a ortopedii, viz obr. 17.17. Díky ní se dalo zkoumat rozložení tlaků na ploskách chodidel a jejich časový

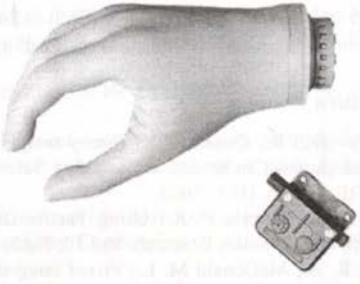
průběh, diagnostika poruch motoriky, ortopedických vad a mnohých onemocnění. Následně byla využita ve fyzioterapii pro rehabilitaci, pro vývoj rehabilitačních pomůcek a protéz. Bližší informace najde čtenář v (Živčák a kol., 1997), (Šimčík a kol., 1996), (Volf a kol., 2001), (Volf a kol., 1997), (Volf a kol., 2003), (Volf a kol., 2006), (Volf a kol., 2002) a (Volf a kol., 2006b).



Obr. 17.17 Měřicí systém Plantograf V05.

Na webovských stránkách firmy Otto Bock je popsána nejnovější jimi vyráběná protéza lidské ruky „SensorHand™Speed", která je řízená EMG signály ve svalech pahýlu. Umožňuje proporcionální řízení uchopovací síly v rozmezí 0–100 N a má proporcionální řízení rychlosti sevření 15–300 mm/s. Tak lze zachytit i rychle letící předměty, např. při sportování. Tato ruka je současně vybavena senzory a funkcí AutoGrasp, která umožňuje automatické zvýšení úchopné síly, jestliže uchopený předmět prokluzuje. Konstrukce ruky je znázorněna na obr. 17.18, její design pak na obr. 17.19.

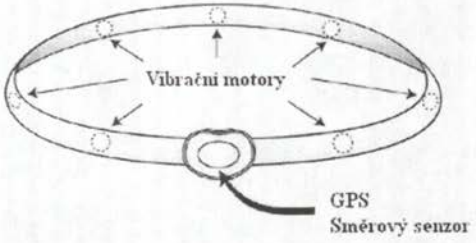
Obr. 17.18 Konstrukce protézy lidské ruky SensorHand™.



Obr. 17.19 Design protézy.

### 17.5.3 Taktilní informace jako dodatečný informační zdroj

Jako dodatečný zdroj informace se taktilní informace může využívat u lidí, u kterých došlo k nějakému postižení. Na obr. 17.19 je zobrazeno schéma vibračního opasku pro slepé. Opasek je spojen s modulem GPS, kde je nastavena uživatelem adresa. Strana, na kterou se má uživatel vydat, je charakterizována vibracemi příslušným vibračním motorem.




Obr. 17.19 Opasek s vibračními motory.

## 17.6 Závěr

V úvodu jsme si položili otázku proč taktilní senzory a snímače. V dalších odstavcích byly popsány jejich různé principy a konstrukce. Řada z nich je vhodných jen pro úchopné hlavice robotů, některé lze požít též jako náhradu lidského hmatu. Tímto směrem se v současnosti upírá největší zájem. Zdaleka však není uspokojivě vyřešen problém v oblasti smykových senzorů, a to jak pro roboty, tak především pro náhradu hmatu. Zde navíc ještě vystupuje problém s připojením na nervový systém člověka.

Velká část je pak věnována užití taktilní informace v různých oblastech lidské činnosti s různým stupněm uplatnění umělé inteligence.

Jak bylo uvedeno, byly zde popsány různé taktilní senzory a užití taktilní informace, vzhledem k omezenému místu a neustálému dynamickému vývoji však nelze mít tento přehled úplný. Velké množství dalších informací nalezne zájemce v citovaných publikacích nebo na uvedených internetových odkazech.

### Literatura

Benhadj-djilali R., Quazzane K: *Binary image coding algorithm using tactile data information.* International Conference on Imaging Science, System and technology (CISST'03), vol. 2, CSREA Press, USA, 2003.

Berger A. D., Khosla P. K.: Using Tactile Data for Real – Time Fedback. The International Journal of Robotics Research. Vol.10, No.2, 1991, s. 88–102

Browse R. A., McDonald M. L.: *Visual integration of tactile information in telerobotic.* Journal Man and Cybernetics, vol. 22, no. 5, 1992.

Duda R., Hart P.: Raspoznavanije obrazov i analyz scen. Mir, Moskva 1976.

Fischler M. A., Wolf H. C.: Locating Perceptually Salient Points on Planar Curves. IEEE Transactions on Robotics and Automation. Vol.5, No.3, 1993. s. 359–368

Forse Sensing Resistors Integration Guide and Evaluation Parts Catalog.: Interlink electronics, Camrillo, 2006, 26 s.

Fu K. S.: Digital Pattern Recognition. Springer-Verlag, Berlín, Heidelberg, New York 1976.

Gaston P. C., Perez T. L.: Tactile Recognition and Localization Using Object Models: The Case of Polyedra on a Plane. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol.6, No. 3, 1984, s. 257–266.

Gen-Ichiro K., Aida S., Mori M.: A pattern classification by dynamic tactile sense information processing. Pattern Recognition, vol.7, no. 4, 1975.

Grajski K. A.: *Tactile information processing: A back-propagation model of pattern recognition in the human somatosensory system*. Neural Networks v 1 n 1 SUPPL 1988.

Greenway R. B. Jr., Faddis T. N.: *Effect of tactile information on telerobotic performance*. The International Society for Optical Engineering , vol. 1833, 1993 s. 317–324.

Greenway R. B. Jr., Faddis T. N.: *Tactile information system for a telerobotic cell*, The International Society for Optical Engineering , vol. 1833, 1993, s. 228–239.

Hasegawa T., Honda K., Kiriki T. a kol.: *Real-time pose estimation of an object grasped by a multi-fingered hand by using sensory fusion of visual and tactile information*. Transaction of the Institute of Electrical Engineers of japan, Part C, vol. 118-C, no. 9. Inst. Electr. Eng. Japonsko, 1998.

Howe R. D.: *Tactile Sensing and Control of Robotic Manipulation*. Harvard, 1994

Hutton R. J. B., Flasch J. M. etc.: *Keeping in touch: kinesthetictactile information and fly-by-wire*. Proceedings of Human Factors and Ergonomics Society. 38th annual meeting, vol. 1. Human Factors & Ergonomic Soc., USA, 1994.

Chalupa V., Mařík V., Volf, J.: *Tactile matrix for Shape Recognition*. In: IX. IMEKO World Congress, Berlín 1982. Preprints vol. V/IV, pp. 303-312.

Chien C. H., Aggarwal J. K.: *Model Construction and Shape Recognition from Ocluding Contours*. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol.II. No.4, 1989. s. 372– 389.

Ivanovic B.: *Diskriminiciona analyza*. Naucna knjiga. Beograd 1963.

Jockush J., Walter J., Ritter H.: *Tactile Sensor System for a Three – Fingered Robot manipulator*. Department of Computer Science. University of Bielefeld, 1997.

KAB.: *Mechanická ruka pro roboty*. Automatizace. roč. 49, No. 10, Praha 2006, s. 639.

Kammermeier P., Buss M., Scmidt G.: *Dynamic display of distributed tactile shape information by a prototypical actuator array*. Proceedings 2000 IEEE/RSJ International Conference on Inteligent Robots and Systems ( Cat. No. 00CH37113 ), vol. 2. IEEE, Piscataway, USA, 2000.

Kanajev E. M., Šnějděr A. J.: *Mechanika mašin*. Nauka, Moskva 1974.

Karason S. P., Srinivasan M. A., Annaswamy A. M.: *Encoding and decoding of static information in tactile sensing*. International Journal of Robotic Research, vol. 18, no.2. Sage Publications, 1999.

Kinoshita, G., Aida S., Mori M.: *Pattern Classification of the Grosped Object by the Artificial Hand*. In: Third IJCAI, Stanford 1973.

Kolektiv: Sborník statej: *Distancionno upravljajemyje roboty – manipuljatory*. Mir, Moskva 1976.

Kučera, J.: *Syntaktický přístup ke zpracování taktilní informace*. Diplomová práce FEL ČVUT, Praha 1981.

Kurze M.: *Interaction of blind persons with virtual worlds, based on two-dimensional tactile information*. Software-Ergonometrie '97. Usuability Engineering. B. G. Teubner, Německo, 1997.

Lindenbaum M., Bruckstein A. M.: *Blind Approximation of Planar Convex Sets*. IEEE Transactions on Robotics and Automation. Vol.10, No.4, 1994, s. 517– 529.

Liu H., Meusel P., Hirzinger G.: *A Tactile sensing for the DLR Three-Finger Robot Hand.* In.: ISMCR 2004. Houston, USA 2004.

Liu H., Meusel P., Hirzinger G.: *A Tactile Sensing system for the DLR Three Finger Robot Hand.* Institut of Robot and system Dynamics. German Aerospace Research Establishment, 1998.

Ljapunov A. A. a kol.: *Problemy kibernetiky.* Gosudarstvennoje izdatelstvo fizikomatematičeskoj literatury, Moskva 1960.

Lu S., Sakai Y., Wu J. L., a kol.: *Measurement and analysis of the object length perceptive characteristics with visual and tactile information for proposal of three-dimension tactile shape.* Transactions of the Japan Society of Mechanical Engineers, Part C v 70 n 9 September 2004.

Mařík V., Štěpánková O., Lažanský, J. a kol.: *Umělá inteligence (1)*, Academia, Praha 1993.

Mařík V., Zdráhal Z.: *Rozpoznávání obrazů a využití strukturálních metod.* In: Strukturální metody rozpoznávání. ČSVTS Praha 1985, s. 3–48.

Mařík V., Zdráhal Z.: *Strukturální metody rozpoznávání.* In: Metody rozpoznávání obrazů a jejich aplikace v diagnostice, robotice a v dalších oblastech. ČSVTS Praha 1982, s. 35–63.

Mařík V.: *Lingvistický přístup k taktilnímu rozpoznávání.* Kandidátská disertační práce. FEL ČVUT, Praha 1979.

Mařík V.: *Tactile Pattern Recognition by Use of Formal Languages.* In: International Conference of Artifitial Intelligence and Information – Control Systems of Robots. Smolenice 1980, s. 152–154.

Mařík V.: *Taktilní rozpoznávání.* Písemná práce k odborné kandidátské zkoušce. FEL ČVUT, Praha 1978.

Matsumiya T., Nakayama S., Miura Y., etc.: *Intelligent control method for robot hand based on tactile information by double-octagon tactile sensor.* IEEE International Conference on Systems, Man, and Cybernetics 'Human Communication and Cybernetics', Tokio, 1999

Mehdian M., Thomas D.: *Tactile Recognition of Solids Object.* Robotica. Vol.13, 1995. s. 169–175.

Miyasato T., Nakatsu R.: *Allowable delay between images and tactile information in a haptic interface.* International Conference on Virtual Systems and Multimedia (Cat. No. 97100182). IEEE Comput. Soc., USA, 1997.

Miyasato T., Noma H., Kishimo F.: *Subjective evaluation of perception of delay time between visual information and tactile information.* IEICE Transaction on Fundamentals of Electronics, Communications and computer Sciences, vol. E79-A, no. 5. Inst. Electron. Inf. & Commun, Anglie, 1996.

MST news: International newsletter on micronano integration, No.2/05.

Murali Krishna G., Rajanna K.: *Tactile Sensor Based on Piezoelectric Rezonance.* Indian Institute of Science, Bangalore, Indie 2002.

Nakamura Y., Hanafusa H., Ueno N.: *A piezoelectric film sensor with uniformly expanded surface to detect tactile information for robotic end-effectors.* Proceedings of '85 International Conference on Advanced Robotics. Japan Ind. Robot Assoc. Japonsko, 1985.

Neděla R.: *Snímání taktilní informace a její zpracování v biomedicíně.* Diplomová práce. FS ČVUT. Praha 2005.

Neděla R.: *Taktilní informace a její využití.* FS ČVUT, Praha 2006.

Niemann H.: *Pattern Analysis.* Springer-Verlag Berlín, Heidelberg, New York 1981.

Nicholls H. R., Lee M. H.: *A Survey of Robot Tactile Sensing Technology.* The International Journal of Robotics Research. Vol. 8, No. 3, 1989, s. 3–30.

Oliva J.: *Snímání a zpracování taktilní informace v biologických systémech.* Teze doktorské disertační práce. FS ČVUT, Praha 2006.

Pliska M.: *Metody rozpoznávání primárníhotaktilního obrazu.* DP FEL ČVUT, Praha 1986

Pugh A.: *Robot Sensors.* Vol. 2 - Tactile and Non-Vision. IFC (Publications) Ltd., Velká británie, Springer-Verlag, New York 1986.

Reynaerts D., Brussel H. V.: *Tactile Sensing Data Interpretation for Object Manipulation.* Sensor and Actuators A. 37–38, 1993. s. 268–273.

Sano A., Nishi K. a kol.: *Multi-fingered hand for telepresence based on tactile information.* 2004 IEEE International Conference on robotic and Automation (IEEE Cat. No. 04CH37508), vol. 2, IEEE, Piscataway, USA, 2004.

Sensors. Puchheim: Sensortechnics, 2005. 36 s.

Shilong J.: *Sensor Based Manipulation for Multifinger Robotic Hand.* August 2000, Department of Electrical and Electronical Engineering, Hongkong, 2000.

Shimojo M., Ishikava M., Kanaya K.: *A Flexible High Resulution Tactile Imager with Video Signal Output.* Reprinted from Proceedings of the 191 IEEE International Conference on Robotics and Automation. Sacramento, California. 1991.

Schmid H. P., Bekey G. A: *Tactile information processing by human operators in control system.* IEEE Transaction on Systems, Man and Cybernetics, vol. SMC-8, no. 12, 1976.

Schneiter J. L., Sheridan T. B.: *An Automated Tactile Sensing Strategy for Planar Object and Localization.* IEEE Transaction on Pattern Analysis and Machine Intelligence. Vol. 12, No. 8, 1990, s. 775–786.

Šimšík D., Živčák J., Volf J.: *Tactile Transducer Development for Rehabilitation and Exercise Machines.* European Journal of Mechanical and Enviromental Engineering, Vol.41, No.1, BSMEE, 1996, s. 19–24.

Teh Ch. H., Chin R. T.: *On the Detection of Dominant Points on Digital Curves.* IEEE Transaction on Pattern Analysis and Machine Intelligence. Vol. 11, No. 8, 1989, s. 859–872.

*Technical documentation of the conductive composite elastomer CS 57-7 RSC,* Yokohama Rubber Co. Ltd., Japonsko, 1980.

Todova V., Christov S.: *A system for processing of tactile information.* Proceedings of the Fourth International Symposium on Measurement and Control in Robotic. Slovak Tech. Univ, Slovensko 1995.

Tsang K. M., Billings S. A.: *Two Dimensional Pattern Analysis and Classification Using a Complex Orthogonal Estimation Algorithm.* IEE Proc.-Vis. Image Signal Process. Vol. 141, No. 5, 1994, s. 339–347.

van Veen H. A. H. C., van Erp J. B. F.: *Tactile information presentation in the cockpit.* Haptic Human-Computer Interaction. First International Workshop. Springer-Verlag, Německo, 2001.

Volf F.: *Hodnotová analýza ve stavebnictví.* SNTL, Praha 1982.

Volf J. a kol.: *Systém pro měření rozložení statických a dynamických tlaků pro biomechanické aplikace.* AUTOMA. No.11/2002, Vol.8.FCC Public s.r.o., s. 17–19.

Volf J., Holy S., Papezova S., Vlcek J.: *Tactile Transducer for Pressure Distribution Measurement and its Practical Test-Proceeding IMEKO XV.* Osaka, Japonsko 1999, s. 153–157.

Volf J., Holy S., Vlcek J.: *Using of Tactile Transducer for Pressure Distribution Measurement on Sole - Sensor and Actuators A62.* Physical, Elsevier Sequoia S.A., Lausanne, Švýcarsko, 1997, s. 556–561.

Volf J., Papežová S., Vlček J., Novák P.: *Orthopaedic Investigation System for medical non-invasive diagnostics.* In. XVIII IMEKO World Congress, Rio de Janeiro 2006b.

Volf J., Papežová S., Vlček J., Otáhal J., Novák P., Holý S.: *Transducer for Pressure Distribution Measurement and its Practical Tests.* In.: The 5th World Multi-Conference on Systemics, Cybernetics and Informatics SCI 2001. Orlando 2001, Florida, USA, s. 575.

Volf J., Papežová S., Vlček J.: *Transducer for Extreme Pressures Differentiation. In: Measurement and Control.* in Robotics ISMCR 03-IMEKO TC17. Madrid: Instituto de automatica industrial, 2003, s. 111–114.

Volf J., Vlček J., Semnicky T.: *The piezoelectric tactile sensor for static force measurement.* In. XVIII IMEKO World Congress, Rio de Janeiro, Brazílie, 2006a.

Volf J., Vlček J.: *New Piezoelectric Tactile Sensors.* In: XV. World Congress IMEKO. Japonsko, Osaka 1999, s. 35–40.

Volf J.: *Classification of Tactile Images by Dicrininatory Analysis.* In: EUROSENSORS X. Belgie, Leuven 1996b, s. 1369–1372

Volf J.: *Evalution Methods of Tactile Information.Elektrotechnický časopis.* Vol. 43 (1992a), No. 1., s. 23–25.

Volf J.: *Methods for Processing the Outline Tactile Information.* In:ISMCR'93, Turín, Itálie, 1993. s. As.IV-7–As.IV-14.

Volf J.: *Methods for Processing the Tactile Image Based on The Solution of Helmholtz,s Equation.* Elektrotechnický časopis. Vol. 42, 1991, No. 11–12, s. 629–637

Volf J.: *Methods for Processing the Tactile Pattern.* In: Second Japanese-Czech-Slovak Joint Seminar on Applied Electromagnetics in Materials. Kyoto, Japonsko, 1994. s. 47–50.

Volf J.: *Methods of Processing Tactile Information based on the Solution of Helmholtz's Equation.* Sensors and Actuators A. Physical, Vol 41, No. 1–3. ELSEVIER SEQUOIA S.A., Lausanne. Švýcarsko. 1994. s. 174–179.

Volf J.: *Metody rozpoznávání taktilních obrazů.* In: AI '87. Praha 1987, s. 270–277.

Volf J.: *Metody zpracování taktilního obrazu s využitím Helmholtzovy rovnice.* Acta Polytechnica, 9(II,3), Praha 1990, s. 93–103.

Volf J.: Metody zpracování taktilního obrazu. In: AI '89. ČSVTS, Praha 1989, s. 329–338.

Volf J.: *Tactile Sensors and Transducers.* In: XIII IMEKO World Congress, Turín, Itálie, 1994, s. 2022–2027.

Volf J.: *Using Dicrininatory Analysis II for Classification of Tactile Images.* In: Fifth International Symposium on Measurement and Control in Robotics ISMCR'96, Belgie, Brusel 1996a, s. 421–425.

Volf, J.: *Methods for Processing the Tactile Pattern. In: Second International Symposium on Measurement and Control in Robotics* (ISMCR '92). Tsukuba, Japonsko, 1992b. s. 107–114.

Volf, J.: *Pattern Recognition of Tactile Images by Discriminatory Analysis. In.:International* Conference on Computational Intelligence for Modelling, Control and Automation CIMCA '2001, Las Vegas 2001, USA, s. 261–270.

Volf, J.: *Tactile Sensors and Transducers for Robotisation.* In: ISMCR '93, Turín, Itálie, 1993. s. As.IV-13-As.IV-18

Volf, J.: *The Influence of Features Sequence on Recognition.* In. Proceedings Volume XI TC 17 of XVI IMEKO World Congress. Vídeň 2000, Rakousko. s. 69–74.

Weed H., Taige P.: *Vibrator matrix for a tactile vision information system.* Bioengineering American Society of mechanical Engineers, Petroleum Division PD v 64 n 4 1994, ASME, USA, 1994.

Yan-Bin J.: *Localization on curved object using tactile information.* Proceedings 2001 IEEE/RSJ International Conference on Inteligent Robots and Systems. Expanding the Societal Role of Robotic in the Next Millenium (Cat. No. 01CH37180), vol. 2. IEEE, Piscataway, USA 2001.

Yuji J. I., Shida K.: *A discriminating method of three mixed tactile information using pressure-temperature sensitive materials.* Transaction of Electrical Engineers of Japan, Part E, vol. 119-E, no.4, Inst. Electr. Eng., 1999.

Zivcak J., Simsik D., Knezo D: *Experimental testing of local muscle disposition sensor.* ISMCR'95, Smolenice, Slovensko, 1995, s. 543–546.

Živčák J., Šimšík D., Kněžo D., Volf, J.: *Electronic Processing of Local Muscle Disposition Sensor Output.* The International Journal of Microcircuits & Electronic Packaging. Vol.20, No.1, IEPS, ISHM, USA, Virginia 1997, s. 41–45.

**Internetové odkazy:**

[1] http://www.interlinkelec.com/

[2] SPI – Sensor Product Inc. [on line]. c2006. Dostupné z http://www.sensorprod.com/
[3] http://www.techstorm.co.kr/e-main.htm
[4] http://search.globalspec.com/Search?query=%22tactile%20sensor%22&n=10&show=total&pg=1
[5] http://robotics.eecs.berkeley.edu/~nicolson/paperlinks/dissert.ps.Z
[6] http://robotics.eecs.berkeley.edu/~nicolson/paperlinks/icra95b.ps
[7] http://robotics.eecs.berkeley.edu/~nicolson/nicolson.html
[8] http://www.soton.ac.uk/~rmc1/robotics/artactile.htm
[9] http://www.dactyl.com/scratchpad/pps/tactileArraySensor.html
[10] http://www.freepatentsonline.com/4492949.html
[11] http://www.globalspec.com/learnmore/sensors_transducers_detectors/pressure_sensing/pressure_tactile_imaging_sensors
[12] http://www.sfb588.uni-karlsruhe.de/publikationen/2003/820%20KerpaR4_2003.pdf
[13] http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve&db=PubMed&list_uids=8584258&dopt=Citation
[14] www.cs.queensu.ca/home/browse/Publications/webstuff/ROGERB.htm
[15] http://www.springerlink.com/content/emy9133pcpg6t5ej
[16] http://www.neuroinformatik.rub.de/PEOPLE/rolf/tactile2006.html
[17] http://www.cs.queensu.ca/home/browse/Publications/webstuff/ROGERC.htm
[18] http://sensors-transducers.globalspec.com/LearnMore/Sensors_Transducers_Detectors/Pressure_Sensing/Pressure_Tactile_Imaging_Sensors
[19] http://search.globalspec.com/search?query=Pressure%20and%20Tactile%20Imaging%20Sensors
[20] http://www.dactyl.com/scratchpad/pps/robotics.html
[21] http://www.fujipress.jp/finder/xslt.php?mode=present&inputfile=ROBOT000900030005.xml
[22] http://mass.micro.uiuc.edu/publications/papers/134.pdf
[23] http://www.rit.edu/%7Eeasi/itd/itdv03n4/article2.htm
[24] http://ieeexplore.ieee.org/search/wrapper.jsp?arnumber=220070
[25] http://www.rit.edu/%7Eeasi/itd/itdv03n4/article2.htm
[26] http://mobiquitous.com/active-belt-e.html
[27] http://www.namrl.navy.mil/TSAS/achievements.htm
[28] http://www.tuat.ac.jp/Tech96e/DenshiJouhou/t66b_e.html
[29] http://www.kajitani.mce.uec.ac.jp/study-shmj/shimojo.html
[30] http://www.sanbi.co.jp/rsj/Conts/Vol_18/Vol18_6e.html
[31] http://www.robotic.dlr.de/
[32] http://www.technovelgy.com/ct/Science-Fiction-News.asp? NewsNum=576
[33] http://www.pressureprofile.com/
[34] http://www.ottobock.com/cps/rde/xchg/SID-3F574DD1-489B767C/ob_com_en/hs.xsl/3633.html
[35] http://epedagog.upol.cz/eped1.2004/clanek11.pdf [cit.27.12.2006]
[36] http://www.mobiquitous.com/active-belt-e.html
[37] http://www.namrl.navy.mil/TSAS/DEFAULT.HTM

# 18 Od jednoduchých robotů k anticipujícím stvořením

## Úvod

Cílem této kapitoly je nejen shrnout z dnešního pohledu průběh, možné příčiny a pozadí celého dosavadního vývoje inteligentní robotiky, ale pokusit se i stručně analyzovat a zdůvodnit současný stav v celém kontextu historie. Navrhnout možná opatření a vývojové proudy pro rychlejší progresi dalšího úspěšného vývoje v blízké budoucnosti. Z podobného úhlu pohledu jako v prvém oddílu se dále věnuji i oblasti umělého života (*ALife*). Z filosofujícího nadhledu se zaměřuji na problematiku umělých stvoření navrhovaných v doméně umělého života a rizika jejich možného vývoje a zneužití v budoucnu. Velkou část výkladu věnuji posléze anticipaci – předjímání; tedy jak anticipující umělé stvoření vyhodnotí vliv chování svého okolí na původní předpověď určité události (*Anticipace = predikce + vliv chování okolí*). Zamýšlím se i nad možnými aspekty koexistence člověk–stroj a jejími riziky.

## 18.1 Vývoj robotů

### 18.1.1 Vyvíjení robotů – začátky robotizace

Myšlenka automatů, které mohou provádět nejrůznější úkoly, pochází ze starověkého Egypta. V historii se opakují pokusy o umělé konstrukce člověku se podobajících zařízení, zejména na mechanickém principu. Tento princip byl ale Karlu Čapkovi cizí; jeho „roboti" ve hře RUR[1] byli koncipováni pouze na biochemické bázi.

První velmi jednoduché průmyslové roboty – spíše z dnešního pohledu manipulátory – se objevily začátkem 50. let minulého století, v době studené války, kdy byl svět stravován hrozbou atomového nebezpečí. Tehdy také *UNIMATION* vytvořil první průmyslově využitelný manipulátor s jedním robotickým ramenem.[2] Počáteční nedůvěra byla velká. Teprve tři roky poté, v roce 1961, bylo několik prvých *UNIMATION 1900* použito úspěšně v General Motors pro lití pod tlakem. Od roku 1962 začala dodávat na trh své roboty konkurenční firma *VERSATRAN*. Firma *ASEA* navrhla prvního robota již se třemi stupni volnosti pohybu. Roboty užívané v 60. létech 20. st. byly však zcela neautonomní jednotky, umístěné v pevných lokalitách do pevných pozic. Byly poměrně

[1] R.U.R. – Rossum´s Universal Robots, Praha 1920.
[2] Bylo to v roce 1958 a „robot" se jmenoval UNIMATE 1900.

pomalé, udělaly nejvýše několik pohybů za minutu. Působily však už tehdy impozantně, i když mohly provádět pouze jednoduché úkoly jako přemisťovat nebezpečné sloučeniny z jednoho místa na druhé. O trochu později přišel *Barrett Electronics* s bezdrátově řízeným elektrickým nákupním vozíkem pro obchodní dům se smíšeným zbožím. Ten byl navigován signály vysílanými několika majáčky na příslušném podlaží. V 70. létech minulého století se začalo již s osazováním výrobních linek roboty, nejprve svařovacími, montážními, tvářecími, stříkacími apod.

Hnacím motorem pro rozvoj robotiky a zavádění robotů do denní praxe byla a je touha po zlepšování životní úrovně, úsilí o rozvoj výroby, růst produktivity práce, sociální požadavky (hygiena, humanizace práce, monotónní a nebezpečná práce), rozšiřování sortimentu, zvyšování kvality výroby, ...

### 18.1.2 Mezigenerační posun v možnostech a myšlení

V době, kdy jsem se narodil, v roce 1946, si vědecká komunita nebyla vědoma faktu, že naše dědičnost je uchovávána uvnitř jádra DNA každé buňky. Tyto řetězce nukleotidů jsou chemické substance vytvořené kombinacemi jen čtyř enzymů, reprezentovaných sekvencemi jejich začátečních písmen *c, g, t, a*. Od přelomu tisíciletí již vědci umí rozpoznat sekvenci úplného řetězce lidské DNA, stejně jako DNA myší, jednoduchých rostlin a četných dalších živočišných druhů a tříd. Každá taková sekvence neboli genom obsahuje miliony až biliony znaků, což naprosto vylučuje, aby mohly být uloženy klasickým způsobem v knihách. Data jsou skladována rutinně v počítačových médiích a nikoho to již neudivuje. Stejně jako existence stále dokonalejších programů a výkonnějšího hardwaru, kde jsou implementovány. Textový editor Word Office 2003, ve kterém píši tento příspěvek, mě sám upozorňuje na případné gramatické chyby, navrhuje jejich opravy či vhodnější větnou stavbu. Internet mi poté umožní odeslat text bezprostředně po jeho dokončení prostřednictvím e-mailu. V roce 1946 neexistoval ani náznak těchto možností výrazně měnících způsob a kvalitu našeho života, naší relaxační i tvůrčí činnosti.

V polovině dvacátého století se technické informace i další vědomosti a fakta uchovávali v knihách. Nejdůležitější díla byla duplikována ve stovkách knihoven. Člověk, který si přál získat přístup k těmto informacím, musel za nimi vážit cestu, poté vyhledat příslušný svazek využitím mechanického kartotéčního systému a nakonec nalistovat ručně požadované stránky odpovídající tématu. Již dnes, o budoucnosti ani nemluvě, vyžaduje stejná operace jen několik málo minut či jen pár kliknutí myší. Získaná kompletní informace je v celé své složitosti (textové, grafické, zvukové, obrazové atd.) ihned uskladnitelná v digitální podobě na CD či DVD nebo na miniaturní flash paměti o kapacitě mnoha GB. Vynechme otázky spojené s bezpečností používané sítě s množstvím rozesílaných spamů, počítačových virů a trojských koní. Abychom dnes získali požadované materiály, musíme mít k dispozici v podstatě jen moderní hardware, přístup k Internetu a sami být poměrně dobře uživatelsky obeznámeni s ovládáním potřebného softwaru. Uskladnění každého dokumentu v počítačích je omezeno pouze do několika málo míst – URL adres dosažitelných pomocí vhodného vyhledávače z jakéhokoliv internetového připojení na této planetě.

Evoluční – vývojová – teorie se občas namáhá vysvětlit nejkomplexnější rysy (znaky) a charakteristiky organismů. Minsky používá lidské oko jako příkladu (Minsky, 1967). To je očividně určeno především pro vidění a sestává proto z několika částí přizpůso-

benych výborně ke svému účelu – to je například čočka, která může zaostřovat na různé vzdálenosti. Ale jak se něco tak komplikovaného, jako je oko, mohlo vůbec vyvinout?

Od časů Charlese Darwina biologové usuzovali, že takové znaky musely vyvstat – vyvinout se během množství přechodných kroků. Kromě toho tyto přechodové struktury možná sloužily ke zcela odlišným účelům, než slouží dnes. Například krystalické proteiny, ze kterých se čočka oka skládá, jsou blízce příbuzné k těm, které slouží k enzymatickým funkcím nesouvisejících s viděním. V tomto případě si evoluce vypůjčila jeden již existující protein a dala mu zcela novou funkci. „*V průběhu času*," říká Minský, „*stará struktura zde mohla být vylepšena pro zavedení a zdokonalení funkce zcela nové. A to je mnohem, mnohem snadnější než vynalézání něčeho naprosto nového.*"

Darwinovská evoluce je teorie, která exaktně neuvádí, jak je vyvíjející se informace kódována. Tento proces spontánně napodobuje DNA i počítačový kód, a to v podstatě stejným způsobem. A právě to nás opravňuje studovat a modelovat evoluci s využitím počítače. Ten sice nemůže být inteligentní, ale může v něm být formou programu vytvořen jakýsi paralelní umělý svět, v němž přirozené formy inteligence mohou být replikovány. Správná cesta k návrhu stroje, který myslí, není ve vyšetřování procesu jeho myšlení. Musí to platit obráceně, nejdříve je nutné vyšetřovat struktury biologických systémů a chování živých mechanismů, protože ty leží mnohem hlouběji než vědomé myšlení.

Mysl společně s emocemi nemůže být vysvětlena aparátem fyziky (Minsky,1986). Přístupy umělého života mohou často řešit problémy mnohem lépe než to umíme my, užívajíc svou vlastní inteligenci, běžné vědomosti a svůj rozum. V současnosti proto mnoho kybernetiků, robotiků a vědců humanitních oborů využívá pro řešení svých složitých problémů evolučních algoritmů, které právě z principů genetiky vychází (Lažanský, 2001).

### 18.1.3 Chování inteligentních strojů a jejich vývoj

Padesát let vývoje robotů rychle uběhlo. Dnes se svět mění mnohem dynamičtěji než kdykoliv předtím. Poměrně inteligentní roboti, vybavení sice jen omezenou mobilitou a autonomností, mohou již vykonávat úkoly určené ještě donedávna pouze člověku. Rodney R. Brooks (Brooks, 1995) z MIT charakterizoval mobilního inteligentního robota takto:

- *Je situovaný v prostředí,* aby mohl být tímto prostředím bezprostředně ovlivňován;
- *Musí mít tělo* (*Embodiment)* pro vnímání dynamiky prostředí a jeho vlastností;
- *Musí mít inteligenci,* aby mohl vnímat, pozorovat a učit se.

Takto definovaný robot musí navenek vystupovat jako jeden celek. Z druhého rysu plyne požadavek, že nesmí být virtuální, nesmí mít ani virtuální senzory. Požadavkem je také situovanost prostředí. Virtuální roboti na Internetu jsou z té definice vyřazeni. Oproti tomu může být za robota pokládán i automobil.

Právě v oblasti dopravy bude *auto blízké budoucnosti* koncipováno jako vysoce inteligentní robot na čtyřech kolech. Když uvážíme, že průměrná chybovost lidských rozhodnutí je $10^{-3}$ a běžně dosahovaná hodnota u technických zabezpečovacích systémů je $10^{-9}$, jedná se při řízení bez těchto bezpečnostních systémů o hazardování se životem. Ve všech, zdaleka již nejen futurologických návrzích se klade důraz především na

bezpečnostní prvky, jako jsou samovyhýbání se neočekávaným překážkám, řešení krizových a život ohrožujících situací apod.[3] Už nyní je v sériově vyráběných automobilech implementováno poměrně velké množství automatických funkcí, které si ani neuvědomujeme. U současných navigačních přístrojů dochází ale paradoxně čím dál častěji k častým dopravním nehodám. Jejich manuální nastavování za jízdy totiž výrazně odvádí řidičovu pozornost; doporučení výrobců při korekcích trasy vždy vůz zastavit, nelze mnohdy (např. v hustém provozu, na dálnici bez odstavného pruhu apod.) ani při nejlepší vůli realizovat.

Ambiciózní projekt technického centra automobilky SEAT *AIDE* (Adaptive Integrated Driver-Vehicle Interface) využívá pro zvýšení bezpečnosti silničního provozu umělou inteligenci. Právě s její pomocí jsou on-line nepřetržitě zpracovávány snímané informace týkající se jak vlastního vozu, tak jeho okolí a řidiče. *Adaptivní rozhraní mezi řidičem a „inteligentním" vozem* SEAT León vychází při plánování a aktualizaci trasy vozidla (včetně případných nutných akčních zásahů) nejen z údajů GPS, ale automaticky bere v úvahu i kritické překážky lokalizované radarem.

Současně sleduje a vyhodnocuje i pohyb a frekvenci mrkání řidičových očí. Kamerový systém v kabině řidiče je vidět na obr. 18.1.

Obr. 18.1 Inteligentní prototyp vozu SEAT León.

Když se například řidič nedívá na silnici a vůz zaznamená hrozící nebezpečí, rozsvítí se na palubní desce varovné světlo a současně zazní varovný zvukový signál. Prototyp také dokáže vysílat varovný zvukový signál a vyvolat vibrace ve volantu, pokud zjistí, že řidič upadá do mikrospánku. V Technickém centru SEAT experti vyvinuli rovněž nové způsoby interakce řidiče s vozem, například možnost ovládat převážnou většinu funkcí v menu přímo z volantu. Většinu z těchto funkcí lze již ovládat také hlasem. Rovněž

[3] Dosud platí Vídeňská úmluva z roku 1968, která mimo jiné požaduje, aby řidič trvale ovládal své vozidlo, za jehož provoz je odpovědný. Automatický systém nesmí de-jure řízení vozidla převzít plně. Řada podpůrných bezpečnostních systémů jako ABS ale v autech již existuje.

zaparkování auta León bude postupnem času až plně automatické. Pro obousměrný tok informací jsou v *AIDE* využívány speciální komunikační kanály.

Systém Siemens snímá kamerou CMOS dění před vozidlem, rozpoznává dopravní značky a palubní počítač na ně upozorňuje řidiče (překročená rychlost, zákazy apod.). V kombinaci s navigačním systémem GPS zjistí, zda je v obci, mimo ni či na dálnici.

Na obdobných projektech, kdy vůz dokáže přizpůsobit svoji komunikační strategii silničním podmínkám, okolí i kvalitě samotného řidiče „inteligentního" vozidla (rozpoznáváním jízdního pruhu, vjetí do protisměru, nočním viděním atd.), intenzivně pracuje většina světových výrobců aut a postupně je také již uvádí do praxe[4].

Stále jsme však závislí na přeplněných silnicích, jejichž princip se vlastně nezměnil od dob antického Říma. Podle prognóz se ale již během dvaceti let dočkáme převratných změn. Počítačové naváděcí systémy a levná aktivní GPS umožní flexibilní individuální přepravu, přičemž odpadne nutnost orientace podle silniční sítě. Intenzivně se pracuje na vývoji tichého, spolehlivého, trojživelného (země, voda, vzduch) individuálního dopravního prostředku. Pro tento nový způsob dopravy budou muset být ale vypracována a přijata nová celosvětově platná pravidla pohybu ve 3D prostoru. Očekávané problémy budou ale víc než vyváženy úsporou finančních prostředků na infrastrukturu (stavbu, rozšiřování a údržbu dálnic a mostů).

Všestranní mnohoúčeloví roboti vstoupili na trh teprve před několika málo lety. Mikromechanický lékařský operační systém *da Vinci* firmy Intuitive Surgical, Inc., je sada velmi přesných „robotických rukou" převádějících věrně i nepatrné pohyby rukou a prstů chirurga na stejný pohyb nástroje v těle pacienta. Přirozený třes je přitom softwarově kompenzován. Systém *da Vinci* s prvky umělé inteligence tak dovoluje přesně nastavit a precizně vést i velmi komplikovaný chirurgický řez. Operatér sedí u ovládací a řídicí konzole a pomocí převodů ovládá pákovými ovladači (joystick) čtyři až šest chapadel robota. Do pacientova těla se přitom protáhne jen centimetrovými rankami. Optický systém, zaváděný endoskopicky jedním z ramen, umožňuje sledovat sterilní operační pole zdálky v mnohonásobném zvětšení s dokonalým prostorovým 3D vjemem. Podle vyjádření chirurgů je přenos pohybu jejich rukou do ramen robotu tak věrný, že mají pocit, jako by měli ruce přímo v těle pacienta. Ramena robotu jsou ale přitom široká jen 8 mm. Dostanou se tak i tam, kam lidská ruka nedosáhne. Umí provrtat špatně přístupnou kost i v nezvyklých úhlech. Jde v podstatě o zdokonalenou laparoskopickou operaci, která je ale méně invazivní a zraňující. Následně je i rekonvalescence pacienta rychlejší, méně bolestivá a léčba vyjde celkově levněji. V současnosti pracuje po celém světě již kolem 500 systémů *da Vinci*. Jejich první instalace v ČR se uskutečnily v ÚVN Praha a v Nemocnici Na Homolce v roce 2005.

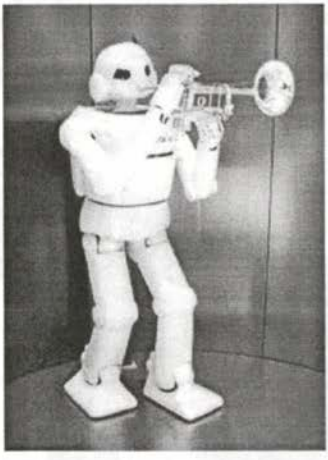
Překvapivě největším novým odběratelem robotů tak už dlouho není průmysl automobilový (i když co do absolutního počtu je zde robotů nesporně stále nejvíce), nýbrž průmysl biotechnologií a velké nadnárodní farmaceutické společnosti.

Takoví a ještě „chytřejší" roboti nejsou ale v podmínkách současné globální ekonomiky ještě dostatečně lákavým artiklem. Trh malých osobních robotů se stává ale pozvolna rozsáhlejším než trh průmyslových pevných a nemobilních robotů.

---

[4] Koncern Volkswagen předpokládá, že od roku 2007 bude všechny své nové typy vozů (včetně SEAT) standardně vybavovat systémem pro navádění při parkování.

Společnost *iRobot* prohlašuje, že v jediném roce (2004) prodala jen v Evropě dalších několik set tisíc domácích úklidových robotů Scooba a Roomba, relativně levných, autonomně řízených, samostatně poháněných vysavačů, byť zatím jen s minimem umělé inteligence.[5] Obdobné robotické vysavače vyrábí dnes sériově řada dalších firem.[6] Existuje prosperující trh robotů i pro stříhání trávníků, čistění bazénů a pro další stereotypní, na inteligenci nenáročné aplikace.



Obr. 18.2 Humanoidní robot firmy Toyota.

Nicméně schopnosti inteligentních robotů překotně vzrůstají. Průměrný lidský mozek (asi 1500 gramů) umí zpracovat okolo sto bilionů instrukcí za sekundu – tedy stále téměř třikrát více, než stihne i ten nejvýkonnější počítač světa *BLUE GENE* firmy IBM. Podle Mooreových zákonů by úroveň výkonu počítačového zpracování mohla vzrůstat až k bodu, kdy stroje budou moci pracovat téměř tak dobře jako lidský mozek (Minsky, 2005).

*"Nechme to proboha, jak to je! Dostatečně užiteční roboti nepotřebují mít přece k dispozici mozkovou kapacitu srovnatelnou plně s člověkem,"* říká profesor Hans Moravec z Carnegie Mellon's University.

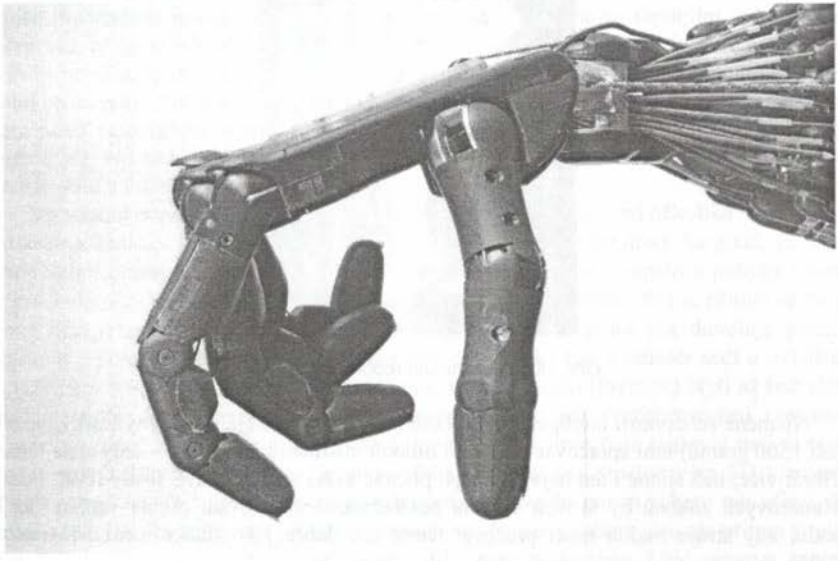
Pomineme-li lidský mozek a jeho těžko dosažitelnou výkonnost, vítězí dnes na trhu antropomorfismus. Humanoidní roboti jsou konstruovány s cílem, poskytovat účelné robotické funkce v takové estetické formě, která zvětší pravděpodobnost jejich přijetí ve společnosti. Musí být vzhlední a své okolí nevyděsit. Proto jsou také například vždy nižší než dospělý člověk. Výhledově by měly napodobovat akce člověka ve fyzickém smyslu nebo v duševním smyslu (*biomimikry*). Prvního humanoidního tančícího robota předvedla firma Schunk a robot firmy Toyota (zachycen na obr. 18.2) nedávno v Praze

[5] Pro porovnání cenového trendu a tím i ekonomické dostupnosti: v České republice poklesla prodejní cena robotického vysavače Roomba během 3 roků ze 33 tisíc na 11 900 Kč, včetně virtuálních zdí a kompletního příslušenství (stav říjen 2006).

[6] Od firmy Electrolux jsou to robotické vysavače Trilobit a Automower v obdobných cenách.

ohromoval svojí hrou na trubku. Firma Shadow pro roboty této kategorie navrhla vysoce sofistifikovaně vyvinutou ruku, analogickou k lidské, jejíž 36 pneumatických svalů umožňuje pohyb ve 24 stupních volnosti, viz obr. 18.3.

Tyto roboti už spadají do rychle se rozvíjející *kategorie sociálních robotů*, kteří reagují podle chování svého okolí *(Behaviour Based Robotics)*. Dokonce již nyní nabízejí některé převážně japonské firmy (Sony, Honda) své robotické výtvory už i za společníka, i když ještě stále na poměrně nízké úrovni možných jednání a akcí. Výhledově by měly poskytovat „inteligentní" umělá stvoření společnost (a také i pomoc v krizových situacích) starším, nemocným a osamělým lidem. Příkladem prvních velmi jednoduchých sociálních robotů jsou Tamagotchi, Sony Aibo, Ludobot a Wakamaru.



Obr. 18.3 Pneumatické svaly umělé ruky firmy Shadow.

Problém napodobování lidského chování je velmi komplexní a stále otevřený; ještě se jím budeme později zabývat. Poptávka po takových robotech zůstává však stále nízká. *„Dnes si už můžete opatřit robota, aby s Vámi hovořil, ale nemůžete po něm chtít, aby s Vámi rozprávěl jako filosof Spinoza"*, říká Moravec.

Nepřítomnost dalších typicky lidských orgánů – dvou plně funkčních nohou je už zjevnější. Zatímco někteří experimentální roboti mají nohy, které napodobují chůzi ještěrek nebo krabů, kola mohou dobře fungovat v téměř všech prostředích a jsou mnohem snadněji naprojektovatelná. *„Nohy – ty jsou vhodné tak pro Hollywood a umělá stvoření jako byl Terminátor"* dodává prof. Moravec.

V poslední době vystupuje oblast *behaviorální robotiky* čím dál zřetelněji jako integrace poznatků z celé řady různorodých vědních oborů a disciplín (Kelemen, 1993). Všechno je vzájemně propojeno – software, hardware, mechanika, elektronická zařízení,

kybernetika, etologie, aplikace progresivních myšlenek a nápadů (Franck, 1985), (Lorenz, 1993) a (Dierk, 1996). To by měl nadstřešovat marketing, především co se týče podpory prodeje nápadů a jejich prvních realizací.

Podle předpovědí by mohla být již za deset až dvacet let sestrojena celá nervová soustava živé bytosti. Bude tedy vyvinut vyživovaný materiál na biologickém principu. Poněkud mě to děsí, že si lidé takto zahrávají. Domnívám se v této souvislosti, že pouhé napodobování člověka nemá smysl. Život je natolik složitý, že vyrobit jeho kopii je téměř nemožné, je to nekonečný proces.

Robotika by si nyní měla dát za prioritní cíl začleňovat svá stále dokonalejší zařízení – inteligentní roboty do běžného života, aniž by se jako roboti vymezovali a aniž by se jim tak stále říkalo.

## 18.2 Umělé formy života

### 18.2.1 Technické předpoklady realizace

1. První krok k vytváření nové generace umělých bytostí (nazývejme je zatím ještě inteligentní roboti) spočívá v *návrhu nových supervýkonných procesorů* na objednávku, minimálně 1000krát rychlejších, než jsou dnešní. Poté už bychom měli být schopni implementovat v reálném čase systémy počítačového vidění, sofistikované rozhodovací systémy, mechanismy výběru akcí *(Action Selection Mechanism)* a vyhodnocování nasnímaných dat z inteligentních senzorů různých typů.To je skutečně důležité.

2. Vystupuje *potřeba nových originálních látek a hmot*, obdobně progresivních, jako byly ty, jež byly vyvinuté původně pro kosmický průmysl (*Smart Fabrics*). Vyžadujeme pohodlně zpracovatelné, měkké a snadno udržovatelné struktury. Vývoj směřuje k integraci elektronických modulů do textilních látek.[7] Jako nejlepší se jeví vetkání vodičů a sběrnicových systémů přímo do textilie. Na jednotlivé vodivé dráhy se potom připojí mikroelektronické moduly, které musí být samozřejmě lehké, omyvatelné a podobně pružné, ohebné a trvanlivé jako běžné tkaniny.Takové vlastnosti má i nově vyvinutý velmi tenký transponderový modul s tkanou anténou, který se po zašití do obleku může i prát.

3. Hledáme *úspornější pohony* robotů. *Umělé svaly* (obr. 18.3) namísto energeticky velmi náročných motorů a převodovek pohybu se zdají být výhodnou cestou.

4. Je třeba se soustředit na otázku *napájení*, což je jeden z hlavních problémů v podstatě všech mobilních elektronických systémů (včetně notebooků). I zde bude asi nutné nastoupit novou cestu. Boty s vysokými, masivními podpatky se sice hodí dobře k umístění přídavných akumulátorů, jsou však příliš těžké i pro antropoidy. Slibné jsou zejména nové, velmi tenké a pružné tzv. *fóliové akumulátory*. I ty se dají integrovat přímo do povrchové vrstvy oděvu. Mohou se trvale dobíjet solárními články umístěnými nejlépe v oblasti ramen, kam dopadá nejvíce denního přímého světla. Aby bylo možné

[7] Tento výzkum je další fází miniaturizace IT a VT s cílem učinit elektroniku lidmi i roboty „oblékatelnou" (*wearable*). Jednotlivé funkční moduly elektronických systémů jsou na různých místech integrovány do povrchového oděvu těla. Odtud jsou přímo propojitelné do konkrétních komunikačních, počítačových a senzorových sítí (přes displej SMS, GPS, TK...).

solární články pružně přimknout k rameni, jsou rozdělené do velmi malých segmentů, zapouzdřené a zapuštěné do pružného substrátu povrchové vrstvy oděvu.[8]

5. Zásadní význam má *vývoj kvalitní senzoriky*, bez které je pokrok v oblasti kvalifikovaného rozhodování značně problematický. Množství stimulů, které je inteligentní robot schopen registrovat a efektivně je zpracovat, přitom silně determinuje jeho inteligenci (a naopak). Analogicky to platí u všech živých tvorů. Vedle inteligentních senzorů, zpřesňujících informaci o prostředí a o realizovaných akcích, je důležité i kvalitní zpracování informací, které z těchto senzorů přichází. Člověk si z ohromného množství redundantních informací, zaznamenaných zrakem, filtruje jen ty důležité a podle toho se rozhoduje. Obdobně je i úkolem inteligentního zpracování informací ze senzorů vybrat ty, které jsou pro další rozhodnutí podstatné. Robot by měl mít také možnost provázat a tím i verifikovat informace ze všech dostupných senzorů. Obdobně i člověk podle potřeby snímá a vyhodnocuje kromě zrakových i další, např. sluchové, čichové, hmatové, popř. i chuťové, informace.

I zde je však nutné nejdříve závazně definovat jednotné, otevřené a na výrobcích nezávislé *komunikační rozhraní* pro inteligentní senzory. Při návrhu robotických celků tak bude snad již možné využívat senzory různých výrobců. Zatím je velký problém právě s jejich nezaměnitelností a vzájemnou nekompatibilitou.

Výzkum UI a umělého života se vůbec rozvíjí překotně a často nekoordinovaně; ale my potřebujeme, aby pokrok v těchto disciplínách už pokročil do fáze jeho praktických aplikací v každodenním životě. Průmysloví roboti se musí učením adaptovat pro práci i v předem nepřipraveném prostředí. A hlavně už musí být schopni nezávislé, autonomní činnosti po dlouhou dobu provozu.

V neposlední řadě je vývoj inteligentních robotů také ekonomická záležitost. Průměrný člověk uvažuje: „Jestliže je to tak snadné najmout si důvěryhodného a spolehlivého člověka, aby uklidil dům, posekal trávník nebo přestěhoval nábytek, proč se zlobit s nákladným a nespolečenským robotem?" Roboti jsou dnes stále ještě vhodnější pro použití pouze tam, kde mohou odevzdat lepší, kvalitnější práci. Jsou vhodní pro precizní svařování, mikrochirurgickou operaci nebo tam kam člověk nemůže nebo nechce jít, protože tato práce je pro něj příliš nepříjemná nebo nebezpečná.[9] Robot je ale nezastupitelný i v prostředí, která by naopak člověk mohl znečistit, například při výrobě krystalů křemíku nebo mikročipů.

Už v blízké budoucnosti dojde nesporně k fúzi biotechnologií, nanotechnologií[10] a informačních technologií do jedné jediné disciplíny. Umělé materiály s vysokou inteligencí budou kvalitnější a pohodlnější než ty přírodní. „Nastoupila epocha syntetičnosti a všeobecné náhražkovitosti", říkal filosof K. Kosík: „Vytváříme umělé látky, ale také umělé příběhy a požitky i umělý život. A už i místo *diference dobra a zla* nastupuje *rozlišení zástupné, náhražkovité*: nikoli dobro a zlo, ale priorita výkonu, úspěchu, profitu, rentability, rychlého vzestupu a obohacení!" Pro kybernetiky je to *memento*!

---

[8] Tato převratná inovace byla vyvinuta pracovníky Frauenhoferova ústavu IZM (*Institut für Zuverlässigkeit und Mikrointegration*) v Berlíně. Poprvé prezentováno v roce 2005.

[9] Práce při čištění jedovatého odpadu, neprůchodné kanalizace, likvidaci výbušnin, v radioaktivním prostředí, práce ve vesmíru (Mars, Měsíc) nebo pod vodou, v hlubinných dolech atd.

[10] Nanotechnologie představuje perspektivní disciplínu, která si klade za cíl navrhovat a realizovat technické systémy, např. nanoroboty, a to v měřítkách $10^{-9}$ metru, tedy ve velikostí srovnatelné s většinou molekul.

### 18.2.2 Historické pozadí vývoje umělého života

Dřívější pionýři prvých umělých (*life-like)* robotických systémů odráželi ve svých názorech a přístupech k řešení skutečně i ducha a atmosféru doby. Začínali totiž svou práci bezprostředně po druhé světové válce v prostředí dominující vojenské organizace založené na příkazech a řízení. I stroje byly tehdy brány jako způsob jak dominovat a ovládnout přírodu přístupem *shora dolů* (top-down). Řízení bylo synonymum pro dominanci.

Věda oné periody se nám z dnešního pohledu zdá velmi jednoduchá až naivní, doufající, že vše lze vysvětlit v pojmech vlastně jen nedostatečně vyvinutého aparátu. Tito lidé dospívali v 30. létech minulého století, v čase utopických snů, kdy sterilní a odosobnění hrdinové nového světa se vlastně zdáli být dobrým nápadem. Inteligence byla pokládána za unifikovaný proces s tím, že by snad mohla být někdy implementována přímo jako nějaký přesný a opakovatelný algoritmus, jako sekvence vysvětlitelných logických kroků. Zastánci klasické umělé inteligence, opírající se výhradně o znalosti, tvrdí i dnes, že intelekt a inteligence jsou zhruba ty samé věci, a proto i schopnost zdůvodňování může být vyčleněna a implementována odděleně od ostatních módů myšlení. Inteligentní systémy musí být prý proto vytvářeny jen shora dolů, jaksi „naslepo".

Už v osmdesátých létech 20. st. se ale objevily zcela opačné názory. Podle nich nejlepším způsobem jak navrhnout stroj, který myslí, už není snažit se nejdříve pochopit proces, či dokonce popsat proces našeho myšlení. Prokazuje se, že daleko efektivnější je zabývat se biologickým systémem z hlediska vyšetřování především jeho struktury a chování celého ústrojí. Bazální mechanismy leží podstatně hlouběji než vědomé myšlení a jsou navíc evolucí ověřeny. Inteligence je vlastností každé populace. Navrhovat inteligentní systémy se musí proto zdola nahoru (bottom-up) jako například u reaktivních agentů, aby se jejich inteligence mohla za určitých okolností vynořit (to emerge) – emergovat (Connell, 1990), (Havel, 1993), (Havel, 2001). Takto nestandardně uvažující vědci vybudovali základy teorie umělého života jako jakéhosi mezioborového mostu mezi evoluční biologií, etologií, sociologií a etikou na jedné straně a umělou inteligencí, počítačovou vědou a matematikou na straně druhé (Nahodil, 1993), (Nahodil, 2005). Prvními „A-Lifers", stojícími v určité opozici proti tehdy většinovým názorům, byli biologové C. G. Langton a Mark Bagleye (Lagton, 1984), v architektuře počítačů to byl Daniel Hillis (Hillis, 1990), v celulárních automatech S. Wolfram (Wolfram, 1985), v genetických algoritmech americký teoretický biolog J. Holland (Holland, 1992) a v genetickém programování John R. Koza (Koza, 1992). Robotik Rodney A. Brooks z MIT (Massachusetts Institute of Technology) kritizoval, že se v mobilní robotice uplatňuje jen klasický přístup k umělé inteligenci opírající se o model světa. „Svět sám je svojí nejvýstižnější reprezentací"[11] píše již v r. 1991 (Brooks, 1991), (Brooks, 1995). Doporučil namísto „géniů, kteří nedokážou přejít místnost", začít stavět umělé pavouky a brouky, kteří reagují instinktivně – a vůbec začít od jednoduchého života. Inteligentní roboti tak dnes nefungují již jen podle strohých a neměnných matematických postupů, ale i na základě stále se vyvíjejících emocí, mají své vlastní ego, potřeby, vrtochy. Učí se mimo jiné obdobnému chování, které je vlastní živým tvorům v různých prostředích a situacích od okamžiku, kdy přijdou na svět.

[11] Intelligence without Representation; Intelligence without Reason.

Využívání uvedených přístupů umělého života „zdola-nahoru", blízkých přírodě a životu skutečnému nejen v UI, ale i v dalších oblastech vědy a techniky, odráží zásadní posun v myšlení celé naší současné společnosti. Idea, že řízení je něco, co musíme za každou cenu, vždy a v každé situaci prosadit již dávno neplatí. Direktivní postupy „shora-dolů" a rigidní principy obecně jsou považovány za méně efektivní než hledání vzájemných spojitostí a relací mezi částmi celků. I přírodu se již snažíme chápat jako celek, pokorně se od ní učit, chránit ji, a ne bojovat proti ní[12].

### 18.2.3 Základní principy a cíle umělého života

Podstatou přístupů umělého života (ALife) je návrh stále ještě relativně jednoduchých umělých entic, definování jak jejich vzájemných interakcí, tak jejich interakcí s okolím a nakonec pozorování takto vytvořeného systému (Kelemen, 2001). Umělý život je název, kterým se začala v 90. létech 20. st. (ne zcela výstižně) označovat nová disciplína, která s použitím počítačů a dalších umělých médií studuje život přirozený a současně se jím inspiruje při vytváření toho umělého. ALife přináší současně analytický přístup tradiční biologie a přístup syntetický (Langton, 1990). V syntetickém pojetí se spíše pokoušíme o vysvětlení společných vlastností systémů, které se jako živé organismy chovají, než o studium biologických fenoménů pomocí dekompozice živých organismů.[13]

Za současného stavu poznání a možností realizací má ALife v podstatě dva poněkud odlišné základní cíle (Csonto, 2002):

- Inspirovat biologii – navrhovat nové hypotézy o procesech přirozeného života, (*life-as-we-know-it*),
- Inspirovat se biologií – vytvářet umělé systémy, které jsou schopny (díky některým vlastnostem charakteristickým pro živé objekty) přispět k efektivnímu řešení praktických úloh, přicházejících z praxe, (*life-as-it-could-be*).

Základními pilíři teorie umělého života jsou: *samoorganizující princip* behaviorálního řízení *„zdola nahoru"*, *emergentní jevy* a *evoluční mechanismy*. Výzkum je zaměřen na chování jednodušších umělých entit, definování jejich vzájemných interakcí, stejně jako jejich interakcí s okolím. Je známo, že vzájemné lokální spolupůsobení dílčích entit vyvolává na globální úrovni zcela nový jev – *emergenci* (Havel, 1993), (Havel, 2000). Z dílčích elementárních chování a vztahů se (obvykle nečekaně a bez jakéhokoliv centrálního řízení) „vynořuje" výsledné chování obvykle složitější (Hillis, 1988), (Kubik, 2003). Skutečná *evoluce s otevřeným koncem* musí mít pro přežití nejúspěšnějších jedinců v dynamicky se měnícím prostředí emergentní kriteriální funkci. V chování například reaktivních agentů už jsou pak patrny fenomény, které lze na určité úrovni abstrakce označit za projevy života (Kelemen, 2001).

Z přírodních věd převzalo ALife celou řadu pojmů, jako jsou genetická informace, DNA, populace, křížení, mutace, přirozený výběr, generační výměna, výběr a ohodnocování jedinců apod. I v přístupech ALife platí, že *genotyp* zahrnuje všechny genetické

[12] Ještě v 60. létech byl zcela vážně míněn text propagandistické písně *„... Poručíme slunci, dešti, kdy má pršet a kdy hřát..."*

[13] Klasický analytický přístup, používaný zejména biology, je založen mimo jiné právě na dekompozici živých organismů s cílem odhalit mechanismus jejich činnosti.

informace uložené v „chromozomech" robota-agenta. *Fenotyp*, popisující stavbu a fyziologické vlastnosti agenta, je pak nejen kombinovaným výsledkem genů,[14] které se skutečně projevily, ale i všech vlivů prostředí ovlivňujících jedince od samého začátku během jeho simulovaného či reálného života (Kadleček, 2001), (Nahodil, 2004).

Výběr následného evolučního kroku u jednodušších simulátorů života vychází vesměs ze zadané kriteriální funkce (*Fitness Function*). Fitness organismu ohodnocuje jeho celkovou schopnost přežít v daném prostředí. Čím vyšší je fitness, tím má organismus větší šanci stát se vzorem pro tvorbu nové generace a šířit tak svoji genetickou informaci dál (schopnost reprodukce). Tento operátor nám simuluje mechanismus přirozeného výběru v přírodě, kde přežívají jen nejsilnější jedinci.[15] Těmito otázkami se zde nechci podrobněji zabývat, čtenáře bych si dovolil odkázat například na důkladně propracované kapitoly Evoluční výpočetní techniky (Lažanský, 2001) a Genetické programování a vybrané problémy EVT (Lažanský, 2003).

Klíčovým principem umělého života je adaptace a evoluční pohled. Evoluci lze chápat i jako neustálé šlechtění genetické informace, které probíhá ve skocích. Proč tomu tak je a že evoluce je vždy mnohem rychlejší a účinnější, pokud k ní dochází v konkurenčním prostředí, zdůvodnili J. Feynmann a W. D. Hillis. Po období stagnace, v němž se přeskupují schémata a navenek (fenotypově) se zdánlivě nic neděje, následuje vždy bouřlivá výměna generací, kdy se prosadí schopnější jedinci na úkor těch méně úspěšných. Evoluce tak zároveň prosazuje robustní a stabilní řešení a postupuje při vynalézání jaksi „zdola nahoru". Vesměs platí, že nový, složitější postup je kombinací několika jednodušších, ověřených a fungujících. Výsledné chování má zpravidla, jak již bylo řečeno, emergentní charakter (Havel, 2001), (Hillis, 1988). Tento *„samoorganizující"* přístup je v protikladu k *„inženýrskému přístupu"* klasické umělé inteligence. Ta, ve snaze přenášet postupně něco ze schopností člověka na agenty – roboty, řeší problémy „shora dolů", často s rizikem, že naslepo.

Celý dosavadní bouřlivý vývoj ALife dává tušit, že existují obecnější přírodní zákony, které nejenže dovolují, ale přímo diktují vznik života, této *„síly směřující proti proudu entropie"*. Člověk je v procesu evoluce i umělého života vlastně skoro nadbytečný; jako by platilo, že čím méně se do něj vměšuje svými „vědomostmi o problému", počátečními podmínkami a omezujícími parametry, tím lépe. Naprosto náhodně (metodou pokusů a omylů) se v procesu evoluce odzkouší, zda se povede kombinací předchozích ověřených postupů sestavit něco nového, úspěšnějšího. Jakmile se to podaří, uloží se nový objev do paměti genofondu a zastaralí, adaptace již neschopní jedinci, jsou nahrazeni (generační upgrade) novými – chytřejšími, či pružnějšími.

Shrneme-li: *Tradiční klasická (majoritní) umělá inteligence vychází z myšlenky reprezentace světa, zatímco (minoritní) přístup převzatý z výzkumů umělého života se zaměřuje přímo na akce v tomto světě.*[16]

---

[14] Fenotyp zahrnuje kromě kombinace všech vnějších vlastností a rysů jedince (vzhled apod.) i jeho chování a další vlastnosti „vnitřní", které nejsou vždy navenek vidět.

[15] Fitness je původně zastřešující pojem pro biologickou zdatnost *w* jedince produkovat (plodné) potomstvo, a to ve srovnání s ostatními členy populace.

[16] Již dnes se však oba tyto dominantní proudy začínají setkávat a vzájemně se doplňovat v nově navrhovaných hybridních architekturách řídicích systémů rozsáhlých společenství kooperujících inteligentních robotů.

### 18.2.4 Agentové technologie a umělý život

Agentové technologie jsou přirozeným rozšířením současných přístupů, jež se opírají o dílčí komponenty. Staly se nejdynamičtější a nejvíc stimulující počítačovou vědou dneška. V dnešní době už čistý výzkum v pevně vymezených oblastech, jakými jsou mobilní robotika, nanotechnologie, sociální a evoluční systémy, v podstatě neexistuje. V praxi, tj. v reálném světě, jsou mnohem zjevnější rozsáhlé mezioborové aplikace výzkumu, zahrnující a vzájemně propojující poznatky od medicíny a řízení procesů až po virtuální podnikání a e-komerci.

Agenti jako obecně opakovaně použitelné vzory budou dále ovlivňovat aplikační oblasti v mnoha směrech: Za prvé jsou určitou metaforou pro návrh velkých celků, distribuovaných výpočetních systémů. Za druhé jsou zásobárnou – zdrojem – technologií pro zmíněné počítačové systémy. Za třetí slouží jako modely i velmi složitých reálných systémů, které můžeme nalézt například v biologii a ekonomii (Lorenz, 1993). To platí obecně i pro oblast výzkumu umělého života.

Problematika inteligentních agentů i multiagentních systémů (MAS) byla v této řadě monografií UI několikrát a dostatečně dobře do hloubky prezentována (Mařík a kol., 1993), (Mařík a kol., 2001), (Mařík a kol., 2003), (Mařík a kol., 2005). Pro kontinuitu výkladu připomeňme proto jen *základní definice inteligentního agenta.*

*Inteligentní agent* je hardwarový nebo mnohem častěji softwarový počítačový systém, který vykazuje podle Wooldridge a Jenningsové (Wooldridge, 1995) v upřesnění (Wooldridge, 2002) následující vlastnosti. Je:

- *Autonomní* (Autonomy) – agent pracuje bez přímého zásahu člověka a do jisté míry řídí své akce a vnitřní stavy.
- *Sociální* (Social ability) – agent interaguje s ostatními agenty (popř. s lidmi) prostřednictvím jazyka pro komunikaci agentů.
- *Reaktivní* (Reactivity) – agent vnímá své okolní prostředí a reaguje včas na změny okolí tak, aby plnil cíle, pro něž byl navržen.[17]
- *Proaktivní* (Pro-activeness) – reakce agenta na podnět z okolí neurčuje pouze momentální stav, ale jeho chování je řízené cíli v tom smyslu, že agent může sám převzít iniciativu při řešení.

Již na konferenci Simulation of Adaptive Behavior SAB'96 „From Animals to Animates" definovala Patty Maes své agenty takto: „ *agent je definován jako umělý živočich, který může být simulován v počítači nebo včleněn do robota, tento živočich musí přežít a adaptovat se v měnícím se prostředí. Tento popis agentů sjednocuje širokou oblast zainteresovaných disciplín od umělé inteligence a robotiky po evoluční biologii, etologii, sociologii a etiku.*"

Luck (Luck, 2003)[18] definuje agenta jako počítačový systém, který je schopen autonomně a flexibilně jednat v dynamickém, nepředvídatelném a otevřeném prostředí. Tato definice je dostatečně flexibilní i na popis umělých tvorů dle Maesové.

---

[17] Prostředím může být fyzický svět, uživatel sám (cestou grafického či jiného uživatelského rozhraní), soubor dalších agentů, Internet, popř. vzájemné kombinace.

[18] V publikaci „*A Roadmap for Agent Based Computing*" vytvořené v 2003 v rámci AgentLink.

Pro úspěšnou realizaci umělého tvora, vyhovujícího uvedeným definicím, je nutná důkladná znalost jeho chování (Artkin, 1999), (Holland, 1991), (Kelemen, 1993), (Schroeder, 2001). Pro návrh a odzkoušení správné funkce jeho řídicího systému musíme v simulacích otestovat chování z různých hledisek a mít možnost jej při vývoji analyzovat. Byla proto vyvinuta řada metod a postupů pro objektivní posouzení chování fyzického robota, či obecně jen agenta (Reynolds, 1987), (Inselberg, 1991), (Mataric, 1997), (Nahodil, 2001), (Nahodil, 2002), (Nahodil, 2004), (Svrček, 2005). Byly realizovány rovněž různé typy architektur řídicích systémů (Csonto, 2002), již i s prvky umělého života (Kadlecek, 2001), (Nahodil, 2002), (Nahodil, 2005). Součástí našeho výzkumu byla i vizualizace chování namodelovaných agentů s možností zpětné analýzy tohoto chování (Řehoř, 2003), (Kohout, 2004), (Nahodil, 2004), (Kadleček, 2004), (Duda, 2005), (Foltýn, 2005).

Protože problematika virtuálních bytostí je v 5. kapitole této knihy uceleně již zpracována Cyrilem Bromem (Brom, 2005), (Brom a kol., 2006), vizualizaci chování umělých bytostí se zde dále nevěnuji. Vede mě k tomu nejen riziko možných duplicit, ale i větší konzistence vlastního výkladu a jeho limitování.

## 18.3 Anticipace a anticipační systémy

### 18.3.1 Srovnání anticipačního chování s reaktivním

Již od konce dvacátého století profitují studie lidského chování na zjištění, že naše *chování je zaměřeno na cíle* (goal oriented). V našem každodenním životě ho výrazně ovlivňuje anticipace[19], jejíž vliv se sám o sobě projevuje nejen v tom, že si sami umíme stanovovat dlouhodobé cíle, ale také, že činíme účelná krátkodobá rozhodnutí. Například tím, že si jako dlouhodobý cíl dáme získat akademickou hodnost, anticipujeme vyšší příjem v budoucnosti. Co se týče příkladu krátkodobého rozhodnutí, anticipace ukojení hladu nás přiměje k jídlu, anticipace pohodlnějšího místa pro odpočinek nám napoví, abychom si lehli raději do postele, než na zem. Krátkodobá rozhodnutí ovlivňují naše jednání velmi silně, (ne nutně jen vědomá) anticipace upadnutí nás například nutí chodit po kluzkém povrchu mnohem opatrněji (Nahodil, 2005), (Kohout, 2006). Anticipace je přítomna ve všech sférách našeho života jako jeho důležitá složka. Doplňuje podstatu reakce o proaktivní kvalitu života.[20] Povaha se vyvíjí nepřetržitým anticipačním chováním zaměřeným na přežití[21]. Někdy si my lidé anticipaci uvědomujeme, pokud předjímáme záměrně. Pokud se však procesy kdesi uvnitř našeho těla a mysli odehrají dřív, než je rozpoznáme, tak si ji ani neuvědomíme. Je tomu tak například i v tenise, kde odpal míčku profesionálního podání může být úspěšný jen díky anticipačnímu mechanismu. Zpracování vědomé reakce by vzhledem ke krátkému trvání akce trvalo příliš dlouho. Anti-

[19] Anticipovat (to anticipate) neznamená jen předvídat, resp. předjímat budoucí události, ale též se podle toho, co se předjímá, i účelně chovat.

[20] Reakci (agenta nebo živočicha) na podnět z okolí neurčuje pouze momentální stav, ale jeho chování je *řízené cíli* v tom smyslu, že on sám může převzít iniciativu při řešení.

[21] Tento mechanismus demonstruje dynamika kmenových buněk. Dělením základní kmenové buňky je možné dostat nekonečné množství různých biologických vzezření, forem a mutací.

cipace stojí i v pozadí kreativity v umění a designu. Rovněž pro jakoukoliv prevenci už ze samé její podstaty jsou anticipační mechanismy nezbytné.

Anticipace je i v oblasti umělé inteligenci chápána pomocí koncepce agenta přijímajícího rozhodnutí založená na předpovědích, očekávání nebo důvěře, že budoucí stav nastane. Je nesporné, že anticipace je nezbytnou složkou složitých, přírodních rozpoznávacích systémů. Opakem anticipace je vlastně reaktivní chování.

Ukážeme si zásadní rozdíl mezi těmito dvěma zcela odlišnými koncepcemi:

Elementární formy umělé inteligence mohou být jak známo, konstruovány na principu, opírajícím se o jednoduchá rozhodovací pravidla typu *if-then-else* (Mařík, 1993). Jako příklad takového systému, by mohl být agent, řídící se tímto pravidlem

*IF venku prší* **THEN** *vezmi si deštník.*
**ELSE** *nech deštník doma*

Definovaný *systém je z podstaty reaktivní*, protože přijetí rozhodnutí je založeno na současném stavu prostředí, bez ohledu na budoucnost.

Naopak jiný *agent používající anticipaci*, by se pokusil predikovat budoucí stav prostředí (v tomto případě počasí) a využít predikce při svém rozhodnutí:

*IF je nebe zamračené* **AND** *tlak vzduchu je nízký* **THEN** *bude pravděpodobně brzy pršet, takže si vezmi deštník s sebou.*
**ELSE** *nech deštník doma.*

Jednoduchá metoda predikce je založena na předpokladu, že i blízká budoucnost bude stejná jako nedávná minulost a přítomnost. Běžným příkladem je pak výrok „*zítra bude přibližně stejné počasí jako dnes*". Tato primitivní metoda předpovídání počasí je správná jen v přibližně 70 % případů. Nicméně, pravděpodobnost správnosti předpovědi počasí založené na tomto „modelu beze změny" klesá tím rychleji, čím delší je čas do anticipovaného okamžiku. Předpověď na delší časové období vyžaduje vzít v úvahu nejen současný stav systému, ale i míru, s jakou se mění. O něco lepší metoda předpovědi je založena na předpokladu, že „míra růstu či poklesu je konstantní." Takovýto model předpovídání je často používán například v demografii. Obecně, předpovědi na delší časová období vyžadují rostoucí komplexnost modelu, pomocí kterého jsou budoucí hodnoty predikovány.

### 18.3.2 Anticipace – základní pojmy a definice

Základní definici anticipujících systémů publikoval americký teoretický biolog Robert Rosen ve své stěžejní knize *Anticipatory systems* (Rosen, 1985):

„*Anticipující systémy jsou systémy, které obsahují model sebe sama a popřípadě svého okolí, na tomto modelu predikují své budoucí možné stavy a na základě této predikce ovlivňují své okamžité aktivity tak, aby jejich budoucí stav odpovídal jejich okamžitým záměrům.*"

R. Rosen vycházel při tom ze svých pozorování živých organismů, zvláště těch, které řadíme mezi vyšší živočichy. Inspirovala ho jejich schopnost předvídat vývoj situace v budoucnu na základě jakéhosi modelu této situace v nich samých. Na tomto modelu sebe sama a okolního světa zjišťují, co všechno by se vůbec mohlo stát, a podle toho

účelně upravují své okamžité reakce tak, aby budoucí stavy odpovídaly jejich záměrům v té chvíli. Tato schopnost živočichů byla známa samozřejmě již mnohem dříve, avšak teprve Rosen z těchto poznatků vytvořil základy teorie, kterou zamýšlel zobecnit pro nejrůznější systémy. Rysy anticipace objevil totiž i u takových „neživých" systémů, kde se to naprosto nedalo očekávat.

K rozšíření definice anticipujících systémů došlo ale až krátce po Rosenově smrti. Belgický matematik a inženýr D. M. Dubois (Dubois, 1999) svým výzkumem poukázal na to, že anticipaci můžeme pozorovat dokonce i v těch systémech, kde vytváření modelů podle Rosenovy definice není jaksi z principu vůbec možné. Jedná se spíše o systémy fyzikální než technické či biologické, zato však v širokém rozpětí – od systémů z mikrosvěta kvantové fyziky, přes laboratorní objekty až po makrosystémy vzdálených světů, o které se dnes zajímají astronomové a astrofyzici. Na základě velmi citlivého rozboru matematických popisů takových systémů se i v těchto oblastech objevují nezpochybnitelné aspekty anticipace. Takové systémy anticipují „sami ze sebe", aniž by nějaký model měly. Sám Dubois upozornil na dva takové fenomény, a to na elektromagnetické systémy (Dubois, 1999) a v roce 2003 na gravitační systémy. U obou jejich anticipace spočívá v tom, že, ačkoliv by se změna elektromagnetického resp. gravitačního pole měla šířit konečnou rychlostí (např. až rychlostí světla), uvedené systémy změnu anticipují, neboť na ni reagují, jako by se k jejich místům dostávala okamžitě.[22]

Z hlediska vědecké metodologie je ovšem třeba vyloučit předpoklad, že by například atomy, elektromagnetické systémy (ať již v laboratoři či v průmyslovém robotu) nebo celé galaxie měly k dispozici modely, z nichž by vycházely při predikci svých budoucích možných stavů. Pracují tedy bez modelu. Proto Dubois zavedl definici *„anticipujícího systému v silném slova smyslu"*, která předpokládá, že takový anticipující systém nemá model, ale anticipuje „sám ze sebe" (Strong Anticipation Systems). K tomu, aby anticipoval, si nemusí žádný model vytvářet, protože ten je explicitně součástí jejich struktury (Events Built by or Embedded in a System). Objekty, vytvářející si model sebe i okolního světa (do této skupiny patří všechny systémy, které splňují základní Rosenovu definici anticipujících systémů), se nyní nazývají *„anticipujícími systémy ve slabém slova smyslu"* (Weak Anticipations Systems).

Výzkum anticipace není zaměřen pouze na fyziku, i když zmíněné, podivuhodně se chovající objekty mají charakter spíše fyzikální než technický či biologický. Bylo dokázáno[23], že anticipace existuje i v živých systémech, které nejsou biology považovány za inteligentní. Japonští vědci referovali nedávno o experimentech s anticipujícím společenství améb a neváhali mluvit i o jakési jejich mimosmyslové komunikaci. Tito primitivní jednobuněční živočichové postupně formovali takovou strukturu, která jim umožňovala v budoucnosti vždy naprosto optimální přístup k potravě (Dubois, 1999).

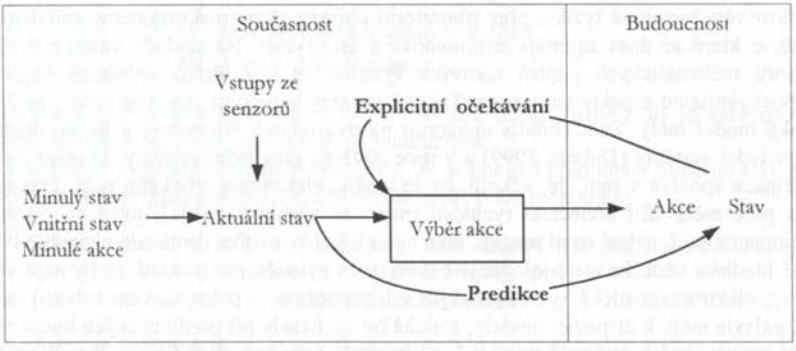
Dalším významným zjištěním je, že téměř každý systém vytvořený, či alespoň ovlivňovaný, člověkem, resp. lidskou společností je vlastně anticipující. Lidské záměry ostatně

[22] Příklad anticipujícího gravitačního systému je i v naší sluneční soustavě. Protože Slunce dostává signály o Merkurově poloze nejvýše rychlostí světla *c*, měl by být *pohyb Merkurova perihelia* (dávný problém astronomů; využit při ověření obecné teorie relativity) větší, než je pozorován. Přijmeme-li předpoklad, že nejen světlo, ale i gravitace se šíří rychlostí světla, pak Slunce – řečeno populárně – musí pohyb Merkura předvídat a podle toho na něj ihned působit.

[23] Mezinárodní konference CASYS (*Computing Anticipatory SYStems)* v létech 1998–2006, periodikum *International Journal of Computing Anticipatory Systems* atd.

vždy směřují spíše do budoucnosti. Proto jakýkoliv výrobní, dopravní, úřední, sociální či jiný systém lidmi sestavený či aspoň jimi řízený, který by vycházel jen ze současného stavu, či nejvýše ze své historie, a ne z alespoň částečné anticipace své budoucnosti (s ohledem na okolnosti), je dětinsky naivní, nefunkční a vlastně nemyslitelný.

Jinými slovy, vytvářené dynamické systémy jsou velmi často anticipující. Pokud ne, jde buď o velmi podstatnou abstrakci reality, a nebo o výjimečný jev. Tento fakt byl zvláště v technickém světě vždy jistým způsobem intuitivně vnímán, avšak teprve postulovaný a systematický výzkum anticipujících systémů posledních dvaceti let dal této představě pevnou racionální formu.




Obr. 18.4 Základní schéma anticipačního chování.

Schéma na obr. 18.4 ukazuje základní myšlenku anticipačního chování. Agent v aktuálním stavu predikuje stav následující a bere v úvahu tento predikovaný stav při výběru akcí.

V prostředí, které obsahuje více agentů s anticipačním chováním, můžeme do predikce budoucího stavu zahrnout i anticipační chování okolních agentů. Takovouto anticipaci lze nazvat *anticipací druhého řádu*. Jinými slovy, anticipujeme ve věci anticipujícího systému. Mluvíme pak o *vnořování anticipace* (Kindler, 2001), (Krivy, 2002), (Butz, 2001). Takto bychom mohli vnořovat stále hlouběji a modelovat anticipaci libovolného kladného řádu. V praxi se však něco, co by odpovídalo řádu tři a většímu, objeví až v budoucnosti. Některé programovací prostředky takovou anticipaci připouštějí, ale je jich mizivě málo. Nejvhodnější se zdá být SIMULA (Simula, 1986).

### 18.3.3 Rozdělení anticipačního chování

Jestliže chceme rozdělit anticipační chování, je nejprve nutné definovat, v jakém prostředí se bude agent s tímto anticipačním chováním pohybovat (Kohout, 2006).

Prostředí definujme jako částečně pozorovatelný Markovův rozhodovací proces POMDP (Partially Observable Markov Decision Process) – šesticí $\langle X, Y, U, T, O, R \rangle$, kde:

$X$ je množina stavů prostředí,

$Y$ je množina možných senzorických vjemů,

$U$ je množina možných akcí, které je možné na prostředí vykonat,

$T : X \times U \rightarrow \Pi(X)$ je přechodová funkce stavů, kde $\Pi(X)$ je množina všech rozložení pravděpodobností množiny $X$,

$O$: $X \rightarrow \Pi(Y)$ je pozorovatel, kde $\Pi(Y)$ je množina všech rozložení pravděpodobností množiny $Y$,

$\mathbf{R} : X \times U \times X \rightarrow \Re^r$ je funkce bezprostřední odměny, kde $r$ je počet kritérií.

V případě, že je splněna Markovova podmínka, která říká, že *„účinek vykonané akce závisí pouze na vstupech"*, jedná se o Markovův rozhodovací proces MDP (Markov Decision Process).

V případě, že částečně pozorovatelný Markovův rozhodovací proces POMDP popisuje každý možný senzorický vjem daného stavu jednoznačně a jedinečně, redukuje se POMDP na MDP. Jinak řečeno, všechny možné senzorické vjemy stavu $x$ ( tzn. všechny $y \in Y$, pro které je pravděpodobnost $O(x)$ větší než nula) jsou možné pouze v tomto stavu. Pokud senzorická pozorování jednoznačně nepopisují pouze jeden stav prostředí, Markovova podmínka je porušena a prostředí spadá do úloh tzv. nemarkovovského problému (Non-Markov problem). V takovém případě optimální volba akce nezávisí pouze na zprostředkovaných senzorických informacích, ale také na předešlých percepcích, akcích a užitku.

V prostředí částečně pozorovatelných Markovových rozhodovacích procesů POMDP rozeznáváme čtyři typy anticipujících agentů, které popíšeme v následujících odstavcích.

### *18.3.3.1 Implicitní anticipace*

První typ agentů s implicitní anticipací (Implicit Anticipation) nevytváří o svých budoucích stavech vůbec žádné predikce, které by jejich budoucí rozhodování a jednání mohly nějak ovlivnit. Senzorické vstupy (doplněné popřípadě informacemi i o vnitřním stavu) jsou propojeny přímo s aktuátory (výkonnými akčními členy) agenta. Predikční model agenta je prázdný (agent si nevytváří žádné předpoklady), nebo na rozhodování o výběru vhodné akce nemají žádný vliv. Navíc u agentů tohoto typu anticipace nemůžeme porovnávat výsledky akcí. Implicitní anticipace je napevno zakódována přímo v řídicím systému, tedy hluboko v programu, který řídí chování agenta.

V přírodě mají už ze své podstaty všechny živé bytosti (a to dokonce i ty, které se chovají čistě reaktivně) v sobě zakódované informace pro implicitní anticipaci. Evoluce sama je vlastně implicitní anticipační mechanismus, který implicitní anticipaci vtiskuje do genů. Je nutné poznamenat, že toto dosti hrubé chápání pojmu anticipace přiřazuje každé živé formě implicitní anticipaci.

Implicitní anticipace je známa a velmi dobře popsána, avšak z hlediska adaptace jsou mnohem významnější ostatní druhy anticipací, které již předpokládají i jisté zabudované mechanismy učení.

### *18.3.3.2 Anticipace odměny*

Pokud agent při rozhodování, kterou akci vykoná, bere v úvahu očekávání odměny (ať již pozitivní nebo negativní) za svou činnost v různých akcích, můžeme ho nazvat agentem s anticipací odměny (Reward Anticipation). Význam jednotlivých akcí je pak kvanti-

fikován a je tak možné provádět mezi jednotlivými akcemi porovnávání z hlediska očekávaného přínosu či naopak ztrát.

#### *18.3.3.3 Anticipace senzorické informace*

Anticipace údajů získaných smyslovými orgány (Anticipation of Sensor's Information) neovlivňuje chování agenta přímo, ale přes subsystémy, které senzorické údaje zpracovají. Predikce budoucích stavů a tím i predikce budoucích podnětů, tak vlastně ovlivňuje senzorický vjem v nejbližších okamžicích.

Aby mohl být agent schopen takovéto predikce, musí použít predikční model prostředí (ten není nutné vytvářet v plném rozsahu). Očekávané senzorické vstupy mohou být zpracovány rychleji než ty neočekávané a agent na ně může zareagovat rychleji. Příkladem je hrozba možného ohrožení. Stejně jako v přírodě, když agent očekává útok, je daleko pozornější a ve chvíli, kdy je pak opravdu napaden, je připraven správně jednat.

#### *18.3.3.4 Anticipace stavů*

Jeden z nejzajímavějších typů anticipace je anticipace stavů (Anticipation of States), kdy agent vytváří explicitní predikce budoucích stavů. Tyto predikce pak přímo ovlivňují jeho rozhodování o výběru následného chování. Obdobně jako u anticipace senzorické informace i zde musí existovat zažitý predikční model, anebo musí být dodatečným naučením zpřesněn. Anticipace stavů umožňuje díky existujícímu vnitřnímu modelu plánovat vývoj situace do více či méně vzdálené budoucnosti.

### 18.3.4 Možné aplikace anticipačních přístupů v praxi

Studium anticipačních systémů splňujících základní Rosenovu definici vyžaduje dostatečně komplexní model systému. Ten musí zahrnovat jak znázornění systému a jeho vazeb, tak všech okolností v rámci celku. Tyto systémy již nemodelují vnější svět, nýbrž se zaobírají vnitřním znázorněním svých relevantních prostředí ve smyslu rozsahu možných budoucích variant vývoje. Možnost anticipace systémů může být považována za analytický důsledek komplexnosti modelu. Tato dodatečná komplexnost není při využívání dimenze času považována za jednou provždy danou, nýbrž jen za další stupeň volnosti, který je systému právě dostupný. Tento stupeň volnosti dovolí zřejmě aktivní adaptaci na změny v prostředí prostřednictvím simulované anticipace s obráceným tokem času. Výzkum anticipačních systémů významně čerpá z pokroku v oblasti umělé inteligence, stejně jako z prohlubujících se znalostí biologického života a emergujícího života umělého. Současnost se vyznačuje rostoucím zájmem o anticipující systémy a především o jejich potenciální aplikace sahající od výroby a řízení dopravy až po inteligentní agenty na Internetu.

Přestože anticipační systémy byly a jsou studovány množstvím vědců v kontextu matematické biologie, etologie, teorie automatů a kybernetiky,[24] teprve až nedávný nástup moderních výpočetních a informačních technologií umožnil jejich nasazení pro komplexní regulaci a anticipované řízení systémů. Intenzivní výzkum s prvními úspěšnými realizacemi probíhá v mnoha oblastech. Automobilový průmysl je jednou z nich. Anti-

[24] Epistemologické otázky se týkají našeho poznání: co můžeme vůbec vědět o přirozeném myšlení, jakými cestami se o něm dovídat a jaké poučení lze čekat od umělého modelování. Robert Rosen zdůrazňoval, že epistemologické kořeny vědomé anticipace sahají možná až k Aristotelově vnímání kauzality.

cipace reakcí řidičů, která je založena na vzorcích chování naučených v minulosti, kombinovaných s odhady trajektorií jízdy a chování ostatních účastníků silničního provozu, může výrazně pomoci v předcházení kolizí a prevenci nehod (Anticipated Collision Avoidance). Jinou oblastí aplikace může být ekonomie a burza. Doposud byly odhady dalšího vývoje cen akcií vytvářeny pouze lidmi – experty. Anticipační systémy mohou být použity k objevení doposud skrytých kauzálních vazeb mezi příčinou a důsledkem.[25]

Ochrana lidských životů a majetku je další z velkých oblastí, kde anticipace může pomoci. Předpovídání zemětřesení, vulkanických aktivit nebo vzniku a pohybu hurikánů může zachraňovat životy. Intenzivně se pracuje na překvapivě účinných aplikacích v rámci boje proti terorismu a ochraně majetku a životů i jedinců obecně.

### 18.3.5 Učící se klasifikační systémy LCS

Pokud chceme uvést ACS (Anticipatory Classifier Systems) v jeho plném kontextu, je třeba nejdříve se zmínit o LCS (Learning Classifier Systems) (Holland, 1976) jako o jednom z odvětví strojového učení, do kterého ACS spadá. LCS pro svou činnost spojují metody posilovaného učení (Reinforcement Learning) a evolučního programování (Evolutionary Programming) a představují produkční systémy, které generují a testují pravidla a spravují svou bázi znalostí tak, aby byla vždy konzistentní s popisovaným světem (Mařík, 1993). Učení a adaptace v LCS lze rozdělit do několika oblastí:

- znalostní struktury,
- rozhodování,
- produkce nových pravidel,
- zobecňování těchto pravidel,
- učení v prostředí se zpožděnou odměnou.

*Generovaná pravidla* jsou opět ve standardním formátu

**IF** *podmínka* **THEN** *proveď akci*,

přičemž nezbytná zpětná vazba z prostředí umožní určitou kvantifikaci užitku, který obdržíme aplikací tohoto pravidla.

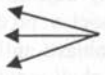
Stručně zopakujme princip učících se klasifikačních systémů: Klasifikační systém obsahuje *seznam klasifikátorů* (obr. 18.5). *Klasifikátor* realizuje generovaná pravidla typu *„podmínka – akce"*. Na začátku je tento seznam inicializován ručně navrženými pravidly. Klasifikátor se skládá ze tří částí: *Podmínková část* určuje, pro kterou kombinaci senzorických vjemů je klasifikátor aplikovatelný. Kriteriální část *Fitness* ohodnocuje velikost zisku, který může přinést vykonání akce z 2. části, nazývané *Akční zásah.*

Každý řádek seznamu představuje jeden klasifikátor. Podmínkovou část představuje řetězec znaků z množiny {0, 1, #}. Jednotlivé znaky představují:

0 – vypnuto (senzor není aktivován),
1 – zapnuto (senzor je aktivován),
# – nezáleží (nezáleží na hodnotě tohoto senzoru).

[25] Z cenového vývoje byť jen jediné komodity na trhu lze odvodit a využitím anticipačních mechanismů vyšších řádů v konečných důsledcích ovlivnit či využít budoucího vývoje cen ostatních komodit.

| Podmínková část | Akční zásah | Fitness |
|---|---|---|
| 1001 | 01 | 0,8 |
| 1#11 | 10 | 5,6 |
| 0#10 | 10 | 2,0 |




Obr. 18.5 Příklad seznamu klasifikátorů systému LCS.

Důležitý je význam znaku „#", neboť je to právě on, díky němuž může být pravidlo generalizováno. Například klasifikátor s podmínkovou částí 1#11 může být aktivován, pokud je senzorický vstup mapován na hodnoty 1011 anebo 1111. Při vytváření klasifikátorů na základě senzorických dat se může stát, že se vytvoří více klasifikátorů najednou. Agent se potom rozhoduje pro výhodnější variantu, podobně jako živočich v přírodě. Učení LCS probíhá v podstatě dvěma různými způsoby. Prvním způsobem se v každém časovém okamžiku, kdy je vykonána nějaká akce, upravuje (v závislosti na velikosti a polaritě odměny) hodnota Fitness pro daný klasifikátor. Druhý způsob učení spočívá ve vytváření nových klasifikátorů evolučními výpočetními technikami (např. genetickými algoritmy) a odstranění z paměti těch, které mají hodnotu Fitness malou, podrobněji například (Lažanský, 2003).

LCS je typickým představitelem řešení pro anticipující agenty typu *anticipace odměny* (viz odst. 18.3.3). První implementace LCS se nazývala *CS1* (Holland, 1975), nejnovější pak *XCS* (Wilson, 1995).

### 18.3.6 Anticipační třídicí systémy ACS

O několik desítek let dříve, než byl navržen první systém LCS, psycholog E. Tolman ve své práci (Tolman, 1932), zabývající se latentním učením u živočichů a člověka, navrhl tzv. **SRE** útvary. Tyto útvary reprezentují základní znalost organismu o tom, že pokud bude na situaci **S** (State) reagovat reakcí **R** (Reaction), bude mít tato za následek efekt **E** (Effect). Edward Tolman ale nijak blíže nespecifikoval, jak by k takovému učení mělo docházet.

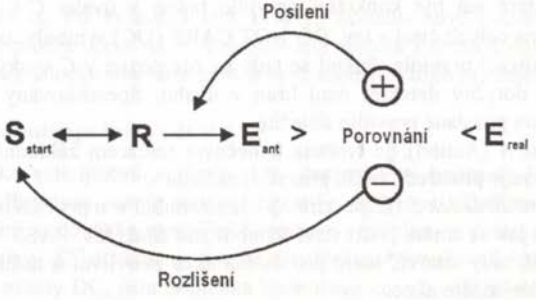
K. Hoffman poté formuloval mechanismus učení (Hoffman, 1993), který nazval *„Chování řízené anticipací"* (Anticipatory Behavioral Control). Celá teorie je založena na předpokladu, že určujícím faktorem účelného chování je znalost jeho důsledků. Konsekvence konkrétního chování je závislá na situaci, ve které se realizuje. Je tedy důležité vědět, k jakému důsledku vede reakce **R** na danou situaci **S**.

Každá akce **R** je svázána s očekávaným (předjímaným) efektem. Tato vazba je posilována v závislosti na výsledku porovnání anticipovaného a skutečného efektu. Pokud je spolehlivost anticipace podmíněna výchozím stavem situace je vytvořena nová **SRE** jednotka.

Anticipační třídicí systém ACS, obr. 18.6, navržený W. Stolzmannem, se od tradičního LCS liší především v tom, že jako určitou formu anticipace zavádí cyklickou predikci příštího stavu prostředí (Stolzmann, 2000). Každé pravidlo pak vedle podmínkové a akční části specifikuje i očekávaný příští stav

**IF** *podmínka* **THEN DO** *akce* **AND EXPECT** *příští stav.*

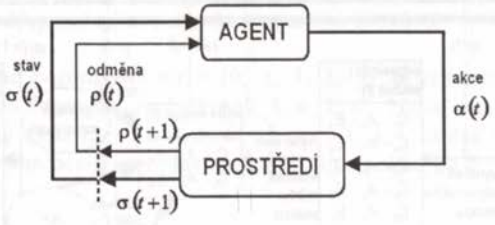
V ACS má pak každé pravidlo přiřazeno dvě kvalitativní hodnoty. Jednak je to již zmiňovaný *očekávaný efekt* (užitek či ztráta) a nově potom *kvalita predikce*, což je vlastně schopnost pravidla správně předpovědět, které atributy prostředí vybraná akce ovlivní. V kontextu základů etologie (Lorenz, 1993) se dá říct, že ACS vlastně realizuje všechny hlavní druhy podmiňování (tj. klasické, operační a apetenční operační) a vyhasínání.[26]




Obr. 18.6 Chování řízené anticipací (Stolzmann, 2000).

### *18.3.6.1 Model prostředí vytvářený ACS*

Předpokládejme anticipačního agenta, pohybujícího se ve světě, kde vývojový čas běží v diskrétních krocích. Interakcí s předem neznámým prostředím se agent postupně učí jeho strukturu. Každá akce, kterou agent provede a která má potenciál změnit některé atributy jeho prostředí, přivede agenta do nové situace (nového stavu).




Obr.18.7 Interakce ACS s okolím (Stolzmann, 2000).

Na obr. 18.7 je načrtnuto schéma interakce agenta s okolím v rámci ACS. V čase $t$ agent vnímá současný stav $\sigma_t \in I = \{N_1, N_2, \ldots, N_m\}^n$, kde $m$ značí počet možných hodnot konkrétního atributu a $n$ je pak počet detekovaných atributů prostředí. Jednotlivé atributy nemusí být nutně kódovány binárně, ale mohou nabývat jen diskrétních hodnot. Agent může získat zpětně informaci od prostředí $p_t \in \Re$ a na základě pozorování pak v souladu

[26] ACS v průběhu času od prvotního Stolzmannova návrhu prošel řadou změn. Jeho současná verze se v literatuře někdy označuje i jako ACS2. Zde hovořím vždy jen o ACS, i když popisuji již jeho novější verzi.

se svou vnitřní strategií vybere nejlepší akci $a_t \in A = \{a_1, a_2, \ldots, a_w\}$, která má být provedena v příštím kroku.

Aparát, kterým agent činí pozorování, sestává ze sady ***n*** detektorů. Každý z nich dává informaci o konkrétním atributu prostředí. Vytvářená pravidla mají šest hlavních parametrů, kterými jsou:

*Podmínková část pravidla C* (Condition) – je to soubor obsahující prvky, určující podmínku, za které má být konkrétní pravidlo bráno v úvahu $C \in \{1, \#\}^n$. Oborem hodnot je množina celých čísel a tzv. DO NOT CARE (DC) symboly, označené # , které umožňují generalizaci pravidla. Pokud se tedy na *i*-té pozici v *C* vyskytuje symbol DC, znamená to, že dotyčný detektor není brán v úvahu. Specifikovány jsou pak jen ty atributy, které jsou pro dané pravidlo důležité.

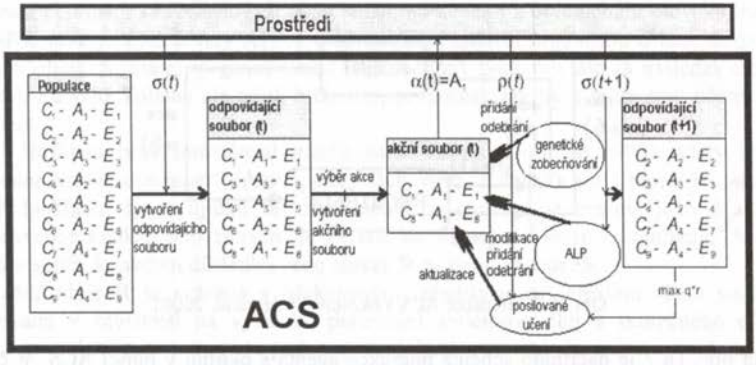
*Akční část A* (Action), je tvořena konečným řetězcem základních akcí, kterými agent ovlivňuje svoje prostředí, takže přesněji „aktivita".

*Očekávaný důsledek E* (Expectation) – je to soubor o *n* prvcích určujících, ve kterých atributech a jak se změní příští stav. Symbol zde značí tzv. PASS THROUGH (PT) atributy prostředí, tedy takové, které provedená akce neovlivní a tudíž zůstanou stejné jako před provedením této akce.

*Kvalita* $q_t \in (0, 1)$ pravidla, je hodnota, která se zvyšuje. Pokud bylo pravidlo aplikováno správně, vedlo k očekávanému efektu. Je to vlastně odhad pravděpodobnosti, že po provedení akce $a$ nastane očekávaný stav.

*Predikce okamžitého výnosu Ir* (Immediate Reward Prediction) specifikuje očekávanou okamžitou odměnu jako zpětnou vazbu od prostředí.

*Predikce dlouhodobého výnosu R* odhaduje výnos nějaké dlouhodobé strategie, uskutečňované agentem.




Obr. 18.8 Jeden cyklus interakce ACS s okolím (Butz, 2001).

Pravidlo, jehož kvalita přesáhne práh spolehlivosti (Reliability Threshold), je bráno jako spolehlivé a je zařazeno do interního modelu prostředí. Naopak, pokud kvalita klasifikátoru podteče práh nespolehlivosti (Inadequacy Threshold), je pravidlo vymazáno.

Schéma na obr. 18.8 slouží jako ilustrace hlavní myšlenky Butzova ACS algoritmu (Butz, 2001).

V čase $t$ učiní agent pozorování $\sigma_t$ o stavu $S_t$, ve kterém se právě nachází. $S_t$ je porovnán s podmínkovými částmi všech pravidel v agentově znalostní databázi – v literatuře je používán pojem *populace pravidel* [$\mathscr{P}$]. Klasifikátory z [$\mathscr{P}$], jejichž podmínkové části $C$ odpovídají $S_t$, tvoří tzv. odpovídající soubor (Match Set) [$\mathscr{M}$]. V souladu s vnitřní strategií agenta je vybrána další akce $\alpha_t$. Obvykle se používá *ε-reedy Algoritmus*. Akční soubor (Action Set) [$\mathscr{A}$] je pak tvořen těmi pravidly z [$\mathscr{M}$], která navrhují vybranou $\alpha_t$. Po vykonání této akce je učiněno nové pozorování, na jehož základě proběhne proces učení ALP, je aplikována některá z technik posilovaného učení a proběhne správa znalostní databáze genetickým zobecňováním pravidel.

### 18.3.7 Příklady anticipačního učení ALP

Základní myšlenka anticipačního učení ALP (Anticipatory Learning Process) vychází z již zmíněné Hoffmanovy *teorie chování řízeného anticipací* (Hoffman 1993). Proces slouží k rozšiřování, testování a specifikaci agentova vnitřního modelu prostředí.

Predikovaný stav $S_{t+1}^{\text{ant}}$ dostaneme tak, že na všechna místa, kde $E$ část použitého pravidla obsahuje symboly DC, jsou doplněna hodnotami korespondujících atributů ze $\sigma_{t+1}$. Anticipační učení ALP je aplikováno na každý klasifikátor v akčním souboru [$\mathscr{A}$] zvlášť.

Nejdříve je třeba vyřešit otázku, jestli byla predikce nového stavu správná. Odpověď dostaneme porovnáním $S_{t+1}^{\text{ant}}$ s reálným stavem $S_{t+1}$.

Pokud bylo anticipováno správně, nastal tzv. očekávaný výsledek (Expected Case). Kvalita pravidla je pak zvýšena v závislosti na parametru $\beta \in (0, 1)$, což reprezentuje tempo učení (Learning Rate)

$$q_{\text{updated}} = q + \beta(1 - q),$$

kde $q$ je zmíněná kvalita pravidla. Větší tempo učení zanáší do učícího procesu šum, ale zároveň pomáhá rychlejší adaptaci na změny v okolním prostředí.

**Příklad 1**. Učiníme pozorování $\sigma(t) = \{0, 1, 1, 1, 0\}$ a vybereme pravidlo, jehož podmínková část mu odpovídá, např. $C = \{0, \#, \#, 1, \#\}$. Nechť toto pravidlo navrhuje akci $\alpha$ a očekávaný efekt $E = \{1, \#, \#, 0, \#\}$, tedy 1. a 4. atribut se změní a ostatní zůstanou stejné. Potom očekávaný stav nastane jedině, pokud po provedení akce agent učinil pozorování

$$\sigma(t + 1) = \{1, 1, 1, 0, 0\}.$$

V případě, že nový stav nesplňuje očekávání, hovoří se o tzv. neočekávaném výsledku (Unexpected Case) a ohodnocení kvality klasifikátoru je následně sníženo

$$q_{\text{updated}} = q - \beta q .$$

V procesu anticipačního učení je dále třeba rozlišit dva případy: korigovatelný a nekorigovatelný. *Korigovatelný případ* nastává v momentě, kdy všechny atributy $C$ a $E$, které $S_{t+1}^{\text{ant}}$ nepředpovídá správně, jsou symboly DC. Jinými slovy jsou to ty atributy prostředí, které se navzdory předpovědi změnily. V takové situaci je možné pomocí souboru $M$ dodatečně specifikovat podmínkovou část nedostatečně fungujícího pravidla tak, aby příště

nebylo v obdobném případě bráno vůbec v potaz. Současně je třeba vytvořit nový klasifikátor, který odpovídá současnému stavu. V *nekorigovatelném případě* je pouze vytvořeno nové pravidlo.

**Příklad 2.** Korigovatelnému neočekávanému efektu z prvního příkladu by odpovídalo pozorování například

$$\sigma(t+1) = \{1, 1, 0, 0, 0\}.$$

Nekorigovatelný efekt by nastal například tehdy, když by platilo

$$\sigma(t+1) = \{1, 1, 1, 0, 0\}.$$

*Aktualizaci odhadu frekvence adaptace* dostaneme použitím přístupu MAMT (Moyenne Adapté Modifié Technique), který zaručuje, že nová pravidla (Updated Rules) jsou adaptována častěji než pravidla starší. Vychází z empirie, že pravidlo, které je po čas svého života často měněno, patrně není s popisovaným světem konzistentní. Je proto pravděpodobné, že bude při genetickém zobecňování vyřazeno z populace. Podrobný popis algoritmu lze najít například již v materiálech SAB'94 (Venturini, 1994).

Převratné anticipační projevy kvantové fyziky, astrofyziky, astronomie a kosmologie ovlivní nesporně vývoj mnoha dalších oborů, umělou inteligenci nevyjímaje. Intenzivně jsou studovány ale nejen systémy, které anticipují pomocí svých modelů implementovaných na počítačích, tedy zejména pomocí simulačních modelů. Anticipace bude v kombinaci s ostatními technikami hrát již v blízkém budoucnu nesporně klíčovou roli při návrzích a realizacích anticipačního chování umělých stvoření.

Trend současné civilizace převést lidský faktor (mj. lidskou představivost, kalkulování a předvídání) na počítače, automaty, roboty podstatně zvyšuje význam výzkumu a kvalitního návrhu anticipujících umělých systémů pro celou společnost. Skutečně, anticipující systémy jsou stále více a více ve středu zájmu.

## 18.4 Budoucnost koexistence člověk–stroj, bezpečnostní rizika

Umělý život usiluje o pochopení základních vlastností života. Jak tvořit živé systémy, jaké mají tyto systémy vlastnosti, zda se dají tyto vlastnosti aplikovat na některé inženýrské problémy. Umělý život by mohl objasnit průběh evoluce (*život-jaký-ho-známe*) a vznik inteligentního života na Zemi. Nesporně pozoruhodná jsou slova Karla Čapka pronesená při premiéře R.U.R.: „*Nesmíte si myslet, že vývoj, který dal vzniknout našemu životu byl jedinou vývojovou možností na této planetě.*"

Nevěřím, že umělá inteligence povede v důsledku svých aplikací, opírajících se o počítač, nutně k degradaci důležitosti člověka oproti výpočetní technice a sofistikovaným strojům obecně. Určitě se ale dočkáme protestů, někdy možná i násilných, proti technologiím, o nichž budou lidé přesvědčeni, že nás degradují. Že nás řadí do stejné kategorie s jednoduchými, lidmi manipulovatelnými výtvory. Faktem je, že ve špičko-

vých světových laboratořích, kde vědci sestavovali donedávna roboty především z „oceli a křemíku", se už dnes sestavují první roboti z „oceli, křemíku a živých buněk". Na předních pracovištích už umíme kultivovat svalové buňky, které pak v těchto zatím jednoduchých umělých mechanismech fungují coby ovladače a akční členy. Za několik málo let budou na stejném principu sériově vyráběny protézy, které budou zcela organickou součástí poškozených lidských těl. Velmi nadějný je *výzkum kmenových buněk* z kostní dřeně či embryonálních buněk (Syková, 2004), ve kterém patříme ke světové špičce. „Zjednodušeně řečeno, dopravíme-li kmenové buňky na poškozené místo v těle, mohou se tam z nich vyvinout například buňky chrupavky, jater, plic nebo jiného orgánu, dokonce i buňky nervové," říká prof. Syková. Pro pacienty s poraněnou míchou nebo po infarktu se tak jejich život může zásadně zkvalitnit. Obrovskou nadějí jsou i pro pacienty, kteří čekají na transplantaci orgánu. V místě postiženého orgánu by se mohly vyvinout v orgán nový nebo by povzbudily starý, poškozený orgán k regeneraci. Naději mají i nemocní s roztroušenou sklerózou, Parkinsonovou a Alzheimerovou chorobou.

Také proces stále rychlejšího začleňování původně jen „křemíkových" a „ocelových" prvků do lidského těla už dávno začal. První pacienty, kteří si tyto technologie nechali implementovat, k tomu vedly klinické důvody. Snažili se prostě kompenzovat svá těžká tělesná postižení. Již dlouho máme kardiostimulátory a umělé kyčle (totální endoprotézy). Objevila se i umělá srdce, umělé ledviny, plíce, ale dnes se běžnou záležitostí stávají i daleko složitější nervové protézy. Desetitisíce lidí s vážně poškozeným sluchem nosí v uchu (v části zvané kochlea – hlemýžď) natrvalo implantovaná zařízení. Ty generují až v 10 kmitočtových pásmech signály jakožto přímou nervovou stimulaci hlemýždě, který tyto frekvence ve zdravém uchu normálně zachycuje. Tito lidé slyší prostřednictvím přímé elektronické stimulace svých periferních neuronů, přesněji prostřednictvím kombinace křemíkových a organických nervových spojení.

Existuje spousta dalších experimentů s implantováním „křemíkových a ocelových prvků" lidem se zdravotními potížemi. Jsou vyráběny přístroje určené k procvičování svalů pacientů po mozkové příhodě nebo s poškozenou páteří atd. Ukazuje se, že adaptabilita kritických oblastí lidského mozku je značná a to dává ochrnutým pacientům velkou naději do budoucna. Nepotrvá totiž dlouho a tyto zatím převážně klinické procedury se začnou uplatňovat v běžném životě. Rodney Brooks z MIT odhaduje, že již během příštích deseti až dvaceti let dojde k zásadnímu kulturně sociálnímu posunu. Lidé začnou postupně vnímat robotiku, informační technologie, křemík a ocel ve svém těle ne jako nucenou náhražku, ale jako cestu k lepšímu pochopení a dokonalejší interakci s okolním světem. Možná, že si pak i někteří vidomí, zcela zdraví lidé nechají do těla implantovat mechanismy umožňující jim vidět i v infračerveném nebo ultrafialovém spektru. Existují futurologické představy, že svět bude jednou globálním superinteligentním kyberprostorem. Jeho zárodkem je dnešní Internet. Nevěřím, že by s ním mohl být člověk pro zvýšení svých intelektuální schopností a znalostí již brzy bezdrátově propojen (přes rozhraní) přímo do mozku. Kdoví však, kam to posune *web sémantický*, na jehož široké nasazení se zatím čeká.

Za zcela reálná pokládám obdobná vylepšování člověka biologického charakteru, profitující z očekávaného prudkého rozmachu genetického inženýrství, evoluční biologie a kvalitativního posunu v bio- a nanotechnologiích již v příštích desetiletích.

Nárůst intelektuální výkonnosti naší civilizace bude pokračovat právě co nejužším spojením a vzájemným provázáním specifik inteligence člověka a umělé „inteligence

stroje" do výrazně efektivnějšího celku. Jsem proto přesvědčen, že lidská civilizace 21. století rozhodně nebude kopírovat jen zjednodušené vize autorů současných příběhů sci-fi a katastrofických filmů, kde dominantní postavení a nadvládu nad naším světem přebírají kyborgové či nadlidsky výkonným umělým mozkem vybavení roboti – k nerozeznání podobní člověku.

Samozřejmě nelze zcela vyloučit, že inteligentní stroje někdy ohrozí člověka. Z dnešního pohledu však neumíme objektivně posoudit, co by v budoucnu takovým „ohrožením člověka" mohlo být. Vždyť lidé už tím, že proměňují své prostředí, mění také sami sebe a své postoje. Často se tak už nebojí toho, co dříve naši předci považovali za smrtelná nebezpečí, např. auta, letadla. A čím dál častěji své úhlavní nepřátele (divoké šelmy aj.) dokonce ochraňujeme. Tak jako jsme postupně novým dopravním prostředkům přizpůsobili legislativu i uspořádání sídlišť a krajiny, budeme si zřejmě zvykat i na mnohem komplikovanější a autonomnější stroje. Jejich vývoj prošel za posledních padesát let dlouhou cestu. Dnes již stavíme roboty s prvky umělého života. Roboti se vyvíjejí přibližně milionkrát rychleji, než jsme se vyvíjeli my lidé. Ještě v polovině 90. let minulého století bylo jejich vědomí jen zhruba na úrovni hmyzu. To už je ale velký skok z pozice, kde byli před padesáti léty. Tehdy byli srovnatelní s jednobuněčnými organismy. Kdyby se i nadále vyvíjeli milionkrát rychleji než druh *Homo sapiens* (a všechno tomu nasvědčuje), trvalo by jim to teoreticky už jen 50 dalších let, aby nás plně dostihli a byli nám svými schopnostmi ve všech oblastech rovni nebo dokonce nadřazeni. Kevin Warvick říká, že „... pro tyto inteligentní stroje budou lidé pak jen tím, čím jsou nyní zvířata pro mnohé z nás, tedy téměř bez významu" (Warwick, 2005). Nabízí se čapkovská otázka: Budou nás někdy roboti ovládat? Já si to nemyslím.

Přesto možnost, že tyto umělé organismy se stanou konkurenty lidí, zde jistě existuje, především díky tomu, že *každý organismus (i umělý) se snaží dosáhnout své cíle.* Evoluce umělých živočichů bude sledovat především cíle svého „živočišného druhu" a ne cíle a očekávání svého „stvořitele". Již dnes se pracuje na legislativě upravující pravidla komunikace s umělým světem.[27] Skutečně nebezpečné by pro nás mohly být zejména takové formy syntetického života, které s námi sdílí společný fyzický prostor (hardware, wetware aj.) a využívají ty samé zdroje, např. energetické (Havel, 2001). Dokonalé *kontrolní a bezpečnostní mechanismy* musíme mít proto k dispozici výrazně dříve, než vývoj ALife umožní takovým formám *samoreprodukci*, či dokonce *evoluci s otevřeným koncem*, což jsou vlastnosti všech živých systémů. Ostatně už Norbert Wiener upozorňoval na možné riziko nevypnutelnosti budoucích kybernetických strojů. Zřejmě proto nikdy nebudou sestrojovány umělé bytosti, které by člověk nemohl sám vypnout. I když kupříkladu robot-voják, který by byl snadno vypnutelný, by tím byl i velmi zranitelný. A jak by to dopadlo s jadernými elektrárnami,[28] kdybychom připustili vypnutí jejich automatických řídicích systémů a všech systémů ochran? Možná budou roboti nesnadno vypnutelní právě proto, že po jejich vypnutí my, lidé, nebudeme už schopni plně nahradit jejich funkce (Hoffmann, 1993), (Nahodil, 2005).

Pomyslným klíčem od Pandořiny skříňky umělého života je nesporně jeho samostatný spontánní vývoj – *umělá evoluce* (Warwick, 2005). S profesorem Warwickem souhlasím v tom, že „vyšlechtěním" současného *Homo sapiens* příroda evoluci zdaleka neskončila.

[27] Zákaz výroby programů s implementovaným pudem sebezáchovy, tedy mimo jiné nevypnutelných.
[28] Ukrajina, Černobyl, 30. 4. 1986

V této souvislosti je na místě připomenout, že vznik nového biologického druhu vůbec neznamená, že bude úspěšný. Biologie zná mnoho příkladů: bílý žralok[29] je velmi starý a úspěšný produkt biologické evoluce, který přežil ohromující počet mnohem mladších výstupů Darwinova mechanismu přirozeného vývoje.

Už v blízké budoucnosti dojde nesporně k fúzi biotechnologií, nanotechnologií a informačních technologií do jedné jediné disciplíny. Dá se očekávat masivní nasazování robotů téměř ve všech oblastech a bude nutné řešit i etické otázky týkající se zneužití robotů. Bylo by naivní se domnívat, že nikdy nedojde ke zneužití robotů k útoku na jiného člověka. Vždyť člověk dokázal proti svému bližnímu obrátit snad každý svůj produkt – počínaje nástrojem a konče náboženstvím. A ve vyhledávání způsobů, jak použít to, co mělo být původně pro člověka a ne proti němu, je více než velmi kreativní. Dnes jsme svědky toho, jak armády ve válečných konfliktech nasazují „inteligentní" robotické zbraně. Morálnost tohoto nasazení je odvozena pouze od morální oprávněnosti celého konfliktu. Ačkoliv dnešní inteligence robotů zdaleka nedostačuje pro posuzování morálních otázek jejich vlastního chování, pro likvidaci nepřítele již dostatečná je (Nahodil, 2002b). Nejhorší je, že armády jsou pouze „vrcholem ledovce". Jak budou pronikat roboti do našeho života, budou vznikat další příležitosti k jejich zneužití. Už dnes nás začíná v této souvislosti zasahovat obecnější problém – jak si udržet dostatečnou úroveň soukromí a svobody života bez neustálého dohledu nejrůznějších plně automatizovaných a robotizovaných zařízení, kamerami počínajíce. Má to sice dnes ještě převahu výhod – znásilnění, loupežná přepadení a jiné zločiny jsou pro pachatele, snímané (pro ně nečekaně) kamerami v „přímém přenosu" stále nebezpečnějším podnikem. Ale už brzy bude možné technicky sledovat člověka skutečně doslova všude, kam se hne.[30] A čím dál, tím více sofistifikovaní roboti budou prostupovat všemi oblastmi. Jsem ale přesvědčen, že to přispěje především ke zvýšení kvality našeho žití, včetně naší bezpečnosti, tolik žádoucí ve světě teroristických útoků a rasové a náboženské nesnášenlivosti. Věřím totiž, že se podaří nalézt *optimální kombinaci zákonů a zvyklostí*, která by účinně zabránila vládám i jedincům v nepřípustném nahlížení do našeho soukromí zneužitím techniky. A obecně vzato, sám robot nemá žádný primární důvod ublížit člověku. K tomu může dojít buď jen lidskou nedbalostí, nebo úmyslem. Je to tedy opět především otázka etiky a morálky v nás, v jednotlivcích i v celém lidském společenství, aby se tak nestalo.

## Literatura

Artkin R. C.: *Behavior-Based Robotics*. Cambridge-Massachusetts: MIT Press, 1999.

Brom C., Lukavský J., Šerý O., Poch T., Šafrata P.: *Affordances and Level-of-Detail AI for Virtual Humans*. In: Proceedings of Game Set and Match 2, The Netherlands, Delft, 2006.

Brom C.: *Hierarchical Reactive Planning: Where is its limit?* In: Proc. of Modelling Natural Action Selection (Bryson J. J., Prescott T. J., Seth A. K., eds), Edinburgh, 2005, s. 235–242.

Brooks R. A.: *Intelligence without Reason*. In: L. Steels and R.A. Brooks (eds) The Artificial Life Route to AI. Lawrence Erlbaum Assoc., Hillsdale, NJ, 1995, s. 25–81.

Brooks R. A.: *Intelligence without Representation*. In: Artificial Intellig. 47, 1991, s. 139–159.

[29] Jeho evoluce se téměř zastavila již před přibližně 130 miliony let.

[30] Soukromí a svoboda bez obtěžování jsou základem „osvícených" demokracií. Jsou-li tato práva porušena, záhy se objeví tyranie. Visí tak nad námi všemi stín Orvellova románu *1984*.

Brooks R. A.: *How to build complete creatures rather than isolated cognitive simulators.* In: K. van Lehn (ed.): Arch for Intellig., Lawrence Erlbaum Assoc, Hillsdale, NJ, 1991, s. 25–39.

Brooks R. A.: *Robust Layered Control System for a Mobile Robot.* In: IEEE Journal of Robotics and Automation, Vol. 2, No. 1, 1986, s. 14–23.

Butz V. M., Sigaud O., Gepard P.: *Internal Models and Anticipations in Adaptive Learning System.* In: Anticipatory Behavior in Adaptive Learning Systems: Butz, M.V., Sigaud, O., Gérard, P. (eds),: Springer-Verlag, Berlín, 2003.

Butz M. V., Stolzmann W.: An Algorithmic Description of ACS2. In *Advances in Learning Classifier Systems: 4th International Workshop* 2001 IWLCS, Springer-Verlag, Berlín, Heidelberg 2001.

Connell J. H.: *Minimalist Mobile Robotics – A Colony Archtecture for a Mobile Robot.* Academia Press, N. Y., 1990.

Csonto J., Palko M.: *Umelý život.* 1. vyd. ELFA, Košice 2002.

Dierk F.: *Etologie.* 2. vyd., Karolinum, Praha 2002.

Dubois D. M.: *Review of Incursive, Hyperincursive and Anticipatory Systems – Foundation of Anticipation in Electromagnetism.* In: D. M. Dubois (ed.), Computing Anticipatory Systems CASYS'99 Woodbury, NY: American Institute of Physics, Vol. 517, 1999, s. 3–30.

Duda F.: *Roboti – agenti v jednoduchém prostředí společného cíle.* Diplomová práce - vedoucí DP Nahodil, P., dipl. práce: ČVUT v Praze, FEL, kat. kybernetiky, Praha 2005. 55 s. + CD.

Foltýn L.: *Realization of Intelligent Agents Architecture for Artificial Life Domain.* diplomová práce - vedoucí Nahodil P.: České vysoké učení technické v Praze, Fakulta elektrotechnická, katedra kybernetiky, Praha 2005, 66 s. + CD s přílohami.

Franck D.: *Verhaltensbiologie-Einführung in die Ethologie.* Thieme Verlag Stuttgart, 1985.

Havel I. M.: *Přirozené a umělé myšlení jako filozofický problém.* In: Mařík V. a kol.: Umělá inteligence 3. Academia, Praha 2001, s. 17–75.

Havel I. M.: *Causal domain and emergent rationality.* In: B. Smith, B. Broguard (eds): Rationality and Irrationality. Proc. of Wittgenstein Symposium, Kirchberg, Rakousko, 2000.

Havel I. M.: *Artificial Thought and Emergent Mind.* In: Proceedings IJCAI'93, Morgan Kaufman Professional Book Center, Denver, USA 1993, s. 758–766.

Hillis D. W.: *Intelligence as an Emergent Behavior*, Daedalus, Journal of the American Academy of Arts and Sciences, Special Issue on Artificial Intelligence, 1988 s. 175–189.

Hoffmann K.: *Vorhersage und Erkenntnis.*: Hogrefe, Göttingen 1993.

Holland J. H.: *Adaptation in Natural and Artificial Systems.* In: Progress in Theory. Biology, Rosen (ed.), Plenum, Londýn 1976.

Holland J. H., Inselberg A., Dimsdale B.: Parallel Coordinates: *A Tool for Visualizing Multivariate Relations.* Plenum Publishing Corporation, New York 1991.

Inselberg A., Dimsdale B.: *Parallel Coordinates: A Tool for Visualizing Multivariate Relations.* Plenum Publishing Corporation, New York 1991.

Kadleček D., Řehoř D., Nahodil P., Slavik P.: *Analysis of Virtual Agent Communities by Means of AI Techniques and Visualization.* In: Rist, T. (ed.): IVA'04 – Intelligent Virtual Agents. LNCS, Vol. 2792. Springer-Verlag, Berlín, Heidelberg, New York 2004, s. 274–282.

Kadleček D., Nahodil P.: *New Hybrid Architecture in Artificial Life Simulation.* In: Kelemen J., Sosík P. (eds): Advances in Artificial Life. ECAL 2001, LNCS Vol. 2159., Springer-Verlag, Heidelberg 2001, s. 143–148.

Kelemen J.: *Multiagent Symbol Systems and Behavior-Based Robots.* App. Artificial Intelligence 7, 1993, s. 419–432.

Kindler E.: *Anticipující systémy ve středu zájmu.* Automa, 2001, roč. 2001, č. 11, s. 23.

Kohout K.: *Anticipatory Behavior for ALife Domain.* Odborná práce v PhD - školitel Nahodil P.: ČVUT v Praze, Fakulta elektrotechnická, katedra kybernetiky, Praha 2006.

Kohout K.: *Simulace chování agentů a jeho vizualizace*, diplomová práce – vedoucí Nahodil P.: České vysoké učení technické v Praze, Fakulta elektrotechnická, katedra kybernetiky, Praha 2004, 78 s. + CD.

Krivy I., Kindler E., Tanguy A.: *Software for Simulation of Anticipatory Production Systems.* IJCAS Int. Journal of Computing Anticipatory Systems Vol. 11, 2002, s. 320–335.

Kubik A.: *Toward a Formalization of Emergence.* Artificial Life, 2003, vol. 9, no. 1, s. 41–65.

Langton C. G.: *Self-Reproduction in Cellular Automata.* Physica 10D, 1984, s. 135–144.

Lažanský J.: *Evoluční výpočetní techniky.* In: Umělá inteligence 3 (Mařík V., Štěpánková O., Lažanský J., eds), Academia, Praha 2001, s. 117–160.

Lažanský J., Kubalík J.: *Genetické programování a vybrané problémy evolučních výpočtů.* In: Umělá inteligence 4 (Mařík V., Štěpánková O., Lažanský J., eds), Academia, Praha 2003, s. 128–171.

Lorenz K.: *Základy etologie.* 1. vyd., Academia, Praha 1993.

Mach M. *Mechanismus dobývání znalostí o prostředí na základě chování a funkčnosti jeho dílčích objektů.* Diplomová práce, vedoucí Nahodil, P.: ČVUT v Praze, Fakulta elektrotechnická, katedra kybernetiky, Praha 2005, 59 s.+ CD.

Mařík V., Štěpánková O., Lažanský J. a kol.: *Umělá inteligence 1.* Academia, Praha 1993.

Mařík V., Štěpánková O., Lažanský J. a kol.: *Umělá inteligence 2.* Academia, Praha 1997.

Mařík V., Štěpánková O., Lažanský J. a kol.: *Umělá inteligence 3.* Academia, Praha 2001.

Mařík V., Štěpánková O., Lažanský J. a kol.: *Umělá inteligence 4.* Academia, Praha 2003.

Mataric M. J.: *Evaluation of Learning Performance of Situated Embodied Agents.* In: F. Moran et al., (eds), Advances in Artificial Life. Springer-Verlag, Berlín, 1995, s. 579–589.

Minsky M.: *Computation. Finite and Infinite Machines.* Prentice-Hall, Engelwood Cliffs, 1967.

Minsky M.: *The Society of Mind.* Simon and Schuster, New York 1986.

Nahodil P.: *From Slow Machines towards Anticipatory Creatures.* In: Mařík V., Jacovkis P., Štěpánková O., Kléma J. (eds): Interdisciplinary Aspects of Human-Machine Co-existence and Co-operation. - International Workshop „e-Golem" to the World IFAC Congress, Prague: CTU Press, 2005, s. 110–119.

Nahodil P.: *Svět umělého života a jeho vytváření.* Automatizace, roč. 48, č. 5, 2005, s. 318–323. (http://www.automatizace.cz/article.php?a=681).

Nahodil P., Slavík P. Kadleček D., Řehoř D.: *Dynamic Analysis of Agents Behaviour Combining ALife Visualization and AI* – Invited Paper. In: *LNCS.* Volume 3071/2004, Springer-Verlag, Heidelberg 2004, s. 346–359.

Nahodil P., Kadleček D.: *Animal-like Control of Intelligent Autonomous Agents.* In Proc. of MIC2004, Grindelwald, ACTA Press, Calgary 2004, s. 154–159.

Nahodil, P.: *From Intelligent Robots to Implementation of Artificial Life.* In: Proc. of CAL´02 – Cognition and Artificial Life, Milovy 2002, s. 141–153.

Nahodil P., Petrus M.: *Behavior Co-ordination in Multi-Robot Group.* In Proc. of the Int. Conf. IASTED MIC2002, ACTA Press, Innsbruck, Calgary 2002, s. 464–468.

Nahodil P., Svatoš V.: *Sense-Action Encoding in Hybrid Decision System.* In: Proc. of the International Conf. IASTED MIC2001, vol. 1, ACTA Press Innsbruck, Calgary 2001, s. 414–419.

Reynolds C.: *Flocks, Herd and Schools: a Distributed Behavior Model.* Computer Graphics, 21, (July 1987), s. 25–32.

Rosen R.: *Anticipatory System Philosophical, Mathematical and Methodological Foundations,* Pergamon Press, 1985.

Řehoř D., Kadleček D., Slavík P., Nahodil P.: *VAT- a New Approach to Multi-agent Systems Visualization.* In: Visualization, Imaging, and Image Processing VIIP 2003 vol. II, ACTA Press, Anaheim 2003, s. 849–854.

Schroeder N. – Noy, P: *Multi-agent Visualization Based on Multivariate Data.* In: Autonomous Agents 2001. ACM Press, Montreal (2001), s. 210–218.

SIMULA Standard. SIMULA a.s., Oslo, 1986.

Stolzmann W.: *An Introduction to Anticipatory Classifier Systems* In: Learning Classifier systems - Applications, Springer-Verlag, Berlín 2000 s. 175–194.

Svrček A.: *Výběr a hodnocení chování robotů – agentů*, diplomová práce – vedoucí Nahodil P.: ČVUT v Praze, FEL, katedra kybernetiky, Praha 2005, 89 s. + CD.

Syková E.: *Diffusion Properties of the Brain in Health and Disease*. In: Neurochemistry International//3.261, roč. 45, č. 4, 2004, s. 453–466,

Tolman E. C.: *Purposive Behavior in Animals and Men*. Appleton, New York1932.

Venturini G.: *Adaptation in Dynamic Environments through a Minimal Probability of Exploration*. In: Wilson, S. W. et all (eds): From Animals to Animates 3: Proc. of the Third International Conference on Simulation of Adaptive Behavior, SAB´94, MIT Press, Cambridge-Massachusetts, 1994, s. 371–381.

Veselovský Z.: *Etologie*. Biologie chování zvířat, Academia, Praha 2005.

Warwick K.: Evolution: *The End Game for Human*. In: Mařík V., Jacovkis P., Štěpánková O., Klema J. (eds): Interdisciplinary Aspects of Human-Machine Co-existence and Co-operation. Accompanying International Workshop „e-Golem" - to the World IFAC Congress, Prague: CTU Press, 2005, s. 49–58.

Warwick K.: *Úsvit robotů – soumrak lidstva*. nakladatelství Vesmír, Praha 1999.

Wilson S. W.: *Classifier Fitness Based on Accuracy*. In: Evolutionary Computation, vol. 3, no. 2, 1995, s. 149–175.

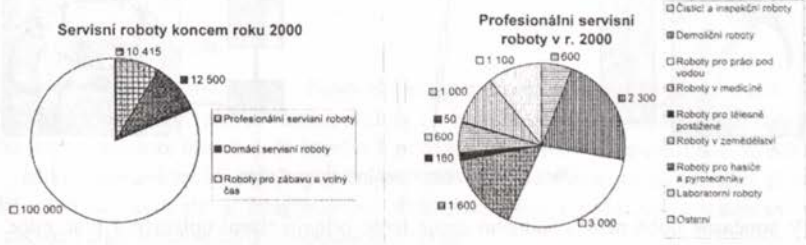
Wooldridge M.: *An Introduction to Multi-Agent Systems*. John Wiley & Sons, 2002.

Wooldridge M. J., Jennings N. R.: *Intelligent Agents*. In Lecture Notes in AI, *LNAI* Volume 890/1995, Springer-Verlag, Heidelberg 1995, s. 244–255.

# 19 Servisní robotika

## 19.1 Úvod

Robotika a umělá inteligence jsou spolu pro většinu lidí nerozlučně spojeny. Celou historií robotiky se totiž vine snaha o vytvoření umělého člověka. Je to vidět na pověstech (např. o Golemovi), mechanických napodobeninách hodinářských mistrů z 18. století i na nejmodernějších strojích, jako jsou japonské roboty ASIMO (Advanced Step in Innovative MObility) nebo QRIO (Quest for cuRIOsity). Zatímco snaha nahradit pohybové a manipulační schopnosti člověka je vcelku úspěšná, náhrada duševních schopností, umělá inteligence, je stále ve svých počátcích. Z poměrně iracionální honby za vytvořením umělé bytosti však vzniká řada velmi racionálních výsledků, robotů, zatím bez inteligence nebo jen s jejími mizivými náznaky. Nejznámější jsou samozřejmě průmyslové roboty, kterých bylo používáno v průmyslu podle statistiky UNECE (United Nations Economic Commission for Europe) na celém světě koncem roku 2000 kolem 750 000 kusů. Tatáž organizace však v témže roce uvádí již statistické údaje o jiném rozšiřujícím se typu robotů. Jde o tzv. *„servisní roboty"*. Tyto roboty jsou rozdělovány na tzv. roboty profesionální (čištění budov, čištění letadel, inspekce potrubí, demoliční roboty, bezpečnostní roboty apod.), roboty pro domácnost (vysavače, sekačky trávy apod.) a „roboty pro volný čas" (leisure and hobby). Počty používaných robotů v jednotlivých kategoriích ke konci roku 2000 jsou uvedeny v následujících grafech.



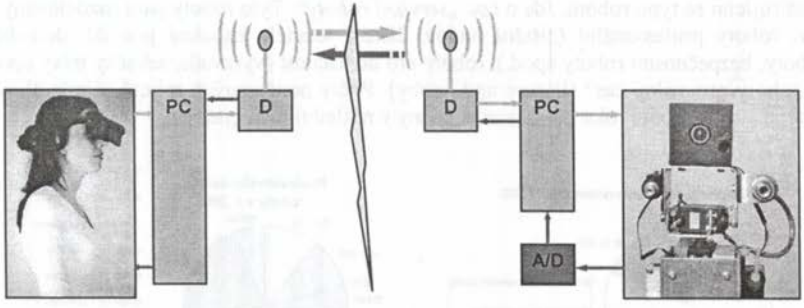

Obr. 19.1 Statistiky servisních robotů podle UNECE z r. 2000.

Jednotlivé kategorie servisních robotů nejsou bohužel ve statistikách přesněji definovány. V současné době se v kategorii servisních robotů začíná vyčleňovat skupina, pro kterou se dá použít název „roboty pro práci v otevřeném terénu" (field robotics) a do níž patří především stroje pro armádu, bezpečnost a záchranné složky. Je zřejmé, že u mnoha

strojů z oblasti servisní robotiky, vzhledem ke složitosti úkolů a prostředí, pro které jsou navrženy, nejde o stroje, které pracují naprosto autonomně. Budeme-li trvat na definici robota uvedené například v (Havel, 1981), je použití termínu robot pro takové stroje sporné. Tyto stroje vyžadují k plnění úkolu lidskou inteligenci a v mnoha případech jsou tak složité, že se u nich název robot zcela vžil. V následujícím textu se pokusíme přiblížit tři kategorie robotů, se kterými jsme měli možnost pracovat a provádět na nich výzkum. Půjde o chirurgické roboty, roboty pro armádu, bezpečnost a záchranné složky a roboty pro zábavu a volný čas.

V poslední uváděné kategorii půjde o robotický fotbal, ve kterém družstva skutečně autonomních robotů hrají hru podobnou fotbalu, s podobnými pravidly, jako má fotbal, a v současnosti i se stejným počtem hráčů jako ve skutečném fotbalu. Do hry člověk nikterak nezasahuje a o použití slova robot zde není sporu. Hra je již natolik složitá, že přímo vybízí k uplatnění prvků umělé inteligence.

U chirurgických robotů a mnohých záchranných robotů jde o tak složité činnosti, že je může v současné době vykonávat pouze člověk s pomocí robota. Jak se tedy principiálně liší takový robot od rádiem dálkově ovládaného autíčka? Kromě výrazně větší složitosti řízeného stroje jde především o způsob spolupráce, který se nazývá *teleprezence*. Jde o takový systém řízení robota člověkem, při kterém člověk dostává z prostředí, ve kterém se pohybuje robot, takové informace, jako by sám v tomto prostředí byl a pracoval. Při plné aplikaci tohoto principu dostává informace pro všechny své smysly. Robot naopak provádí všechny úkony, které v bezpečném prostoru, vzdáleném od pracovního prostoru robota, vykonává člověk.




Obr. 19.2 Princip vizuální teleprezence.

V současné době není u žádného stroje tento princip řízení uplatněn v plné míře. Je však zřejmé, že použití principu teleprezence s kombinací autonomního chování a principů umělé inteligence povede k nejrychlejším praktickým aplikacím robotiky v nestrukturovaném prostředí.

Dále se budeme věnovat detailnějším informacím o vybraných aplikacích či aplikačních oblastech. Konkrétně se jedná o

- *mikromechanický lékařský operační počítačový systém da Vinci* firmy Intuitive Surgical, Inc., používaný mimo jiné v Centru robotické chirurgie na I. Chirurgické

klinice, Fakultní nemocnice u sv. Anny v Brně – autory této části jsou L.Veverková, F.Pikeš , I..Čapov, P.Vlček, J. Doležel,

- *mobilní robotický systém Orpheus*, vyvinutý na FEKT VUT Brno, autorem příslušné části kapitoly je L. Žalud,
- *roboty pro armádu*, přehled připravený T. Neužilem a
- *robotický fotbal*, kde úvod do této zajímavé oblasti byl zpracován autorem J. Hrabcem.

## 19.2 Mikromechanický lékařský operační počítačový systém da Vinci

### 19.2.1 Robotická chirurgie

Zlomové vývojové etapy v operační technice všeobecné chirurgie jsou zřídkavé. Zavedení laparoskopie v 90. létech minulého století způsobilo doslova revoluci. Dnes není v České republice jediné chirurgické pracoviště, které by nemělo k dispozici alespoň jednu laparoskopickou jednotku. Robotická chirurgie představuje nové odvětví chirurgie vycházející z miniinvazivních operačních přístupů.

Vývoj robotických systémů pro chirurgii začal zhruba od poloviny 80. let 20. st. – tehdy šlo o projekty amerického úřadu pro vesmír a kosmonautiku NASA, který přišel s konceptem telechirurgie. Cílem bylo zajistit řešení urgentních stavů kosmonautů při meziplanetárních letech, a to právě pomocí dálkově ovládaných robotických systémů. Přibližně ve stejné době měla americká armáda podobnou představu o léčebných zásazích na vojácích zraněných v průběhu válečných a bojových operací. Na konci devadesátých let jsou vyvíjené systémy postupně uvolňovány do civilního sektoru, a tak vzniká nový obor robotické chirurgie, který bude mít v budoucnu velký význam v léčbě pomocí miniinvazivních metod. Minimálně invazivní robotika spojuje odbornou práci chirurga s moderními technologiemi – mikromechanikou, trojrozměrnou vizualizací a počítačovým řízením.

### 19.2.2 Popis systému

Systém da Vinci® firmy Intuitive Surgical, Inc., je určen k ovládání endoskopických nástrojů během výkonů ve velkých tělních dutinách v hrudníku (torakoskopické operace) a břicha (laparoskopické operace). Je navržen k použití profesionály v prostředí operačních sálů.

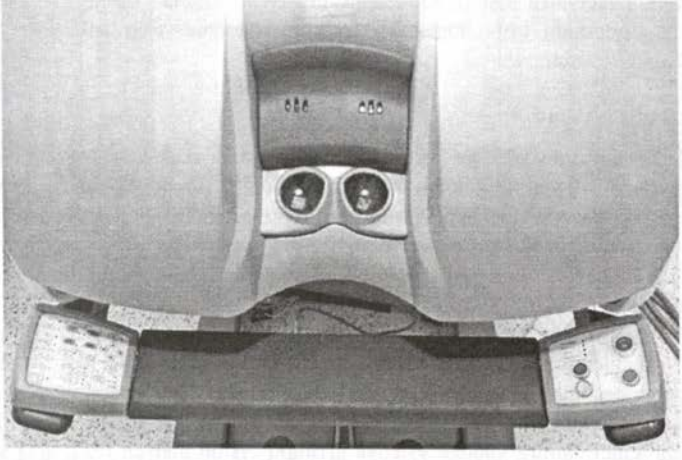
Mikromechanický lékařský operační počítačový systém daVinci se skládá z chirurgické ovládací konzoly s integrovanou obrazovou částí umožňující prostorový vjem a zařízení u pacienta se čtyřmi robotickými rameny.

Systém da Vinci, obr. 19.3 (též barev. příl.), umožňuje chirurgovi přímé řízení pohybu nástrojů v reálném čase. Používá kinematické struktury umožňující použití otevřených chirurgických technik. Tyto techniky otevřené chirurgie jsou okamžitě převáděny na pohyb nástrojů realizující zákrok technikou miniinvazivní chirurgie (MIS). Při použití systému da Vinci chirurg vhodně využívá přístupu cestou malých řezů, aniž se vzdává obratnosti, přesnosti a intuitivnosti pohybu rukou v otevřené chirurgii.

Elektronika systému da Vinci umožňuje převod velkého rozsahu pohybů rukou chirurga. Pohyby rukou jsou převáděny v nastaveném měřítku na odpovídající pohyb instrumentu v operačním poli. Odlišné nastavení umožňuje optimalizaci převodu pro různé klinické aplikace. Přirozený třes rukou chirurga je zredukován, což podporuje stabilní a předvídatelné ovládání instrumentů.

Nástroje mají celkově šest stupňů volnosti plus uchopení, o dva více než tradiční endoskopické instrumenty. Volnost pohybu čelistí napodobuje směrem nahoru či dolů i stranově flexibilitu lidské ruky. Toto kloubové řešení povyšuje možnosti operatéra

Obr. 19.3 Pohled na systém da Vinci, ©(2007) Intuitive Surgical, Inc.

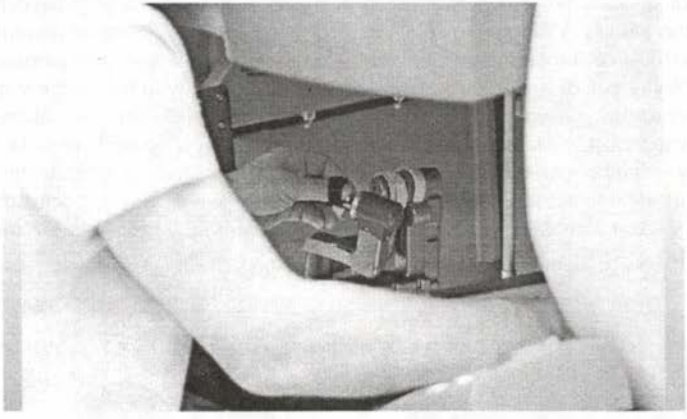


Obr. 19.4 Řídicí konzola, ©(2007) Intuitive Surgical, Inc.

miniinvazivní chirurgie na novou úroveň, která umožňuje provedení komplexní, rekonstrukční chirurgie pomocí malých řezů (incizí).

Chirurg sedí při provozu za ovládací konzolou, která je umístěna mimo sterilní operační pole. Trojrozměrný obraz operačního pole je s vysokým rozlišením zobrazován stereoskopickým systémem integrovaným v konzole.

Chirurg drží držadlo hlavních ovladačů a ovládá chirurgické instrumenty a endoskopickou kameru, které jsou připevněny na ramena chirurgického vozíku. Rozsah pohybu hlavních ovladačů umožňuje plné, přirozené pohyby rukou chirurga. Hlavní ovladače umožňují přesné převedení těchto pohybů na chirurgická ramena.



Obr. 19.5 Ovládání ramen z ovládací konzoly, ©(2007) Intuitive Surgical, Inc.

Dva nezávislé optické kanály jsou zobrazené na (CRT) monitorech, které jako součásti stereoskopického systému vytvářejí 3D obraz chirurgického pole. Systém da Vinci předává vzkazy o stavu systému pomocí textu a ikon přes stereoprohlížeč.

Chirurg a asistent ovládají rozhodující funkce systému da Vinci pomocí tlačítek a nožních spínačů. Funkce, které chirurg potřebuje mít přístupné během operování, jsou lokalizovány na konzole nožních ovladačů. Hlavní vypínač, pohotovostní zastavení, pohotovostní režim a další využívaná tlačítka jsou lokalizována na loketní opěrce, a tak jsou vždy dostupná.

Chirurgický vozík se skládá ze sloupce a základny, ke které jsou připevněny kamera a ramena instrumentů. Vozík je kabelově propojen s chirurgickou konzolou. Jakmile chirurg pohne hlavním ovladačem, ramena na chirurgickém vozíku reagují na pohyby chirurga. Každý vozík obsahuje jedno kamerové rameno, na které je připevněna kamera, a dále dvě nebo tři instrumentální ramena, která podporují plný rozsah nástrojů nabízených výrobcem. Systém má ramena čtyři – jedno kamerové a tři instrumentální.

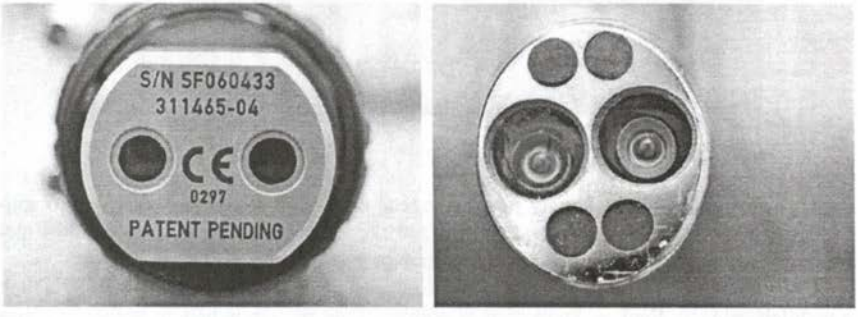
Každé rameno je připevněno na svislou kolejnici, která je napojena na sloup. Nastavením polohy v kolejnici a dalších kloubech je rameno umístěno do nejvhodnější pozice během operace. Zobrazovací rameno je vybaveno dvěma tříčipovými kamerami, které vytvářejí stereoskopický pohled na operační prostor. Další ramena nesou mikro-

chirurgické nástroje, kterými se operuje. Na konci těchto ramen jsou pohyblivé klouby, tzv. endoskopická zápěstí. Nástroje umožňují širokou škálu úkonů – stříhání, řezání, stavění krvácení, šití a další.

Nástroje mají i vlastní paměť, takže při dosažení určitého počtu zákroků se automaticky pomocí čipu zablokují. Přesný pohyb nástrojů je zajištěn pohony s krokovými motory a lankovým převodem. Díky mechanickému „zápěstí" mají nástroje vysokou pohyblivost, kterou lze srovnat s pohyblivostí ruky lidské.

### 19.2.3 Princip robotické operace

Robotická operace prováděná pomocí systému da Vinci usnadňuje provedení laparoskopického zásahu. Významnou odlišností je možnost precizního manévrování nástroji, které umožňují větší rozsah pohybů a dosah do míst, kde již klasickou laparoskopii nelze použít. Přesný pohyb ruky operatéra je přenášen na stejný pohyb nástroje v operačním poli. Operačními přístupy jsou malé řezy (do 10 mm), přes které je trokary zaveden optický systém a robotická ramena s nástroji. Nástroje jsou pomocí převodů a pohonů připojeny k počítačem řízenému mechanismu, který je ovládán operatérem. Chirurg v průběhu celé operace sedí u řídicí a ovládací konzoly. Optický systém zprostředkovává obraz s vysokou hloubkou rozlišení, zachovává prostorový 3D vjem a umožňuje výrazné zvětšení (až 15×).



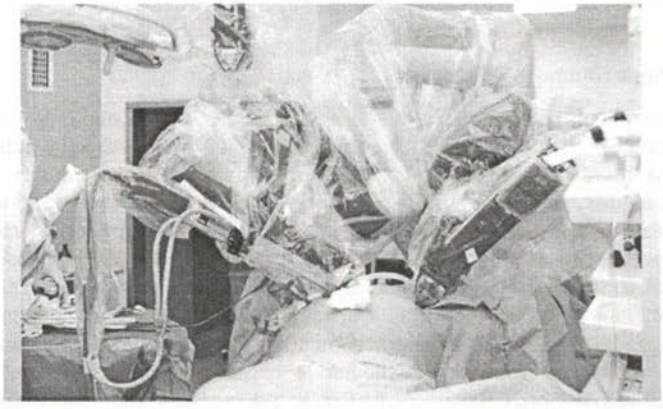

Obr. 19.6 Kamery na zobrazovacím rameni pro stereoskopický pohled na operační prostor, ©(2007) Intuitive Surgical, Inc.

Pokud chirurg odhlédne z okuláru konzoly, nástroje se automaticky zastaví. Pohyb rukou operatéra se přenáší z ovládací konzoly přes robotické rameno na nástroje v operačním poli. Všechny pohyby rukou a prstů chirurga jsou zpracovány a interpretovány počítačem tak, aby zpřesnily a zabezpečily bezpečný pohyb chirurgických nástrojů v těle pacienta. To vše přispívá k eliminaci pooperačních rizik.

Výhody použití robotického systému da Vinci jsou na straně pacienta i chirurga. Lékař nemusí vykonávat často velmi časově náročný výkon fyzicky sám a je osvobozen od negativních jevů, jako je stres, třes rukou a únava. Na druhé straně pacientům zákrok způsobí menší trauma organismu, menší ztrátu krve, menší jizvy, minimální riziko infekce a v neposlední řadě i zkrácení hospitalizace, rychlejší uzdravení a návrat do běžného života.

Operace je sice dražší než klasický výkon, avšak výsledný efekt je nevyčíslitelný. Zkrácení doby hospitalizace (až o 2/3) a minimalizace komplikací pak přináší i významný pozitivní ekonomický efekt.

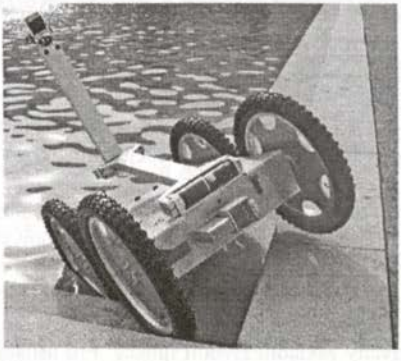
Robotickými systémy lze již v současnosti provádět více než 60 druhů operací. Centrum robotické chirurgie řeší problematiku břišní a hrudní chirurgie, cévní chirurgie, urologie a gynekologie. V době přípravy knihy bylo provedeno přes 100 operací.



Obr. 19.7 Zavedení robotických ramen s nástroji do těla pacienta, ©(2007) Intuitive Surgical, Inc.

## 19.3 Mobilní robotický systém Orpheus

Orpheus (viz obr. 19.8, též barev. příl.) je dálkově ovládaný mobilní průzkumný robotický systém určený zejména k průzkumu oblastí pro člověka nebezpečných nebo nepřístupných. Typickými aplikacemi pro takový stroj jsou: robot záchranář, pyrotechnický robot, vojenský průzkumný robot.



Obr. 19.8 Orpheus.

Robotický systém je složen ze dvou základních částí – mobilního robotu samotného a vzdáleného řídicího či operátorského stanoviště. Řízení robotu je prováděno operátorem v reálném čase, a to pomocí teleprezenčních vizuálních technik. Datový přenos je obousměrný a bezdrátový.

Systém Orpheus je vyvíjen od února 2003 na Fakultě elektrotechniky a komunikačních technologií Vysokého učení technického (FEKT VUT) v Brně a od této doby zaznamenal již tři vývojové etapy – Orpheus, Orpheus-X1 a Orpheus-X2.

### 19.3.1 Popis robotu

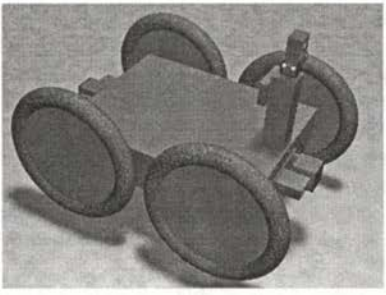
Vlastní tělo robotu má tvar kvádru o rozměrech 430 × 540 × 112 mm (š × v × h), na něj jsou připevněna čtyři shodná kola o průměru 410 mm. Pro speciální mise, kde je požadována menší výška robotu, je možné použít kola menšího průměru. Celkové maximální rozměry robotu jsou: 550 × 830 × 410 mm. Hmotnost robotu je 43 kg.

Pro pohon robotu jsou použity dva stejnosměrné 24voltové motory s integrovanou keramickou planetovou převodovkou a magnetickým inkrementálním enkodérem. Dvě kola na každé straně robotu jsou spojena ozubeným řemenem a otáčejí se tedy vždy stejným směrem a stejnou rychlostí. Chování podvozku je velmi podobné chování podvozku pásového se všemi jeho výhodami i nevýhodami. Hlavní výhodou je jednoduchost a robustnost podvozku. Výhodná je rovněž značná průchodivost terénem a vysoká stoupavost podvozku. Nevýhodou je pak zvýšená energetická spotřeba při zatáčení a nutnost dimenzovat všechny prvky pohonného systému na vyšší hodnoty. Pro robotické aplikace lze za velkou nevýhodu považovat také zatáčení podvozku smykem a tím obtížnou a nepřesnou sebelokalizaci či navigaci robotu v prostoru z dat o pohybu jednotlivých kol (diferenciální odometrie).

Robot je vybaven čtyřmi kamerami (viz obr. 19.9). Přední a zadní kamera je kompaktně spojena s tělem robotu. Tyto kamery jsou černobílé a velmi citlivé, přičemž přední kamera má infračervené přisvícení pro případ úplné tmy. Kamery jsou pohyblivé v jedné ose v rozsahu 180° a mohou tedy být nasměrovány i přímo dolů nebo vzhůru, což je výhodné například pro průzkum otvorů v zemi či podvozků vozidel (výbušniny apod.). Vrchní kamery jsou barevné kamery s vysokým rozlišením a mohou se pohybovat ve dvou stupních volnosti. Rozsahy i rychlosti pohybu hlavních kamer jsou srovnatelné s pohyby lidské hlavy, což je výhodné pro vizuální teleprezenci. Vzhledem k výšce, ve které je kamera umístěna, vzniká poměrně přirozený zrakový vjem a díky její pohyblivosti je možné stroj dobře ovládat i v těsných a složitých prostorách. Od verze Orpheus-X1 je robot vybaven manipulátorem s dalšími dvěma stupni volnosti (viz obr. 19.8), který umožňuje prodloužení dosahu hlavních kamer o 95 cm a podstatně zvyšuje univerzálnost robotu.

Robot může pracovat kromě kamer i s dalšími senzory. Jedná se například o senzor pro dálkové měření teploty. Jde o velmi přesný a citlivý průmyslový senzor pracující na principu vyhodnocování infračerveného vyzařování objektů. Tento senzor je například možné s úspěchem použít pro identifikaci stavu potenciálních obětí v případě neštěstí, protože člověk, který vykazuje zvýšenou teplotu vůči chladnému prostředí, má s velkou pravděpodobností zachovány základní životní funkce. Pro usnadnění orientace operátora je robot také vybaven miniaturním náklonoměrem se dvěma stupni volnosti, vyrobeným

technologií MEMS. Robot rovněž měří napětí baterií a je schopen předat operátorovi informace o svém stavu. Je rovněž vybaven dvěma optickými proximitními skenery, které mohou měřit vzdálenosti neprůhledných objektů do vzdálenosti asi 5 metrů. Tyto senzory mohou původně měřit pouze v jedné polorovině, jsou však upraveny tak, aby poskytovaly prostorová data z oblasti přibližně 150° horizontálně a 60° vertikálně.



Obr. 19.9 Schematické znázornění umístění kamer na robotu.

Jádro elektronického subsystému robotu tvoří vestavěný osobní počítač spolu s řídicí deskou na bázi mikrokontrolérů.

Celý robot je až na kamerový manipulátor plně symetrický, což představuje velkou výhodu zejména ve velmi těžkém terénu, kde může stroj pokračovat v činnosti i po případném převrácení. Pro tento případ je veškeré vybavení robotu pevně připevněno a je rovněž otestována funkčnost převráceného robotu.

### 19.3.2 Operátorské stanoviště

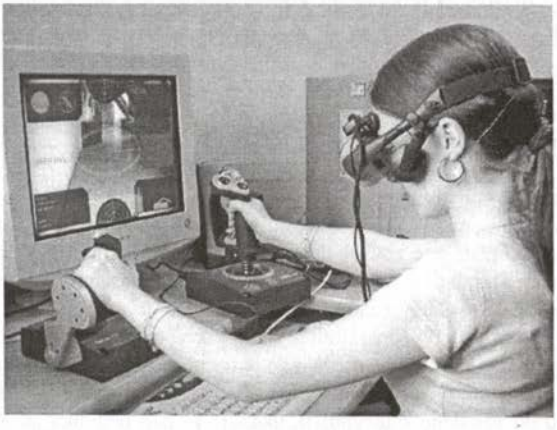
Pro ovládání robotu slouží tzv. operátorská stanice. Jde v zásadě o počítač (běžné PC nebo notebook) s ovládacím programem, komunikačními moduly a vstupními zařízeními pro ovládání funkcí robotu. Zajímavým je ovládání robotu pomocí PDA se systémem PocketPC. Zde jsou pochopitelně funkce omezeny (např. 3D zobrazení není zatím podporováno vůbec), ale přesto má operátor k dispozici poměrně plynulé ovládání s obrazem z vybrané kamery.

Pro ovládání robotu byla zvolena technika již zmiňované vizuální teleprezence. Operátor, řídící robot v reálném čase, má na hlavě helmu virtuální reality se speciálním senzorem snímajícím pohyby hlavy ve dvou nebo třech osách. Data odpovídající pohybům hlavy jsou přenesena k robotu, transformována a způsobí vychýlení pohyblivého kamerového systému přibližně způsobem odpovídajícím pohybu hlavy operátora. Obrazová data z kamery jsou digitalizována, kompresována, přenesena z robotu k operátorovi a zobrazena do zobrazovačů v jeho brýlích. Výsledek je tedy takový, že kamera na robotu kopíruje pohyby hlavy operátora a ten ve svých brýlích vidí obraz z kamery. Operátor by se tedy měl cítit být na místě, kde je robot. Přesto, že tento vjem nemůže být za současného stavu dokonalý (v ideálním případě by operátor neměl být schopen rozpoznat realitu od vzdálené skutečnosti), ukazuje se tato technika jako velice vhodná pro dálkově řízené průzkumné robotické systémy hned z několika důvodů. Nejdů-

ležitějším důvodem je snadnost a intuitivnost obsluhy pohybu kamer. U teleprezenčního ovládání kamer se totiž operátor nemusí zabývat současným natočením kamery ani způsobem jak dosáhnout nového požadovaného natočení – jednoduše otočí hlavou požadovaným směrem. Tím dochází současně k úspoře ovládacích prvků a ke značnému zjednodušení obsluhy. Velmi důležitou pro časově kritické mise je také časová nenáročnost tohoto typu obsluhy.

Pro ovládání celého robotického systému Orpheus je obvykle používána kombinace ovládání pohybů kamer pomocí senzoru na hlavě operátora a speciálního pákového ovladače pro všechny ostatní funkce robotu (viz obr. 19.10, též barev. příl.). Technika ovládání všech funkcí pomocí speciálního křížového ovladače pro obě ruce je známa z uživatelských prostředí některých stíhacích letadel, nazývá se HOTAS (Hands on Throttle and Stick). Tato technika umožňuje při vhodném návrhu ušetřit nezanedbatelné množství času během mise, protože operátor se nemusí starat o přesuny rukou mezi jednotlivými ovládacími zařízeními (křížový ovladač, klávesnice, myš). Kromě ušetřeného času během obvykle časově kritických misí se použitím této metody také zvýší koncentrace operátora. Je totiž nutné si uvědomit, že pro jmenované změny mezi ovládacími zařízeními je operátor nucen sledovat pohyby svých rukou, což způsobuje problémy zejména při použití brýlí virtuální reality. Operátor je nucen nejen přeostřit zrak na jinou vzdálenost, ale zároveň přechází mezi virtuálním prostředím v okolí robotu, které vidí v brýlích, a svým reálným okolím. Pozornost operátora je přitom během jmenovaných typů misí velmi důležitá, protože jde obvykle o mise s primárním cílem vyhledat určité objekty (např. osoby, výbušniny, toxické látky apod.) a samotný proces vyhledávání představuje zpravidla pro operátora značnou zátěž. Operátor musí mít tedy určitá data během mise neustále k dispozici, jeho pozornost by však měla být maximálně koncentrována na samotné okolí robotu.

Z množství praktických testů a experimentů na raných verzích systému Orpheus vyplynula nutnost použít jakousi rozšířenou verzi vizuální teleprezence, která byla později nazvána jako *senzoricky podporovaná vizuální teleprezence* (Sensory Supported



Obr. 19.10 Operátorská stanice.

Visual Telepresence). Tato metoda si bere jako základ popsanou vizuální teleprezenci, rozšiřuje ji však podstatným způsobem o intuitivní zobrazení dalších senzorických dat. Operátor tak má k dispozici mnohem více informací než pouhý obraz z aktivní kamery – může rovněž sledovat údaje z proximitních skenerů, teplotních čidel, čidel náklonu aj. Princip senzoricky podporované vizuální teleprezence je velmi jednoduchý. Videosignál z aktivní kamery je zobrazen přes celou zobrazovací jednotku tak, jak je tomu u vizuální teleprezence. Do obrazu jsou však přidána další data ve formě tzv. *virtuálních průhledových displejů* (VHUD – Virtual Head-up Displays). Ty je možné si představit jako průsvitky s dalšími informacemi pro operátora. Vzhledem k překrývání obrazu z kamer je podstatnou vlastností použitých VHUD jejich průhlednost, v ideálním případě by měla být zcela nezávisle definovatelná průhlednost všech objektů na obrazovce vyjma obrazu z kamery. Navíc je možné přidávat do obrazu libovolná trojrozměrná data.

### 19.3.3 ARGOS – systém pro teleprezenční ovládání

Pro ovládání systému Orpheus a dalších teleprezenčních robotických systémů byl na FEKT VUT navržen a vyvinut programový systém ARGOS. Jde o systém pro řízení robotů v reálném čase s pokročilými možnostmi zobrazení videa a dat s podporou uvedených technik vizuální teleprezence a senzoricky podporované vizuální teleprezence. Systém je naprogramován v jazyce C++ v prostředí Microsoft Windows XP pomocí knihoven Microsoft DirectX 9.0c. Pro 2D zobrazování byla na VUT vyvinuta rychlá grafická knihovna od úrovně bodu. Pro zobrazení 3D dat (např. měření proximitních skenerů) je možné použít kromě běžného zobrazení pomocí vrcholů rovněž zobrazení pomocí krychliček či rychlé zobrazení pomocí tzv. bodových sprajtů. Veškerá 3D data mohou být zobrazena jak přímo do obrazu z kamery, tak do samostatného VHUD. V současnosti je zkoumána možnost stereovizního zobrazení signálů ze dvou kamer se současným zobrazením 3D dat, přičemž každá z těchto metod již byla samostatně experimentálně vyzkoušena.

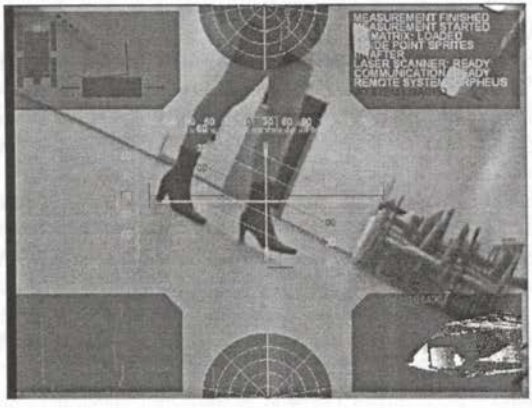
Jak bylo popsáno v předchozím výkladu, je důležitou vlastností systémů pro senzoricky podporovanou vizuální teleprezenci schopnost zobrazovat objekty překrývající videosignál s definovatelnou průhledností. V systému ARGOS může být libovolně definována průhlednost jakéhokoli 2D či 3D objektu. U 3D objektů jde o běžné vlastnosti povrchů v Direct3D, u 2D kreslení má každý bod definovatelnou průhlednost. Průhlednost u případně se překrývajících objektů různých VHUD je řešena samostatnými vrstvami každého displeje a jejich mícháním.

Další důležitou vlastností systému ARGOS je možnost téměř libovolné změny vzhledu uživatelského rozhraní kdykoli během mise. To se osvědčilo při simulacích praktických misí (např. při soutěži Robocup Rescue), kdy se jeví jako výhodné mít alespoň dva typy předdefinovaného zobrazení – jedno komplexní se značným množstvím dat a jedno zjednodušené obsahující pouze nejdůležitější informace. Typická mise pak vypadá tak, že operátor používá většinu času zjednodušené zobrazení a soustředí se na obraz z kamery a v případě potřeby přepne na komplexnější zobrazení, kde vyhledá příslušný údaj.

Ukázka jedné ze současných konfigurací systému ARGOS zobrazující data z robotu Orpheus je na obr. 19.11 – jde o přesnou kopii obrazu, který vidí operátor ve svých brýlích.

Je zde vidět praktická ukázka jmenovaného principu senzoricky podporované vizuální teleprezence, kde je na pozadí obraz z aktivní kamery a do něj jsou přidány další důležité údaje ve formě sedmi VHUD.

Pro aktuální informovanost operátora o současném stavu robotu slouží VHUD v nejdůležitější, tedy prostřední, části obrazu. Vzhledem k umístění tohoto displeje přímo doprostřed obrazu byl tento navržen tak, aby nebyl potřebný podkladový obdélník ztmavující obraz. Displej obsahuje informace o natočení aktuální kamery vzhledem k tělu robotu – viz tmavý nitkový kříž se světlým retenčním křížem. Tato informace je důležitá zejména v případech, kdy není použito ovládání kamer pomocí helmy virtuální reality, tedy například v případech, kdy její použití není možné například z prostorových důvodů. I při použití VR brýlí se však tento ukazatel velmi osvědčil pro rychlou orientaci operátora zejména vzhledem k určitému teplotnímu driftu použitého snímače pohybů hlavy operátora.




Obr. 19.11. Ukázka systému ARGOS.

Dalším, velmi důležitým, ukazatelem v centrálním VHUD je tzv. umělý horizont. Jde o intuitivní zobrazení naklonění těla robotu na základě informací z inklinometrů na jeho palubě. Tato informace se ukázala jako velice užitečná v případě ovládání robotu v těžkém terénu, kdy má operátor vlivem nedokonalosti zobrazení špatnou vizuální kontrolu o naklonění platformy. Přesnou hodnotu náklonů je možné vyčíst z posuvných stupnic v levé a horní části centrálního displeje.

V levé horní části obrazovky se nachází displej informující o poloze mechanických částí robotu. Je zde možné odečíst polohu senzorického ramena se dvěma stupni volnosti a aktuální náklon proximitních skenerů. V tomto displeji je navíc graficky zobrazena aktuální kamera.

Pravý horní displej slouží ke zobrazení textových hlášení robotu a systému ARGOS. Zprávy jsou zobrazovány na jakýsi nekonečný posuvný podklad a jejich důležitost může být znázorněna různou barvou hlášení. Tento způsob zobrazení je vhodný pro zobrazení jednorázových zpráv, je tedy například zobrazován přechod do chybového stavu, nikoli chybový stav, protože by docházelo k zahlcení tohoto informačního kanálu.

V pravém spodním displeji jsou zobrazena 3D data z předního proximitního skeneru. V tomto případě je použita zobrazovací metoda prostorových bodových sprajtů. Displej je pochopitelně rovněž možné použít pro zobrazení diskrétních 3D map, které jsou velmi vhodné pro okamžitou navigaci operátora.

Poslední z okrajových VHUD se nachází v pravém spodním rohu obrazovky a je v současnosti použit pro účely ladění. Do budoucna se počítá i se zobrazováním 2D map, které jsou přehlednější pro dlouhodobější navigaci.

Dva půlkruhové displeje nahoře a dole v prostřední části obrazu zobrazují aktuální měření z proximitních skenerů a slouží k navigaci operátora zejména za špatných světelných podmínek.

### 19.3.4 Robocup 2003

Tříčlenný tým z FEKT VUT v Brně se ve dnech 4. až 11. 7. 2003 zúčastnil světového šampionátu robotů s názvem *Robocup 2003* (viz též další výklad) konaného v italské Padově. Tým soutěžil se systémem Orpheus v robotické soutěži nazvané *Rescue Robot League* – tedy záchranářská liga. Cílem soutěžících je za pomocí libovolného počtu robotů prozkoumat v časovém limitu co největší prostor předem neznámých ruin stavby zřícené po zemětřesení. Uvnitř stavby je množství potenciálních obětí představovaných figurínami lidí. Figuríny mohou vykazovat několik známek života – např. zvýšenou teplotou, pohyb končetin, emisi $CO_2$, vydávání zvuků. Soutěžící roboty musí vytvořit mapu prostředí, zaznamenat polohu jednotlivých obětí, co nejpřesněji popsat jejich vzezření a určit jejich stav (při vědomí, bezvědomí apod.). Za každé správně určené příznaky jsou kladné body, za chybné určení jsou body odečítány. Záporné body jsou také například za nechtěný kontakt robotu s figurínou apod.

Brněnskému týmu se podařilo, navzdory pozici naprostého outsidera, v soutěži zvítězit, a to v konkurenci dalších dvanácti týmů z celého světa (např. Japonska, USA, Německa ...).

## 19.4 Roboty pro armádu

Vojenské konflikty v několika posledních létech a snaha o maximální omezení rizika pro lidské osádky vozidel, pyrotechnické jednotky a jednotky zabezpečující ostrahu objektů urychlily proniknutí technologií spojených s robotikou do armádního sektoru.

Roboty jsou využívány zvláště pro transport materiálu, průzkumné mise a střežení rozsáhlých objektů. Jsou konstruovány většinou jako teleprezenčně ovládané stroje.

Jednou z hlavní náplní práce průzkumných robotů je získávání vzorků vzduchu, půdy a vody v oblastech, u nichž je předpokládáno zamoření nebezpečnými látkami, mapování oblastí a vyhledávání nepřátelských cílů. Pro zaminované oblasti jsou využívána robotická odminovávací zařízení a mobilní robotická vozítka pro hledání nástrah a jejich ničení.

V posledních několika létech se začínají objevovat také teleprezenčně ovládané stroje osazené zbraňovými systémy.

Robot je chápán jako stroj, který je schopen vykonávat činnost jako člověk, dokáže získávat informace o prostředí a popřípadě prostředí i aktivně ovlivňovat.

Vojenský robot se této nejčastěji užívané definici vymyká. Jeho úkolem ve většině případů není nahradit člověka, ale techniku člověkem řízenou. V současné době této skutečnosti vyhovuje mnoho prostředků využívaných armádou. Spadají sem například i rakety, které korigují svoji dráhu letu v závislosti na informacích získaných ze satelitního systému GPS.

Z uvedených důvodů se pojem robot v odborné vojenské literatuře nevyskytuje příliš často. Namísto toho jsou používány spíše opisné termíny. Tyto termíny mohou zároveň posloužit pro základní klasifikaci podle prostředí, ve kterém stroje vykonávají svoji činnosti:

UGV bezpilotní (bezosádkový) pozemní prostředek (Unmanned Ground Vehicle),
UGCV bojový bezosádkový prostředek (Unmanned Ground Combat Vehicle),
RSTAV průzkumný bezosádkový prostředek (Reconnaissance, Surveillance and Target Acquisition Vehicle),
UAV létající bezpilotní prostředek (Unmanned Aerial Vehicle),
UCAV bojový bezpilotní prostředek (Unmanned Combat Aerial Vehicle),
URAV průzkumný bezpilotní prostředek (Unmanned Reconnaissance Aerial Vehicle),
USV (UUV) námořní bezosádkové plavidlo (Unmanned Sea Vehicle, Unmanned Underwater Vehicle).

V následujícím textu jsou uvedeny parametry typických zástupců pozemních robotů. V současnosti je však nejrozšířenější skupina vzdušných prostředků, které jsou využívány již od druhé světové války (projekt Mistel – rádiově naváděné bombardéry Ju 88).

Stále více se však začíná technologie bezpilotních, bezosádkových vozidel prosazovat i v pozemním vojsku, kde jsou vozidla využívána zvláště k průzkumu nebezpečných oblastí a k přepravě munice a materiálu.

### 19.4.1 Pozemní bezosádkové prostředky

Obrněná robotická vozidla (ARV – Armed Robotic Vehicles) je možné rozdělit do dvou skupin na vozidla útočná (UGCV) a vozidla určená pro průzkum, sledování a zjišťování cílů (RSTA – Reconnaissance, Surveillance, and Target Acquisition).

Obr. 19.12. Průzkumný typ bezosádkového vozidla.

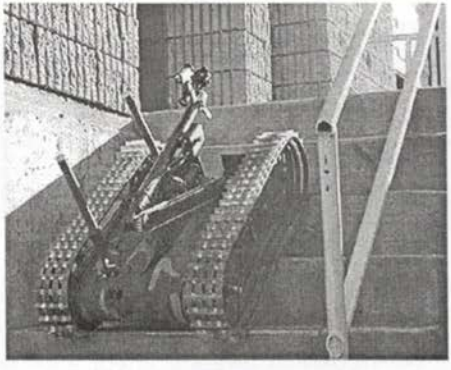
Útočná vozidla jsou schopna provádět dálkové rozmisťování snímačů, zbraní a munice do budov a bunkrů. Jsou určena pro vyhledávání překážek, bunkrů a tunelů. Mohou být využívána pro vyhodnocování stavu na bojišti, pro předávání zpráv. Útočná vozidla je možné využít jako mobilní zařízení umožňující střelbu z různých druhů zbraní a pro obsazení důležitých pozic a jejich následné palebné krytí.

Průzkumná vozidla jsou určena pro dálkový průzkum při vojenských operacích. Jsou schopna plnit stejné průzkumné úkoly jako útočná vozidla, nejsou však určena pro využívání zbraňových systémů.

### *19.4.1.1 TALON™*

Jedním ze zástupců skupiny dálkově řízených mobilních robotů aktivně využívaných ve službách armádě je robot TALON™ (www.foster-miller.com). Robot byl navržen pro manipulaci s minami a nevybuchlou municí.

Robot se skládá z pásového podvozku, který je schopen překonávat terénní překážky, je schopen zdolávat schodiště a může se pohybovat i v sypkém a kluzkém materiálu (sněhu, jemném prachu, písku apod.) Na tomto podvozku je umístěn manipulátor umožňující uchopení a manipulaci s předměty o maximální hmotnosti 11 kg. V základní verzi jsou na rameni manipulátoru umístěny kamery s možností infračerveného přisvícení.



Obr. 19.13. Průzkumná verze robotu TalonTM.

Obr. 19.14. Průzkumná verze robotu TalonTM.

Robot je řízen pomocí pákového ovladače z operátorského pracoviště, se kterým může být spojen bezdrátově, nebo pomocí optického kabelu.

Operátorské pracoviště má podobu odolného kufru, ve kterém je umístěn displej, pákový ovladač a další ovládací prvky.

V poslední době začaly být roboty TALON™ vybavovány různými zbraňovými systémy, jako jsou zadýmovací zařízení, granátomety, protitankové zbraně apod. Roboty TALON™ jsou využívány také pro průzkum člověku nebezpečných oblastí. Pro tuto úlohu je možné robot osadit snímači, jako jsou: detektor plynů, detektor chemických látek, detektor záření a snímač pro měření teploty.

### *19.4.1.2 EXPERT – PIAP*

Jedná se o mobilní robot navržený pro operace uvnitř objektů se speciálním zřetelem na dopravní systémy, jako jsou letadla, autobusy, vlaky atd. [1]. Hlavní pracovní náplní robotu je práce s nebezpečnými materiály a s nevybuchlou municí. Za tímto účelem je pásový podvozek vybaven manipulátorem, který umožňuje uchopení předmětu. Na tomto manipulátoru je umístěna kamera, pomocí které obsluha mobilní robot ovládá.

Operátorské pracoviště lze s robotem propojit bezdrátově nebo pomocí kabelu, kterým je robot napájen.

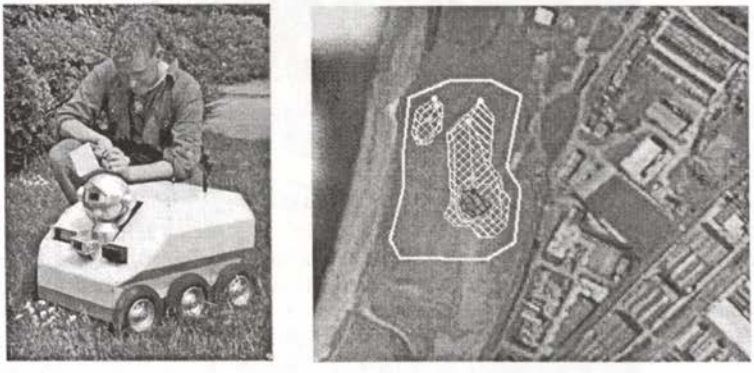
Obr. 19.15 Robot PIAP.

### *19.4.1.3 MoonBuggy*

Britská firma MoonBuggy se specializuje na dálkově řízená vozítka [7]. Ve své nabídce má celou škálu dálkově řízených robotů určených pro různé úkoly, jako jsou například dálkově ovládaná rypadla, zařízení pro konání dozoru nad určenou oblastí a podobně. Vozidla se liší nejen svou výbavou (měřicí přístroje, počet kamer, pracovní nástroj apod.), ale také pohonným systémem. V nabídce jsou vozítka s dieselovými, benzinovými a elektrickými motory. Zvláštností robotických vozítek firmy MoonBuggy je velká nosnost užitečného nákladu.

Pro monitorování a vyhodnocování nebezpečných situací je určeno vozítko s označením Mini Electric MoonBuggy UGV. Dálkově řízený robot pro průzkum zamořených oblastí – radiačně a chemicky. Výsledkem průzkumné mise jsou měření zpracovaná do podoby map zamořených oblastí. Hodnoty měření jsou zobrazovány do mapy pořízené snímkováním z letadla, popřípadě do klasické mapy terénu v digitální podobě.

Obr. 19.16 Transportní vozidlo MoonBuggy.



Obr. 19.17 Vozidlo MoonBuggy určené pro průzkum nebezpečných oblastí a měření transponované na leteckou mapu.

### *19.4.1.4 MDARS – Mobile Detection Assesment and Response System*

Jedná se o projekt americké armády a námořnictva spadající do skupiny servisní robotiky. Tento projekt se zabývá automatickými systémy proti vniknutí do objektů skladišť. Projekt je rozdělen do dvou skupin podle oblasti použití, na vnější a vnitřní.

Roboty pro vnitřní použití jsou používány v nemocnicích, skladech a kancelářských budovách. Robot je schopen se autonomně pohybovat po budově, provádět ostrahu proti vniknutí, vizuální kontrolu skladovaných předmětů a akustickou kontrolu prostředí.

Roboty pro vnější použití jsou rozšířením systému o možnost pohybovat se v částečně strukturovaném prostředí (cesty, definované oplocené prostory apod.). Jsou určeny zvláště pro využití v prostředí, jako jsou skladiště, doky a letiště.

Mechanická konstrukce je založena na vozidle s náhonem na všechna čtyři kola, poháněném dieselovým motorem. Toto vozidlo je osazeno snímači, které umožňují autonomní provoz. Pro navigaci a lokalizaci využívají vozidla senzory, jako jsou: radar, stereovizní kamerový systém a diferenciální zapojení GPS.

Celý strážní mobilní systém (obr. 19.18) je možné začlenit do skupiny společně s roboty systémů COUGAR, PerceptOR a dalších [2], [3].

Obr. 19.18 Strážní mobilní systém MDARS.

### *19.4.1.5 ECA – COBRA M.R.*

Dálkově ovládaný mobilní robot určený pro inspekci v těžko přístupných oblastech [4]. Robot je vybaven dvěma barevnými kamerami na naklápěcím zařízení, kamery jsou vybaveny přisvícením jak ve viditelné části spektra, tak i v infračervené části. V základním vybavení je také mikrofon. Jako volitelné zařízení je možné na robotický podvozek

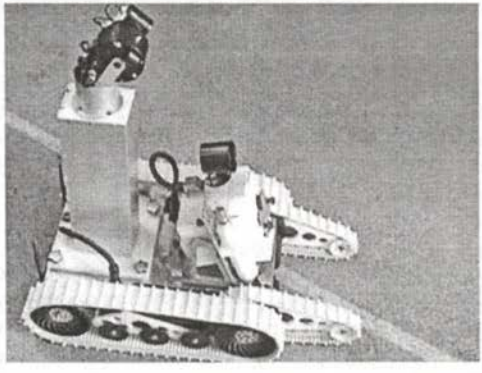
Obr. 19.19 Průzkumný robot ECA – COBRA MR.

připevnit vodní pušku. Konstrukce podvozku je symetrická a velmi robustní. Jedním ze způsobů použití je vhození robotu do sledované oblasti (např. oknem). Videosignál z robotu je bezdrátově přenášen do pracoviště operátora, které je vybaveno pákovým ovladačem a LCD displejem. Jako volitelné příslušenství lze k operátorské stanici připojit brýle virtuální reality.

### *19.4.1.6 ECA – PRM*

Je modulárním mobilním robotem pro využití při zabezpečovacích a obranných operacích [4]. Robotický systém se skládá z pásového podvozku, na který je možné připevnit požadované zařízení. Těmito zařízeními jsou:

- citlivá barevná kamera umístěná na manipulátoru umožňující přehled prostoru v 360° okolí robotu,
- mikrofon a reproduktor pro oboustrannou komunikaci,
- přisvícení ve viditelné a infračervené oblasti.



Obr. 19.20 Průzkumný robot ECA – PRM.

Mezi nesporné výhody robotu PRM patří jeho vysoká pohyblivost, schopnost šplhat do schodů a v neposlední řadě také celková nízká hmotnost zařízení (26 kg). Robot je ovládán pomocí přenosného zařízení s vestavěným LCD displejem.

Francouzská firma ECA se zabývá vývojem a výrobou několika dalších typů mobilních robotů, které jsou určeny pro využití zejména pyrotechnickými týmy při manipulaci s nevybuchlou municí.

### *19.4.1.7 RobuROC6*

RobuROC6 je mobilní platforma určená pro průzkumné a bezpečnostní operace v těžkém terénu [5]. Díky konstrukci je robotický podvozek schopen uvézt až 100 kg užitečného nákladu. Pro plnění stanovených úkolů je robot možno vybavit následujícími typy zařízení:

- kamerovými systémy – barevnými kamerami, kamerami pro noční vidění,
- ultrazvukovými snímači,
- bezdrátovou komunikací,

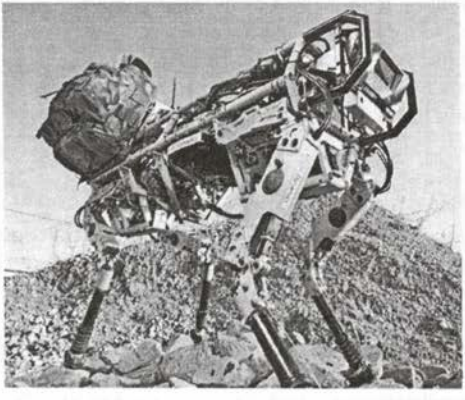
- různými typy manipulátorů,
- laserovými proximitními snímači (SICK),
- přijímači signálu GPS,
- inerciálními snímačí.

Mobilní robot je řízen z operátorského stanoviště pomocí pákového ovladače. Operátorské stanoviště je osazeno průmyslovým PC a ovládací program je možné stanovit přesně dle požadavků uživatele.

Obr. 19.21 Průzkumný robot RobuROC6.

### *19.4.1.8 BigDog*

Netradiční přístup při konstrukci robotu pro armádní účely zvolila firma Boston Dynamics [6]. Základní myšlenkou bylo vytvořit stroj pro přepravu nákladu, který je schopen se vypořádat téměř s jakýmkoliv terénem. Na tento přepravní robot jsou však zároveň kladeny vysoké nároky na jeho nosnost. Velikost robotického nosiče je srovnatelná s velikostí zvířat dodnes využívaných pro přepravu materiálu do oblastí pro stroje těžko dostupných.



Obr. 19.22 Transportní robot Big Dog.

Výsledkem vývoje je čtyřnohý kráčející robot určený pro přepravu materiálu. Robot je schopen se pohybovat v nestrukturovaném terénu, jeho svahová dostupnost činí až 35°. Maximální hmotnost nesené zátěže je 250 kg. Robot je schopen dosáhnout maximální rychlosti až 5 km/h.

Energii celému robotu dodává jednoválcový benzinový motor. Pro řízení stability a generování chůze je využíván laserový optický gyroskop a soustava akcelerometrů. Vlastní nohy robotu mají hydraulický pohon.

Robot je řízen dálkově (pomocí kabelů připojených k robotu), do budoucna se však počítá se zcela autonomní verzí, která bude mít větší nosnost a až dvojnásobnou maximální rychlost pohybu.

## 19.5 Robotický fotbal

Myšlenka robotického fotbalu vznikla v 90. létech 20. století. Jako první publikoval tuto ideu prof. Alan Mackworth z University of British Columbia v Kanadě v roce 1992. Přibližně ve stejné době vznikla v Japonsku iniciativa, která vyústila v založení Robot J-League (tedy robotická obdoba nově vznikající profesionální fotbalové ligy v Japonsku). Tomu však předcházelo ještě několik kongresů zabývajících se možnostmi využití robotického fotbalu jako seriózního prostředku pro výzkum umělé inteligence a robotiky. Vzhledem k velkému zájmu z celého světa byla rozšířena působnost této iniciativy a organizace byla přejmenována na *Robot World Cup Initiative*, neboli zkráceně *RoboCup* [13].

O několik let později, přesněji v roce 1995, prof. Jong Hwan-Kim z KAIST (Korea Advanced Institute of Science and Technology) přišel s ideou uspořádat mezinárodní soutěž v robotickém fotbale. O rok později se tak stalo, a to pod názvem MiroSot'96 (MIcroRObot World Cup Soccer Tournament). V roce 1997 byla založena organizace FIRA (Federation of International Robot-soccer Association [8]).

Uvedené organizace představují hlavní, ne však jediné, organizátory robotických fotbalových soutěží. Hlavní událostí bývá každoroční mistrovství světa. Mimo to existují kontinentální i národní soutěže. V soutěžích RoboCup platí pravidlo, že vítězný tým musí po skončení soutěže uveřejnit popis svého řešení (technickou dokumentaci).

Obě organizace pak prosazují myšlenku, že robotický fotbal je ideálním prostředkem pro rozšiřování „ducha vědy", povědomí o robotice a umělé inteligenci mezi laiky i odbornou veřejností. A také je třeba dodat, že i pro samotné týmy, zabývající se vývojem systémů robotického fotbalu, se jedná o velmi atraktivní a zábavnou oblast.

### 19.5.1 Kategorie robotického fotbalu

Robotický fotbal má několik kategorií. Pravidla se u jednotlivých organizací v detailech liší, s jistou mírou zjednodušení lze však nalézt tyto kategorie:

- systém s centrálním viděním a řízením,
- systém s autonomními agenty,
- simulační ligu,
- humanoidní ligu,
- ostatní.

Jak je zřejmé, každá z těchto kategorií má své zvláštnosti. Systém s centrálním viděním a řízením umožňuje použít konstrukčně jednodušší roboty, veškerá „inteligence" systému je soustředěna v řídicím počítači. Kategorie s autonomními agenty naopak vyžaduje roboty se složitějším uvažováním. Simulační liga má výhodu v relativně nízkých pořizovacích nákladech, simulátory jsou poskytovány zdarma ke stažení [9], [10], možnosti sehrávat utkání na dálku a nezanedbatelná není ani možnost zpětného přehrání vybraných situací i urychlení času simulace (zrychlení zápasu). Humanoidní (člověku podobné) roboty jsou zřejmě nejzajímavější, dynamicky se rozvíjející kategorií. Cílem organizace RoboCup totiž je, aby v roce 2050 tým složený z robotických humanoidních fotbalistů porazil mistra světa ve fotbale (tom „lidském"). Do kategorie ostatních lze zařadit například roboty Aibo (Sony), upravené osobní transportéry Segway aj.

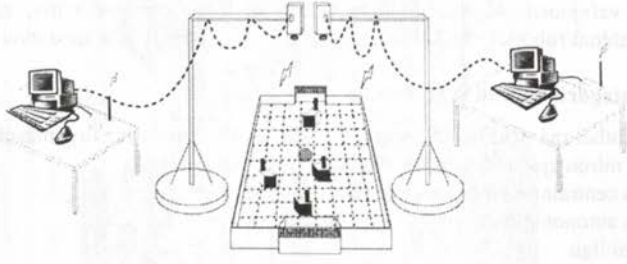
V dalším textu se zaměříme na popis robotického fotbalu kategorie FIRA MIROSOT (Micro Robot Soccer Tournament). Tato kategorie spadá do systémů s centrálním viděním i řízením. Konkrétní příklady řešení některých popisovaných problémů pocházejí od robotického týmu RoBohemia (ÚAMT FEKT VUT v Brně [11]).

### 19.5.2 Systém robotického fotbalu FIRA MIROSOT

Hrací systém kategorie FIRA MIROSOT je znázorněn na obr. 19.23. Hřiště o příslušných rozměrech (viz následující výklad) má černou barvu a je vybaveno mantinely 5 cm vysokými. Herní scéna je snímána kamerou připojenou k řídicímu počítači a poskytuje tak zpětnou vazbu pro bezdrátové řízení robotů. Volba bezdrátové komunikace je v režii týmů. Zápas se hraje s golfovým míčkem oranžové barvy. Povrch tohoto míčku je pokryt sítí důlků, a to kvůli letovým vlastnostem (doletu) při golfu. Nehomogenní povrch však negativně ovlivňuje trajektorii míčku při valení po hrací ploše (zejména při nižších rychlostech) a odrazech od mantinelů.

Důležitým aspektem je striktní zákaz zásahu lidského operátora do hry v průběhu zápasu. Pravidla jsou podobná lidské variantě hry (existují tedy fauly, penalty, ...), na jejich dodržování dohlíží rozhodčí – člověk. Délka zápasu je 2 × 5 minut čistého času, avšak celková doba utkání bývá několikanásobně delší (přerušování hry z důvodu gólu, blokování míče, time-outy, ...).

V kategorii MIROSOT existuje několik lig. Ligy se liší počtem robotů v týmu a také velikostí hřiště. V počátcích se jednalo o ligu S-MIROSOT (Single MIROSOT), která



Obr 19.23 Systém robotického fotbalu kategorie FIRA MIROSOT.

byla soubojem jednoho robotu v každém týmu. Poté byla uvedena kategorie Small League se třemi roboty v týmu (hřiště 130 × 150 cm). V dalších létech byly uvedeny i ligy Middle (s pěti roboty) a Large (se sedmi roboty v týmu). Vše směřovalo k „pravé" fotbalové jedenáctce. Kategorie má název X-Large. Ta je již velmi náročná po stránce technického vybavení. Mimo značného počtu robotů potřebných k sestavení týmu je také nutné z důvodu velkých rozměrů hřiště (400 × 280 cm) používat dva kamerové systémy nebo kameru s vysokým rozlišením a dostatečnou snímkovou frekvencí.

Každý tým má vlastní vybavení, které není standardizováno. Proto se lze setkat se širokou škálou řešení jednotlivých subsystémů.

Celý systém lze rozdělit na dvě oblasti – hardware a software. Do oblasti hardware spadá nejen technické vybavení řídicího počítače s počítačovým viděním, ale i roboty a jejich komunikace s řídicím počítačem (většinou jednosměrná – od řídicího PC směrem k robotům). Softwarovou část lze dělit na tři velké celky – počítačové vidění (zpracování obrazu), strategii rozhodování a řízení robotů.

### 19.5.3 Počítačové vidění

Pro identifikaci jednotlivých robotů pomocí systému počítačového vidění jsou určeny barevné štítky umístěné na horní části („střeše") robotů. Pravidla určují pouze minimální plochu štítku v tzv. týmové barvě (modrá nebo žlutá). Druhý tým nesmí mít na svých robotech štítky v týmové barvě soupeře. Existuje ještě jedna výjimka v barevném označení robotů, a tou je barva míčku (oranžová). Pro možnost rozlišení jednotlivých robotů navzájem jsou používány následující metody: různé tvary štítků v týmové barvě, štítky v doplňkových barvách, popř. kombinace uvedených přístupů (rozlišování tvarů štítků spolu s jejich barvami). Dalším řešením je detekce jednotlivých robotů na začátku hry a pak jejich sledování – tracking (všechny roboty pak mají stejné označení).

Počítačové vidění také vyžaduje kvalitní osvětlení hrací plochy. Mimo dostatečné intenzity se jedná zejména o stálost a rovnoměrnost osvětlení. Ne vždy jsou však tyto podmínky bezezbytku splněny, proto bývá do systému počítačového vidění implementována korekce nehomogenity osvětlení hřiště.

U této části systému je důležitá především rychlost zpracování. Snímková frekvence může totiž dosahovat až 100 snímků/s. Vzhledem k tomu, že je nutné, aby „zbyl čas" na ostatní části řídicího softwaru (zejména strategii) a vyslání vypočítaných řídicích povelů jednotlivým robotům, musí se celé zpracování obrazu odehrát v časovém intervalu menším než 10 ms. Algoritmy zpracování obrazu jsou proto optimalizovány především na rychlost. Také je nutné korigovat nedokonalost snímacího řetězce, např. optické vady objektivu, šum obrazu atd.). Výstupem zpracování obrazu jsou souřadnice jednotlivých štítků a míče na hřišti.

Poté, co jsou známé pozice štítků a míče, je nutné z nich rekonstruovat jednotlivé roboty a jejich polohu. Problematické z tohoto pohledu jsou zejména situace, kdy je více robotů shromážděno na jednom místě.

### 19.5.4 Strategické rozhodování

Z hlediska umělé inteligence (UI) se jedná o nejzajímavější část celého systému. Vzhledem k multiagentnímu charakteru celé hry se zde nabízí široké pole působnosti pro řešení problematiky přístupy z oblasti UI a teorie multiagentních systémů.

Tabulka 19.1 Atributy rozhodovacího stromu akcí

| Atribut | Typ atributu |
|---|---|
| vzdálenost týmového robotu od protivníka | spojitý |
| vzdálenost protivníka k míči | spojitý |
| vzdálenost týmového robotu k míči | spojitý |
| úhel od týmového robotu k míči | spojitý |
| úhel od týmového robotu k protivníkově bráně | spojitý |
| vhodný úhel ke střelbě | binární |
| volná dráha do protivníkovy brány | binární |
| možnost střelby bez bránění | binární |
| vlastníme míč | binární |
| jsme za míčem | binární |
| diference vzdáleností týmového robotu a protivníka k míči | spojitý |

Jako příklad řešení strategické části systému může posloužit jedna z variant vytvořená týmem RoBohemia. Strategie je rozdělena na tři vrstvy. První rozděluje jednotlivým robotům tzv. role. Rolí se rozumí herní zařazení fotbalisty v týmu, tedy zda se jedná o útočníka, obránce atd. Brankář je určen pevně, ačkoli pravidla toto nevyžadují. Některé týmy využívají možnosti dynamického střídání brankářů při hře (tedy zmocní-li se brankář míče, vyváží jej od brány a na jeho místo se postaví jiný robot). Tento přístup má nevýhodu v tom, že pokud se nastupující brankář nedostane včas na místo (například po kolizi se soupeřem), zůstává tým bez brankáře.

Po rozdělení rolí se rozdělí jednotlivým robotů akce. Akcí se rozumí např. najetí na pozici, střelba na bránu atd. Rozhodovací algoritmus je poměrně jednoduchý – jedná se vlastně o rozhodovací strom vytvořený pomocí algoritmu C5.0. Názvy atributů a jejich typ jsou uvedeny v tab. 19.1 (Hrubý, 2001).

### 19.5.5 Řízení robotů

Třetím krokem je plánování trajektorií jednotlivých robotů. Tato část je řešena pomocí vektorových polí (artificial potential fields). Každá akce zvolená v předchozím kroku má vlastní vektorové pole. Řízení po tomto vektorovém poli je zajišťováno modifikovanou variantou regulátoru uvedeného v (Veloso a kol., 1997).

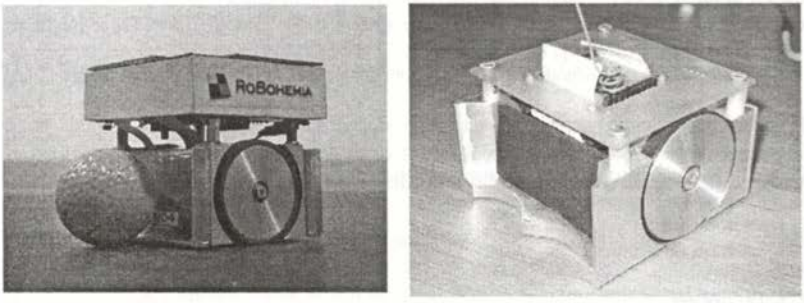
### 19.5.6 Robotický fotbalista

Robot pro fotbal kategorie FIRA MIROSOT musí splňovat několik základních kritérií. V současné době je robot v pojetí většiny týmů bez vlastní inteligence (pokud pomineme například detekci a korekci prokluzu kol). Kladeny jsou hlavně vysoké nároky na jeho jízdní vlastnosti (dynamiku, adhezi, rychlost ...). Tomu je přizpůsobován i charakter konstrukce – malá výška robotu, robustní základní deska (kvůli umístění těžiště nízko nad hrací plochu). Řídicí elektronika v minimální variantě sestává z přijímací části,

řídicího procesoru a výkonové části. Důležitá je vysoká spolehlivost a robustnost konstrukce (vzhledem k „tvrdosti" zápasů, resp. srážek mezi stroji).

Rozměry robotu jsou omezeny krychlí o hraně 75 mm (resp. 80 mm, pokud má robot zakrytí horní části – „dres"). Na obr. 19.24 je srovnání starší a nové varianty konstrukce fotbalového robotu.

Typickým konstrukčním řešením je diferenciálně řízený podvozek se stabilizačními body. V české literatuře může čtenář nalézt některé užitečné informace v (Kárník a kol., 2000) a (Novák, 2005).




Obr. 19.24 Srovnání dvou generací fotbalových robotů FIRA MIROSOT.

## 19.6 Závěr

V kapitole jsou popsány podrobněji některé praktické aplikace z oblasti servisní robotiky, které je v blízké budoucnosti předpovídán rychlý nárůst.

Zavedení robotické technologie je přelomem v chirurgických technikách. V současné době se navíc stále vyvíjejí nové chirurgické nástroje, které se miniaturizují a zvyšuje se jejich pohyblivost. V blízké budoucnosti lze očekávat i jejich propojení s navigačními systémy. To znamená, že operační nástroje budou automaticky navigovány na místa, kde jich je zapotřebí. Tyto systémy jsou schopny nahradit přímý kontakt rukou chirurga s orgány pacienta, nikdy však nenahradí jeho znalosti a zkušenosti.

Popisovaný mobilní robotický systém Orpheus je i nadále vylepšován na základě zkušeností získaných z mnoha praktických experimentů. Do budoucna se počítá zejména s ještě větším využitím dat z proximitních skenerů a jejich fúze s obrazovými daty, nově je aplikována i termovize.

V textu je uveden přehled typických zástupců armádních robotů. Jedná se o roboty určené pro transport materiálu, pozorovací činnosti a pyrotechnické práce. Tyto roboty jsou většinou konstruovány jako dálkově řízené s rozdílnou mírou inteligence. Škála možností je opravdu široká: od vozítek na dálkové ovládání osazených kamerou pro zkoumání a případné odpalování nevybuchlých nálož, až po sofistikované navigační systémy u průzkumných letounů a řízených střel, které jsou schopny vytvářet mapy sledovaných oblastí a vyhodnocovat typy objektů a vozidel.

Významnou a perspektivní skupinu robotů z oblasti servisní robotiky tvoří soutěžní roboty. Roboty pro fotbal poskytují ideální platformu pro výzkum kooperativních stra-

tegií v multiagentních systémech. Simulační ligy jsou výhodné svými nízkými pořizovacími náklady.

## Literatura

Armádní technický magazín.
Havel I. M.: *Robotika*, SNTL, Praha 1980.
Hrubý D.: *Herní strategie pro roboty MIROSOT* (diplomová práce), VUT v Brně, 2001.
Kárník L., Knoflíček R., Novák-Marcinčin J.: *Mobilní roboty*. Márfy Slezko, Ostrava 2000.
Novák P.: *Mobilní roboty – pohony, senzory,řízení*. BEN, Praha 2005.
Veloso M., Stone P., Han K., Achimm S.: *The CMUnited-97 Small Robot Team* [techreport], Carnegie Mellon University, Pittsburgh 1997.
URL:http://www.cs.cmu.edu/afs/cs/usr/pstone/public/papers/97springer/robot/robot.ps.gz

[1] www.army-technology.com/contractors/mines/piap/
[2] www.globalsecurity.org/military/systems/ground/mdars.htm
[3] www.globalsecurity.org/military/systems/ground/fcs-arv.htm
[4] www.eca.fr
[5] www.robosoft.fr
[6] www.bostondynamics.com
[7] www.moonbuggy.com
[8] www.fira.net
[9] www.fira.net/soccer/simurosot/overview.html
[10] http://sserver.sourceforge.net/
[11] www.robohemia.cz
[12] www.foster-miller.com
[13] www.robocup.org

# 20 Od osamocených robotů ke kolaborativní robotice

## 20.1 Úvod

Ivan M. Havel v první české monografii z oblasti kognitivní robotiky z roku 1980 definoval robot jako „... počítačem řízený integrovaný systém, schopný autonomní a cílově orientované interakce s reálným prostředím v souladu s instrukcemi člověka." (Havel, 1980).

Havlova definice byla orientována na jednotlivý robot, ale již zde vidíme, že silná vazba na spolupráci s člověkem v ní hraje důležitou součást chápání robotu a robotiky. Tato skutečnost je důležitá i pro obsah naší kapitoly a ukazuje mimo jiné stálou živost Havlovy definice.

Pod pojmem interakce se implicitně předpokládá, že robot jako stroj bude

- vnímat prostředí,
- rozhodovat se a
- provádět akční zásahy ve vnějším světě.

V řadě případů technické praxe se setkáváme čím dál tím častěji s úlohami, které nejsou řešitelné jedním robotem, nýbrž vyžadují silně distribuované vnímání a distribuovanou akci, tedy implicitně i distribuované cílově orientované rozhodování. Jedná se například o úkoly spojené s montáží složitého zařízení v těžko dostupném terénu či kosmu, aktivity při poruše jaderné elektrárny či při záchranných a vojenských operacích. Skupiny robotů je rozumné uvažovat i tam, kde je efektivní fyzicky a prostorově rozdělit úlohy vnímání, rozhodování a akce – pak například nepříliš veliké specializované roboty provádějí průzkum zamořeného či těžko dostupného prostředí, informaci zpracovávají jiné roboty (často neschopné se pohybovat v prostředí) a vlastní náročný fyzický zásah (odstranění překážky, přerušení potrubí, vyhrabání miny atd.) realizují roboty s velkým výkonem, event. v koordinované součinnosti.

Zajímavé jsou i úlohy, v nichž skupina robotů simuluje nebo nahrazuje týmy specializovaných lidí. Jedná se například o simulaci a napodobování týmových her (robotický fotbal), kde hráči s různými či stejnými schopnostmi formulují společnou taktiku, někdy dokonce na základě identifikace a modelování taktiky týmu protihráčů, event. organizují společné akce (obranu branky s účastí záložníků atd.). Jiným příkladem jsou záchranné operace, při nichž musí spolupracovat specialisté různých profesí se specializovanými roboty, které zasahují především v místech nebezpečných pro člověka a chovají se jako nedílní členové záchranářského týmu, složeného z lidí i robotů.

Pro naznačené úlohy je pak nutné počítat s nasazením většího počtu robotů, popř. s vytvořením kombinovaného týmu robotů a lidí, což je problém, jak v dalším výkladu

uvidíme, ještě o něco složitější. Nezbytnost *distribuovaného vnímání, distribuovaného rozhodování* a *distribuovaných*, avšak koordinovaných, *akčních zásahů* nás staví před celou řadu nových teoretických i technických problémů, spojených například s komunikací, distribuovaným rozhodováním na základě částečné informace, zřetelným oddělením lokálních dat a globálních znalostí, s koordinovaným pohybem v dynamicky se měnícím prostředí atd. Další množina problémů vyvstává při společných aktivitách smíšených týmů robotů a lidí – lidé a roboty vnímají, komunikují a rozhodují zcela jiným způsobem ve zcela různých teoretických i realizačních rámcích. Všechny tyto problémy se snaží řešit disciplína, často označovaná jako *kolaborativní robotika*.

Pro kvalifikované distribuované rozhodování potřebuje skupina robotů

- vědomě sdílet společné cíle, a to buď úplně, nebo částečně,
- mít schopnost koordinovat akce či dokonce aktivně kooperovat k dosažení sdíleného cíle,
- mít schopnost plánovat a flexibilně přeplánovávat akce,
- mít schopnost komunikovat, nejlépe v rámci jednotného softwarového prostředí s jednotným jazykem a jasně stanovenými principy vyjednávání, tedy užívat jednotné komunikační a vyjednávací platformy,
- schopnost delegovat úlohu či podúlohu,
- sdílet znalosti o okolí, o schopnostech robotů-partnerů (tzv. *sociální znalosti*), udržovat tyto znalosti aktuálními a konzistentními a minimalizovat tak nároky na komunikační kanály v časově kritických momentech rozhodování,
- mít schopnost uvažovat o zájmech, záměrech a vizích ostatních robotů s prioritou týkající se existenčních zájmů, současně ale mít schopnost uvažovat i o sobě samém, své pozici a úkolech ve skupině,
- mít schopnost zabezpečit přežití skupiny i v případě selhání jednoho robotu.

Obdobné požadavky se jeví přirozenými i pro distribuované vnímání, kde se v podstatě jedná o globální interpretaci lokálních vjemů s explicitní kombinací (a) lokálního využívání lokálně získaných dat a (b) lokálně sdílené globální znalosti o prostředí, která vzniká globální integrací lokálně získaných informací. To znamená, že některá data ze senzorů jsou zpracovávána jen lokálně daným robotem a (některé) výsledky tohoto zpracování jsou zasílány dalším robotům pro potřeby tvorby globálnější vize o prostředí. Robot obvykle zpracovává data z mnoha svých senzorů a vytváří si prvotní představu například o charakteru překážky před ním. Tuto představu, např. ve formě jednoduchého obrázku, předává dalším robotům k jejich úvaze, kritickému zhodnocení či pro kompletaci jejich globálnějšího obrazu celé scény. Ostatní roboty mohou vést o zaslané informaci určitou diskusi, dohadovat se a vyjednávat o získávání dalších podkladů mezi sebou. Po vytvoření globálnější představy roboty pak obvykle předávají relevantní část globálnějšího pohledu zpět danému robotu tak, aby si na základě tohoto globálnějšího pohledu upřesnil výsledky svého lokálního vnímání. Výměna informací mezi roboty může být výrazně zefektivněna, pokud roboty pracují ve stejném sémantickém rámci, tedy pokud sdílejí příslušné *znalostní ontologie* (tj. základní významové koncepty a relace mezi nimi).

Při distribuovaných akcích jsou roboty navíc nuceny sledovat a vyhodnocovat lokální prostředí a realizovat lokální akce tak, aby nedošlo – ať již při akci, nebo při vyčkávání – k poškození robotu elementy prostředí nebo jinými roboty v týmu.

Skupiny robotů se jako tým musejí zabývat řešením problémů na třech úrovních:
- na *úrovni strategické*, na níž se realizuje výběr cílů a podcílů a vytvářejí se strategické plány nejvyšší úrovně pro dosažení sdíleného cíle – práce na této úrovni se může zúčastnit jeden nebo několik málo robotů schopných abstraktnějšího (symbolického) uvažování,
- na *úrovni taktické*, na níž jsou řešeny úlohy takticko-operační povahy, tedy kde se rozpracovávají globální strategické plány na konkrétnější plány operační s uvažováním momentální lokální situace,
- na *úrovni akční*, tj. *motorické*, kde jsou zabezpečovány vlastní pohyby, fyzické zásahy do prostředí a automatické vyhýbání se překážkám či jiným robotům (dekonflikce).

Na své cestě ke kolaborativní robotice se dnešní robotika stále ve větším měřítku opírá o výsledky jiné oblasti, a to *oblasti multiagentních systémů*. Tato oblast dnes poskytuje celou řadu teoretických výsledků, experimentálních implementací a nástrojů, které jsou zaměřeny na modelování a realizaci chování souboru autonomních kooperujících subsystémů. Nicméně výzkum v oblasti agentních systémů se zaměřuje především na informační (softwarové) agenty, kteří nebývají bezprostředně spojeni – s výjimkou holonických výrobních systémů (Brennan, Norrie, 2001), v nichž občas participují i roboty – s fyzickým světem a fyzickými akcemi (Mařík, McFarlane, 2005). Nicméně oblast multiagentních systémů přináší řadu technik, technologií a nástrojů bezprostředně aplikovatelných v kolaborativní robotice. Jedná se například o architekturu znalostních modelů (acquaintance models, viz (Mařík a kol., 2001a)) pro reprezentaci lokálních znalostí, o techniky komunikace, vyjednávání, uvažování a (sebe)reflexe, o struktury pro reprezentaci ontologických znalostí, o řešení úloh při komunikační nedostupnosti, řešení konfliktních pohybových plánů robotů či algoritmy pro stanovování důvěryhodnosti zpráv zasílaných mezi roboty. Svět kolaborativní robotiky může naopak oblast multiagentních systémů motivovat k dalším řešením, např. díky uvažování nepřímé komunikace (zprávy vysílané jako výsledek vedlejšího efektu fyzického přemístění objektu) apod. Lze proto právem předpokládat, že užší vazby mezi oblastí kolaborativní robotiky a multiagentních systémů budou nabývat na intenzitě.

## 20.2 Robot – základní stavební prvek kooperujících skupin

### 20.2.1 Co je to inteligentní entita, robot?

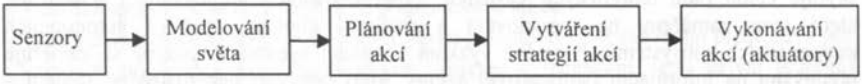
Pojem robot ve svém obecném pojetí obvykle evokuje představu jakéhosi stroje nebo obecněji entity vykazující chování s prvky inteligence.

Inteligentním robotem je zařízení, od kterého na základě samostatného získávání informací o prostředí očekáváme vytváření posloupností akcí (plánů) a jejich fyzické vykonání, čímž robot realizuje cílově orientované chování. Cílově orientovaným chováním rozumíme takovou posloupnost akcí, jež vede ke splnění zadaného úkolu jistým výhodným postupem.

Věnujme se nyní úvahám o sdružování odlišných entit. Základní funkční schémata popisující vazbu mezi inteligentní entitou a její fyzickou realizací mohou být odlišného charakteru. Klasický a dlouhou dobu používaný přístup k dosažení jednoduchého inteligentního chování ve spojení s robotem (Nilsson, 1969) byl založen na jednoduchém

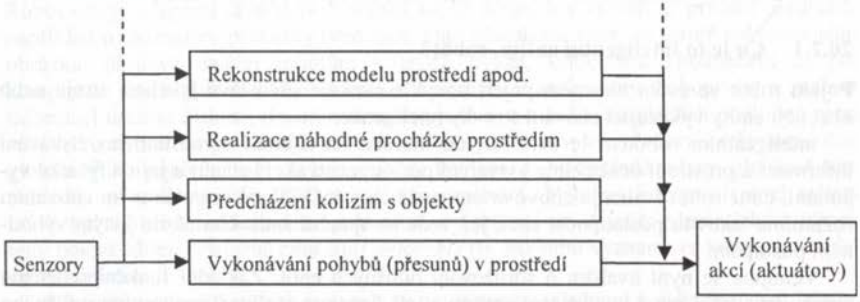
konceptu nazývaném *SPA* (Sense–Plan–Act). Jeho myšlenka spočívá ve vytváření plánu činností na základě interpretace senzorických informací v modelu světa, jenž následně realizují. Popsaný proces je vykonáván přísně sekvenčně, jak je naznačeno na obr. 20.1.

Nicméně s rostoucí složitostí, a to jak ve smyslu množství zpracovávaných dat a rozlehlosti systému, tak ve smyslu náročnosti postupů pro jejich zpracování, se u takto řízených systémů ukazuje omezení plynoucí z výpočetního výkonu, jenž je typicky svým způsobem a vždy limitovaný. Z obecného hlediska je velmi obtížné udržet a spravovat složité reprezentace interního strojového modelu světa robotu v reálném čase a udržet je konzistentními s realitou. Tím se klasický způsob aplikace umělé inteligence ve složitých systémech stává nevýhodným až nemožným. Obtížnou začíná být nejenom reprezentace modelu, ale i údržba a použití složitých (obecných) reprezentací znalostí, které jsou obvykle potřebné pro účely plánování a rozhodování v takovém systému.




Obr. 20.1 Paradigma SPA (Sense–Plan–Act).

Ve snaze o eliminaci omezení klasického postupu vznikla odlišná metodologie vycházející z průlomové práce R. Brookse (Brooks, 1991), jejíž výsledky byly aplikovány například v (Maes, Brooks, 1990), (Mataric, 1992), (Horswill, 1993), ale i jinde. Principem odlišného pohledu na problém a jeho řešení bylo hledání souvislostí se způsobem, jakým jednoduché organismy dosahují rychlého, účinného a cílově orientovaného chování, např. pro účely přežití. Výsledkem vývoje byl jazyk dovolující modelovat chování analogická chování hmyzu užitím uzavřených smyček *vjem–akce* (Sense–Act Loop) realizovaných množinou asynchronních konečných automatů uspořádaných do hierarchické struktury. Jednotlivé automaty jsou schopny realizovat určité druhy chování, kdy k rozhodování o konkrétních činnostech (vesměs na základě vyhodnocení priorit chování) dochází až v aktuátorech (akčních členech) robotu. Podstatou této myšlenky je



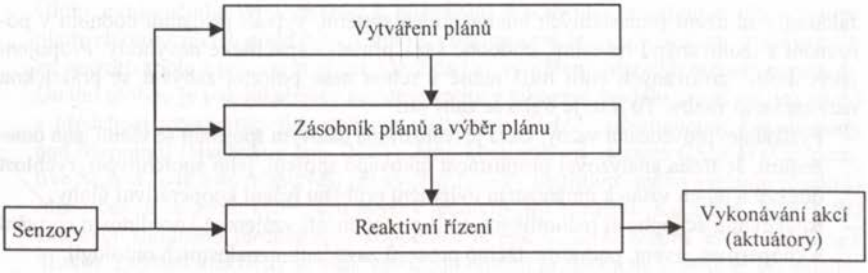

Obr. 20.2 Ilustrace změny mechanismu *SPA* na Brooksův model, který využívá paralelismu samostatných procesů.

vybudování schopností robotu z „knihovny" jednotlivých druhů chování, tedy způsobu odezvy na určitý vjem. Brooksův postup rekonfiguruje klasický přístup *SPA* na tzv. *paralelní behaviorální architekturu.*

Ačkoliv takto realizovaná inteligence robotů vykazuje mnoho pozitivních vlastností, je velmi obtížné metodologii formalizovat. Roboty řízené naznačenými mechanismy jsou často schopny realizovat chování vedoucí k efektivnímu přežití v prostředí, tedy chování podobná lidskému způsobu řešení. Naopak, obtížná formální postižitelnost vlastností řešení nedává možnost predikce chování takového systému, stejně jako možnost cíleně ovlivnit vykonávání některých specifických akcí.

Dalším rozvojem postupů s důrazem na činnosti související s *vnímáním* a navazující *akcí* při omezení plánovací činnosti systému je pak možné dospět k architekturám označovaným jako *AOP* (Action–Oriented Perception), jež jsou užitečné pro řešení manipulačních a navigačních úloh (Arkin, 1987) a fúze senzorických dat (Murphy, Mali, 1996). Takto řízený systém má již distribuovanou realizaci s tím, že většina činností *vjem–akce* probíhá v bezprostřední fyzické blízkosti jednotlivých senzorů. Nicméně takové systémy, i přes svoji reaktivitu, rychlost a schopnost reakce bez silného prvku globálního pohledu, jsou jen zřídkakdy užitečné pro řešení úloh v běžném životě.

Z předchozích příkladů vyplývá, že nejlepšího možného chování inteligentní entity – robotu – není zřejmě možné dosáhnout striktní aplikací ani jednoho z uvedených postupů. Tento fakt byl navíc mnohokrát potvrzen i analýzou výsledků z různých praktických realizací takových systémů (Mahadevan, Connell, 1991), (Nemzow, 1999) a soutěží – viz adresy [1], [2]. Východiskem se ukazuje použití třívrstvé architektury (Bonasso a kol., 1997), (Gatt, 1992) kombinující klasické plánování na nejvyšší úrovni, reaktivní techniky vykonávání akcí na úrovni senzorů a aktuátorů, a využívající přitom mezilehlou vrstvu, která obsahuje dynamické plány činnosti. Posloupnost plánů je zde výsledkem nejvyšší úrovně plánování a vstupem do reaktivního mechanismu vykonávání. Naznačená hybridní architektura je nazývána též *P–SA* (Plan first, then Sense–Act). Vede k systémům s reaktivním chováním, reagujícím na senzorická data, jež je koordinováno plány připravenými na vyšší úrovni centrální jednotkou. Volné uvažování a rozhodování je pak úkolem centrální jednotky, která je zodpovědná za funkci celého systému.




Obr. 20.3 Mechanismus *P–SA* (Plan first, then Sense–Act).

### 20.2.2 Motivace pro sdružování entit

Každý technický systém, stejně jako živý organismus, disponuje určitými, předem danými, vlastnostmi, respektive schopnostmi. Podle svého charakteru a vlastností pak taková entita obvykle dokáže svoje schopnosti modifikovat podle okamžitých podmínek (adaptace) a rozvíjet jejich kvalitu na základě zkušenosti (učení). Z tohoto pohledu nic nebrání představě, že dvě nebo více inteligentních, ale fyzicky distribuovaných entit může projevit snahu adaptovat nebo zdokonalit svoje chování při řešení daného problému. Propojení fyzicky izolovaných entit umožní vzájemné *sdílení zdrojů* (energie, výpočetního výkonu, senzorů, aktuátorů atd.) stejně jako *sdílení dat a znalostí* (reprezentovaných aktuálním stavem paměti, obsahem databáze či nejrůznějších integrovaných znalostních a datových bází v různých typech znalostních modelů (Mařík a kol., 2001a) stejně jako dříve odvozenými plány řešení úloh atd.) a jejich *dynamickou aktualizaci*.

Poněkud méně očekávanou vlastností, nicméně podstatným přínosem, je pak *získání nových schopností* jednotlivých entit, které se podílejí na společném řešení problému. Propojením fyzicky distribuovaných entit si celá skupina rozšiřuje třídy řešitelných úloh o takové, k nimž individuální entity nemají přístup, např. vyhovující manipulační schopnosti, dostatek energie, potřebné pozorovací schopnosti atd. Dalšími úlohami řešitelnými jenom v rámci množiny entit jsou úlohy *dynamické rekonfigurace* a *delegace úloh* jiným entitám.

Získání nových a výhodných vlastností nelze dosáhnout bez *kooperace* a *koordinace* podílejících se entit. V dané souvislosti to znamená, že všechny nebo vybrané entity podílející se na společné činnosti jsou nuceny do jisté míry podřizovat svoje rozhodování ostatním entitám, globální situaci i globálním kritériím optimality. Obdobně pak je očekáváno, že část svých zdrojů, informací a znalostí jsou tyto schopny nabídnout k dispozici ostatním a sdílet je. Aby bylo možné právě uvedené vize realizovat, musí existovat mechanismus, který dokáže zajistit globální kooperaci a koordinaci činnosti jednotlivých entit. Tento nadřazený mechanismus se mimoto stává i prostředkem k dosažení kvantitativně, ale především kvalitativně dokonalejších vlastností skupiny entit jako celku, tzv. *globálních vlastností*.

### 20.2.3 Výhody, nevýhody a rizika sdružování

Jakákoliv sdružení samostatných inteligentních systémů vytváří přidanou hodnotu v porovnání s izolovanými řešeními, obdobně však přináší i specifické nevýhody. Propojení jakýchkoliv izolovaných entit totiž nutně s sebou nese potřebu zabývat se praktickou realizací této vazby. To jest, je třeba se zabývat:

- Fyzikálním provedením vazby, která je realizována datovým spojením se všemi jeho omezeními. Je třeba analyzovat propustnost takového spojení, jeho spolehlivost, rychlost odezvy a jejich vztah k možnostem ovlivnění průběhu řešení kooperativní úlohy.
- Rozšířením schopností jednotlivých entit o schopnosti vzájemně koordinovat chování a kooperovat, event. podporou těchto procesů zavedením znalostních ontologií.
- Spotřebou určité části zdrojů jednotlivé entity pro účely komunikace. Tento aspekt může být významným v úlohách, které pracují s nízkopříkonovými a výpočetně slabými systémy, jakými jsou například vesmírné aplikace, senzorové sítě, robotická hejna (robot swarms) apod.

– Bezpečnostními aspekty dané úlohy a jejími komunikacemi, které vyplývají z možnosti úmyslného, často skrytého narušení spojení nebo činnosti, jež působí cíleně proti podstatě vykonávaného řešení.

Je třeba zkoumat i robustnost samotného řešení úlohy. Takový pohled na věc zahrnuje situace, kdy dojde k neočekávanému selhání části ve skupině společně pracující komunity entit, např. výpadkem některého jedince. Zde je potřebné zkoumat reakci zbytku skupiny z hlediska dosažitelnosti a efektivity dosažení původního cíle, jež zřejmě záleží na povaze úlohy, způsobu jejího řešení a zdrojích, které jsou k dispozici. Jedná se pak o úlohy *rekonfigurace* skupiny a *realokace* cílů a zdrojů. Uvedená situace je v kooperujících systémech častým jevem a svoji povahou formuluje jednu ze základních tříd problémů pro tuto oblast.

Dalším nebezpečím může být přecenění schopností kooperující množiny inteligentních entit při řešení nadměrně komplikovaných úloh. Odhad náročnosti řešení úlohy pro skupinu entit s danými vlastnostmi je nelehkým úkolem. Proto vznikají i situace, kdy zadaná úloha nebude danými prostředky řešitelná vůbec nebo jen velmi neefektivně. Obdobně může nastat situace, kdy v průběhu řešení dojde k omezení zdrojů skupiny, jež úlohu řeší.

### 20.2.4 Nové úlohy spojené se sdružováním

Jak již bylo naznačeno, schopnost kooperace a koordinace činnosti autonomních systémů přináší efektivnější řešení standardních úloh, které jsou běžně řešeny nekooperujícími systémy. To zahrnuje především:

– Úlohy reaktivního typu, jakými jsou *předcházení kolizím*, jež jsou obvykle řešeny při interakci aktivní entity (robotu) s pasivním objektem prostředí v jednoduché verzi. Pro situaci s více spolupracujícími systémy je možné úlohu rozšířit o řešení interakcí i mezi aktivními účastníky úlohy navzájem. Prosté reaktivní chování je pak rozšířeno směrem k postupům založeným na vyjednávání přípustného nebo požadovaného chování – *dekonflikce* – mezi jednotlivými entitami k zabránění vzájemných kolizí. *Pohybová dekonflikce* při současném pohybu více entit v omezeném prostoru (např. vyhýbání se autonomních transportních vozíků, automatická koordinace letadel v letovém prostoru apod.) patří k prioritním úlohám kolaborativní robotiky.
– Úlohy explaračního typu vedoucí k poznávání pracovního prostředí entity a v něm platných omezujících podmínek. Činnost s tím spojená realizuje fúzi dat, získávaných ze senzorů spolu s procesem učení, které ústí ve vytváření interního modelu prostředí. Duální úlohou je pak lokalizace samotné entity v takovém modelu prostředí, jež spolu s předchozí interpretací dat realizuje algoritmus *SLAM* (Simultaneous Localization and Mapping) (Haehnel a kol., 2003). Jeho kooperativní varianta *CSLAM* (Cooperative SLAM) pak tuto úlohu řeší při využití fyzicky distribuovaného systému (často i) různorodých senzorů, které jsou neseny jednotlivými entitami (Ko a kol., 2003). Při existenci vhodného mechanismu kooperace je pak *CSLAM* zřejmě efektivnější než SLAM. Zvýšení efektivity je však vyváženo novým problémem, vzniklým při skládání informací získaných od fyzicky oddělených senzorů do jediné společné reprezentace – modelu prostředí.
– Problém *vzájemné lokalizace zúčastněných entit* je rozšířením klasického postupu pro lokalizaci pohyblivé entity relativně, vůči statickému nebo velmi pomalu se měnícímu

prostředí. Vzájemná dynamická lokalizace aktivních entit je pak prostředkem k efektivnímu řešení úloh typu *CSLAM*, ale i například k řešení problému retranslačního spojení přes komunikační most tvořený roboty.

- Problém *kolaborativní montážní, výrobní či demontážní akce*, založené na kooperativní spolupráci více entit, které komunikují přímo či nepřímo a vyjednávají nejen mezi sebou, ale i s prvky prostředí (např. kooperace robotů v pružném výrobním úseku, kde nakonec i dopravník lze považovat za specifický typ robotu).
- *Udržování formace*, kdy se skupina spolupracujících entit snaží udržet v dané geometrické formaci, jež je vyžadována pro zajištění činnosti systému, splnění úkolu apod.
- *Vytváření formace*, tj. úlohu, ve které je určení a nastavení vhodného postavení jednotlivých entit samotným řešením daného problému. Příkladem může být vytváření komunikačních mostů – kanálů pro uskutečnění přenosu informace v podmínkách s omezeními.
- *Přidělování aktivit a zdrojů*, jejichž vhodné užití vede k dosažení řešení úlohy, která byla skupině entit zadána. Úloha je chápána jako obecný problém globální optimalizace v systémech s více zúčastněnými entitami a více aktivitami. Obvykle se jedná o různorodé systémy s odlišnými schopnostmi.
- *Úlohy v prostředí s informační nedostupností*, které si kladou za cíl ošetřovat případy, kdy se jednotlivé entity v systému vyznačují omezenou komunikační dostupností. Ta může být zapříčiněna jednak fyzikálními podmínkami, popřípadě i cílenou strategií utajení. Řešením jsou postupy zachovávající některé požadované vlastnosti takových systémů a garantující dosažení řešení dané úlohy. Používá se zde často tzv. „stand-in" modelů nahrazujících po omezenou dobu nedostupnou část komunity.
- *Integraci v hybridních týmech* sdružující řešení problémů vznikajících při integraci inteligentních systémů v hybridních týmech spolupracujících robotických a lidských entit. Úloha zahrnuje základní roviny zprostředkování specifické komunikace na rozhraní člověk–stroj a řešení problémů přidělování aktivit, zdrojů a eventuálně dalších podúloh.

## 20.3 Sdružování shodných a odlišných druhů entit

### 20.3.1 Entity shodného druhu (roboty s roboty)

Návrhový a implementační pohled na způsob svázání jednotlivých druhů autonomních entit musí respektovat jejich funkční podstatu. Základním dělením podle způsobu činosti a zpracování informace v systému je nepochybně rozlišení autonomních inteligentních zařízení (robotů) a biologických entit (člověk).

Jednodušším případem je integrace entit stejného druhu, tedy robotů navzájem. V důsledku toho, že charakter činnosti takových systémů je založen na obdobném nebo shodném principu, často postačuje pouhé vytvoření jednotné definice formátu či protokolu vzájemné komunikace. Velmi výhodné je využívat předností objektově orientovaného přístupu. Vlastní propojení je obvykle realizováno užitím identického média – datového spojení.

Na realizaci fyzického propojení navazuje mechanismus, jehož úkolem je zajistit výběr, přenos a zpracování sdílené informace, a to oběma směry vzhledem k dané entitě. Tento mechanismus tedy zajišťuje samotnou funkcionalitu spolupráce, to jest je zodpovědný za koordinaci a kooperaci.

Charakteristickým rysem použití řídicího postupu pro řešení kooperativní úlohy je v daném případě fakt, že veškeré entity mohou být v tomto nejjednodušším případě řízeny sdílenými či všeobecně známými pravidly, jež reprezentují deterministický postup (algoritmus). To přináší možnost zjednodušení při tvorbě modelu chování takového systému nebo jeho části, predikci jeho stavu atd. Přímým důsledkem pak je snadnější řešení například situací s omezenou komunikační dostupností, utajování vnitřního stavu části koooperujícího systému atd.

Uvedenou generickou úlohu lze přidáváním vhodných anebo vyžádaných omezení na sdílení informací učinit podstatně složitější. V takových situacích je pak budování modelu chování, popřípadě jeho odvození na základě pozorování jednotlivých částí skupiny, podstatně obtížnější či dokonce nemožné.

### 20.3.2 Sdružování rozdílných systémů (roboty a lidé)

K získání schopnosti řešit široké spektrum úloh může být výhodné mít k dispozici více entit s různorodými schopnostmi. Jedná se o úlohy, kdy tým robotů může být účelně doplněn jedním člověkem či několika lidmi, či naopak, kdy tým lidí potřebuje podporu v podobě skupiny robotů. V obou případech se pak jedná o tzv. *hybridní týmy*.

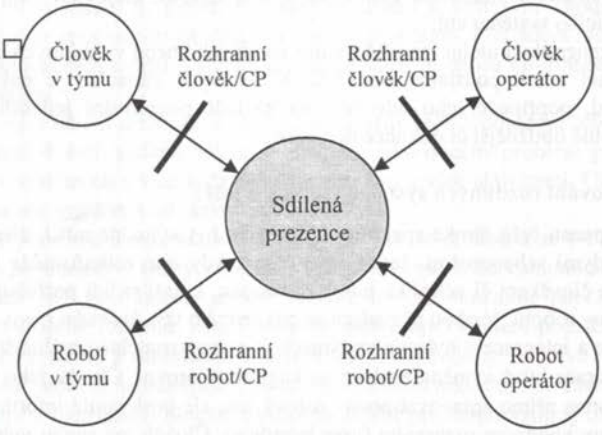
Komunikace a integrace v hybridních týmech je neporovnatelně obtížnější. Prostředkem jejich integrace bývá výměna znalostí na kognitivní úrovni. Člověk jako biologická entita není schopna přímo zpracovat prostý datový tok, ale poskytnuté informace musejí být konvertovány vhodným rozhraním (user interface). Člověk má snahu vyhodnocovat data samostatně. Objevuje se proto riziko nesprávné interpretace, způsob vyhodnocení je nedeterministický a závislý na mnoha neovlivnitelných a obtížně predikovatelných faktorech (psychickém stavu, fyzické únavě atd). Navíc, tok informací, jenž je člověk schopen vyhodnotit, je poměrně nízký (ve srovnání s robotem), a tudíž může snadno dojít k výpadku ve zprostředkování informace (člověk „nestíhá").

Další třída problémů související s činností hybridních týmů pramení z relativně velmi vysokého stupně inteligence člověka a jeho schopnosti se samostatně rozhodovat a velmi rychle adaptovat. Tím je obecné vytváření modelů chování lidské entity pro naznačené účely velmi omezené a téměř nemožné. Výjimku snad mohou tvořit jen velmi úzce zaměřené úlohy a specifické situace.

Faktická realizace hybridních systémů souvisí s vytvářením *prezence* (Barfield a kol., 1995) prováděné na úrovni CP (*Common Presence*) pro participující systémy (Suomela a kol., 2003). Modely CP jsou založeny na představě vytváření společných modelů prostředí, znalostí o nich a popisu schopností či funkcionalit jednotlivých entit. Přístupy k informacím, jejich změny atd. jsou prováděny prostřednictvím rozhraní, které je vždy specifické pro daný druh entity. Tím vzniká určitý druh společného nosiče informace, jenž obsahuje znalost společnou všem entitám (robotům a lidem) při řešení úloh spolupráce. Fyzická implementace scénáře CP pak může být provedena jak v centralizované podobě, tak plně distribuovaně.

Společné použití uvedeného scénáře CP odlišnými entitami navzájem může být přísně místní, kdy se zúčastněné entity vyskytují společně v jedné fyzické lokalitě, vnímají stejné prostředí a jsou schopny ovlivňovat jeho stav společně. V takovém případě jde o pouhou realizaci sdílení prezence.

Další zvláštní třídu tvoří situace, kdy ke sdílení prezence dochází na určitou, významnou vzdálenost. Takové případy, kdy fyzická vzdálenost užitá pro sdílení modelu CP neumožní společné vnímání a fyzické ovlivňování stejné části prostředí všemi nebo některými vybranými entitami současně, označujeme pojmem *teleprezence*.




Obr. 20.4 Koncept společné prezence CP (Common Presence) pro biologické a robotické entity v rolích člena týmu nebo řídicího operátora.

Vznik teleprezence ve své podstatě reprezentuje situace, kdy jsou mezi zúčastněnými entitami a prostředím, ve kterém operují, popř. mezi jednotlivými entitami navzájem, dálkově zprostředkována senzorická pozorování a akční zásahy schopné ovlivnit stav tohoto prostředí. Realizaci teleprezence je možné principiálně rozdělit na dva současně řešené základní problémy. Prvním je úloha přenosu a realizace povelů a akčních zásahů od entity do vzdáleného prostředí nebo k jiné entitě. Taková úloha, ve své nejjednodušší podobě, může být reprezentována například systémy dálkového ovládání. Druhou úlohou je pořizování zpětnovazební informace ze vzdáleného prostředí, která je reprezentována sběrem senzorických měření v odlehlém místě a extrakcí znalostí z nich. Na základě získaných informací je pak cílem dosažení schopnosti rekonstruovat vjemy obdobné fyzické přítomnosti člověka nebo jiného systému (robota) v místě, což postihuje právě pojem teleprezence vybraného systému ve fyzicky odlišné lokalitě.

Tak, jako CP představuje koncepci sdílení společných informací, znalostí a funkcionalit různých a odlišných entit, teleprezence je pak speciálním případem její realizace na velkou vzdálenost.

Rozhraní pro přístup jednotlivé entity do společného modelu CP je obvykle jednodušší pro neživou část skupiny. Roboty realizují prosté čtení nebo zápis kvantitativních informací (například o svojí poloze v prostoru, pozorování pracovního prostředí, existenci a tvaru překážek v prostoru atd.).

Složitější je situace pro lidskou entitu, neboť jí při přístupu do modelu CP musí být získaná data nejprve doručena (například hlasově nebo zobrazením pomocí grafického uživatelského rozhraní) a poté vyhodnocena (porozumění). Výsledky rozhodnutí člověka musí být – obvykle opět vícestupňově – převedeny zpět do podoby elementárních instrukcí a dat.

Odlišný charakter informace, s níž je člověk schopen pracovat, vzhledem k informaci uskladněné v modelu CP, vyžaduje použití podpůrných technických systémů. Jejich úkolem je konvertovat lidskou formu informace a znalostí do podoby vhodné pro uložení do modelu CP. Funkcionalita podpory je součástí rozhraní člověk–kooperativní systém.

V předchozím výkladu naznačené principy budování prezence, jejího sdílení v modelech CP, popř. její vzdálené realizace v teleprezenčních systémech, jsou postupy zabývající se způsoby komplexního porozumění senzorickým informacím s cílem vytvořit reálnou „iluzi" přítomnosti nějakého systému (člověka nebo robotu) ve zvoleném kontextu. Nicméně nástrojem k jejich realizaci stále zůstávají technické prostředky z oblasti tzv. *telematiky*, jež kombinuje telekomunikační technologie a postupy z oblasti informatiky. V kontextu robotiky je pak dnešní telematika reprezentována především dálkově povelované (semi)autonomní roboty, či jednoduše ovládané manipulátory (tzv. teleoperátory) určené k ovlivňování stavu prostředí na dálku. K jejím nedílným součástem stejně tak patří postupy k získávání a zpracování dat z jejich senzorů, na jejichž základě lze tyto systémy efektivně řídit.

I když se zdá být sdružování biologických entit s neživými roboty komplikované, ve svém důsledku přináší do oblasti kooperujících týmů zcela nové vize a rozvíjí netradiční postupy. Základní motivací k řešení této úlohy však stále zůstává snaha substituovat nedokonalost neživých autonomních entit, stále neschopných vykonávat veškeré druhy činnosti. V dnešní době, ale i v blízké budoucnosti si lze na základě zmíněných technologií například představit vývoj telematických systémů nové generace, kdy se člověk-operátor stává nedílným a rovnocenným partnerem robotu při vykonávání úlohy, kdy takový systém dokáže optimalizovat míru stresu lidského operátora, minimalizovat chybovost, dobu vykonávání úkolu apod.

## 20.4 Agentový přístup k multirobotickým systémům

V této části se budeme věnovat problematice návrhu architektury systémů tvořených entitami stejné funkční podstaty. V případě, že jsou těmito entitami autonomní inteligentní roboty, jsou takovéto systémy v literatuře často označovány jako *multirobotické systémy*.

Tyto systémy nacházejí čím dál tím větší uplatnění v oblastech, jako jsou průmyslová výroba, záchranné operace, průzkum země a vesmíru (rovery, sondy), vojenské operace, zemědělství, zábavní průmysl a další.

Použití většího počtu robotů pro řešení úloh z těchto oblastí přináší množství výhod ve srovnání s řešením spoléhajícím pouze na jednoho robota:

- **Paralelismus**. V mnoha aplikacích umožňuje nasazení většího množství robotů snížení celkové doby řešení úlohy. Příkladem může být mapování a prohledávání prostoru nebo úklid.
- **Robustnost**. V případě selhání některého z robotů může jeho funkci převzít jiný robot disponující stejnou funkcionalitou a celý systém tak pokračuje v plnění svých cílů. V některých aplikacích (zejména vojenských) je dokonce počítáno s možností zničení některých robotů, a proto je využití většího množství vzájemně zastupitelných entit přímo vyžadováno.
- **Cena**. Náklady na výrobu jednoho velkého složitého robota mohou převýšit cenu většího počtu menších, jednoduchých robotů. U složitých robotů navíc nastává problém s rekonfigurovatelností. Zpravidla bývají navrhovány pro jednu konkrétní úlohu, zatímco jednodušší roboty mohou být opakovaně použity jako součást týmu v různých aplikacích.
- **Škálovatelnost**. Tento faktor úzce souvisí s cenou. V případě změny parametrů úlohy (např. zvětšení plochy určené k prohledávání) je často jednodušší zvětšit tým přidáním jednoho nebo více robotů než upravit vlastnosti jednoho robota tak, aby splňoval požadavky na řešení úlohy.
- **Týmová rekonfigurace**. Je schopnost dynamicky měnit úlohy jednotlivých robotů, přerozdělovat cíle a zdroje na základě vzájemné komunikace a koordinace, a to jak za měnících se podmínek, tak zejména při výpadku některých robotů nebo některých jejich funkcionalit.

Některé úlohy ze své podstaty nemohou být splněny jedním, byť velmi složitým, robotem. Jedná se zejména o úlohy, které vyžadují fyzickou přítomnost robotů na více místech v prostředí současně, jako je například přenos informace na větší vzdálenost v reálném čase, kdy roboty musí vytvořit komunikační most. Na druhou stranu jsou výhody použití multirobotického systému zpravidla vyváženy větší složitostí při návrhu, implementaci a testování takového systému.

### 20.4.1 Vztah mezi multiagentovými a multirobotickými systémy

Pro návrh organizačních struktur a řídicích algoritmů multirobotických systémů se často používají metody a principy známé z oblasti *multiagentových systémů* (Ortiz a kol., 2005). Tyto dvě oblasti výzkumu se vyvíjely relativně odděleně, nicméně technologický vývoj v posledních létech (menší a levnější robotické platformy, rychlejší procesory, lepší senzory, bezdrátová komunikace, nové výsledky v multiagentní oblasti apod.) vede k jejich vzájemnému sbližování. Pro řízení činnosti skupiny robotů se tak často využívají algoritmy vyvinuté původně pro oblast vysoce distribuovaného agentového rozhodování. Jedná se zejména o algoritmy distribuovaného plánování akcí, vyjednávací mechanismy a komunikační jazyky a protokoly pro vzájemnou komunikaci mezi agenty (Mařík a kol., 2003).

Vlastnosti multiagentních systémů, podporující smysluplnost jejich použití v kolaborativní robotice, se vztahují jednak k *distribuovaným a kolaborativním aspektům* chování skupiny agentů, jednak k *aspektům podporujícím autonomii* rozhodování jednoho agenta.

- **Kolaborativní chování.** Je podporováno metodami spolupráce, např.
  - za situace, kdy je část agentů komunikačně nedostupná,
  - v úlohách, kdy úplné sdílení znalostí není možné nebo žádoucí, tedy například v kompetitivním prostředí,
  - v úlohách, pro něž je kritická rychlost reakce a robustnost řešení,
  - tam, kde je vhodné simulovat celé řešení před vlastní fyzickou realizací, např. z důvodu studia realizovatelnosti postupu nebo při neznalosti celkového charakteru chování celé skupiny,
  - tam, kde se počítá s nasazením různorodých agentů a kde se pro potřeby integrace požaduje poměrně vysoký stupeň jejich interoperability,
  - v tak složitých úlohách, že žádný agent není schopen vyřešit úlohu samostatně – pak je nutné použít metod dekompozice úloh, robustních metod usuzování a současně metod vyjednávání mezi entitami.

- **Autonomní chování.** Je podporováno především metodami delegování, autonomního usuzování a samostatného rozhodování při nedostatku informace o stavu a chování ostatních.

Multirobotická úloha je ve své podstatě úlohou kolaborativní, neboť vždy jde o maximalizaci užitku ze společného chování. Jak je patrné z předchozího výčtu, multiagentní systémy navíc podporují i *kompetitivní* či *dokonce adversariální (nepřátelské) chování.*

Při návrhu multiagentového systému, který z hlediska informatiky vlastně reprezentuje komunitu robotů a komunikační interakce mezi nimi, je především nutné vyjít z důkladné analýzy umožňující realizovat vhodnou organizaci komunity agentů. Agent by měl vždy reprezentovat viditelně nezávislou funkční část, tedy například robot jako celek nebo jeho funkční část (pak je robot reprezentován více agenty).

Dále je vždy nutné zvolit softwarové prostředí, v němž budou agenti reprezentující roboty nebo jejich funkční části existovat. Jedná se tedy o volbu vhodné multiagentní platformy, u níž se agenti musejí registrovat (platforma musí zabezpečovat funkci „bílých stránek" – viz (Mařík a kol., 2003)) a která by měla přirozeně zabezpečovat i funkci „žlutých stránek", byť v distribuované podobě.

S volbou platformy je spojena otázka volby vhodného komunikačního jazyka, volba vhodných komunikačních agentů, vyjednávacích algoritmů a scénářů. Ke vzájemnému vyjednávání se obvykle vybírá buď dnes již klasický vyjednávací mechanismus CNP (contract-net-protocol) (Smith, 1980) či některý z typů aukcí.

Mimořádně důležitým aspektem pro zjednodušení komunikace mezi agenty v multirobotickém prostředí (která je v tomto případě komplikována mobilitou robotů a mnohdy nezbytností dynamicky budovat komunikační kanály, často o nízké přenosové kapacitě) je otázka sdílení sémantické informace, zejména o okolním prostředí, např. v podobě map prostředí, a o schopnostech jiných robotů, např. ve formě jejich strukturálních popisů a mnohdy grafických znázornění jejich možností v reálném 3D prostoru, grafů či rovnic dynamických vlastností apod. Tedy *znalostní modely* (acquaintance models) obsahující sociální informaci ve smyslu prací (Mařík a kol., 1997), (Mařík a kol., 2001a) jsou velmi bohaté s relativně vysokým podílem jiné než pouhé textové informace. Udržování aktuálnosti těchto modelů může být poměrně náročné a již samo o sobě představuje jednu z prvotních, a to velmi náročných, úloh pro multirobotické systémy (viz např. kolaborativní explorace prostředí v dalším textu).

V pokročilejších multirobotických úlohách lze s výhodou využívat výsledků výzkumu v oblasti multiagentových systémů, např. lze pracovat s měrami důvěry ve schopnosti partnerských robotů, realizovat složité vyvozovací a usuzovací procesy, učit se na základě vlastních zkušeností (ať již individuálně nebo v týmu) apod. Teprve při využívání této „inteligenční nadstavby" multiagentových systémů, která podporuje přirozené trendy v multirobotických systémech směrem ke zvyšování inteligenční úrovně kooperace a synergických akcí, se v plné míře projeví výhodnost agentové reprezentace v multirobotické oblasti.

Nasazení multiagentových technologií do oblasti robotiky přináší jisté problémy a omezení, na které je třeba myslet v okamžiku návrhu architektury systému a řídicích algoritmů. Tato omezení jsou důsledkem faktu, že roboty jsou fyzicky existujícími entitami a operují ve fyzicky existujícím prostředí, tedy:

- při plánování a provádění akcí je nutné uvažovat i o komunikaci a fyzické poloze robotu – zatímco informační agenti obvykle nejsou omezeni v tom, s kým budou komunikovat, v robotických aplikacích je často omezen komunikační dosah, a tím i výběr možných komunikačních partnerů,
- roboty operují v reálném čase – v některých případech je tedy nutné preferovat rychlost reakce na vzniklou situaci na úkor složitosti uvažování o problému a hledání optimálního řešení, jindy je vhodné omezit intenzitu komunikace v důsledku relativně pomalejších reakcí robotu ve fyzikálním prostředí,
- agenti obvykle pracují se znalostí reprezentovanou v symbolické podobě, zatímco roboty získávají informaci o okolním prostředí zpravidla pouze pomocí svých senzorů a je tedy nutné tuto informaci nejprve zpracovat a převést do symbolické podoby,
- reálný svět se dynamicky mění a tyto změny často nelze predikovat – musí být proto kladen větší důraz na robustnost algoritmů při selhání některého zdroje nebo akce, při vzniku nečekané překážky apod.
- roboty musí zároveň s plněním svých cílů provádět i velké množství „podpůrných" aktivit jako je lokalizace v pracovním prostředí, mapování a tvorba modelu prostředí, přičemž všechny tyto aktivity jsou paměťově i výpočetně náročné. Tato skutečnost by měla být uvažována při optimalizaci rozsahu komunikace a algoritmů vyjednávání mezi roboty.

### 20.4.2 Základní vlastnosti multirobotických systémů

Při hledání vhodné struktury multirobotického systému pro řešení zadané úlohy je klíčovým problémem velké množství možných struktur takových systémů. Na návrh systému je možné nahlížet z několika různých hledisek. Dudek se spolupracovníky navrhli následující taxonomii umožňující vzájemné porovnání různých multirobotických systémů (Dudek a kol., 1996):

- **Velikost systému**. Přestavuje celkový počet robotů (entit) ve skupině. Skupiny mohou být jednočlenné, dvojčlenné, malé až skutečně velké co do počtu entit.
- **Komunikační vzdálenost**. Ve většině systémů je omezena vzdálenost, na kterou mohou roboty přímo komunikovat. Podle tohoto kritéria rozlišujeme tři různé systémy:
  - *bez přímé komunikace* – roboty nemohou přímo komunikovat s ostatními,

– *s omezenou komunikací* – roboty mohou komunikovat pouze s těmi roboty, které se vyskytují v jejich blízkosti; důvodem jsou zejména fyzická omezení daná použitou komunikační infrastrukturou,
– *bez komunikačních omezení* – robot může komunikovat s ostatními roboty bez ohledu na vzdálenost od nich.

• **Komunikační topologie**. Možnost komunikovat s ostatními roboty nemusí být omezena pouze fyzickou vzdáleností mezi roboty, ale také logickými vazbami v systému. Komunikace s ostatními může být robotu umožněna pouze v rámci určité hierarchie systému nebo jen se specifickým typem robotů. Uveďme základní rozdělení:
– *všesměrové vysílání* (broadcasting) – roboty mohou zaslat zprávu všem ostatním robotům současně,
– *adresné vysílání* (unicasting) – robot může předat zprávu libovolnému jinému robotu na základě znalosti jeho jména nebo komunikační adresy,
– *stromová komunikační topologie* – hierarchie systému má podobu stromu a roboty mohou komunikovat pouze v rámci této hierarchie (tedy předávat zprávy funkčně nadřazeným nebo podřazeným robotům),
– *grafová komunikační topologie* – funkční uspořádání skupiny robotů je popsáno pomocí obecného grafu. Takováto topologie umožňuje robustnější propojení systému než v případě stromu, protože redundantní linky zvyšují potenciální schopnost interakce mezi roboty, což je významné zejména při poškození topologie (ztráty komunikace či dokonce celého robota).

• **Šířka komunikačního pásma**. Komunikace mezi roboty není charakterizována pouze fyzickou vzdáleností a topologií systému, ale také cenou za vyslání informace. Tato cena je kombinací několika faktorů, jako je například strojový čas nutný k vy-slání a zpracování zprávy nebo blokací dalších akcí během vysílání jiné akce. Lze uvažovat následující varianty:
– *neomezenou šířku pásma* – v tomto případě lze cenu komunikace ignorovat. Tento přístup se uplatňuje zejména v teoretických modelech a může vést k podobnému chování skupiny robotů, jako by v systému byla centrální entita. Neomezená šířka pásma umožňuje výměnu úplné informace mezi všemi členy skupiny;
– *omezenou šířku pásma* – při komunikaci mezi roboty je pak nutné počítat s cenou každé zprávy. Robot je schopen vyslat pouze omezené množství zpráv nebo je limitován objemem přenesených dat;
– *implicitní komunikaci fyzickou interakcí* (implicit motion communication) – roboty si předávají zprávy prostřednictvím své interakce s okolím, cena komunikace je pak dána cenou pohybu. Příkladem může být způsob, jakým komunikují včely, když „tancem" předávají zprávy družkám ve své blízkosti. Tento způsob komunikace je detailněji popsán v dalším výkladu;
– *nulovou šířku pásma* – roboty nejsou schopny navzájem komunikovat. Tento případ zpravidla znemožňuje koordinaci jejich akcí.

• **Rekonfigurovatelnost systému**. Je mírou schopnosti měnit organizační strukturu systému. Autoři publikace (Dudek a kol., 1996) pohlížejí na rekonfigurovatelnost systému jako na změnu fyzické struktury skupiny entit v prostředí (např. změnu formace) či změnu hierarchických vazeb v systému či změnu schopností jednotlivých entit.

Systémy mohou být rekonfigurovatelné zcela libovolně, nebo jsou možnosti rekonfigurování v určitém rozsahu omezeny. V nejhorším případě mohou existovat skupiny entit bez možnosti jakékoliv rekonfigurace.

- **Skladba entit ve skupině**. Může být nejrůznější: může se jednat například o zcela identické entity, skupinu homogenních entit (stejný hardware, jiný software) či entit zcela heterogenních.

Tento výčet hledisek není samozřejmě úplný, nicméně jejich srozumitelnost umožňuje rychlé a snadné porovnání různých systémů a usnadňuje návrh architektury systémů nových.

Různí autoři přidávají často další hlediska. Zajímavým hlediskem je například stupeň centralizace či decentralizace systému (Cao a kol., 1997). Lze tak rozlišovat distribuované systémy se silně centrálním rozhodováním, s částečně distribuovaným rozhodováním a řízením či bez jakékoliv centralizované aktivity.

### 20.4.3 Výměna informací mezi roboty

Jedním ze základních projevů kolektivní inteligence je vzájemná výměna informace mezi entitami tvořícími tento kolektiv. Stejně jako v případě skupin živých organismů se i v multirobotických systémech můžeme setkat s různými způsoby vzájemné interakce mezi jednotlivými roboty. Možnost vzájemné výměny informací je zde základním předpokladem k tomu, aby při plnění úlohy mohly vzájemně koordinovat své akce a postupovaly tak efektivně ke splnění svých cílů.

Dle Junga a Zelinského je vzájemná interakce mezi entitami charakterizována následujícími vlastnostmi (Jung, Zelinsky, 2000):

- *Interakční vzdáleností*: fyzickou vzdáleností mezi entitami během vzájemné interakce. Tato vzdálenost je nulová při přímém fyzickém kontaktu, nebo může být omezena přímou viditelností senzorů či při větších vzdálenostech je dána dosahem vytvořeného komunikačního kanálu.
- *Simultánností interakce*: časovou prodlevou mezi vysláním a příjmem informace. Prodleva může být minimální v případě přímého kontaktu nebo naopak velmi dlouhá, pokud je komunikační kanál kanálem s významným zpožděním (entita komunikuje např. zanecháním pachové stopy v prostředí).
- *Explicitností komunikace*: tato vlastnost vyjadřuje zřetelný záměr vysílající entity vyslat podnět. Vedlejším efektem vykonání určitých akcí může být totiž i vyslání podnětu, aniž by to bylo jakýmkoliv záměrem entity.
- *Kvalitou interpretace*: tato vlastnost se vztahuje na vysílající i přijímající entitu. Zachycuje a ohodnocuje složitost interpretačního procesu, který přiřazuje podnětu význam. Je možné, že konkrétní podnět bude mít odlišný význam pro vysílající i přijímající entitu, popřípadě nebude mít pro nikoho jiného než vysílající entitu význam vůbec žádný. Příkladem složitého interpretačního procesu je například zpracování zprávy v přirozeném jazyce a její porozumění.

#### *20.4.3.1 Implicitní komunikace*

Implicitní komunikace představuje předávání informace skrze prostředí, ve kterém se roboty pohybují. Roboty mohou svými akcemi toto prostředí modifikovat a předávat tak

právě prostřednictvím změn v prostředí ostatním robotům informaci. Aby se tato informace k ostatním robotům v systému dostala, musí být tyto roboty vybaveny senzory, které jim umožní příslušné změny prostředí detekovat.

Implicitní komunikaci se často nelze vyhnout. Tím, že robot interaguje s okolním prostředím, v něm zanechává informace o svém působení, aniž by to bylo jeho záměrem. Implicitní komunikace je tedy velmi robustní – nelze ji vypnout.

Pro vlastnosti implicitní komunikace platí, že vzdálenost interakce závisí na schopnostech senzorů, jimiž je robot vybaven. Simultánnost interakce je značně variabilní, explicitnost signalizace je obvykle poměrně nízká.

Výhodou implicitní komunikace je její jednoduchost, často se děje automaticky. Je nezávislá na komunikační infrastruktuře. Mezi její nevýhody naopak patří to, že je omezena pouze na informaci, kterou lze předat skrze prostředí a roboty musí být vybaveny příslušnými senzory.

#### *20.4.3.2 Komunikace prostřednictvím detekce stavu (stavová komunikace)*

Stavová komunikace je založena na přenosu informace prostřednictvím sledování a vyhodnocování chování ostatních robotů (často s využitím nejrůznějších modelů chování), event. na komplexním vyhodnocování celkové situace (stavu řešení úlohy).

K tomu, aby mohl robot využívat stavovou komunikaci, musí splnit dva předpoklady:

a) musí být schopen odlišit roboty od ostatních objektů v prostředí a

b) musí být schopen vnímat a interpretovat jejich akce.

Obzvláště interpretace akcí je velmi složitý problém, zvláště pokud je výsledné chování robota tvořeno kompozicí většího počtu jednodušších, elementárních typů chování. Pro přesnou interpretaci chování je také zpravidla potřeba sledovat chování robotu po delší dobu tak, aby bylo možné vytvořit (ať již na základě statistického vyhodnocení, či pomocí strojového učení) vhodné, dostatečně přesné modely chování entit. Nedostatek času k pozorování však bývá hlavní překážkou tvorby efektivních modelů.

Zajímavé aplikace stavové komunikace nalezneme zejména v kompetitivních robotických soutěžích, jako je například RoboCup Soccer. V těchto aplikacích přináší schopnost robotu rozpoznat a předvídat akce ostatních robotů velkou výhodu v duelu.

Vlastností stavové komunikace je, stejně jako u implicitní komunikace, závislost interakce na vlastnostech senzorů. Nejčastěji používaným senzorem pro stavovou komunikaci je v současné době kamera. Ta umožňuje využít pro rozpoznávání akcí metody a algoritmy z oblasti počítačového vidění. Z hlediska explicitnosti signalizace může být stavová komunikace záměrná i neúmyslná. Záměrná stavová komunikace je například rozsvícení světla na robotu v průběhu plnění úkolu. Naproti tomu časová prodleva mezi vysláním signálu a jeho přijetím je zanedbatelná, simultánnost interakce je tedy vysoká. Protože stavová komunikace se týká pouze aktuální informace o aktuální činnosti robotu, tato informace musí být také ihned zpracována.

Výhodou stavové komunikace je její robustnost a nezávislost na komunikační infrastruktuře. Nevýhodou je, že informace, kterou dokáže robot o ostatních získat, je omezena vlastnostmi jeho senzorů. Proces interpretace této informace je zpravidla velmi složitý.

#### *20.4.3.3 Explicitní komunikace*

Pojmem explicitní komunikace se rozumí záměrné vysílání a přijímání zpráv. Tato komunikace bývá zajišťována speciální hardwarovou infrastrukturou. V multirobotických aplikacích se zpravidla jedná o využití komunikačních mechanismů jako jsou bezdrátové LAN (definované standardy IEEE 802.11x), infračervené vysílání nebo nejnověji například Bluetooth.

Možnosti adresného i všesměrového vysílání zpráv mají omezení daná:

- fyzickými vlastnostmi použité komunikační infrastruktury a protokolu přenosu zpráv,
- logickými omezeními vyplývajícími z hierarchického uspořádání systému, kde jeden z robotů může například představovat vedoucího týmu robotů a komunikuje se všemi členy týmu současně pomocí všesměrového vysílání, zatímco členové týmu se mohou obracet pouze na svého vedoucího pomocí adresného vysílání.

Vlastností explicitní komunikace je závislost vzdálenosti interakce na vlastnostech použité komunikační infrastruktury. V případě použití bezdrátových LAN dle standardů IEEE 802.11x se vzdálenost pohybuje v desítkách metrů při komunikaci uvnitř budov a ve stovkách metrů ve venkovních prostorách. Vysílající i přijímající robot se aktivně podílí na komunikaci, ta je tedy záměrná (signálová explicitnost). Odezva na vysílání musí být okamžitá, robot musí přijímat zprávu ve stejném okamžiku, kdy ji jiný robot vysílá (vysoká simultánnost interakce).

Výhodou explicitní komunikace je snadnost jejího použití a možnost předat libovolnou informaci. Robot není omezen množinou signálů jako v případě implicitní a stavové komunikace. Nevýhodou explicitní komunikace je její závislost na použité komunikační infrastruktuře a nebezpečí zahlcení komunikačních kanálů.

### 20.4.4 Zvláštnosti návrhu multirobotických systémů

Typ úlohy, kterou má multirobotický systém řešit, přímo ovlivňuje jeho organizační strukturu i nároky na použitou komunikační infrastrukturu.

Pokud roboty operují v omezeném, předem známém prostoru (jako jsou například sklady, letištní a tovární haly), je možné zde připravit takovou komunikační infrastrukturu, která umožní využít centrální prvky v robotické komunitě. Takovou komunikační infrastrukturu může zajistit například větší množství přístupových bodů k bezdrátové síti pokrývající celý prostor. Tím vlastně vytvoříme systém s neomezenou komunikační vzdáleností, kde může každý robot v každém okamžiku komunikovat s libovolným jiným robotem.

V případě průmyslových systémů může být centrálním prvkem počítač zpracovávající objednávky, který poté na jejich základě rozdělí práci mezi jednotlivé roboty. Pokud však roboty operují v neznámém prostředí nebo v prostředí, kde není zajištěna komunikační infrastruktura, je vzájemné předávání informací obvykle zajištěno pomocí *ad-hoc komunikačních sítí*. V takových systémech zpravidla nelze využít centrální prvky, protože topologie sítě se dynamicky mění a mohlo by tak dojít k přerušení komunikační linky mezi robotem a centrálním prvkem. Příkladem takových úloh mohou být záchranné a vojenské operace nebo planetární průzkum.

Použití centrálních prvků přináší některé výhody, jako například možnost globální optimalizace při plánování a rozvrhování akcí, soustředění problémově orientované znalosti do jednoho místa nebo možnost sběru a fúze dat. Centrální prvek může tvořit

globální plán pro všechny roboty a zjednodušuje tak mechanismy koordinace a kooperace (Riley, Veloso, 2006).

Přítomnost centrálního prvku v systému ale na druhou stranu přináší i množství omezení. S rostoucím počtem entit v systému nastávají problémy s přenosem dat do centrálního uzlu a s jejich zpracováním. Důvodem jsou omezené kapacity komunikačních linek a omezený výpočetní výkon. Také hledání optimálního řešení je v případě komplexních úloh výpočetně velmi náročné a čekání na výsledek může velmi snižovat reaktivitu systému na dynamické změny okolního prostředí.

Dále je v případě selhání centrálního prvku nebo komunikace s ním ohrožena či přímo znemožněna funkcionalita celého systému. Důvodem takového selhání může být zničení tohoto prvku, poškození komunikačního zařízení nebo i – v nejprozaičtějším případě – nedostatek energie v bateriích.

Organizační struktury bez centrálního prvku přinášejí větší robustnost a flexibilitu. Selhání jednoho prvku nevede ke zhroucení celého systému. V systémech s takovou strukturou jsou na jednotlivé entity kladeny vyšší nároky na schopnosti vzájemně vyjednávat, distribuovaně tvořit plány a koordinovat své aktivity.

Při návrhu organizační struktury systému, který bude řešit konkrétní úlohu, je tedy nutné přihlížet i k tomu, jaká komunikační infrastruktura bude v průběhu řešení k dispozici. Možnost vzájemné komunikace totiž přímo ovlivňuje přiřazení rolí jednotlivým entitám v systému či to, jakou problémově orientovanou znalost musí mít k dispozici.

#### *20.4.4.1 Řešení problematiky komunikační nedostupnosti*

V případě, že při řešení úlohy není k dispozici komunikační infrastruktura pokrývající celý prostor a roboty tak musí být propojeny pomocí ad-hoc sítí, mohou nastat situace, kdy se robot stane pro ostatní roboty v systému komunikačně nedostupný. Důvodem mohou být mimo poruchy robota a selhání či přetížení komunikačních linek také dynamické změny v topologii ad-hoc mobilních sítí. Nemožnost vzájemné komunikace přináší celou řadu problémů s tvorbou plánů a koordinací aktivit. Problematika komunikační nedostupnosti je tak předmětem intenzivního výzkumu v oblasti multirobotických i multiagentových systémů.

Dle (Šišlák a kol., 2005a) je *komunikační nedostupnost* definována jako situace, kdy si dva roboty (které vzájemně vědí o své existenci) chtějí vyměnit zprávu, ovšem tento záměr se nepodaří uskutečnit. K tomu, aby v systému došlo k situaci komunikační nedostupnosti, musí současně platit všechny tři podmínky, které definice obsahuje:

- *Znalost o existenci robota* – může nastat případ, kdy se robot rozhodne zaslat zprávu jinému robotu, který už není součástí systému (byl zničen, odregistroval se apod.). Proto je nutné, aby si robot udržoval co nejaktuálnější informace o ostatních robotech. K účelu registrace existujících robotů v komunitě obvykle slouží mechanismus, známý z oblasti multiagentních systémů jako mechanismus bílých stránek (*white-pages*).
- *Záměr komunikovat* – komunikační nedostupnost se totiž nemůže projevit, pokud robot neměl v úmyslu komunikovat.
- *Selhání komunikace* – důsledkem toho, že se vysílajícímu robotu nepodaří doručit zprávu, je to, že se přijímající agent nedozví o jeho záměru komunikovat. Součástí mnoha komunikačních protokolů je proto také zaslání „potvrzující" zprávy, která

informuje odesílatele o úspěšném doručení zprávy. Nicméně pro to, aby si obě strany mohly být úplně jisté tím, že celý komunikační akt proběhl korektně, by bylo třeba těchto potvrzujících zpráv zaslat více, ideálně nekonečné množství.

Existuje několik důvodů vzniku komunikační nedostupnosti v multirobotických systémech s multiagentním typem komunikace:

- *Fyzický pohyb robota a následné změny ad-hoc komunikační sítě* – nedostupnost může nastat v případě, že roboti používají ke vzájemné komunikaci rádiové spojení s omezeným dosahem (IEEE 802.11x). Robot pak může komunikovat pouze s roboty nacházejícími se v jeho blízkosti, přesněji v dosahu tohoto spojení. Pohyb robota může způsobit dynamické změny v topologii komunikační sítě a tím i výpadky spojení.
- *Nespolehlivost komunikačních linek* – tento případ zahrnuje selhání linek a jejich přetížení. Příčinou může být například rušení rádiových vln jiným vysílačem pracujícím na stejné frekvenci. Nedostupnost je v tomto případě charakterizována ojedinělými, periodickými či trvalými výpadky nebo velkým zpožděním zpráv.
- *Migrace agentů* – v okamžiku, kdy agenti migrují (jejich kód i stav je přenášen na jiný hostitelský počítač popř. na jiný robot), nejsou schopni přijímat a zpracovávat zprávy. Migrace agentů také způsobuje změnu jejich komunikační adresy a vysílající entita se nemusí o této změně dozvědět.
- *Nepřátelské prostředí* – pokud roboty operují v nepřátelském prostředí, mohou nastat situace, kdy je nutné zachovat rádiový klid, aby nedošlo k odhalení jejich přítomnosti nebo polohy. Takový robot je pak považován za nedostupného. Existuje i možnost, že robot sám nemůže vysílat, ale je schopen informace pasivně přijímat.
- *Nedostupnost informací* – mohou nastat situace, kdy se agenti navzájem dohodnou pouze na výměně určitého specifického typu informací a utajují své privátní informace. Neúspěšný pokus o přístup k těmto privátním znalostem je pak považován za specifickou formu nedostupnosti. Tato situace nastává zejména v prostředí, ve kterém neoperují pouze kooperativní roboty, ale systém obsahuje i roboty s vlastními (mnohdy utajovanými) zájmy.

Existuje několik přístupů k tomu jak řešit problematiku komunikační nedostupnosti, a to jak pro případy, kdy s cílovým robotem nelze komunikovat přímo, ale pouze prostřednictvím mezilehlých (retranslačních) robotů, tak i situace, kdy se robot úplně oddělí od skupiny a není tedy komunikačně dosažitelný žádným jiným robotem:

- *Přeposílání zpráv* (relaying) – klasický přístup, kdy je zpráva postupně šířena komunikační sítí. O přeposílání zpráv se často stará speciální skupina retranslačních agentů, jejichž úkolem je vytvářet a udržovat komunikační propojení mezi prvky sítě, mezi kterými neexistuje přímá komunikační viditelnost. Protokoly pro šíření zpráv v takových systémech jsou v současnosti intenzivně zkoumány, zejména v souvislosti s prudkým rozvojem mobilních komunikací. Přístup založený na přeposílání zpráv má několik nedostatků:
  - omezenou životnost baterií, kdy všechny zprávy jsou posílány vícekrát, což je energeticky náročné,
  - sníženou informační propustnost komunikačních kanálů, protože přeposílání zpráv může vést až k zahlcení komunikačních linek; v některých případech je nutné zavést systém priorit zpráv, kdy zprávy s vyšší prioritou jsou propagovány sítí přednostně,

– obtížnou údržbu komunikačních linek v dynamických systémech, neboť protokoly pro šíření zpráv potřebují pro svou efektivní činnost co nejpřesnější informaci o aktuální topologii systému. Informace o změnách topologie se tak musí neustále šířit systémem, což samo o sobě spotřebovává část zdrojů.

• *Vzdálené povědomí* (remote awareness) – mechanismus umožňující nedostupnému robotu přístup k informacím o zbytku komunity. Jednou z možností, jak toho dosáhnout, je udržovat si sociální modely s informacemi o schopnostech, cílech, závazcích, zdrojích a adresách ostatních robotů. Znalost obsažená v těchto modelech pak může být využita v situacích, kdy není možné získat aktuální informace prostřednictvím komunikace.

• *Mechanismus zprostředkovatelských služeb* – zajišťuje možnost působit vzdáleně i v místech, která nejsou pro robota aktuálně komunikačně dostupná. Tento mechanismus zpravidla využívá různých typů zprostředkujících agentů (tzv. middle agentů), jako jsou například *matchmakeři* nebo *brokeři* (facilitátoři). U matchmakerů si agenti mohou zaregistrovat služby, které nabízejí ostatním členům komunity. V těchto záznamech pak mohou agenti vyhledávat informace o dostupných zdrojích, i když agent, který je nabízí, není dostupný. Brokeři fungují jako zprostředkovatelé a mohou předávat požadavky na služby nebo zdroje vzájemně nedostupným agentům. Brokeři mohou spouštět nové brokery a udržovat tak jejich počet v systému na takové úrovni, aby byla zajištěna jeho globální funkcionalita.

Speciálním druhem zprostředkujících agentů jsou tzv. *zástupní agenti* (stand-in agents) (Šišlák a kol., 2005a). Každý zástupný agent reprezentuje svůj originál. Díky funkcím klonování a migrace se pak tito agenti šíří systémem. Obsahují část nebo celou sociální znalost svého originálu a algoritmy pro práci s touto znalostí. Zástupní agenti pak mohou vyjednávat s ostatními agenty namísto svého nedostupného originálu. Fakt, že každý zástupný agent reprezentuje pouze jeden originál, předchází potencionálnímu konfliktu zájmů – originální agent si může být jistý, že zástupný agent vždy sleduje jeho cíle. Stejně jako u všech ostatních typů zprostředkujících agentů může nastat problém se synchronizací závazků. To, že více zástupných agentů vyjednává na různých místech systému současně, může způsobit, že přidělují více zdrojů, než má jejich originál v daném časovém okamžiku k dispozici. Tento problém je nutné řešit například pomocí úmluv, závazků nebo vícestupňového vyjednávání.

#### *20.4.4.2 Organizační struktury multirobotických systémů*

Struktury multirobotických systémů vycházejí z vlastností jednotlivých robotů a dostupné komunikační infrastruktury. Organizační struktura systému má významný dopad na jeho výslednou efektivitu. Obecně se problematikou analýzy organizačních struktur a jejich vlastností zabývá obor teorie organizace (Organization Theory).

Organizací systému se obvykle myslí soubor rolí jednotlivých entit v systému, jejich pravomocí a oprávnění a vazeb mezi nimi. Organizace zpravidla předurčuje způsob, jakým spolu budou jednotlivé entity interagovat, a to nejen při řešení okamžitých problémů, ale také z hlediska plnění svých dlouhodobých cílů. Jednotlivé typy organizačních struktur jsou vhodné pro různé aplikace, v některých případech je dokonce nutné kombinovat v rámci systému několik různých struktur současně.

Ucelený přehled různých typů architektur multiagentových (multirobotických) systémů lze najít například v (Horling, Lesser, 2005), (Kolp a kol., 2006) nebo (Pěchouček a kol., 2002); zde uvedeme nejpoužívanější z nich:

- *Hierarchie či hierarchická organizace* agentů (robotů) je patrně nejstarší typ struktury použitý v multiagentových systémech a architekturách distribuované umělé inteligence.
- *Holarchie či holonické organizace* reprezentují systémy založené na vnořených vzájemně si podobných organizacích. Tyto systémy jsou analogií fyzikálních systémů (biologických, astronomických, sociálních) tvořených víceúrovňovou skupinovou hierarchií (Brennan, Norrie, 2001). Například vesmír je tvořen galaxiemi, které se skládají ze slunečních soustav atd. až po subatomární částice. Charakter každého uskupení v těchto systémech je odvozen od vlastností členů této skupiny. Zároveň celé toto uskupení přispívá k vlastnostem jedné nebo více nadřazených uskupení. Koncept holonů a holarchií byl zkoumán zejména ve výrobních a obchodních doménách, kde holarchie a holonické organizace reprezentují systémy, které odpovídají představě vnořených, vzájemně si podobných systémů. Například korporace, firmy, divize, pracovní skupiny a jednotlivci mohou být považovány za holony zajímající místo v holarchii.
- *Aliance* představuje množinu entit sledujících stejné cíle, sdílejících informace o sobě a připravených intenzivně spolupracovat pro dosažení svých cílů v konkrétní úloze. Aliance jsou obvykle dlouhodoběji organizovaná „zájmová sdružení" entit připravených vytvořit dočasnou koalici pro efektivní řešení vybrané úlohy.
- *Koalice nebo tým* je obvykle dočasná skupina entit, orientovaná na dosažení krátkodobého cíle, přičemž po jeho dosažení koalice přestává existovat. Koalice mohou být složeny z kooperativních, ale i soutěživých (kompetitivních) entit. Organizační struktura v koalici je jednoduchá – koalice obsahuje vedoucího, který se chová jako zástupce ostatních členů a zprostředkovává členům koalice kontakt s okolím. Jakmile je koalice zformována (obvykle z členů jedné aliance – není však podmínkou), může být považována za jedinou dále nedělitelnou entitu. Koalice netvoří hierarchickou strukturu, ale je možné, aby jedna koalice byla obsažena ve druhé, nebo aby byla jedna entita členem více koalic zároveň. Předpokládá se, že entity v koalici budou koordinovat a plánovat aktivity (*team-action planning*) tak, aby svými akcemi podporovali dosažení společného cíle (Tambe, 1997). Jednotlivé entity v týmu na sebe vzaly určité role, které jsou potřebné ke kolektivnímu dosažení daného cíle. Obecně se mohou role jednotlivých agentů v týmu v průběhu činnosti měnit.

Z dalších struktur jmenujme *kongregace* tvořené jednotlivci majícími stejné nebo vzájemně zastupitelné vlastnosti; *společenství* (societies) založená za účelem sdílení informací a vzájemné interakce, jejichž chování se řídí společně přijatými pravidly, normami a závazky; *federace*, jejichž členové udělili určitou míru autonomie jednomu delegátu, který skupinu reprezentuje, nebo *maticové organizační struktury*, kde mohou na rozdíl od hierarchické struktury být jednotlivci ovlivňováni více entitami než jen jejich nadřízeným agentem.

### 20.4.5 Správa znalostí v multirobotickém sytému

Aby mohl robot efektivně vykonávat své akce, musí mít k dispozici model okolního prostředí. Model si každý robot, pokud není součástí skupiny, tvoří na základě zpracování dat z vlastních senzorů.

Pokud je ovšem robot součástí kolektivu, může si s ostatními roboty části modelu vzájemně vyměňovat. Model tak bude lépe odpovídat aktuálnímu stavu prostředí, popř. bude budován rychleji. To, jaké informace si mezi sebou roboty vyměňují, závisí na skladbě entit v systému. V případě že jsou entity homogenní ve smyslu reprezentace dat tvořících model prostředí, mohou si zasílat přímo části tohoto modelu na symbolické úrovni. V tom případě lze mluvit o tom, že se komunita snaží budovat společný globální model prostředí.

Zde je nutné rozlišit dva případy:

1. Robot je vsazen do zcela neznámého prostředí a z údajů svých senzorů si buduje model prostředí (který pak může sdílet s ostatními).
2. Model prostředí je dopředu znám (např. pevně zabudované trasy pro roboty či automatické vozíky AGV v tovární hale) a tento model je při inicializaci předán robotu (vozíku). Vozík pak může během pohybu prostředím zaznamenávat dynamické změny v modelu, tj. zvláště překážky v pohybu (např. rozbitý vozík blokující segment trasy) a o nich pak informovat ostatní vozíky (takto je to implementováno například v multiagentovém simulačním systému MAST (Vrba, 2003)).

Obecně pro potřeby této kapitoly poznamenejme, že AGV je speciální případ robotu, který se může pohybovat pouze po předem definovaných trasách a relevantní je tedy pouze jeho lokalizace v rámci tras (nemůže se odchýlit mimo trasu). Odhad souřadnic polohy je tedy jednodušší, limitován pouze na hrany grafu.

Pokud roboty používají různé reprezentace dat v modelu prostředí (entity jsou heterogenní), mohou si zasílat přímo data ze senzorů. Podmínkou je, aby byly vybaveny algoritmy umožňujícími zpracovat i data ze senzorů, jimiž nejsou samy vybaveny. Dalším nedostatkem tohoto přístupu je, že přenos velkého objemu dat ze senzorů klade značné nároky na komunikační infrastrukturu a jejich zpracování je taktéž výpočetně náročné.

Výhodou použití globálního modelu prostředí u robotů se stejnou reprezentací znalostí je, že roboty mohou uvažovat a navzájem vyjednávat za použití shodných informací, což všechny procesy velmi zjednodušuje. Existence globálního modelu prostředí nevyžaduje, aby každý robot s ním byl obeznámen v plném rozsahu – naopak často má k dispozici jen jeho relevantní část. Tvorbě modelů prostředí se ještě podrobněji vrátíme v dalším textu.

Mimo znalosti o prostředí si jednotlivé entity udržují ještě znalost nutnou k řešení problémů (problem-solving knowledge) a sociální znalosti, které obsahující modely chování ostatních agentů, jejich schopnosti, závazky a cíle. Obvykle se jedná o tzv. *znalostní modely* (acquaintance models) – podrobněji viz (Mařík a kol., 2003). Znalostní modely je možné využít k vyhledávání partnerů bez nutnosti přímé komunikace s nimi. Pokud lze chování robotu popsat jednoduchým výpočetním či znalostním modelem, pak údržba modelu jeho chování přímo umožňuje předvídat jeho budoucí akce.

V dalším textu budeme přísně oddělovat *znalostní modely* (zachycující znalosti potřebné například k budování a údržbě modelu prostředí) od vlastních *modelů prostředí*, tedy od vnitřní reprezentace okolního světa v paměti robota.

## 20.5 Schopnosti robotu pro řešení společných úloh

### 20.5.1 Lokalizace robotu v prostředí

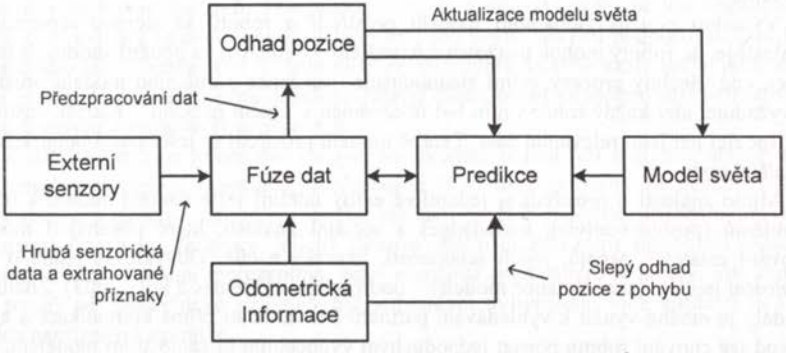
Mobilní robot jako autonomně pracující stroj musí být vybaven systémy, které mu zajistí bezpečnou a bezkolizní interakci s prostředím, ve kterém se pohybuje. Z tohoto důvodu musí být schopen poznávat prostředí, získávat informaci a o své vlastní poloze, vytvářet si interní podobu (model) prostředí a pohybovat se v něm.

Úloha získávání informace o vlastní aktuální pozici (*lokalizace* robotu) v prostoru a vytváření modelu světa (*mapování*) bezesporu patří mezi základní problémy v oblasti autonomních systémů (Cox, 1991). Obě úlohy jsou do velké míry vzájemně svázány, navíc jsou v jistém smyslu komplementární, proto se často řeší současně. Svoji povahou je mapování a globální lokalizace představitelem problému třídy „slepice a vejce". Na přesnou tvorbu modelu prostředí má vliv nejen přesnost prostorových senzorických měření, z nichž se mapa buduje, ale i přesnost určení pozice, odkud jsou senzorická data sejmuta.

Z toho vyplývá, že přesný model prostředí nelze vybudovat bez znalosti přesné pozice senzorických systémů a naopak spolehlivost a přesnost určování aktuální pozice v prostředí je do velké míry podmíněna existencí co možná nepřesnějšího modelu prostředí, vůči kterému se pozice určuje. Tím se kruh uzavírá.

Obecně lze říci, že chyba odhadnuté trajektorie robota silně koreluje s chybami vybudované mapy, čili přesná mapa nemůže být vytvořena bez dostatečně přesné, spolehlivé a robustní lokalizace.

Vzájemná provázanost úloh lokalizace a mapování se v literatuře označuje jako úloha SLAM (Simultaneous Localization and Mapping), vazby jednotlivých úloh jsou ilustrovány na obr. 20.5.




Obr. 20.5 Vzájemné vazby mezi lokalizací a mapováním.

Lokalizací je myšlen proces odhadu aktuálních souřadnic (polohy) robotu na základě informací z dostupných senzorů, jimiž jsou zejména dálkoměrné a odometrické senzory. Pro počáteční odhad pozice se používá odometrická informace získávaná z inkrementálních senzorů, které měří počet otáček koleček robotu odpovídající ujeté vzdálenosti. Hlavním senzorem pro určení pozice jsou pak nejčastěji laserové dálkoměry měřící vzdálenost překážek, podle nichž se robot orientuje. Souřadnice robotu v pracovním prostředí mohou být vztaženy buď k existujícímu modelu prostředí (mapě), nebo k vztažné referenci. Referencí často bývá například počáteční poloha robota.

Je-li výchozí pozice robotu známa, nebo pokud jsou důležité pouze relativní souřadnice mobilního robota vzhledem k výchozí pozici, mluvíme o *úloze sledování polohy* (position tracking). Tyto lokalizační úlohy korigují odometrické chyby vznikající během jízdy, čímž zpřesňují aktuální pozice robotu. V opačné situaci, kdy výchozí poloha není známa a potřebujeme znát absolutní souřadnice robota v modelu světa, mluvíme o *úloze globální lokalizace*. Úloha globální lokalizace je někdy v literatuře nazývaná též jako „*lost robot problem*", kdy robot ví o neznalosti své pozice, či „*kidnapped robot problem*", kdy robot předpokládá, že svoji pozici zná, nicméně byl násilně přemístěn. V druhém případě tak robot musí navíc detekovat vzniklou chybu lokalizace. Jedním ze základních rozdílů mezi oběma úlohami je požadavek na existenci předem známého modelu světa. Zatímco globální lokalizace model světa potřebuje, pro sledování polohy není model prostředí nezbytně nutný, popř. lze využít průběžně vytvářeného modelu během pohybu.

Pokud uvažujeme pohyb robota ve vnitřních prostorech, úloha lokalizace řeší zpravidla odhad tří parametrů: dvou kartézských souřadnic a úhlu natočení robotu. V případě, že lokalizujeme robot v terénu či v prostředí, kde pohyb robota není omezen jednou horizontální rovinou, je nutné odhadovat až šest nezávislých parametrů. Tyto parametry definují polohu pevného tělesa v prostoru, tj. jedná se např. o tři kartézské souřadnice $(x, y, z)$ a tři směrové úhly často označované jako „*roll/pitch/yaw*".

Lokalizační metody lze obecně rozdělit do dvou velkých skupin, a to na metody pracující s neurčitostí polohy a na metody nevyužívající neurčitost.

První skupina metod pracující s neurčitostí je obecně vhodnější pro třídu úloh globální lokalizace. Lokalizace spočívá v odhadu distribuce pravděpodobností výskytu robotu v dané pozici a je tedy založena na teorii pravděpodobnosti (popř. i jiných teorií pracujících s neurčitostí jako Dempsterovou–Shaferovovou, fuzzy atd.). Výsledkem užití těchto metod je množina možných poloh robota spolu s přidruženou mírou jistoty, že se robot v dané pozici skutečně nachází. Výhodou těchto metod je, že poskytují relativně spolehlivé výsledky i v situacích, kdy není možné na základě omezených senzorických měření jednoznačně rozhodnout, v jaké pozici se robot skutečně nalézá. Tato situace obvykle nastává v prostředích, která mají přibližně symetrickou cyklickou strukturu, nebo obsahují výrazné uniformní prvky (chodba s mnoha stejnými dveřmi). Typickým představitelem metod pracujících s neurčitostí jsou metody markovovské lokalizace (Fox a kol., 1999a), s implementací metod Monte Carlo (Thrun a kol., 2001) nebo metody vícehypotézových Kalmanových filtrů (Cox, Leonard, 1994), (Jensfelt, Kristensen, 1999).

Druhá skupina metod zpravidla využívá teorii optimalizace pro určení jedné konkrétní pozice robotu, tyto metody jsou vhodnější pro sledování pozice. Tato pozice může

být určena velmi přesně, ovšem za cenu, že určená poloha může často odpovídat jen lokálnímu extrému. Uvedený fakt tyto metody předurčuje zejména pro sledování pozice a nikoliv pro globální lokalizaci. Metody jsou založeny na hledání translačně-rotační transformace pro optimální sesazení (*matching*) několika po sobě následujících senzorických měření z různých míst v prostoru. Sesazení se provádí na základě určování korespondencí naměřených dat popisujících polohu a tvar každého předmětu tak, aby vzdálenost obrazu předmětu v datech naměřených z různých míst byla minimální. Představitelem těchto metod jsou postupy založené na metodě ICP (*Iterative Closest Point*) (Lu, Milios, 1994) (Lu, Milios, 1997) nebo na korelačních principech (Weiss, Puttkamer, 1995). V principu lze druhou skupinu metod považovat ze speciální případ řešení úlohy s neurčitostí za předpokladu, že se vezme v úvahů nejpravděpodobnější poloha.

### 20.5.2 Mapování prostředí

Pokud mluvíme o úloze výstavby modelu světa, máme v podstatě na mysli proces převádění popisu (tvaru) reálného prostředí do datové reprezentace, která je vhodná pro další zpracování.

Během stavby modelu prostředí, tj. mapování, je třeba řešit několik základních úloh. Jednou z nich je volba vhodné trajektorie pohybu, která zajistí, že senzorický systém robotu bude mít k dispozici dostatečné množství dat z celého mapovaného prostoru. Tuto problematiku řeší plánovací algoritmy pro průzkum a pokrývání prostoru. Volbu vhodné trajektorie lze nahradit vzdálenou teleoperací, během které operátor sám rozhoduje, do kterých míst v mapovaném prostoru je potřeba robot navést, aby vytvořený model pokryl prostor zájmu s dostatečnou kvalitou a rozlišením.

Problematika budování a udržování lokálního modelu světa je popisována (Crowley, 1989) jako třístupňový proces:

- vytvoření abstraktního popisu nejaktuálnějších dat ze senzorů (senzorický model),
- určení korespondencí a sesazení aktuálních dat s existujícím lokálním modelem světa,
- modifikace modelu světa s ohledem na rozdíly získané během sesazování původního modelu světa s novými daty.

Způsob reprezentace reálného světa v datové podobě je určen zejména požadavky, pro jaký účel bude model používán. Z hlediska navigace mobilního robotu je třeba rozlišovat mezi modely podporující lokalizaci robotu, plánování a řízení pohybu. Tyto modely jsou nazývány mapy prostředí. Dle způsobů reprezentace prostředí lze mapy rozdělit na následující druhy:

- senzorické,
- geometrické (čárové, křivkové, polygonální),
- topologické,
- symbolické.

*Senzorické mapy* pracují přímo s hrubými senzorickými daty nebo jen s relativně jednoduše zpracovanými daty. Hrubá senzorická data se využívají na úrovni reaktivních zpětných vazeb v subsystémech předcházení a řešení kolizí, kdy je potřeba rychlých reakcí (Jung, Zelinski, 1996), uplatní se tedy zejména na úrovni řízení pohybu. Typickým příkladem senzorické mapy jsou (pravděpodobnostní) *mřížky obsazenosti* (occupancy

grids) (Elfes, 1990). Mřížkou se rozumí dvourozměrné pole buněk, kde každá buňka reprezentuje čtverec reálného světa a určuje pravděpodobnost, s jakou se v dané oblasti vyskytuje překážka.

Výpočet pravděpodobnosti v každé buňce se skládá ze dvou kroků. Nejprve se vytvoří pravděpodobnostní model senzoru, který umožní určit hodnotu pravděpodobnosti v každé buňce pro jednotlivé senzorické měření, a takto získané pravděpodobnosti se poté kombinují (např. pomocí Bayesova pravděpodobnostního přístupu, věrohodnostního přístupu Dempsterovy–Shaferovy teorie nebo teorie fuzzy množin). Mřížky obsazenosti (Moravec, Elfes, 1985) jsou jedním z modelů využitelným vybranými lokalizačními technikami (Fox a kol., 1999b). Své uplatnění nacházejí zejména v oblasti středně a nízkoúrovňového plánování pohybových trajektorií robotu. Mřížky obsazenosti jsou též často využívány pro vizualizaci prostředí.

Dalším příkladem senzorických map jsou bodové mapy, které jsou vhodné zejména pro podporu lokalizačních technik, neboť jejich interpretace je z hlediska lokalizace poměrně jednoduchá a rychlá.

Velmi efektivní reprezentací prostředí jsou *geometrické mapy*, které patří do skupiny modelů se střední mírou abstrakce popisující objekty v prostředí pomocí geometrických primitiv v souřadném systému. Jako geometrická primitiva jsou brány úsečky, polygony, části oblouků, splinů, či jiných křivek (Shaffer, 1995).

Na nižším stupni abstrakce se pracuje pouze s vektorizovanými daty, která popisují prostředí pomocí jednoduchých primitiv, tj. pomocí úseček bez další přídavné informace o jejich vzájemných relacích. Speciálním případem geometrické mapy je polygonální mapa, která je vyžadována jako vstup vybraných plánovacích algoritmů pro hledání cest v prostoru či pokrývání prostoru.

*Polygonální model prostředí* vychází z vektorového popisu, avšak prostředí a objekty jsou v něm reprezentovány mnohoúhelníky, které se skládají z úseček reprezentujících skutečné hrany objektů v prostředí či hranice dosud neprozkoumaných oblastí. Každá úsečka rozděluje prostor na dvě části, kde jedna část prostoru je tvořena volným prostorem a druhá část obsazeným, popř. neznámým, prostorem. Ohodnocení obsazenosti prostoru je zahrnuto v orientaci jednotlivých úseček, přičemž volný prostor se nachází vždy na levé straně úsečky ve smyslu její orientace. Z orientace úseček následně vychází i orientace polygonů z nich složených, přičemž levotočivý polygon obvykle ohraničuje pracovní prostředí robotu a pravotočivý polygon popisuje překážky umístěné uvnitř vnějšího, hraničního polygonu (tzv. boundary polygon).

Jednotlivé hrany mohou být ohodnoceny mírou spolehlivosti, statistickou významností či počtem měření potvrzujícím skutečnou existenci hrany. K jednotlivým mnohoúhelníkům lze přidat i informaci o typu objektu (stěna, dveře, mobilní překážka atd.), čímž se geometrická mapa začne blížit mapě symbolické.

Pokud mezi hranami jednotlivých objektů zavedeme vzájemné relace, dostáváme polygonální grafový popis prostředí, které je pak možné popsat formou planárního topologického grafu (Přeučil, Štěpán, 1996) viz obrázky 20.6 a 20.7. Graf obsahuje hrany s atributy:

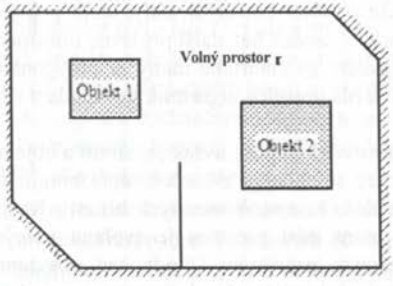
- reálná hrana (může být viditelná i neviditelná),
- virtuální hrana (je vždy neviditelná)

a uzly popisujícími 2D souřadnice hran.

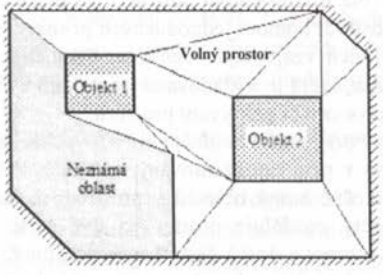
Reálné hrany vždy reprezentují skutečné překážky, kterými není možné fyzicky projet, atribut viditelnosti určuje, zda překážka byla detekována senzory či nikoliv. Hrany v neznámé oblasti prostoru jsou obvykle označeny jako neviditelné. Virtuální hrany jsou určeny pro spojování reálných hran a zajišťují tak spojitost grafu. Virtuální hrany mohou rozdělovat volný prostor na konvexní oblasti, které se následně dají použít jako elementární buňky pro budoucí plánování trajektorií pohybu robotu.

Velkou výhodou polygonálního modelu je snadná reprezentace překážek, ovšem výraznou nevýhodou je poměrně velká složitost automatického budování takovéto mapy, neboť během procesu stavby takové mapy je často nutné senzorická měření správně interpretovat v kontextu reálného světa.

Tvorbou geometrického popisu prostředí se zabývá práce (Kulich, 2003). Autor používá dvou postupů, matematické morfologie a metody rostoucího neuronového plynu. První postup primárně ze senzorických dat buduje mřížky obsazenosti, které následně segmentuje a podrobuje operacím založených na matematické morfologii. Tím dosáhne extrakce hranice z mřížky obsazenosti, eliminace šumu a spojení původně nesouvislých objektů do celku. Výsledkem operací matematické morfologie je posloupnost buněk, které odpovídají kostře hranice obsazených částí prostoru. Posloupnost buněk je pak vektorizována do podoby geometrické mapy prostředí.




Obr. 20.6 Reálné prostředí.




Obr. 20.7 Polygonální grafová reprezentace.

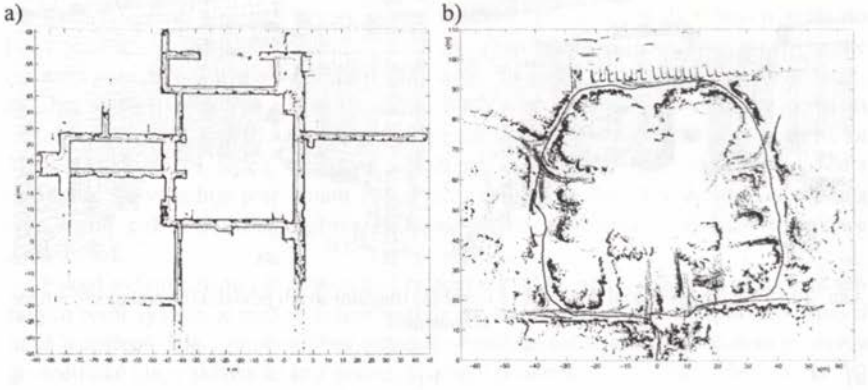
Druhý postup využívající rostoucího neuronového plynu je založen na aproximaci naměřených dat samoorganizující se strukturou, která umožňuje dynamicky měnit počet uzlů struktury. Uzly a hrany neuronové síťové struktury se stávají základem pro následující aproximaci hranic překážek úsečkami.

Geometrické mapy lze použít též pro lokalizaci robotu v prostředí, nebo je lze exportovat do CAD nástrojů, ve kterých se následnými korekcemi vytvoří kompletní CAD model prostředí pro jiná než čistě robotická využití.

*Topologické mapy* neuchovávají informace o absolutních souřadnicích objektů, ale zaznamenávají topologické vztahy mezi objekty v prostředí (Dudek a kol., 1993), (Lefevre a kol., 1994), (Thrun, Bucken, 1996). Mapa pak má formu grafu, kde uzly grafu představují významné objekty v prostoru. Hrany grafu spojují ty uzly v grafu, mezi kterými existuje určitá relace, např. průjezdná cesta. Své uplatnění nalézají topologické mapy též v oblasti vysokoúrovňového plánování ve velmi rozsáhlých prostorech.

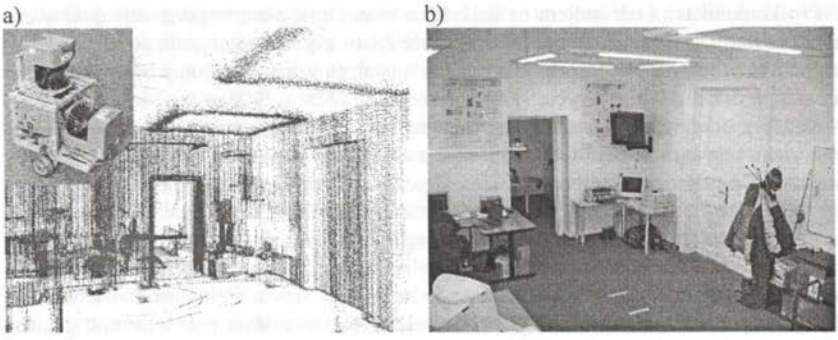
Pro komunikaci s uživatelem na nejvyšším stupni abstrakce se používají *symbolické mapy* obsahující informace, které robot nemůže zjistit svými senzory, ale pomocí kterých lze zadávat příkazy v přirozeném jazyce. Implementace tohoto typu map bývá prováděna v některém deklarativním jazyce – Prolog nebo LISP (Malý, 2005).

Každý z uvedených přístupů k reprezentaci mapy prostředí má svoje klady i zápory. Senzorické reprezentace umožňují pracovat s nepřesnou senzorickou informací, ale jsou naopak velmi náročné na paměť. Topologické reprezentace lze používat pro popis velkých prostorů, neboť jejich paměťová náročnost je malá. Navíc jejich tvorba není na rozdíl od geometrických a senzorických map závislá na znalosti polohy. Na druhou stranu jsou nepoužitelné pro přesnou lokalizaci a zjišťování metrických informací o objektech. Výhodou geometrických map je to, že jsou srozumitelné pro člověka, který tak může apriorní informace o objektech v okolí robota zadávat právě formou geometrické reprezentace. Navíc pro mnoho budov je geometrická mapa hotova ve formě architektonických plánů. Stručně řečeno, každý model prostředí má v robotice své nezastupitelné místo, pro každou úlohu je třeba volit vhodný model prostředí a jeho reprezentaci. Bohužel, převody mezi jednotlivými reprezentacemi jsou zpravidla poměrně obtížné.

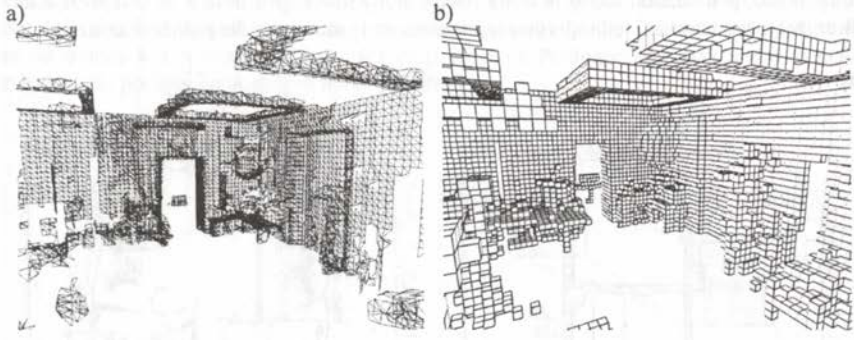



Obr. 20.8 Výsledek souběžné lokalizace a mapování neznámého prostředí – (a) systém chodeb kancelářské budovy (b) venkovní prostory – park na Karlově nám. v Praze.

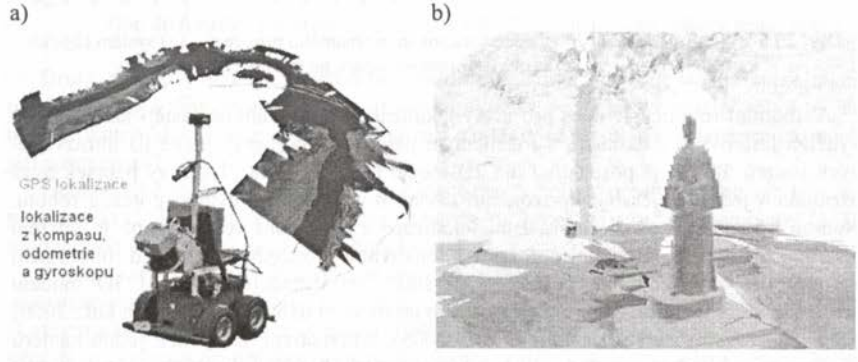
V mobilní robotice se dnes pro účely lokalizace a mapování nejčastěji jako senzorů využívá laserových dálkoměrů s rozmítaným paprskem (skenerů), méně již ultrazvukových sonarů. Pokud je postačující jen 2D model (viz obr. 20.8), laserový paprsek bývá rozmítán v jedné, nejčastěji horizontální rovině rovnoběžné s rovinou pojezdu robotu. Nutnou podmínkou přesné a robustní lokalizace a mapování ve 2D mapě je udržení laserové roviny v horizontální poloze. Ve venkovním prostředí, kde pojezd robotu není rovinný, lze tuto podmínku dodržet jen stěží. To vyžaduje sestavení 3D modelu prostředí, který je nejčastěji tvořen s využitím laserového dálkoměru (Thrun a kol., 2000) nebo kamerového sterea (Lemaire a kol., 2006). Alternativně lze použít jednu kameru a informaci o hloubce prostředí získat z posloupnosti obrázků sejmutých z různých míst navštívených během pohybu robotu (Fitzgibbon, Zissermann, 1998). V případě použití laserového dálkoměru je nezbytné rozmítání paprsku laseru do prostoru buď kýváním




Obr. 20.9 (a) Zobrazení 3D dat získaných robotem z (b) reálného kancelářského prostředí.




Obr. 20.10 Vytvořený model prostředí ve formě (a) triagulovaných povrchů (b) prahované mřížce obsazenosti.




Obr. 20.11 Model venkovního prostředí s informací o odrazivosti povrchu (a) šikmý skener (b) výkyvný skener.

celým rovinným skenerem, nebo použitím konfigurace s konstantním naklopením skeneru. Při konstantním naklopení musí být snímací rovina jiná, než je rovina pojezdu robotu, například k ní kolmá, jak je vidět na obr. 20.9(a). Během pohybu robotu tak snímací rovina protne celý prostor mapovaného prostředí, který je předmětem zájmu. Nevýhodou je nutnost použít jiný senzor k lokalizaci robotu, neboť šikmo nakloněný skener slouží pouze k mapování. V případě použití výkyvného skeneru se doba získání jednoho 3D snímku pohybuje kolem 10 s, což vyžaduje zastavení robotu a komplikuje správnou rekonstrukci pohybujících se objektů. Použití kamer přináší problémy s nestálým osvětlením scény. Nutnost spolehlivě vyhledávat korespondující oblasti v obrazech v reálném čase také vyžaduje značný výpočetní výkon. Při použití jedné kamery nastává problém s přesností určení hloubky prostředí, protože optická osa kamery míří ve směru pohybu robotu. Data získaná kamerami jsou také „řidší" než data z laserových skenerů. V poslední době se také na trhu 3D senzorů objevily třírozměrné (TOF – Time Of Flight) kamery (May a kol., 2006). Jejich relativně nízké rozlišení (asi 250 × 250 pixelů), nepříliš vysoká přesnost a nespolehlivost určení vzdálenosti na hranách snímaných objektů zatím brání jejich širšímu nasazení.

Ve všech zmíněných případech mají získaná data formu 3D mraku bodů, které mohou mít navíc informaci o barvě, odrazivosti nebo vlastnostech jejich okolí. Obvykle je těchto bodů značné množství, což vyžaduje redukci jejich počtu při nízké ztrátě informace o daném prostředí. To je možné přímo (Moening, Dogstone, 2003), nicméně se ukazuje, že před vlastní redukcí je vhodné z těchto bodů nejprve extrahovat některé vlastnosti měřených povrchů. Nevýhodou většiny známých algoritmů pro rekonstrukci povrchů ze 3D mraku bodů je jejich výpočetní náročnost a nutnost zpracovávat všechna data najednou. To vyžaduje přepočítání veškerých nasbíraných dat vždy, když jsou sejmuta data nová, což znemožňuje přímé nasazení těchto algoritmů v úlohách lokalizace a mapování.

Pokud ovšem nejsou požadavky na přesnost a kvalitu rekonstrukce povrchu z nasbíraných bodů vysoké, je možné známé metody modifikovat tak, aby umožňovaly dodávat nově naměřená data z prostředí bez nutnosti přepočítat celou mapu. Příkladem je stavba geometrické mapy skládající se z rovinných ploch (Haehnel a kol., 2003). Senzorické 3D mapy mohou být také konstruovány v reálném čase, problém s paměťovou náročností se v případě 3D map prohlubuje; prostorová mřížka obsazenosti na obr. 20.10(b) se skládá z milionu buněk.

### 20.5.3 Plánování pohybu

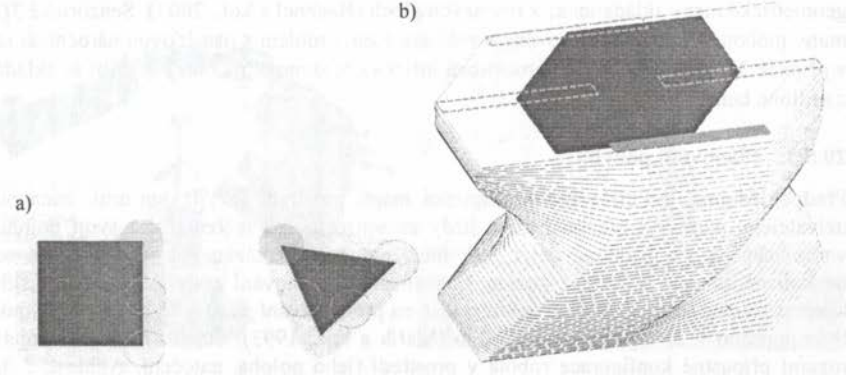
Předpokládejme, že má robot k dispozici mapu prostředí (ať již apriorní, zadanou uživatelem, nebo vytvořenou během jízdy ze senzorických měření), zná svoji polohu v této mapě a cíl, kam má dojet. Pak musí nutně řešit otázku, jakým způsobem se bezkolizně dostane do cílové pozice, tj. řešit úlohu plánování cesty. V nejobecnějším kontextu lze na plánování cesty pohlížet jako na prohledávání stavového prostoru tak, jak bylo popsáno v 2. kapitole prvního dílu (Mařík a kol., 1993). Stavy se v tomto pojetí rozumí přípustné konfigurace robota v prostředí (jeho poloha, natočení, rychlost, ...), přičemž přechody mezi nimi jsou realizovány pohybem robota (za předpokladu, že lze takový pohyb provést). O takto definovaném stavovém prostoru se mluví jako

o *konfiguračním prostoru*. Jelikož je konfigurační prostor příliš rozsáhlý a jeho vytvoření a prohledání by bylo příliš časově náročné, používají se různá zjednodušení.

Nejčastějším zjednodušením je redukce tvaru robotu na hmotný bod a zanedbání jeho dynamických a kinematických vlastností, čímž se problém plánování omezuje na nalezení cesty v mapě 2D prostředí. Plánovací algoritmus pak silně závisí na použitém modelu prostředí. Nejjednodušší je to v případě topologických map majících formu grafu, kde je plánování realizováno hledáním nejkratší cesty v tomto grafu, například algoritmem A* nebo Dijkstrovým algoritmem (Demel, 2002).

Pro plánování v geometrických mapách se nejčastěji aplikuje některý z tzv. kombinatorických algoritmů. Jejich hlavní myšlenkou je získat z geometrické reprezentace graf, v němž se následně realizuje hledání cesty podobně jako u topologických map. Jednotlivé přístupy se tak liší právě způsobem tvorby grafu. K oblíbeným patří zejména *graf viditelnosti*, kde vrcholy odpovídají vrcholům překážek, zatímco hrany spojují pouze vrcholy s přímou viditelností (de Berg a kol., 2000). Jinou možností je rozložit mapu prostředí na oblasti a plánovat na grafu sousednosti těchto oblastí. Z typických dekompozic jmenujme trojúhelníkovou (Bern, 2004), lichoběžníkovou (Collins, 1975) nebo cylindrickou (Chazelle, 1987). Zatímco graf viditelnosti umožňuje zkonstruovat nejkratší cestu, ostatní metody generují cesty daleko od překážek, a tedy bezpečnější, ale nikoliv optimální.

Také na mřížky obsazenosti lze po vhodné diskretizaci aplikovat model plánování na grafu, v němž vrcholy plánovacího grafu jsou reprezentovány volnými buňkami. Hranou jsou pak spojeny ty vrcholy, jejichž odpovídající si buňky spolu sousedí. Jiný přístup používá potenciální pole či jeho vylepšení, tzv. harmonické potenciální pole. Základní princip metody vychází z fyzikální motivace a lze si jej představit jako výsledek určení koncentrace plynu, který se šíří volným prostorem mřížky obsazenosti ze zdroje umístěného ve startovní pozici. Vlastní algoritmus pak vede k řešení soustavy diferenciálních rovnic metodami numerické matematiky (Connolly, Grupen, 1993). Na rozdíl od základní verze, která může uvíznout v lokálním extrému, harmonické potenciální pole garantuje nalezení globálně optimálního řešení.




Obr. 20.12 Konfigurační prostor vytvořený pomocí Minkowského sumy
(a) pro kruhový robot s polygonálními překážkami, (b) pro robot s natočením.

Popsané metody lze použít i pro robot s kruhovým půdorysem. V takovém případě není možné plánovat přímo v zadané mapě, ale v konfiguračním prostoru robotu, který získáme z mapy „nafouknutím" překážek o poloměr robotu. Nafouknutí se provádí pomocí Minkowského sumy překážek s robotem, jejíž výsledek si lze představit jako množinu všech pozic robotu (respektive jeho těžiště) kolidujících s některou překážkou. Důležitým faktem je, že vzniklý konfigurační prostor má tvar 2D mapy, díky čemuž lze v tomto konfiguračním prostoru plánovat pomocí některé z uvedených metod.

Pokud má robot obecný tvar (a je tedy nutné uvažovat i jeho orientaci) aplikuje se Minkowského suma pro každé jeho natočení. Konfigurační prostor je tak trojrozměrný a lze si ho představit jako množinu na sobě naskládaných „nafouknutých" map (každý horizontální řez odpovídá Minkowského sumě překážek s robotem v daném natočení). V tomto případě tedy bod $(x, y, \varphi)$ konfiguračního prostoru určuje, zda robot na pozici $(x, y)$ a natočením $\varphi$ koliduje s překážkou či nikoliv. Bohužel, přidáním třetího rozměru se problém plánování stává NP-těžkým, a proto se řeší aproximačními technikami (Latombe, 1991).

Úloha se stane ještě složitější, pokud budeme chtít vzít v úvahu dynamická a kinematická omezení robotu. Bod konfiguračního prostoru bude totiž kromě pozice a natočení robotu určen dalšími parametry, jako například velikostí a směrem aktuální rychlosti či zrychlení robotu. To vede obecně k $n$ rozměrnému konfiguračnímu prostoru. Kromě toho musí být přechody mezi stavy definovány s ohledem na daná omezení, což konstrukci konfiguračního prostoru, a tím i problém plánování, dále ztěžuje.

Možné řešení spočívá v použití metod založených na vzorkování konfiguračního prostoru, které pracují na iteračním principu. Vzorkovací algoritmus začíná ve startovní konfiguraci a v jejím okolí náhodně vygeneruje nové konfigurace. Pokud existuje bezkolizní cesta mezi těmito dvěma konfiguracemi, potom jsou spojeny hranou. V dalším kroku se náhodně vybere některá z vygenerovaných konfigurací, která se opět expanduje (tj. generují se jí blízké konfigurace). V tomto procesu se pokračuje do té doby, dokud se nenalezne spojení mezi počáteční a koncovou konfigurací, popř. dokud se neusoudí, že takové spojení neexistuje. Bylo popsáno množství metod založených na vzorkování, jež se liší ve způsobu generování nových konfigurací, výběru konfigurace pro expanzi a testování dostupnosti dvou konfigurací. O těchto i mnoha dalších plánovacích algoritmech je možné se dozvědět v přehledové publikaci (LaValle, 2006).

### 20.5.4 Plánování a koordinování akcí

Pro efektivní a robustní řízení kolektivních akcí vícero distribuovaných a autonomních účastníků (robotů, programů nebo i lidí) je potřeba navrhnout vhodné mechanismy plánování akcí, delegace, rozvrhování, koordinace a možného přeplánovávání dohodnutých plánů v případě neočekávané změny okolních podmínek. Zatímco plánování, rozvrhování a delegace úkolů se provádějí zpravidla v čase před zahájením kolektivní akce, koordinace a přeplánovávání probíhá v reálném čase současně s realizací kolektivní akce. Je zjevné, že s detailnějším plánem klesají nároky na koordinaci a přeplánování a naopak.

Základní architektura distribuovaného plánování je definována (DesJardins, Wolverton, 1999) pěti klíčovými fázemi:

1. *Hierarchickým plánováním* – dekompozicí řešené úlohy na dílčí cíle a podúlohy a uspořádáním podúloh do částečně uspořádaného grafu (Blum, Furst, 1997), (Ephrati, Rosenschein, 1997), (Erol a kol., 1994).
2. *Delegací* – přiřazením úloh jednotlivým robotům (Bar-Noy a kol., 2001), (Boutilier a kol., 1999), (Hunsberger, Grosz, 2000).
3. *Detekcí konfliktů* – analýzou dostupnosti zdrojů, identifikací sdílených prostředků a cílů, identifikací kauzálních vazeb mezi jednotlivými akcemi.
4. *Individuálním plánováním a rozvrhováním* – hierarchickou dekompozicí, plánováním a časovým rozvrhováním jednotlivých podúloh přiřazených individuálním robotům.
5. *Spojováním plánů* – integrací a koordinací individuálních plánů a řešením dílčích konfliktů, které vznikly individuálním plánováním (de Weerdt a kol., 2003), (Cox, Durfee, 2005).

Ačkoliv lze všech pět fází provádět centrálně pomocí jednoho komplexního algoritmu, v praxi kolaborativní robotiky a distribuovaných multiagentních systémů je potřeba proces plánování alespoň částečně distribuovat. Primárním důvodem pro distribuci plánovacího procesu je distribuovanost potřebných znalostí a informací. Fáze hierarchického plánování pracuje s poměrně obecnými znalostmi a bývá zpravidla prováděna centrálně. Je-li fáze delegace zajišťována centrálně (a centralizovaný proces pracuje s neaktualizovanými znalostmi o schopnostech individuálních robotů), pak je velmi pravděpodobné, že v následující fázi bude zjištěno velké množství konfliktů. V takovémto případě je nutné předchozí fáze opakovat (*backtracking*). Alternativním přístupem je totální distribuce procesu delegace za použití multiagentních vyjednávacích technologií, např. CNP (Smith, 1980), aukční řízením (Sandholm, 2002), (Hunsberger, Grosz, 2000) apod. Spolupracující roboty nabízejí robotu zodpovědnému za delegaci přesně aktualizované informace o dostupnosti bezkonfliktních zdrojů. Fázi individuálního plánování musí řešit každý z robotů individuálně a fáze spojování plánů je opět řešena s pomocí multiagentního vyjednávání.

V případě nepřesného plánu (například vytvořeného bez fáze detekce konfliktů nebo spojování plánů) či v dynamickém prostředí, kdy se informace, na jejichž základě bylo naplánováno, mění velmi rychle, rostou nároky na proces koordinace a přeplánování v reálném čase. Vzhledem k tomu, že pro koordinaci a přeplánování je k dispozici významně méně času než pro plánování, musejí být koordinační mechanismy navrženy obzvláště efektivně a doménově specificky (pro některé koordinační přístupy viz např. (Mařík a kol., 2001a)).

Důležitým faktorem pro efektivní koordinaci a realizaci distribuovaného plánu jsou mechanismy, které zajistí persistenci chování jednotlivých robotů vzhledem k dohodnutým cílům, a hlavně proces notifikace v případě odchýlení se od plánu. Jedním z možných přístupů je zavedení koordinační autority, která by rovněž monitorovala dodržování individuálních plánů. Vzhledem k tomu, že existence centrálního řídicího prvku významně snižuje robustnost, ideální alternativou jsou distribuované koordinační algoritmy (ovšem s vyšší náročností na komunikační provoz). V multiagentní komunitě byly teoreticky velmi dobře zpracovány a v praxi použity technologie závazků a úmluv (Mařík a kol., 2001b), (Excelente-Toledo a kol., 2001), které formálně specifikují okolnosti, za jakých lze odstoupit od individuálního plánu. Zatímco v kompetitivních komunitách se jedná především o formu penalt (tj. náhrad za škody způsobené odstoupením od plánu),

v kooperativních komunitách robotů se jedná výhradně o technologii notifikace o odstoupení od individuálního plánu.

## 20.6 Simulační nástroje pro vývoj

Realizace úloh z oblasti inteligentní robotiky zahrnuje řešení podproblémů téměř ze všech oborů umělé inteligence, od vnímání okolního prostředí, řízení, uvažování s neurčitou informací až k plánování, rozvrhování a koordinaci akcí. Programování robotických systémů se navíc potýká s celou řadou systémových problémů, např. s omezeními danými fyzickými parametry robotů, jejich dynamikou, omezeným výpočetním výkonem a paměťovou kapacitou řídicích jednotek, omezenou přenosovou kapacitou komunikačních kanálů a jejich nespolehlivostí.

Tato kombinace problémů z oblasti umělé inteligence a systémového inženýrství vede k tomu, že výsledné systémy jsou obvykle velmi složité. Tvorba některých částí takovýchto systémů může být ovšem velmi zdlouhavá a jednotvárná. Uvažujme například, že se systém bude skládat z velkého počtu mobilních robotů různých typů. Přestože jejich výsledná funkcionalita a schopnosti mohou být obdobné, rozdíly ve velikosti, tvaru, komunikačních protokolech a přístupu k senzorům a aktuátorům vedou k tomu, že je obvykle nutné program napsaný pro jeden robot před jeho přenesením na robot jiného typu výrazně upravit. V takovém případě pak i řešení jednoduchých úloh představuje velké množství práce. Výzkumníci se tak často namísto hledání efektivního řešení zadané úlohy soustřeďují na problémy implementační.

Hlavním důvodem je nedostatek univerzálních knihoven a rozhraní, které by umožnily přepoužití již jednou vyvinutých programů a algoritmů bez ohledu na použitou platformu nebo programovací jazyk, což by vedlo k rychlejšímu a efektivnějšímu vývoji robotických systémů.

Významnou podporu urychlení realizačních procesů poskytuje simulace. Zvláštní skupinu simulačních nástrojů tvoří simulační nástroje pro agentové systémy, které přirozeně využívají agentovou architekturu. Pro multiagentně organizované multirobotické systémy je simulace mimořádně důležitá: Takováto simulace totiž umožňuje predikovat chování celé skupiny robotů, volit, modifikovat a testovat nejen reprezentaci znalostí v jednotlivých agentech, ale i vyvíjet optimální algoritmy a scénáře vyjednávání, koordinace a kooperace bez nutnosti provádět experimenty s reálnými roboty. V ideálním případě lze vyzkoušené algoritmy komunikace, řízení, využívání znalostí a vyjednávání přímo přenést do fyzických robotů, a to i krok po kroku (kdy část skupiny kooperujících robotů je simulována, jiná už fyzicky existuje a funguje).

V poslední době vznikla řada projektů zaměřených na tvorbu simulačních nástrojů různé úrovně:

- *systémy pro nejnižší úroveň simulace fyzických vlastností senzorů a akčních členů* – asi nejznámějšími a nejčastěji používaným jsou výsledky projektu Player/Stage/ Gazebo (USC Robotics Research Lab, http://robotics.usc.edu).
- *obecné otevřené nástroje vyšší úrovně*, vhodné pro simulaci procesů plánování, vyjednávání, koordinace a kooperace v nejrůznějších (nejen přímo robotických) úlohách – zde může být příkladem poměrně kvalitní multiagentní nástroj A-globe z ČVUT FEL (Šišlák a kol., 2005b).

• *specializované agentově-orientované simulační nástroje nejvyšší úrovně* umožňující simulovat poměrně komplikované úlohy ve velmi složitém prostředí s využitím problémově orientovaných knihoven – zde je možné jmenovat jako významného představitele této skupiny simulační systém MAST firmy Rockwell Automation, umožňující simulovat agentové systémy pro přepravu materiálu, ale zejména pro řízení průmyslové výroby ve výrobních úsecích s větším počtem robotů a autonomních vozíků (Vrba, 2003). Vyvinuté algoritmy, užité v simulaci, je možné přímo převést do fyzických systémů řízení výroby, včetně nasazení kooperujících robotů.

### 20.6.1 Player, Stage, Gazebo

Cílem tohoto projektu je vytvořit standardní vývojovou platformu, která urychlí vývoj robotických systémů. Projekt je volně přístupný a přispívají do něj vývojáři z mnoha předních univerzit i průmyslových firem zabývajících se robotikou. Projekt se skládá ze dvou nezávislých částí. První z nich je Player – síťové rozhraní umožňující přístup k různým typům mobilních robotů a senzorů. Druhou částí jsou robotické simulátory Stage a Gazebo.

Další výhodou použití tohoto softwaru je mimo usnadnění vývoje i možnost replikace experimentů, a tím i vzájemného porovnání vyvinutých algoritmů.

**Player**. Projekt zahrnuje čtyři hlavní části:

• *Player Abstract Device Interface (PADI)* – tato vrstva definuje standardní rozhraní a tím zajišťuje možnost přepoužití kódu pro různé typy zařízení. PADI definuje syntaxi a sémantiku dat, která se vyměňují mezi hardwarem robotu a řídicím algoritmem.

Pro přístup k jednotlivým hardwarovým zařízením používá Player jejich abstrakci do podoby znakových zařízení (koncept převzatý z oblasti tvorby operačních systémů). Standardní rozhraní ke znakovým zařízením nabízí pět operací – otevření a zavření pro kontrolu přístupu k zařízení, čtení a zápis pro manipulaci s daty a ioctl (input/output control) pro synchronní řídicí operace. Tato abstrakce tedy definuje způsob přístupu k zařízení, nepopisuje však vlastní formát datových proudů. Každé hardwarové zařízení může využívat jiný formát dat a jeho řídicí kód musí být tedy napsaný speciálně pro toto zařízení. Player proto definuje model rozhraní–ovladač. Rozhraní specifikuje obsah datového proudu, tj. zobrazuje vstupní datové proudy na data ze senzorů, výstupní datové proudy na příkazy aktuátorům a na příkazy týkající se konfigurace zařízení. Kód, který implementuje rozhraní, tj. konvertuje data mezi nativním formátem zařízení a formátem, který vyžaduje rozhraní, se nazývá ovladač. Ovladač je většinou specifický pro konkrétní zařízení nebo skupinu zařízení od jednoho výrobce.

Model rozhraní–ovladač sdružuje zařízení do skupin podle jejich logické funkcionality (např. sonary, kamery), takže zařízení z jedné skupiny nabízejí z pohledu uživatele přibližně stejné funkce.

• *Protokol přenosu zpráv* – tento protokol implementuje PADI spolu s dalšími datovými a řídicími strukturami umožňujícími inspekci a řízení chování samotného Playeru běžícího na cílovém zařízení. Protokol definuje také standardní hlavičku, kterou obsahuje každá zpráva, a metadata pro správnou interpretaci zpráv.

- *Přenosovou vrstvu* – zajišťuje přenos zpráv mezi jednotlivými zařízeními a řídicími algoritmy. V současné době je v rámci projektu implementováno několik mechanismů přenosu – TCP, JINI, CORBA.
- *Implementaci* – dosud bylo implementováno PADI, protokol přenosu zpráv, transportní mechanismy a mnoho ovladačů zařízení. Všechny tyto části jsou k dispozici jako volně dostupné knihovny. Jejich nejčastější použití je pomocí spustitelného serveru nazývaného Player. Tento server po svém spuštění načte konfigurační soubory, vytvoří instance příslušných ovladačů zařízení a spravuje připojení klientských programů k těmto ovladačům. Řídicí program vytvořený uživatelem běží obvykle jako klient mimo tento server a k jednotlivým zařízením tedy přistupuje pomocí přesně definovaného rozhraní. To spolu s protokolem přenosu zpráv umožňuje tvorbu klientských řídicích algoritmů v libovolném programovacím jazyce.

Pro usnadnění přístupu k serveru Player existují knihovny v mnoha programovacích jazycích (C, C++, Python, Java, LISP apod.). Player umožňuje také současné připojení více klientských programů k jednomu zařízení a tím vytváří další možnosti pro distribuované zpracování dat a kolaborativní řízení.

**Simulátory Stage a Gazebo**. Podstatnou část projektu zabírají simulační programy. Použití simulátorů přináší oproti experimentům s reálnými roboty několik výhod:
- je pohodlnější – práce se simulovanými roboty je snazší, není třeba kontrolovat stav baterií a senzorů, experimenty je možné kdykoliv zastavit a znovu spustit,
- je levnější – je možné simulovat velký počet robotů s rozličným senzorovým vybavením, jejichž cena by mohla být velmi vysoká,
- je bezpečnější – roboty jsou v reálných aplikacích často nasazovány právě do prostředí nebezpečného člověku. Experimenty v takovém prostředí jsou velmi nákladné, nebo nejsou vůbec možné a není tak jiná možnost než příslušné prostředí nasimulovat.

**Stage** je simulátor mobilních robotů a senzorů ve 2D prostředí. Prostředí může mimo robotů obsahovat i jiné objekty, které roboty vnímají a mohou s nimi i manipulovat. Stage nabízí množství modelů senzorů a aktuátorů, např. sonary, laserové dálkoměry, kamery, platformy robotů s odometrií atd.

Tyto modely jsou poměrně jednoduché s nízkými nároky na výpočetní výkon a umožňují tak v simulátoru spustit simulaci většího množství (desítek až stovek) robotů a senzorů.

**Gazebo** je další simulátor robotů, na rozdíl od Stage simuluje reálnou dynamiku a kinematiku ve 3D prostředí. Umožňuje simulovat nepřesnost dat z reálných senzorů i data virtuální kamery. Cenou za tuto realitu jsou vyšší nároky na výpočetní výkon. Gazebo se proto obvykle používá pro simulaci chování malých populací robotů (menších než 10). Oba simulátory se tedy doplňují, uživatel si může zvolit, který z nich více odpovídá jeho požadavkům.

Simulátory Stage i Gazebo jsou obvykle využívány v kombinaci s rozhraním Player. Řídicí programy interagují pouze s tímto rozhraním a simulované roboty a senzory jsou tedy z jejich pohledu stejné jako reálná zařízení. Na řídicích programech tedy není nutné před nasazením na reálné roboty provádět žádné úpravy.

Simulátory jsou nicméně vyvíjeny jako samostatné knihovny a je tedy možné je využít i bez propojení na rozhraní Player.

### 20.6.2 Multiagentní platforma A-globe

A-globe je platforma určená pro vývoj multiagentních systémů. Tato platforma byla vytvořen a v Gerstnerově laboratoři ČVUT FEL (Šišlák a kol., 2005b). Cílem agentních platforem je usnadnit a zrychlit vývoj distribuovaných systémů. Platformy nabízejí vývojářům množství funkcí, které jsou přítomné ve většině distribuovaných aplikací a jejichž implementace by byla časově náročná a odváděla by pozornost od řešení vlastního problému. Další výhodou použití těchto platforem je vzájemná kompatibilita. Velký význam měla v tomto směru snaha o zavedení standardů do oblasti multiagentních systémů, zejména pak aktivity organizace FIPA. Z hlavních funkcionalit agentních platforem jsou významné zejména:

- *kontrola životního cyklu agentů v systému (Agent Management System)* – tato služba má na starosti spouštění, registrování, odpojování a rušení agentů běžících v rámci dané platformy,
- *systém registrace služeb poskytovaných jednotlivými agenty (žluté stránky)* – poskytuje seznam všech agentů v systému a jimi nabízených služeb,
- *podpora vzájemné komunikace mezi agenty* – platforma poskytuje agentům fyzickou infrastrukturu nezbytnou pro vzájemné zasílání zpráv.

Další funkcionality jsou specifické pro jednotlivé platformy. Jedná se například o podporu zabezpečení při přenosu zpráv, možnost migrace agentů nebo nástroje pro sledování komunikace mezi jednotlivými agenty.

V porovnání s dalšími agentními platformami, které jsou v současné době k dispozici (JADE, Cougaar, JACK a další), nabízí A-globe vývojáři několik specifických funkcí, kterými jsou:

- **Simulační podpora** – platforma je primárně zaměřena na simulaci rozsáhlých systémů z reálného prostředí tvořených jak statickými, tak i mobilními entitami. Systém tak netvoří pouze agenti představující reálné entity, ale i speciální agenti simulující prostředí, v němž tito agenti operují. Každá reálná entita je v systému představována jedním tzv. kontejnerem, v němž může běžet jeden nebo více agentů. Simulační agenti běží na centrálním serveru a zprostředkovávají ostatním agentům informace o prostředí pomocí speciální komunikační služby. Při přechodu na reálný systém je pak tato simulační vrstva nahrazena skutečnými daty ze senzorů.
- **Simulace komunikační nedostupnosti** – v reálných aplikacích může často docházet k vzájemné komunikační nedostupnosti mezi agenty. To klade vyšší nároky na robustnost řídicích algoritmů a komunikačních protokolů. A-globe umožňuje tuto komunikační nedostupnost simulovat pomocí speciálních agentů. Tím je tedy možné vyvíjet algoritmy, které jsou určené pro nasazení například v dynamických ad-hoc mobilních sítích. Simulační mechanismus zajišťuje, že agent nemůže zaslat zprávu nikomu, kdo se nenachází v jeho komunikačním dosahu, a dále umožňuje nastavit i pravděpodobnost selhání komunikačního kanálu. Všechny tyto služby přitom zajišťuje sama platforma a pro vývojáře systému jsou skryty; tvůrce systému tudíž musí pouze vytvořit příslušného simulačního agenta.

- **Škálovatelnost systému** – platforma byla vyvíjena s ohledem na co nejmenší nároky na systémové zdroje. Důraz byl kladen na nízký výpočetní výkon potřebný na běh samotného agenta a zejména na efektivní přenos zpráv mezi agenty. Platforma podporuje adresné i všesměrové vysílání zpráv, uživatel si může zvolit, zda bude využívat komunikační protokol TCP/IP nebo UDP. Vzhledem k těmto vlastnostem umožňuje spuštění i několik tisíc agentů současně na jednom PC (záleží ovšem na složitosti jejich vlastní funkcionality).
- **Migrace agentů** – agenti mohou migrovat mezi jednotlivými kontejnery a přenášet svůj aktuální stav, kód i potřebné knihovny pomocí komunikační infrastruktury. K tomuto účelu byla implementována pokročilá správa verzí jednotlivých knihoven. Migrace může být využita například pro vyvážení výpočetního zatížení uzlů v síti nebo pro řešení problematiky komunikační nedostupnosti.
- **Vizualizační podpora** – platforma obsahuje množství grafických nástrojů usnadňujících vývoj, ladění a kontrolu běhu multiagentních systémů. Mezi tyto nástroje patří například vizualizátor logické struktury systému nebo program pro sledování komunikace mezi agenty (sniffer). Významnou částí platformy je pak podpora 3D vizualizace simulovaného světa. Platforma umožňuje připojení jednoho či více vizualizátorů současně. Ty mohou sloužit nejen ke zobrazení scény, ale díky implementaci zpětné vazby také jako operátorská stanoviště pro řízení činnosti agentů.
- **Snadné nasazení do reálných aplikací** – díky uvedeným funkcím je zajištěn snadný přechod od simulovaných k reálným aplikacím. Vzhledem k použitým komunikačním protokolům mohou agenti navzájem komunikovat i prostřednictvím bezdrátových sítí. Přechod k reálnému nasazení je tak zpravidla realizován pouze nahrazením simulační vrstvy reálnými senzory a změnou konfiguračních parametrů.

Agentní platforma A-globe je napsána v programovacím jazyce Java a lze jí tedy používat na různých operačních systémech. Je volně dostupná ke stažení na stránce http://agents.felk.cvut.cz/aglobe.

### 20.6.3 Simulační nástroj MAST

Systém MAST je multiagentní simulační nástroj zaměřený na oblast průmyslové automatizace. Je jedním z prvních nástrojů svého druhu, který kombinuje řídicí algoritmy implementované na bázi multiagentních systémů, softwarovou simulaci řízeného systému a grafické uživatelské rozhraní pro návrh a vizualizaci multiagentního řídicího systému.

Systém MAST (Manufacturing Agent Simulation Tool) byl vyvinut ve Výzkumném středisku Rockwell Automation v Praze jako multiagentní simulační nástroj pro flexibilní a robustní řízení systémů přepravy materiálu (Vrba, 2003). Veškeré rozhodování je v maximální možné míře distribuováno mezi agenty reprezentující jednotlivé dopravní či výrobní jednotky, které spolu vzájemně komunikují a dynamicky, v závislosti na aktuálních podmínkách, hledají optimální trasy pro přepravu výrobků.

Systém MAST je implementován v jazyce Java a používá volně dostupnou multiagentní platformu JADE (http://jade.tilab.com/) jako platformu pro vývoj a běh agentů. Skládá se z těchto částí:

- *Agentního subsystému*, který poskytuje knihovnu agentů reprezentujících různé typy dopravních a výrobních komponent, jako je dopravní pás, výrobní jednotka (stroj), senzor, robot, RFID čtečka atd.

- *Emulačního subsystému*, který poskytuje softwarovou simulaci fyzického dopravního systému. Tento subsystém ve virtuálním prostředí simuluje pohyb přepravovaných výrobků, aktivaci senzorů jednotlivých komponent a funkce příslušných akčních členů.
- *Runtime rozhraní* poskytujícího obousměrné propojení agentního a emulačního subsystému. Tímto rozhraním je agent informován o tom, že došlo k aktivaci senzoru odpovídající komponenty v emulačním subsystému (tj. například výrobek vjel do prostoru výhybky). Současně se po tomto rozhraní přenášejí řídicí zásahy agenta k odpovídajícím akčním členům v emulačním subsystému (tj. například přehození výhybky do požadovaného směru). Vzhledem ke snadné integraci s průmyslovými řídicími systémy je toto rozhraní založeno na sdílení hodnot vstupních a výstupních veličin přímo v datové paměti PLC (zkratka pro programovatelný logický automat) typu ControlLogix (Vrba, Mařík, 2005). To umožňuje jednoduše zaměnit emulační subsystém fyzickým zařízením připojeným do PLC a použít tak systém MAST pro účely reálného řízení.
- *Grafického uživatelského rozhraní* ke grafickému návrhu dopravního systému a vizualizaci průběhu simulace. Interakce se systémem (např. zavádění změn struktury dopravního systému, umělé zavádění poruchy) je možné provádět za běhu systému bez nutnosti simulaci zastavit či systém složitě rekonfigurovat, viz (Vrba, 2003) a též obrázek 20.22 (též bar. příl.).

## 20.7 Příklady multirobotických experimentů

### 20.7.1 Prohledávání neznámého prostředí skupinou robotů

*Kooperativní explorace* (prohledávání neznámého prostředí) patří k fundamentálním úlohám inteligentní mobilní robotiky. Tato úloha je velice komplexní, neboť v rámci jejího řešení je nutné použít většinu základních robotických dovedností od řízení, přes plánování trajektorií, lokalizaci, mapování až po vyjednávání a plánování vyšší úrovně. Z tohoto důvodu slouží exploračí úloha jako dobrá testovací a demonstrační platforma pro nově vyvíjené algoritmy. V následujících odstavcích představíme jeden z možných přístupů k této úloze a na jeho příkladě popíšeme jednotlivé problémy, s kterými je nutné se vypořádat při řešení složitější robotické úlohy.

Prvořadým úkolem kooperativní explorace je prozkoumání předem neznámého prostředí a vytvoření modelu (mapy) tohoto prostředí se snahou optimalizovat prostředky nutné k řešení. Prostředky můžeme rozumět například dobu nutnou k provedení úlohy, součet délek trajektorií ujetých roboty během explorace nebo délku trajektorie nejvíce cestujícího robotu. Alternativně lze exploračí úlohu chápat jako nalezení definovaných statických objektů v neznámém prostředí.

Klíčová otázka kooperativní explorace zní: kam v každém kroku (při aktuální znalosti okolního prostředí a pozic všech robotů v něm) poslat jednotlivé roboty tak, abychom získali co nejvíce nové informace o prostředí. Její zodpovězení vyžaduje odpověď na dvě jiné otázky:

- Kde se v prostředí nacházejí „zajímavá" místa, tj. místa, jejichž navštívením vzroste úroveň znalostí o prostředí?
- Jak tento průzkum „zajímavých" míst přidělovat jednotlivým robotům? Jinými slovy, který robot navštíví to či ono vybrané místo?

Pro úlohu explorace se jako vhodná jeví reprezentace prostředí pomocí mřížek obsazenosti, zejména kvůli relativně jednoduché implementaci, schopnosti pracovat se šumem senzorických měření a možnosti skládání několika map do jedné.

Důležitým předpokladem úspěšné kooperativní explorace je to, aby roboty pracovaly se společnou (globální) mapou sdružující senzorickou informaci od všech robotů. To se zajistí tak, že každý robot si buduje vlastní mapu ve formě mřížky obsazenosti ze senzorů, které má k dispozici. V okamžiku, kdy je vyvolána akce na rozdělení cílů, získá lokální mřížky obsazenosti od ostatních robotů a tyto mřížky vhodným způsobem sloučí tak, že sloučí odpovídající si buňky mřížek způsobem podobným tvorbě mřížek obsazenosti (Burgard a kol., 1999).

Vraťme se nyní k položeným otázkám. Na první z nich existuje poměrně jednoduchá odpověď – místa, ve kterých získáme novou znalost o prostředí, leží na rozhraní volného a neprozkoumaného prostoru. Jak však tato místa nalézt? K tomu použijeme právě mřížky obsazenosti (připomeňme, že každá buňka mřížky obsahuje pravděpodobnost výskytu překážky v místě prostředí, který buňka reprezentuje). Nejprve jsou jednotlivé buňky klasifikovány podle pravděpodobností do tří tříd:

- v o l n o: pravděpodobnost výskytu překážky je menší než práh,
- n e z n á m o: pravděpodobnost výskytu překážky je rovna prahu,
- o b s a z e n o: pravděpodobnost výskytu překážky je větší než práh.

Poté je aplikován proces analogický detekci hran a extrakci regionů známý z počítačového vidění (Hlaváč, Šonka, 1992), který nalezne hranice mezi volným a neznámým prostorem: každá volná buňka je označena jako „zajímavá", pokud sousedí s alespoň jednou buňkou, která je klasifikována jako neznámá. Dále budeme tyto buňky nazývat hraniční (frontiers).

Pro určení, který robot pojede do konkrétní buňky, použijeme dvě kritéria stanovující „cenu buňky":

a) jaká je vzdálenost daného robotu od buňky a
b) kolik nové informace lze získat navštívením buňky.

Pro výpočet délky trajektorie z jeho aktuální polohy do dané buňky lze použít některou z metod pro plánování pohybu uvedených v 20.5.3. S výhodou lze aplikovat algoritmus A* nebo harmonické potenciální pole, které jsou schopny spočítat vzdálenosti od všech hraničních buněk v jednom běhu algoritmu.

Odhad užitečnosti buňky (tj. kolik nové informace se získá jejím navštívením) je těžší, neboť silně závisí na struktuře okolního prostředí buňky. Navíc, pokud robot navštíví nějakou buňku, její užitečnost se pro ostatní roboty zřejmě sníží. A jelikož robot svými senzory typicky pokryje určitou oblast kolem své pozice, nesníží se pouze užitečnost navštívené buňky, ale i užitečnost jejího okolí viditelného senzory. Užitečnost buňky tak lze vyjádřit jako funkci vzdáleností od buněk, ke kterým jsou naplánovány cesty ostatních robotů (Burgard a kol., 2005).

Výběr cílů pro jednotlivé roboty (buněk, do nichž mají roboty dojet) je realizován s použitím multiagentních technologií. Na každém robotu běží agent, který vyjednává o cílích s ostatními agenty a společně tak přiřazují cíle robotům. Konkrétně toto vyjednávání probíhá následujícím způsobem: v okamžiku, kdy nějaký (libovolný) robot-agent vyvolá akci na rozdělení cílů (například, pokud dojel k předchozímu cíli, nebo vypršel definovaný časový limit), dojde k výměně lokálních map robotů, přičemž každý robot si

poté vytvoří globální mapu obsahující informace od všech robotů. Každý robot-agent poté vygeneruje množinu hraničních buněk (uvědomme si, že všechny roboty pracují se stejnou mapou, a tedy i množina hraničních buněk bude pro všechny roboty stejná). Poté každý robot ohodnotí každou hraniční buňku s ohledem na délku trajektorie a tu pro sebe nejvýhodnější (tj. nejbližší) spolu s ohodnocením pošle agentovi, který vyjednávání vyvolal (iniciátorovi) jako svoji nabídku. Iniciátor všechny nabídky vyhodnotí, vybere tu s nejlepším ohodnocením a rozhodnutí všem oznámí. Každý agent, jehož nabídka nebyla vybrána, podle tohoto rozhodnutí upraví svoje ohodnocení všech hraničních buněk (odečte užitečnosti buněk s ohledem na vybraný cíl) a opět pošle iniciátorovi svou nabídku. Tento proces pokračuje do té doby, než je přidělen každému robotu cíl. Samozřejmě takovýto postup není globálně optimální (úloha explorace je NP-těžká!), nicméně experimentální výsledky dokazují dostatečnou kvalitu generovaných řešení.

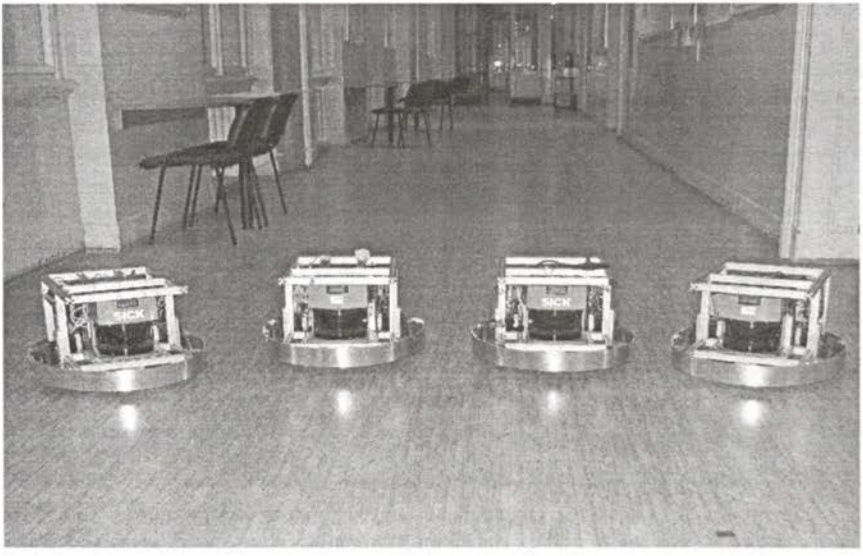
Požadavek, aby vyjednávání o rozdělení cílů mohl zahájit kterýkoliv robot, souvisí s tím, že vzhledem k povaze úlohy může dojít k přerušení komunikačního spojení mezi roboty. Z tohoto důvodu nemůže přiřazování cílů řídit jeden centrální robot. Důvodem pro vznik komunikační nedostupnosti může být například příliš velká vzdálenost mezi roboty (pokud je prohledáván rozsáhlý prostor), výskyt překážky mezi roboty nebo zdroj rušení signálu v určité oblasti prohledávaného prostoru. V těchto případech mohou nastat situace, kdy roboty spolu mohou komunikovat pouze prostřednictvím mezilehlého robotu, popř. se skupina robotů řešících úlohu rozdělí na několik disjunktních týmů. Tyto případy kladou i vyšší nároky na algoritmy řešící přenos a synchronizaci znalostí mezi roboty. Algoritmus pro rozdělování cílů musí být na tuto možnost připraven a nemůže spoléhat pouze na informace získané komunikací s roboty ve svém nejbližším okolí pomocí všesměrového nebo adresného vysílání zpráv.

Další možností je využít pro přenos a synchronizaci znalostí mezilehlou vrstvu tvořenou agenty specializujícími se pouze na distribuci znalosti v dynamické síti (Šišlák a kol., 2005a). Tato vrstva slouží k zajištění lokálního přístupu ke znalostem bez ohledu na aktuální topologii komunikační sítě. Zmiňovaný mechanismus zajistí přenos zpráv mezi agenty i přes mezilehlé uzly, ale ani on nezajistí synchronizaci aktivit s roboty v týmech, jež nejsou komunikačně dostupné. V takovém případě lze aktivity těchto robotů modelovat (tvořit jejich sociální modely) a při plánování vlastních aktivit využít dat z těchto modelů namísto přímé komunikace.

Algoritmus přiřazení hraničních buněk je tak možné rozšířit o využití sociálních modelů ostatních robotů například tak, že se bude uchovávat informace o jejich posledních známých pozicích a přiřazených hraničních buňkách. Na základě této informace pak robot vybírá buňku v oblasti, která je od těchto hraničních buněk vzdálená, a tak prohledává jinou oblast než ostatní roboty i bez možnosti vzájemné komunikace a koordinace aktivit.

Úkolem agentů operujících na jednotlivých robotech je také zjišťovat informace o nových robotech, které se vyskytnou v jejich komunikačním dosahu. K tomu slouží mechanismus distribuovaných žlutých stránek doplněný o službu, která zkouší periodicky kontaktovat ostatní roboty pomocí speciální zprávy zasílané na adresy v předem určeném komunikačním rozsahu. K tomu, aby se mohl nový robot zapojit do týmu v průběhu plnění úkolu, je nutné, aby byl vybaven komunikačním zařízením schopným spojit se s ostatními roboty (v našem případě musí například využívat shodně nakonfigurovanou bezdrátovou síť).

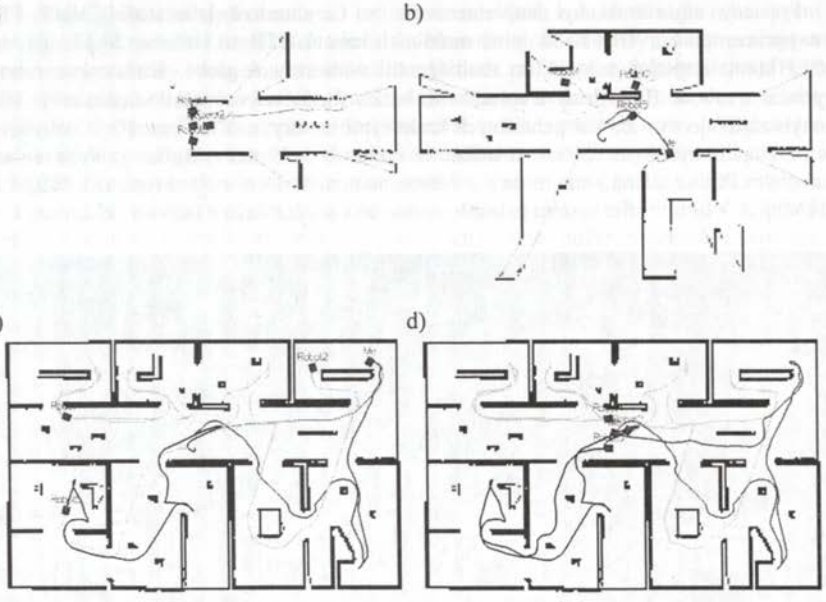
Popsaný algoritmus byl implementován v Gerstnerově laboratoři ČVUT FEL a experimentálně ověřen na skupině mobilních robotů G2Bot (viz obr. 20.13, též bar. příl.) komunikujících s využitím multiagentní platformy A-globe. Konstrukce robotu vychází z robotu ER1 firmy Evolution Robotics (http://www.evolution.com/er1). Jeho pohyb zajišťuje dvojice kol poháněných krokovými motory, což mu umožňuje pohybovat se maximální rychlostí 0,5 m/s a otáčet se rychlostí 3,75 rad/s. Robot je vybaven vestavěným PC, vzdálená komunikace je řešena pomocí WiFi připojení typu IEEE802.11b (11Mbps). Vlastní řídicí systém je implementován v jazyce Java a běží na OS Linux.



Obr. 20.13 Skupina mobilních robotů G2Bot.

Jednotlivé funkční bloky, senzory a periferie jsou hvězdicovou architekturou propojeny sériovými komunikačními linkami na bázi rozhraní RS232 a USB. Jako základní senzor slouží laserový dálkoměr SICK LMS 200 umožňující zejména kontinuální lokalizaci robota, reaktivní vyhýbání se překážkám a stavbu modelu prostředí.

Obrázek 20.14 demonstruje běh algoritmu pro čtyři roboty startující z blízkých pozic. V počátečních fázích jsou robotům naplánovány cíle relativně daleko od sebe (a), což vede k tomu, že každý robot prozkoumává jinou část prostředí (b). Poté, co je explorace dokončena (neexistuje dostupná hraniční buňka) (c), vrátí se roboty na svou původní pozici (d). Všimněte si, že jeden z robotů ujel na první pohled kratší dráhu, než ostatní roboty. To je způsobeno neúplnou informací o prostředí, v důsledku čehož nelze cesty robotů rozvrhnout takovým způsobem, jako kdyby znalost prostředí byla dokonalá. Existuje však celá řada překážek, zabraňujících dosažení zcela optimálního řešení: Například při snaze o řešení NP-těžké úlohy přidělování cílů aproximačním algoritmem se projevují překřížení trajektorií robotů *Robot2* a *Me* (zamyslete se nad tím, že odstranění překřížení vede ke zlepšení nalezeného řešení).




Obr. 20.14 Experimentální běh algoritmu kooperativního prohledávání čtyřmi roboty.

### 20.7.2 Hybridní telematický systém PeLoTe

#### *20.7.2.1 Definice úlohy záchranné mise v hybridním systému*

Koncepce projektu PeLoTe viz adresa [3] se zaměřuje na oblast záchranných misí. Tým záchranářů se neskládá pouze z lidí, ale zahrnuje i mobilní roboty.

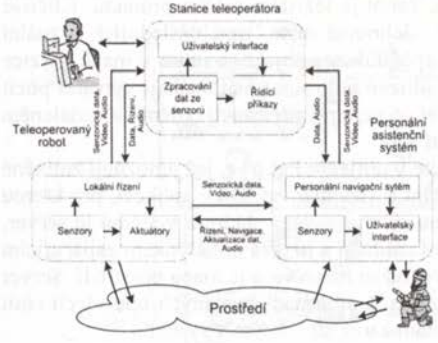
V klasickém scénáři záchranné mise mají lidé k dispozici spoustu různých pomůcek a technických prostředků. Komunikují spolu zpravidla pomocí vysílaček a navíc mají možnost snímat akční scénu pomocí miniaturní kamery. Snímaný obraz může být buďto přenášen přímo na vzdálené pracoviště, nebo je pouze zaznamenán na přenosné médium a později použit k vyhodnocení zásahu. Záchranné mise jsou zpravidla řízeny velitelem zásahu. Velitel zásahu však má jen hrubou představu o rozmístění svých lidí v zásahovém prostoru. Představme si, že záchranáře vybavíme systémem, kdy velitel zásahu bude moci vidět na monitoru mapu zásahového prostoru, jež se neustále aktualizuje a zobrazuje aktuální polohu členů záchranného týmu. V případě, že záchranář lokalizuje objekt zájmu (oběť, nebezpečnou látku, požár, zavalený průchod apod.), velitel zásahu může okamžitě reagovat a poslat na dané místo potřebnou pomoc. To však platí pouze za předpokladu, že kromě velitele zásahu jsou i ostatní záchranáři vybaveni technikou umožňující vidět mapu prostředí, svoji pozici na mapě, ale také pozice ostatních členů týmu, aby přesně věděli, kde zasáhnout nebo kterým místům se vyhnout. Ve chvíli, kdy je život záchranáře ohrožen, nebo cílové místo je pro záchranáře nedostupné, vstupují na scénu mobilní roboty. Jejich hlavní úlohou je zastoupit člověka v nebezpečných situacích, kdy se v prostředí vyskytují například nebezpečné chemikálie, radioaktivita, nebo je

cílová oblast pro člověka nedostupná. Malý robot má šanci do cílového prostoru proniknout a poskytnout člověku řadu užitečných informací.

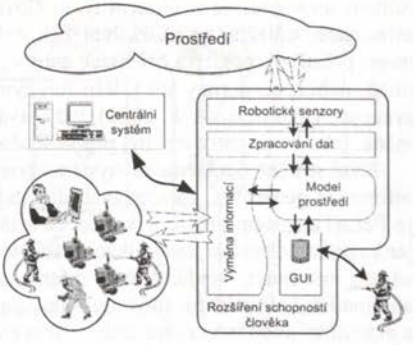
V experimentálním scénáři záchranářské mise projektu PeLoTe omezujeme pracovní prostor na objekt (budovu), k níž máme konstrukční plány. K dispozici máme informace o velikosti budovy, rozmístění pater, místností, schodů, dveří atd. Nemáme však žádné informace o obsazenosti prostoru překážkami. Víme, že uvnitř budovy se mohou vyskytovat lidé, kteří potřebují pomoc, některé části budovy mohou být člověku nepřístupné (zavalený průchod), životu nebezpečné (únik plynu či chemikálie). Kromě (elektronické) mapy prostředí však nemáme žádnou informaci o počtu lidí uvnitř budovy ani o jejich pozici.

Cílem mise je prohledat celou budovu, detekovat co největší počet postižených osob a dovést je k východu, lokalizovat místa požáru, detekovat nebezpečné prostory. Všechna důležitá zjištění jsou ohlášena operátorovi a zanesena do sdílené mapy. Nové informace ve sdílené mapě jsou ihned distribuovány všem členům záchranné mise (lidem i robotům). Všichni tedy mají k dispozici aktuální informace o stavu pracovního prostředí.

Systém PeLoTe, kromě uvedených schopností, umožňuje optimální plánování tras pro hybridní skupinu záchranářů tak, aby byla splněna podmínka prohledání celého prostoru dle aktuální mapy. Pokud dojde během mise ke zjištění, že některá trasa je neprůchodná, systém provede automatické přeplánování.




Obr. 20.15 Datové komunikace.




Obr. 20.16 Rozšíření schopností člověka.

Dosud jsme popisovali pouze scénář experimentální záchranné mise a některé vlastnosti systému PeLoTe, ale nezmiňovali jsme se o prostředcích a způsobech, jakými je celý projekt realizován. Jedny z mnoha klíčových problémů celého projektu jsou například otázky: Jak získávat informace z prostředí? Jak tyto informace efektivně přenášet a sdílet s ostatními členy týmu? Jak realizovat komunikaci mezi člověkem a robotem? Těmito a dalšími dílčími problémy projektu se zabývají následující odstavce.

### *20.7.2.2 Řešené podúlohy a jejich vzájemné vazby*

Vzhledem k tomu, že v záchranné misi kooperuje tým záchranářů a robotů, je třeba jejich aktivity účinně koordinovat. Koordinace vyžaduje vzájemnou vzdálenou komunikaci. Vzdálená komunikace vyžaduje přenosové médium a komunikační protokol. Vzhledem

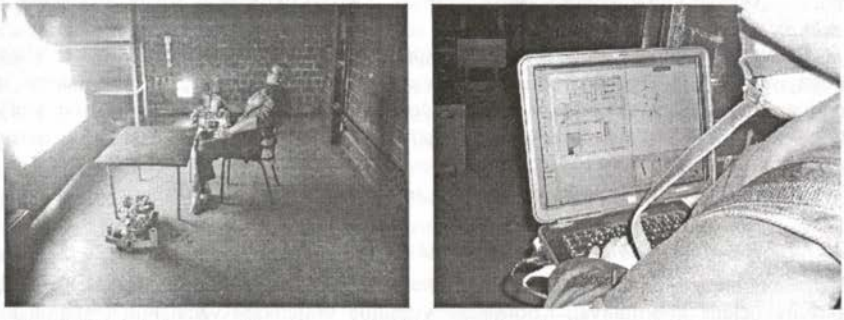
k tomu, že záchranné mise se účastní lidé i roboty, je třeba definovat vhodné způsoby komunikace mezi těmito odlišnými entitami. Typ přenášených dat je ovlivněn schopnostmi jednotlivých entit vnímat prostředí a tyto vjemy definovaným způsobem interpretovat. Obrázek 20.15 zobrazuje hlavní komponenty systému PeLoTe a datové komunikace mezi nimi.

Je evidentní, že člověk i robot vnímají a zpracovávají vstupní informace naprosto odlišným způsobem. Člověk je vybaven jistým druhem senzorů od přírody (zrak, čich, sluch, hmat, chuť). Svoje vjemy zpracovává na základě naučených zkušeností. Důležitá je však schopnost rychlé reakce a přizpůsobivosti v nečekaných situacích. Lidské vnímání okolního prostředí, přestože je v mnoha ohledech dokonalé, má však i své nedostatky. Například nevidíme ve tmě, neumíme přesně určit vzdálenost k předmětům, těžko si zapamatujeme mapu města s názvy všech ulic atd. Proto se snažíme tyto nedostatky kompenzovat různými technickými přístroji a zařízeními, viz obr. 20.16. Informace ze senzorů je zpravidla prezentovaná člověku v grafické podobě pomocí určitého zobrazovacího zařízení. Důraz je kladen na přehledné zobrazování pouze důležitých nebo požadovaných informací. Granularitu zobrazovaných informací si volí sám uživatel. V projektu PeLoTe tyto funkce poskytuje člověku zařízení, jež nazýváme PAS (Personal Assistance System) spolu se systémem PeNa (Personal Navigation System).

### *20.7.2.3 Sdílení informací ve smíšeném týmu*

Sdílení informací ve smíšeném týmu člověk–robot je těžištěm celého projektu. Klíčové informace, důležité pro efektivní běh celé záchranné mise, jsou následující: aktuální mapa prostředí, pozice a orientace entit v mapě, důležité objekty a místa v mapě (pozice obětí, nebezpečné zóny atd.). Všechny tyto sdílené informace napomáhají vytvářet pocit prezence (Suomela a kol., 2003), tedy utvářet co nejlepší představu o dění na vzdáleném místě, jako by pozorovatel byl přímo součástí dění.

Nyní stručně popíšeme subsystémy použité v projektu PeLoTe, jež umožňují zmíněné informace generovat, zpracovávat, distribuovat a vizualizovat. Díky aplikaci, pro kterou je PeLoTe systém určen, je koncepce sytému centralizovaná. Jádrem systému je server, jenž zprostředkovává komunikaci mezi všemi entitami a je také prostředkem zajišťujícím sdílení informací. Nejdůležitější sdílenou strukturou na serveru je mapa prostředí. Server automaticky aktualizuje sdílenou mapu na základě informací zasílaných ode všech entit a aktualizovanou mapu distribuuje všem entitám zaregistrovaným v systému.

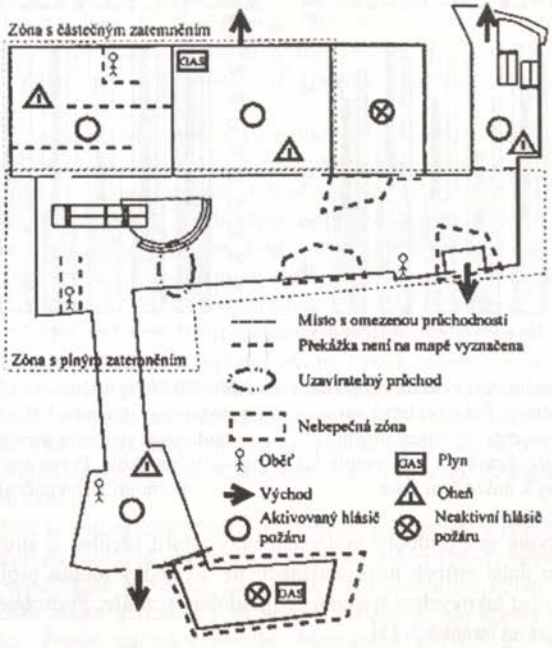


Obr. 20.17 Obrázky z experimentů prováděných ve výcvikovém domě požárníků ve Würzburgu.

Robot v projektu PeLoTe je entitou, jež je přirozeně vybavena potřebnými senzory a subsystémy pro získávaní informací z prostředí (mikrofonem, kamerou, teploměrem, laserovým dálkoměrem, sonary atd.) a výměnu informací se vzdáleným počítačovým systémem (komunikačním modulem). Robot nepotřebuje ke své činnosti zobrazovací jednotku, veškerá data jsou zpracovávána v nativní podobě. Zpravidla však robot zobrazovací jednotkou vybaven je, a to z důvodu sledování vnitřních stavů robotu člověkem.

Pro člověka je důležitá především hlasová komunikace a vizuální informace zprostředkovaná obrazem z kamery záchranáře nebo robotu, jež je vztažena k dané pozici a orientaci dané entity. Člověk však přirozeně není vybaven žádným rozhraním, jež by přímo umožnilo jeho integraci do zmíněné koncepce. Proto byl pro člověka (v roli záchranáře) navržen a realizován již zmiňovaný robotický subsystém PAS, jenž umožňuje získávat informace o prostředí a vzdáleně komunikovat stejně jako robot. Navíc však tento systém zabezpečuje vizualizaci sdílených informací ve srozumitelné podobě a také realizuje rozhraní pro přímou interakci se systémem. Systém PeNa (Personal Navigation System) provádí průběžnou lokalizaci člověka ve známé mapě. Lokalizace je prováděna na základě fúze dat ze senzorů, jež vyhodnocují pohyb člověka (akcelerometrů, inklinometrů, odometrů) a tvar okolního prostředí (laserového dálkoměru).

Člověk v roli operátora má k dispozici operátorské pracoviště, jež umožňuje hlasovou i obrazovou komunikaci se vzdálenými entitami. Člověk-operátor může přímo sám ovládat vzdáleně robota, poskytovat podporu pro lidské záchranáře, vyhodnocovat průběh mise a dle potřeby modifikovat sdílenou mapu prostředí.



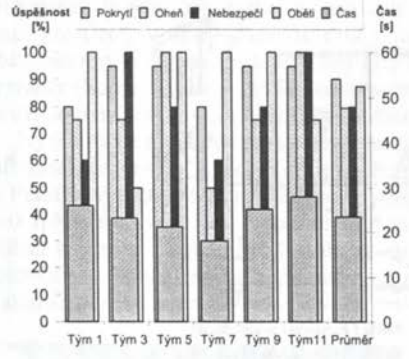

Obr. 20.18 Popis experimentálního prostoru dle předepsaného scénáře.

### *20.7.2.4 Experimentální výsledky*

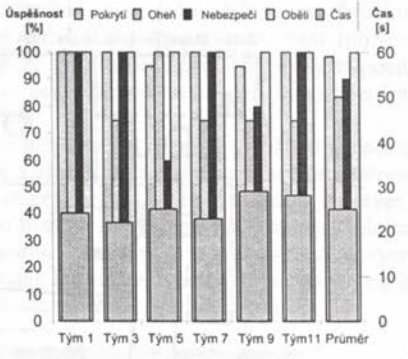
Prototypové řešení systému PeLoTe bylo testováno ve výcvikovém požárním středisku ve Würzburgu, viz obr. 20.17. Pro porovnání výsledků probíhaly testy ve dvou fázích. V první fázi byla testována skupina s podporou systémem PeLoTe a ve druhé fázi skupina bez podpory tímto systémem. Obrázek 20.18 ukazuje popis experimentálního prostoru s vysvětlivkami, přesně dle předepsaného scénáře. V plánku jsou označena místa, kde se nacházejí východy, nebezpečné oblasti, nebezpečí úniku plynu, požár, oběť, aktivní požární alarm, neaktivní požární alarm, průchody atd.

Po vyhodnocení experimentů (viz obrázky 20.19 a 20.20) se ukázalo, že průměrná doba potřebná k dokončení mise byla delší v případě týmů s podporou systémem PeLoTe. To lze vysvětlit tím, že týmy s podporou zkoumaly podrobněji též nepřístupné nebo životu nebezpečné zóny za pomoci robotů. Ostatní ukazatele hodnotící průběh mise vycházely lépe pro týmy s podporou systémem PeLoTe.

Týmy s podporou prováděly mnohem promyšlenější zásahy, lokalizovaly více požárů, našly více obětí a nebezpečných zón, lidé byli mnohem více relaxovaní a byli schopni se lokalizovat vůči mapě mnohem snadněji než týmy bez podpory. Operátor s podporou systémem PeLote většinou měl vždy přehled o situaci během zásahu a poskytoval svému týmu užitečné informace.




Obr. 20.19 Vyhodnocení výsledků týmů *bez podpory* systémem PeLoTe. Levá osa *y* zobrazuje procenta úspěšnost plnění jednotlivých úkolů. Pravá osa *y* zobrazuje čas potřebný k dokončení mise.




Obr. 20.20 Vyhodnocení výsledků týmů *s podporou* systémem PeLoTe. Levá osa *y* zobrazuje procenta úspěšnost plnění jednotlivých úkolů. Pravá osa *y* zobrazuje čas potřebný k dokončení mise.

Operátor v týmu bez podpory systémem brzy ztratil přehled o situaci a po krátké době se cítil pro další průběh mise neužitečným. Výsledky tohoto projektu tedy jasně naznačují užitečnost takovýchto systémů pro podobné scénáře. Podrobnější informace o projektu lze nalézt na stránkách [3].

### 20.7.3 Aplikace systému MAST pro řízení multirobotické linky

V rámci spolupráce Výzkumného střediska Rockwell Automation Praha s Centrem pro distribuovanou automatizaci a řízení (Center for Distributed Automation and Control – CDAC), Cambridge University, bylo rozhodnuto rozšířit systém MAST tak, aby poskytoval multiagentní simulaci robotické balicí linky, jež slouží jako fyzická testovací platforma pro experimenty s moderními automatizačními technologiemi, jako je například distribuované řízení či radiofrekvenční identifikace (RFID) (Fletcher a kol., 2003), (Vrba a kol., 2006).

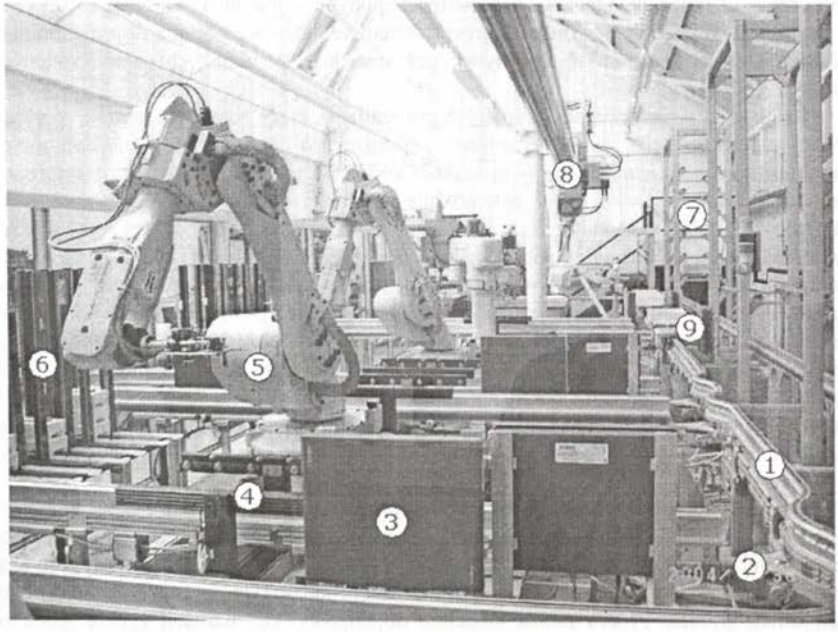
Prototypová výrobní linka je sestavena z komerčně dostupných průmyslových komponent, jako je kolejový dopravní systém Montech, dva roboty Fanuc M6i, policový skladovací systém, pojízdný jeřábový manipulátor, RFID čtečky CheckPoint a Samsys a další (viz obr. 20.21). Jednou z hlavních činností linky je balení dárkových balíčků Gillette, do kterých na základě uživatelovy objednávky vkládají roboty (jeden z nich je označen číslem 5 na obr. 20.21) zvolenou kombinaci tří ze čtyř dostupných holicích potřeb (holicí gel, pěnu na holení, holicí strojek a deodorant), které jsou uloženy ve dvou čtyřsegmentových skladovacích jednotkách (č.6). Prázdné krabice jsou uloženy v policovém skladovacím systému (č.7), kam jsou vkládány, popř. vykládány, pojízdným jeřábovým manipulátorem (č.8). Transport krabic zajišťují samohybné vozíky pohybující se po kolejovém dopravním systému, který se skládá z jedné hlavní smyčky (č.1) a dvou na ni kolmých vedlejších smyček vedoucích k robotům. Přesto, že jde o samohybné vozíky, jejich inteligence je omezena pouze na schopnost zastavit, pokud před sebou detekují překážku. O jejich navigaci, tj. přesměrování pohybu z hlavní na vedlejší smyčky, se starají dvě výhybky (č.2). Tento dopravní systém se tedy více podobá dopravě pomocí klasických dopravních pásů a výhybek než systému automatických samonaváděných vozíků AGV.

Jedním z hlavních principů, použitých pro realizaci, který posouvá agentově založený řídicí systém do další roviny, je aplikace *produktových agentů*, kdy kromě dopravních a výrobních komponent je i každý produkt či výrobek reprezentován vlastním (informačním) agentem. Takovýto agent pak udržuje informaci o stavu rozpracování výrobku a především autonomně řídí výrobu sebe sama prostřednictvím dohadování se a vyjednávání s výrobními agenty. K flexibilitě tohoto řešení přispívá další z principů používaných v multiagentních systémech, a to mechanismus tzv. *žlutých stránek*, kde si jednotliví agenti registrují služby, které nabízejí ostatním agentům. Dotazem do žlutých stránek agent získá seznam jmen ostatních agentů, jež poskytují požadovanou službu, přičemž pak obvykle dojde již k přímému dohadování mezi žadatelem a poskytovateli o detailech provedení služby. Zde se nejčastěji uplatňuje protokol CNP *contract-net* (Smith, 1983), kdy poskytovatelé služby nabídnou cenu, za níž jsou schopni provedení služby realizovat (cena je určena například na základě vlastní kapacity a vytížení), a žadatel vybere poskytovatele s nejnižší cenou.

Navržené řešení je možné chápat jako multirobotický systém, v němž dva stacionární roboty spolupracují na společné úloze balení krabice, přičemž spolu nepřímo komunikují prostřednictvím modifikace prostředí, tak jak vkládají jednotlivé holicí potřeby do krabice. Vlastní proces balení je řízen a koordinován produktovými agenty reprezentujícími jednotlivé krabice. Proces začíná vložením objednávky, kdy uživatel vybere jeden ze dvou různých typů krabic (typ E a T) a určí, které holicí potřeby vyžaduje v jednotlivých

přihrádkách krabice. S objednávkou je automaticky vytvořen *agent objednávky*, který zahájí vyjednávání se všemi dostupnými agenty krabic (jejich seznam získá předem ze žlutých stránek) o tom, který z nich je ochoten spolupracovat na splnění objednávky. Pokud agent krabice již například pracuje na vyřízení předchozí objednávky, nebo je krabice jiného typu než požadovaného, agent krabice zašle agentu objednávky zamítavou odpověď.

V okamžiku, kdy dojde k dohodě mezi objednávkovým a produktovým agentem, zahájí produktový agent vyjednávání s roboty o tom, který z nich je schopen naplnit krabici dle objednávky. Opět je použit vyjednávací protokol CNP, jako v předchozím případě, přičemž znovu dojde nejprve k dotazu do žlutých stránek na jména dostupných robotů. Agent krabice ve své žádosti předává robotu informaci o současném stavu naplnění krabice (některé přihrádky již mohou být naplněny) spolu s informací, kterými holicími potřebami mají být naplněny zbylé přihrádky. Agent robotu potom dotazem do skladovací jednotky zjistí, kolik z požadovaných výrobků je k dispozici (čím více tím lépe) a tuto informaci zkombinuje s počtem dříve nasmlouvaných krabic, které budou robotem plněny (čím méně tím lépe). Výslednou cenu pak zašle jako nabídku zpět produktovému agentu. Ten si nabídky od všech robotů seřadí dle ceny od nejnižší po nejvyšší. Společně s cenou ještě robot zašle seznam pracovních stanic v jeho dosahu, kde vozík s krabicí musí zastavit, aby mohl být robotem obsloužen. Například pro Robot 1 na obr. 20.22 (též bar. příl.) jsou to stanice D1 a D2. V dalším kroku agent krabice vybere takový robot, jehož nabídka je nejvýhodnější a současně některá z blízkých pracovních stanic je v do-



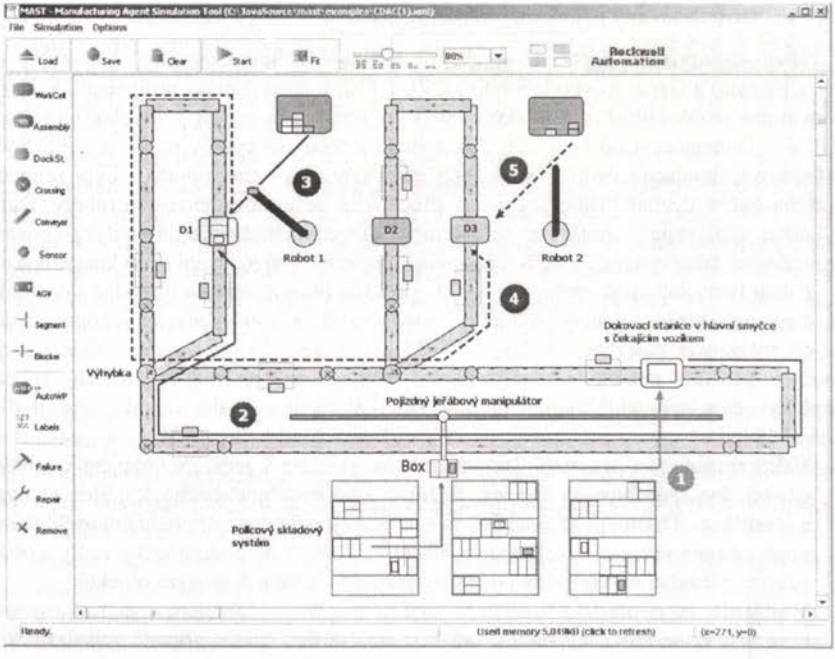

Obr. 20.21 Robotická balicí linka v centru CDAC na Univerzitě v Cambridgi.

pravním systému pro něj dostupná. Vzhledem k tomu, že se krabice nachází v tomto okamžiku v policovém skladovacím systému, požádá některý z vozíků, aby dojel a zastavil v pracovní stanici v hlavní smyčce. Následuje pak posloupnost akcí znázorněných na obrázku 20.22 (též bar. příl.).

1. Agent krabice požádá pojízdný jeřáb, aby vyjmul krabici z police a položil jej na vozík.

2. Agent krabice požádá vozík, aby jej dopravil do vybrané pracovní stanice (v tomto případě D1). Během dopravy je vozík navigován agenty výhybek, tj. jsou využívány všechny schopnosti agentního řešení transportní úlohy, které systém MAST nabízí a jež byly popsány dříve.

3. Po té, co vozík dorazí do pracovní stanice D1, požádá produktový agent agenta robotu, aby naplnil ty prázdné přihrádky, na nichž se oba agenti dohodli během počátečního vyjednávání.




Obr. 20.22 Simulace výrobní linky centra CDAC v systému MAST s výraznčnou posloupností akcí při balení krabice.

V situaci zachycené na obrázku 20.22, viz též barevná příloha, zněla kompletní objednávka na gel (žlutý obdélníček), holicí strojek (modrý obdélníček právě nesený Robotem 1) a deodorant (oranžový obdélníček). Ve skladu obsluhovaném Robotem 1 však byly pouze první dvě ze tří požadovaných holicích potřeb (oranžový obdélníček zde chybí) a tedy po ukončení operace robota není splnění objednávky kompletní. Zbývající

holicí potřeby však mohou být doplněny jiným robotem. Produktový agent tak zopakuje celý proces vyjednávání s roboty od začátku s tím, že v žádosti opět uvede, které přihrádky již byly naplněny jinými roboty a které na naplnění stále čekají. Zbývající deodorant (oranžový obdélníček) do třetí přihrádky je nabídnut Robotem 2 a následuje tak cesta vozíku s krabicí do stanice D3 (krok č.4 v obr. 20.22, též bar. příl.) a doplnění deodorantu do krabice (krok č.5). Nyní je objednávka splněna a krabice se vrací do policového skladovacího systému. O výsledku je současně informován agent objednávky.

Na naplnění krabice tedy spolupracuje více robotů, přičemž mezi nimi probíhá nepřímá komunikace zásahem do výrobního prostředí tak, jak jsou postupně plněny jednotlivé přihrádky krabice. Implementované řešení je značně flexibilní. V kterémkoliv okamžiku může být do systému přidán nový robot společně se skladovací jednotkou a dopravní smyčkou. Tím, jak zaregistruje svoji přítomnost ve žlutých stránkách, bude automaticky zahrnut do dalších vyjednávání s produktovými agenty.

### 20.7.4 Distribuovaný průzkum podmořského dna

V rámci spolupráce s IHMC (Institute for Human-Machine Cognition, University of West Florida) a Gerstnerovou laboratoří ČVUT FEL byla zkoumána problematika distribuovaného prohledávání podmořských oblastí týmem autonomních podmořských robotů (UUV – Unmanned Undersea Vehicle). Jednou z úloh řešených v rámci projektu byla softwarová simulace tohoto scénáře. Důvodem pro vývoj této simulace byla zejména potřeba ověřit vyvinuté řídicí algoritmy před jejich nasazením na reálné roboty. Další výhodou vyplývající z možnosti využít simulátor je možnost snadné analýzy chování jednotlivých částí systému, jako je vzájemná koordinace, vyjednávání nebo komunikace.

Cílem týmu robotů je prohledat předem určenou oblast, nalézt a případně i odstranit všechny nebezpečné objekty. Jednotlivé roboty však nejsou vybaveny schopností detailně analyzovat nalezené objekty, tuto činnost provádí člověk-operátor. Operátor je součástí posádky ponorky operující poblíž prohledávané oblasti. Pro to, aby roboty neplnily celou misi plně autonomně, ale součástí týmu byl také člověk, existují dva hlavní důvody:

- Velká rozmanitost předmětů, které se mohou nacházet v prostoru určeném k prohledávání, by způsobovala značnou složitost algoritmu potřebného k jejich analýze a klasifikaci. Dalším problémem je možnost částečného zakrytí předmětů usedlinami, popř. snížená viditelnost způsobená zakalením vody. Tyto podmínky by vedly k příliš velkým nárokům na algoritmy počítačového vidění určené k analýze objektů.
- V případě, že je objekt klasifikován jako nebezpečný, je úkolem operátora rozhodnout, jak velké riziko by přinesl pokus o jeho odstranění. V případě velkého rizika může operátor rozhodnout, že se s objektem nebude manipulovat, ale pouze se zaznamená jeho poloha do mapy. Pokud se operátor rozhodne objekt odstranit, přepne se robot do teleoperovaného módu a operátor může vzdáleně ovládat teleskopické rameno, jímž je robot vybaven.

K tomu, aby mohl operátor rozhodnout o povaze objektu a případně převzít ovládání robotu, je nutné vytvořit mezi ním a robotem komunikační kanál. Tímto kanálem se směrem od robota k operátorovi přenáší plnohodnotný videoobraz a opačným směrem jsou vysílány příkazy ovládající robota.

Aby byl vytvořen takovýto komunikační kanál, je obvykle třeba posílat zprávy přes mezilehlé roboty. Důvodem je omezený komunikační dosah jednotlivých robotů, neboť pro komunikaci v podmořském prostředí se užívají akustické modemy.

Při řešení tohoto typu úloh není možné použít pro plánování akcí centrální prvek. Roboty se mohou z důvodu omezené komunikační dostupnosti dostat mimo dosah této entity a musely by se často vracet do její blízkosti pro nové úkoly nebo, aby jí předaly aktuální informace o stavu prostředí. Vzhledem k povaze úlohy (roboty operují v nebezpečném prostředí) může také dojít k poškození centrálního prvku. Z tohoto důvodu jsou všechny roboty nezávislé a každý z nich může převzít roli koordinátora. Roboty jsou tak vzájemně zastupitelné a každý z nich může být v případě poruchy nebo zničení nahrazen.

Při řešení úlohy se autoři nesoustředili na simulaci senzorů a akčních členů jednotlivých robotů (k tomu mohou být využity jiné simulátory, např. Gazebo či Stage, nebo simulační prostředky samotného A-globe), ale hlavně na problematiku komunikace a synchronizace znalostí v prostředí s částečnou komunikační nedostupností, na vyhledávání spolupracovníků a podporu vizualizace pro operátora.

Pro tvorbu simulačního nástroje byla použita multiagentní platforma A-globe. Ta umožňuje modelovat pohyb jednotlivých entit v systému a zároveň i jejich vzájemnou komunikační dostupnost. Vzhledem k vlastnostem akustických modemů byly rozlišovány dvě komunikační vzdálenosti.

Při použití akustických modemů s rostoucí vzdáleností vysílání velmi rychle klesá šířka komunikačního pásma. Na větší vzdálenosti je tak možné zasílat zprávy s malým datovým objemem (např. informace o poloze robota nebo jeho vytížení). Tento datový objem je nedostatečný pro vysílání zpráv nesoucích obrazovou informaci (komprimovaný videoobraz). K tomu, aby mohly vysílat video k operátorovi, se tak roboty musí k sobě přiblížit na menší vzdálenost (a tím zvýšit šířku komunikačního pásma). Agent simulující vzájemnou komunikační dostupnost tak nemodeluje jen to, zda spolu mohou dva roboty komunikovat, ale i sílu signálu. Aby bylo možné zahájit vysílání videa je nutné, aby síla signálu dosáhla určitého stupně.

Každý z robotů je reprezentován pomocí několika agentů zastupujících jeho jednotlivé části. Jsou jimi:

- **Koordinátor** – představuje řídicí jednotku robota a skládá se z následujících modulů:
  - *Řídicí jednotky* – obsahuje řídicí algoritmus agenta. Dále pak zajišťuje komunikaci s ostatními agenty reprezentujícími robot.
  - *Komunikačního modulu* – simuluje komunikační omezení akustických modemů při zasílání zpráv koordinátorům na ostatních robotech. Všechny zprávy jsou udržovány v zásobníku typu FIFO. Odtud je modul postupně vybírá, odešle a zablokuje na určitou dobu komunikační kanál – tím se simuluje skutečná doba vysílání zprávy. Ta závisí na délce zprávy, kapacitě kanálu a vzdálenosti mezi roboty.
  - *Databázového modulu* – udržuje znalosti o prohledané oblasti, nalezených objektech a ostatních robotech. Tato znalost může být upravována na základě vlastních pozorování nebo synchronizací s ostatními roboty.
  - *Synchronizační jednotky* – udržuje databázový modul aktuální a konzistentní s ostatními roboty. Dva roboty si vzájemně synchronizují své databáze v okamžiku, kdy se dostanou do komunikačního dosahu, nebo pokud si jeden z nich změní obsah své databáze (buď na základě vlastního pozorování, nebo obdržením synchronizační

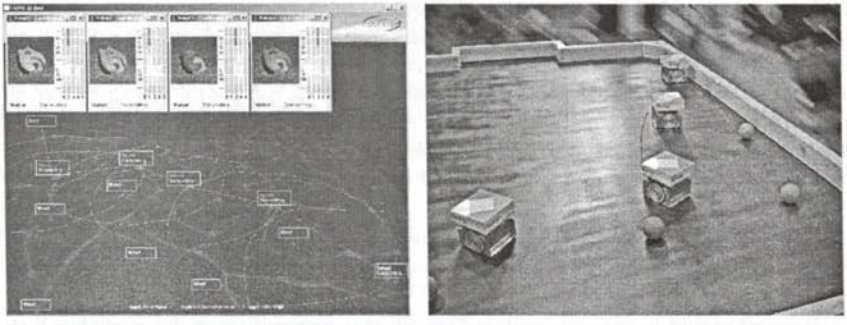
zprávy od třetího robotu). Každý koordinátor si udržuje záznamy o čase poslední synchronizace s každým robotem. Tento záznam je využit v případě obnovení komunikační linky – koordinátoři si vzájemně zašlou databázové záznamy, které byly modifikovány nebo nově přidány od okamžiku poslední vzájemné synchronizace. Synchronizační jednotka řeší možné konflikty mezi aktuálním záznamem v databázi a údaji v synchronizační zprávě.

- *Plánovače pohybu* – plánuje pohyb robota v prostředí. Výsledkem jeho činnosti je výběr části oblasti k prohledání a seznam bodů v prostoru, kterými musí robot projet. Výběr oblastí je prováděn dynamicky na základě vyjednávání s ostatními roboty.
- *Plánovače vysílací trasy* – aktivuje se v okamžiku, kdy robot detekuje podezřelý objekt. Vyhledá vhodné roboty a určí jejich rozmístění v prostoru tak, aby mohl být vytvořen komunikační kanál pro přenos videa k operátorovi.

- **Podvozek** – řídí pohyb robota, počítá jeho aktuální pozici a tuto informaci prostřednictvím služeb platformy A-globe předává ostatním simulačním prvkům. Řídicí příkazy do podvozku zasílá koordinátor.

- **Detektor** – simuluje komponentu zjišťující přítomnost podezřelých objektů v detekčním dosahu senzorů. Informace o nalezených objektech předává koordinátorovi.

- **Videomodul** – zajišťuje příjem a vysílání video signálu. Tento modul získá data ze zdroje poskytnutého simulátorem a připraví vysílací trasu od zdroje k operátorovi pomocí naklonování vysílajících agentů.



Obr. 20.23 Příklady komunikačních kanálů při prohledávání podmořských oblastí.

Oblast, kterou mají roboty prohledat, je předem známa. Roboty si vzájemným vyjednáváním postupně vybírají části této oblasti a oznámí to ostatním. Tato informace je využívána jednak při dalším výběru (robotem vybraná část oblasti je označena jako zabraná) a dále může sloužit při pokusu o komunikační spojení s těmito roboty – získává se tak znalost, v které oblasti se má robot přibližně nacházet. Tuto znalost lze využít při vyhledávání spolupracovníků pro tvorbu komunikačního kanálu a umožňuje také v určitých případech zjistit i případné poškození či zničení robota (nepohybuje se ve zvolené oblasti a žádný z robotů nemá informace o jeho aktuální činnosti), čímž se zvyšuje robustnost algoritmů pro řešení úlohy.

Po výběru části oblasti si roboty vytvoří plán pohybu tak, aby pokryly celou zadanou oblast svými senzory, a začnou tento plán vykonávat. Pokud naleznou podezřelý objekt, který ještě nebyl prověřen operátorem (informaci o takových objektech vzájemně sdílejí a mají ji uloženu ve svých databázových modulech), pokusí se připravit komunikační kanál pro odeslání videoobrazu. V případě, že jsou mimo komunikační dosah od operátora, potřebují ke splnění této úlohy pomoc ostatních robotů. Tento proces se skládá ze dvou fází:

- *Nalezení minimálního počtu robotů potřebných pro vytvoření komunikačního kanálu.* Robot, jenž nalezl objekt, se stává koordinátorem plánovacích aktivit. K nalezení dalších robotů lze použít následující strategie:
  - *Přeposílání zpráv*, kdy robot zašle žádost všem ostatním robotům ve svém komunikačním dosahu. Tyto roboty zprávu přepošlou svým sousedům atd. (je kontrolována možnost vzniku cyklů). Pokud robot nemá žádné sousedy, kterým by zprávu přeposlal, nebo již dostal od všech odpověď, pošle odpověď svému předchůdci. Tímto způsobem získá koordinátor informace o všech dostupných robotech.
  - *Vyhledávání robotů v prostoru*, pokud se v některých případech nepodaří nalézt dostatečný počet spolupracovníků ani předchozí metodou (obzvláště v případě velkého prostoru a malého počtu robotů). V takovém případě může robot opustit nalezený objekt a hledat další roboty. K tomu využívá znalost o přiřazení částí prostoru jednotlivým robotům. Všichni dosud nalezení spolupracovníci se pohybují společně ve formaci, aby pokryli komunikačním signálem co největší oblast. Roboty pokračují v hledání, dokud nenaleznou dostatek spolupracovníků nebo neprohledají celý prostor.

  Pokud robot nalezl dostatečný počet spolupracovníků, určí koordinátor pozice spolupracovníků v prostoru pro zajištění videopřenosu. V případě, že se dostatečný počet spolupracovníků nepodaří nalézt, nahraje robot videozáznam podezřelého objektu do své paměti. Tento záznam je použit pro analýzu operátorem po dokončení mise. Záznam může být dostatečný pro analýzu objektu, v opačném případě může operátor zadat příkaz k opakovanému pokusu o vytvoření komunikačního kanálu.
- *Rozmístění robotů v prostoru.* Koordinátor vybírá z množiny dostupných spolupracovníků ty, jejichž použití zajistí optimální vytvoření komunikačního mostu podle určitého kritéria. Tímto kritériem může být například minimalizace spotřebované energie pro přesun na místo vysílání, minimalizace času přerušení aktuální aktivity nebo pouze snaha použít co nejmenší počet robotů. Jednotliví spolupracovníci se poté přesunou na určené místo v prostoru a pokoušejí se kontaktovat své sousedy. V okamžiku, kdy jsou všichni sousedé vzájemně v komunikačním dosahu, je zahájeno vysílání (příklady řešení simulovaného na počítači a modelovaného souboru robůtků jsou uvedeny na obr. 20.23, též barev. příl.).

Vyvinuté algoritmy je možné použít v různých aplikacích podmořského průzkumu. Příkladem může být odminování pobřeží před vyloděním jednotek při vojenských operacích, průzkum přístavů a mol pro možnost zakotvení lodí s humanitární pomocí při záchranných misích. Jsou obecně využitelné i pro hydrografický a geofyzikální průzkum. Operátor může podle aktuální úlohy a vlastností prostředí volit použité algoritmy a strategie a nastavovat jejich parametry.

### 20.7.5 Řízení autonomních bezpilotních letadel

Stále se zvyšující počty letadel operujících v letovém prostoru vedou k nutnosti vývoje nových technologií pro jejich řízení. Stávající přístup založený na využití pevně definovaných letových hladin a koridorů totiž neumožňuje optimální využívání celého letového prostoru. Řešením tohoto problému je koncept volného letu (free flight). Ten na rozdíl od klasicky předdefinovaných koridorů používá autonomní algoritmy plánování letových tras. Letové trasy jsou tak unikátní pro každé letadlo a pro aktuální podmínky v letovém prostoru. Tento přístup je plně decentralizovaný a odstraňuje zároveň i problém s omezenou kapacitou řídicích věží a vzájemnou synchronizací mezi věžemi při předávání kontroly nad jednotlivými letadly.

Protože při autonomním plánování nemusí existovat přesná znalost budoucích tras všech letadel v prostoru, nemusí být ani odhalena možnost budoucí kolize. Pro řešení tohoto problému jsou používány různé metody *odhalování konfliktů* (collision detection) a metody *řešení konfliktů* (collision avoidance) v průběhu letu, tzv. *dekonflikční algoritmy*. Tyto algoritmy mohou být založeny na vzájemném vyjednávání mezi letadly. V takovém případě si letadla vymění části svých budoucích letových plánů a společně se snaží o jejich úpravu tak, aby se předešlo případným kolizím – *kooperativní dekonflikce*. Pokud spolu letadla nemohou komunikovat (např. mají nekompatibilní komunikační protokoly nebo algoritmy či při poruše komunikace), při nedůvěře k druhému letadlu či jeho nepřátelském chování, je nutné použít metody nekooperativního řešení konfliktu – *nekooperativní dekonflikce*.

Vzhledem k velkém počtu letadel v prostoru je zpravidla nutné řešit několik konfliktů současně použitím obou přístupů.

#### *20.7.5.1 Systém Air Traffic Control (ATC)*

Air Traffic Control (ATC) je multiagentní systém pro simulaci letového provozu, vyvinutý v Gerstnerově laboratoři ČVUT FEL. Simulovaná letadla, s nimiž systém pracuje, jsou (až na výjimky) bezpilotní letouny (UAV). Systém je vybudován nad multiagentní platformou A-globe.

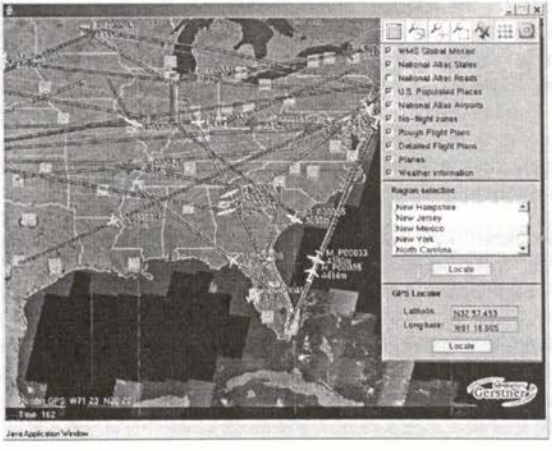
Každé simulované letadlo reprezentují dva agenti:

- *Agent letadla* – reprezentuje technické systémy letadla a pracuje s aktuálním letovým plánem tvořeným množinou význačných bodů, jimiž musí letadlo proletět (tzv. waypointy). Přitom respektuje důležité fyzikální parametry letadla, jako jsou minimální a maximální rychlost a zrychlení, úhly stoupání a klesání, poloměry zatáčení apod. Zprostředkovává také informace o poloze objektů detekovaných palubním radarem a synchronizační časové značky. Na základě příkazů od agenta pilota provádí vlastní aplikaci nově navržených letových plánů.
- *Agent pilota* – simuluje činnost pilota a je hlavní řídicí jednotkou simulovaného letadla. Odpovídá za tvorbu a modifikaci letových plánů. Při prostorovém plánování využívá informace o zakázaných (bezletových) zónách, které si udržuje ve své paměti. Tyto zóny mohou být statické, zadané předem (např. obalové oblasti horského terénu, oblasti okolo jaderných elektráren a vojenských objektů), nebo dynamické (obalové zóny nekooperativních letadel). O řešení kolizních situací se stará dekonflikční modul. Jedná se o systém pro detekci kolizí, výběr, spouštění a přepínání různých metod jejich řešení (kooperativních i nekooperativních).

Zprávy od agenta pilota směrem k agentu letadla odpovídají požadavkům na změnu letového plánu a naopak zprávy jdoucí od agenta letadla k agentu pilota odpovídají informacím o aktuálním pozici a situaci v okolí letadla. Součástí systému ATC je i soubor agentů, kteří se starají o simulaci prostředí, v němž letadla operují. Tato část systému je centralizovaná a běží na speciální části platformy zvané simulační server. Jednotlivá letadla v tomto případě představují klienty, kteří získávají ze serveru informace o aktuálním stavu letadel i okolního prostředí. Základními simulačními agenty jsou:

- *Konfigurační agent* – nahrává, předzpracovává a předává dále všechny konfigurace systému ATC.
- *Správce entit* – spravuje simulovaná letadla připojená do systému, spouští nová, ruší existující, předává výchozí plány novým letadlům atd.
- *Simulátor entit* – počítá skutečné pozice simulovaných letadel na základě letových plánů a příkazů od agentů letadel.
- *Simulátor komunikační dostupnosti a kolizí* – vyhodnocuje vzájemné vzdálenosti letadel. Tato informace je využita platformou A-globe pro správu vzájemné komunikační dostupnosti (znemožňuje zaslání zpráv mezi příliš vzdálenými letadly). Dále detekuje kolize mezi objekty a simuluje jejich zničení při střetu.
- *Senzorický agent* – reprezentuje všechny detekční systémy letadel, jako jsou například palubní radary.

Pro vizualizaci akcí v systému slouží specializovaný agent-operátor, používaný také jako uživatelské rozhraní systému, a webový klient, který umožňuje sledovat simulaci ze vzdálených počítačů.




Obr. 20.24 Ukázka simulace dekonflikce letadel ve webovském prostředí.

Stěžejní datová struktura využívaná při řešení konfliktů je letový plán. Popisuje trajektorii, po které letí letadlo v průběhu simulace. Plán je vytvářen agentem pilota, předáván do agenta letadla, který na jeho základě udržuje letovou trajektorii. Skutečný fyzický pohyb je počítán v simulátoru entit, který na základě aktuálního času simulace a letového plánu aktualizuje pozici letadla v prostoru a vrací ji zpět agentovi letadla.

Letový plán je tvořen významnými body, kterými musí letadlo proletět, aby splnilo svou misi (way-pointy) a segmenty (části trajektorie mezi těmito body). *Waypointy* definující misi nemohou být přeplánovány. Dekonflikční algoritmy ale mohou do letového plánu přidávat další pomocné body, čímž mění letový plán a předcházejí tak kolizím.

### *20.7.5.2 Algoritmy řešení kolizí*

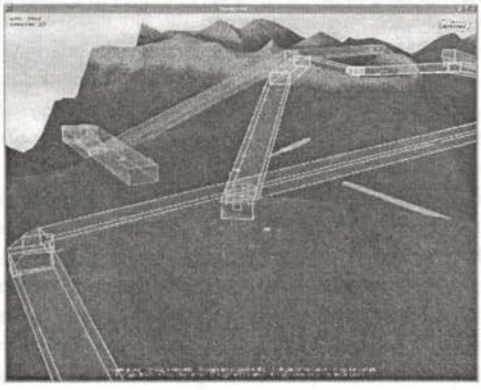
**Kooperativní dekonflikce.** Algoritmy kooperativního vyhýbání jsou založené na vyjednávání mezi letadly. Lze je tedy použít pouze v případě, že spolu letadla mohou vzájemně komunikovat. V případě přiblížení letadel se agenti pilotů začnou vzájemně informovat o svých lokálních letových plánech. Po obdržení této části plánu se spustí proces detekce kolize, který porovnává přijatý lokální letový plán s vlastním plánem letadla. Pokud je nalezen určitý časový okamžik, kdy je vzdálenost mezi letadly menší než předem určený bezpečnostní rozestup, spustí se dekonflikční mechanismus, který modifikuje letové plány. Pro kooperativní řešení kolizí jsou implementovány dva postupy:

- *Pravidlový systém* – využívá systém pravidel, která modifikují letový plán. Tato pravidla jsou vybírána v závislosti na směru, kterým se k sobě obě letadla přibližují. Okolí letadla je rozděleno na čtyři sektory reprezentující typické směry přiblížení (zepředu, zezadu, zprava a zleva). Pro každý sektor je definována skupina pravidel, která letadla aplikují na svůj letový plán a vyřeší tak kolizní situaci. Pravidla jsou složena z elementárních akcí typu zahni vpravo, zrychli, zpomal. Změny plánu jsou okamžitě porovnány s plány ostatních letadel a v případě detekce nové kolize jsou provedeny další změny. Všechna letadla v systému musí pro řešení kolizí využívat stejnou množinu pravidel.
- *Utilitární systém* – na každý z plánů aplikuje sadu operátorů (vybočení do stran, změnu rychlosti, změnu výšky apod.). U operátorů mohou být dynamicky nastaveny parametry, jako je velikost vybočení nebo zrychlení. To urychluje hledání vhodné kombinace plánů – začíná se s malou množinou operátorů a v případě neúspěšného pokusu o řešení kolize se změnou parametrů operátorů tato množina rozšiřuje. Aplikací operátorů vznikne sada možných nových letových plánů pro každé ze dvojice letadel. Poté jsou vzájemně otestovány všechny dvojice plánů a podle optimalizační funkce je vybrána nejlepší dvojici plánů, které pak obě letadla aplikují.

**Nekooperativní dekonflikce.** Na rozdíl od ostatních přístupů je toto řešení založené na principu plánování trasy pomocí algoritmu A* v okolí dynamických zakázaných zón. Nekooperativní objekty jsou detekovány pomocí palubního radaru letadla, na základě sledování jejich pohybu jsou predikovány možné polohy objektu v blízké budoucnosti. Množina těchto bodů představuje dynamickou zakázanou zónu. Tyto zóny jsou průběžně aktualizovány s novými informacemi z radaru. Detailní popis tohoto algoritmu lze nalézt v (Komenda, 2007). Příklad řešení kolizní situace algoritmem nekooperativního vyhýbání je uveden na obr. 20.25.

**Vícevrstvá dekonflikční architektura.** V reálných prostředí se pohybuje velké množství objektů, které používají různé dekonflikční algoritmy a často mění svoje plány. Mnoho z nich je vzájemně nekompatibilních nebo nespolupracujících. Proto byla vyvinuta vícevrstevná dekonflikční architektura, která umožňuje současné použití kooperativních i nekooperativních algoritmů. Tato architektura je implementována jako modul CSM (Collision Solver Manager). Modul je navržen jako doménově nezávislý a je tak

možné jej využít nejen pro řešení kolizí mezi letadly, ale i například pro auta, roboty nebo lodě. Pro přizpůsobení vybrané doméně je třeba implementovat konkrétní reprezentaci plánu pohybu.



Obr. 20.25 Výsledek řešení kolizní situace pro více letadel při použití nekooperativní dekonflikce.

### 20.7.6 Další zajímavé multirobotické projekty ve světě

V posledních deseti letech vzniklo ve světě množství projektů zabývajících se nasazením týmu robotů v reálných aplikacích. Účelem této kapitoly není podat kompletní přehled všech takových projektů. Zmíníme se pouze o těch, které jsou z nějakého důvodu zajímavé a které zároveň ukazují různorodost nasazení multirobotických systémů v různých situacích.

Projekt *Martha* řešilo na přelomu tisíciletí několik evropských univerzit společně s průmyslovými partnery (Alami a kol., 1998). Jeho cílem bylo vyvinout systém pro správu skupiny robotů překládajících náklad v přístavech, na letištích a seřaďovacích nádražích. Klíčovým prvkem systému byla autonomnost jednotlivých robotů umožňující jim se vypořádat s nečekanými událostmi a nepřesným modelem prostředí. Proto byly roboty vybaveny senzory pro lokalizaci, detekci překážek a mapování. Vzhledem k tomu, že prostředí úlohy je vysoce dynamické a nelze korektně předpovídat dobu uskutečnění akcí (roboty mohou měnit svou rychlost kvůli objíždění překážek nebo lokalizačním akcím, mohou nastat prodlevy při nakládání či vykládání atd.), nebylo možné centrálně vytvářet optimální a detailní dlouhodobé nebo střednědobé plány. Proto byly vyvinuty speciální metody umožňující robotům efektivně komunikovat o svých plánech a koordinovat tak svou činnost.

Projekt *Centibots* (www.ai.sri.com/centibots) byl financován agenturou DARPA (Defence Advanced Projects Agency) a na jeho řešení spolupracovaly SRI International, Stanford University, University of Washington a ActivMedia. Jeho cílem bylo vytvořit systém řídící velké skupiny robotů schopných koordinovaně se pohybovat v neznámém prostředí, kooperativně ho prohledávat a plánovat v něm své akce. V první fázi projektu vznikl tým mapovacích robotů Pionner vybavených laserovými dálkoměry provádějící

dohled dané oblasti a vytvářející její distribuovaný model. Následovala druhá skupina robotů Amigobots, které hledaly zajímavé objekty v určeném území, detekovaly a sledovaly „vetřelce" a sdílely informace mezi sebou a s řídicím centrem. Roboty byly plně autonomní a nezávislé na komunikační infrastruktuře, neboť používaly vlastní komunikační síť, díky které byly schopny koordinovat své akce a optimalizovat své chování. Pokud některý robot nemohl splnit zadaný úkol, ostatní byly schopny jej nahradit.

V létech 2003–2005 byl řešen devíti partnery z Finska, Francie, Velké Británie a Německa projekt *SWARM* (www.seaswarm.com), jenž se zabýval nasazením robotů pro podvodní monitorovací systém v mělkých vodách Baltského moře. Systém se skládal z množství podvodních robotických sond schopných provádět dvoutýdenní mise autonomně. Sondy řídily svůj vztlak, ale jinak se nechaly unášet proudem. Byly schopné komunikovat navzájem i s řídicí stanicí (pomocí modemu, rádia, GSM a satelitu). Kromě měření standardních fyzikálních a chemických ukazatelů (tlak, teplota, vodivost) byly sledovány populace mořských řas s cílem předpovídat jejich další výskyt a chování (vytváření vodního květu), což by umožnilo lepší využití vodních zdrojů.

Jiným zajímavým projektem byl projekt *Swarm-bots* (www.swarm–bots.org) řešený v létech 2000–2005 v rámci Evropského programu IST–FET institucemi v Belgii, Švýcarsku a Itálii, který se zabýval návrhem a implementací samoorganizujících a samosestavujících se robotů. Konkrétním výstupem byla skupina asi 30 autonomních robotů samostatně se pohybujících v prostředí. V případě potřeby se roboty byly schopny fyzicky spojit a vytvořit tak geometricky složitější struktury pracující jako celek.

V současné době je pod koordinací Univesität Karlsruhe řešen další IST–FET projekt *I-Swarm* (www.i-swarm.org) zabývající se vývojem mikrorobotického systému sestávajícího se až z tisíce heterogenních, velmi malých robotů (2 × 2 × 1 mm) s různými druhy senzorů, manipulátorů a s různou výpočetní kapacitou. Očekávaným výstupem projektu je realizace efektu samoorganizace tak, jak je v přírodě pozorován například v mraveních nebo včelích koloniích, což umožní větší flexibilitu, přizpůsobivost a robustnost systému. Takovéto společenství lze následně nasadit v množství aplikací zahrnující mikromontáž nebo biologické, medicínské a uklízecí úlohy.

Vhodnou platformou pro testování a porovnání multirobotických systémů jsou robotické soutěže. K nejznámějším a mezi robotickými pracovišti ve světě nejoblíbenějším patří bezesporu skupina soutěží *RoboCup* (www.robocup.org) organizovaných od roku 1996. V *RoboCupSoccer* soutěží vždy dva týmy v několika kategoriích robotické kopané. Jednotlivé kategorie se liší použitou platformou (liga simulátorů nebo reálných robotů), velikostí robotů a hrací plochy, počtem robotů na ploše a typem robotů (kolové, čtyřnohé psy Aibo, humanoidní). Úkolem soutěžících a jejich robotů v *RoboCupRescue* je prohledat budovu postiženou zemětřesením, najít v ní zraněné osoby, zjistit jejich stav a zanést jejich polohu do mapy, kterou si robot samostatně vytváří. Soutěž probíhá jednak v simulovaném prostředí (kde se řeší zejména spolupráce a koordinace více robotů) a jednak s reálnými roboty, které mohou být buď autonomní, nebo teleoperované. Ve všech zmíněných kategoriích se každoročně koná řada národních i mezinárodních soutěží a mistrovství světa. V roce 2006 mistrovství světa hostily Brémy a zúčastnilo se jej rekordních 440 týmů z 36 zemí celého světa.

## 20.8 Závěr – vize dalšího vývoje

Tato kapitola si klade za cíl ukázat základní myšlenkové postupy a prvotní experimenty v oblasti kolaborativní robotiky, tedy oblasti v níž očekáváme v nejbližší budoucnosti prudký rozvoj. Na velmi jednoduchých úlohách kolektivní lokalizace a identifikace prostředí, simulace a robotické teleprezence jsme jednak naznačili, kudy a kam se ubírá kolaborativní robotika, ale současně jsme dokladovali, že výzkum v ČR nestojí stranou tohoto proudu. Jsme však skutečně jenom na samém počátku dlouhé cesty, která na výzkumníky čeká.

Výzkum bude motivován především praktickými potřebami v několika aplikačních oblastech pro některé typické úlohy. Konkrétně půjde o následující typy úloh (ale zdá se, že nejenom o ně):

- *Distribuované montážní a jiné operace v těžko dostupném nebo nebezpečném prostředí vyžadující přesnou časovou synchronizaci.* Takové úlohy se vyskytují například v rámci kosmického programu, v jaderné energetice, při těžbě surovin z mořského dna apod.
- *Záchranné a preventivně-diagnostické operace smíšených týmů specializovaných robotů a lidí-operátorů.* Bude se jednat vlastně o inteligentní systémy telepresence, opírající se o průzkumné roboty (distribuované vnímání) a výkonné manipulátory (distribuované aktuátory), přičemž zásadní rozhodovací činnost bude ponechána člověku. Takové úlohy se vyskytují například při vyhledávání osob v sutinách domů, při diagnostice potrubních systémů apod.
- *Vojenské operace s využitím dočasných aliancí a koalic inteligentních bojových systémů* (mnohdy se vůbec nepodobajících robotům v klasické antropomorfní vizi) a vojáků s cílem násobit údernou sílu nasazováním robotů do nebezpečných situací při ponechání si práva vrcholového rozhodování lidským bytostem.
- *Inteligentní vytváření dostatečně propustných komunikačních kanálů* optimálním automatickým rozmísťováním robotů-retranslačních buněk, a to ať již v oblastech po živelné katastrofě, na bojišti či pod mořskou hladinou.
- *Integraci robotů a jejich týmů do průmyslové výroby*, a to nikoliv jako pouhých manipulátorů, ale přímo jako součásti inteligentní obsluhy výrobních strojů a zařízení.
- *Kolektivní hry*, které budou sloužit jako motivační pobídka a současně testovací platforma pro nové algoritmy kolaborativní robotiky. Lze očekávat vyhlášení nových typů soutěží. Předpokládá se, že kolem roku 2050 by týmy robotů mohly porážet reprezentační fotbalové jedenáctky.
- *Kolaborativní aplikace nanorobotů pro medicínské použití*, které po vpravení do těla pacienta vyhledávají a poté opravují nebo naopak ničí poškození na úrovni jednotlivých buněk.

Vize úspěšného řešení vyjmenovaných, ale i dalších úloh povede v nejbližším období k nezbytným výzkumným trendům:

- Budou rozvíjeny univerzální unifikované softwarové platformy umožňující budovat speciální datové a znalostní architektury jednotlivých robotů (sociální modely) a poskytující unifikované, pro robotiku specifické, algoritmy vyjednávání, usuzování

a rozhodování. Zde lze očekávat velmi těsnou spolupráci s dnes již relativně rozvinutou oblastí multiagentních systémů, pro niž se problémy kolaborativní robotiky stávají významným motivujícím prvkem. Lze očekávat, že informační agenti pro potřeby robotiky se více „přiblíží hmotě" a budou se ve svém usuzování standardně zabývat otázkami fyzické realizovatelnosti navrhovaných řešení v reálném světě.

- Budou rozvíjeny multiagentní simulační platformy pro multirobotické a smíšené týmy s přímou a bezproblémovou přenositelností kódu ze simulátorů na roboty či jiné automaty.
- Mimořádná pozornost musí být věnována výzkumu a vývoji adaptivních a učících se algoritmů, které budou umožňovat postupné nahrazování člověka roboty ve smíšených týmech. Chování člověka ve složitých situacích nelze prostě popsat „ručně" vytvořeným kódem. Zdá se však, že dosavadní metody učení nestačí na mimořádně náročné požadavky kolaborativní robotiky. Lze očekávat, v souladu například s názory nositele Nobelovy ceny za fyziku (1998) Roberta B. Laughlina, že kolaborativní robotika přinese díky nezbytně novým algoritmům učení a automatického programování doslova průlom v softwarovém inženýrství.
- Intenzivní výzkum se soustředí na tvorbu unifikovaných rozhraní člověk–stroj s cílem vytvořit automatizované (později i učící se) obousměrné „překladače" mezi člověkem a robotem. Jedině s takovýmito rozhraními bude umožněna skutečná a efektivní spolupráce robotů a lidí ve smíšených týmech.

Celkový vývoj v oblasti kolaborativní robotiky je dlouhodobě motivován golemovskou vizí existence učících se antropomorfních strojů, inteligentních ve smyslu Minského definice umělé inteligence, která vychází z Turingova testu: Pokud po určité době činnosti nebude možné rozeznat, která zpráva byla vyslána nebo akce motivována člověkem či strojem, dosáhl robot stejné inteligence jako člověk. A právě ve vizi plnohodnotné náhrady lidských týmů, posádek a jednotek kolektivy inteligentních strojů a touze lidstva tuto vizi naplnit lze spatřovat záruku dynamického rozvoje kolaborativní robotiky.

## Literatura


Alami R., Fleury S., Herrb M., Ingrand F., Robert F.: *Multi-robot Cooperation in the MARTHA project*. IEEE Robotics and Automation Magazine, Vol. 5, No. 1, 1998, s. 36–47.

Arkin R. C.: *Motor Schema Based Navigation for a Mobile Robot: An Approach to Programming by Behavior*. In: Proceeding of the IEEE International Conference on Robotics and Automation, IEEE Computer Society Press, 1987, s. 264–271.

Barfield W., Sheridan T., Zeltzer D., Slatter M.: *Presence and Performance Within Virtual Environments*. In: Virtual Environments and Advanced Interface Design, Oxford University Press, 1995.

Bar-Noy A., Bar-Yehuda R., Freund A., Naor J. S., Schieber, B.: *A Unified Approach to Approximating Resource Allocation and Scheduling*. J. ACM, 48(5), 2001, s. 1069–1090.

Bern M.: *Triangulations and Mesh Generation*, In: Handbook of Discrete and Computational Geometry, Chap. 25, (J. E. Goodman a J. O'Rourke, eds), CRC Press, 2004.

Blum A. L., Furst M. L.: *Fast Planning through Planning Graph Analysis*. Artificial Intelligence, 90, 1997, s. 281–300.

Bonasso R. P. a kol.: *Experiences with an Architecture for Intelligent Reactive Agents*. Journal of Experimental and Theoretical Artificial Intelligence, 9(2), 1997, s. 237–256.

Boutilier C., Goldszmidt M., Sabata B.: *Sequential Auctions for the Allocation of Resources with* Complementarities. In: Proc. of IJCAI '99, Morgan Kaufmann Publishers Inc., 1999, s. 527–523.

Brennan, R. W., Norrie, D. H.: *Agents, Holons and Function Blocks: Distributed Intelligent Control in Manufacturing.* Journal of Applied Systems Studies, 2(1), 2001, s. 1–19.

Brooks R. A.: Intelligence Without Representation. Artificial Intelligence 47 (1–3), 1991, s. 139–160.

Burgard W., Fox D., Jans H., Matenar C., Thrun S.: *Sonar-based Mapping of Large-scale Mobile Robot Environments Using EM.* In: Proc. of the 16th International Conference on Machine Learning (ICML '99), 1999, s. 67–76.

Burgard W., Moors M., Stachniss C., Schneider F.: *Coordinated Multi–Robot Exploration.* IEEE Transactions on Robotics, 21 (3), 2005, s. 376–378.

Cao Y. U., Fukunaga A.S., Kahng A.B.: *Cooperative Mobile Robotics: Antecedents and Directions.* Autonomous Robots, Vol. 4 (1), 1997, s. 17–23.

Collins G. E.: *Quantifier Elimination for the Elementary Theory of Real Closed Fields by Cylindrical Algebraic Decomposition.* LNCS No. **33**, Springer Verlag, 1975, s. 134–183.

Connolly C., Grupen C.: *The Application of Harmonic Functions to Robotics.* Journal of Robotic Systems 10(7), 1993, s. 931–946.

Cox I. J., Leonard J. J.: *Modeling a Dynamic Environment Using a Bayesian Multiple Hypothesis Approach.* Artificial Intelligence, 66(2), 1994, s. 311–344.

Cox I. J.: *Blanche – an Experiment in Guidance and Navigation of an Autonomous Robot Vehicle.* IEEE Transaction on Robotics and Automation, 7(2), 1991, s. 193–204.

Cox J. S., Durfee E. H.: *An Efficient Algorithm for Multiagent Plan Coordination.* In: Proc. of AAMAS '05, ACM Press, 2005, s. 828–835.

Crowley J. L.: *World Modeling and Position Estimation for a Mobile Robot Using Ultra-sonic Ranging.* In: Proc. of the IEEE International Conference on Robotics and Automation, volume 2, 1989, s. 674–680.

de Berg M., van Kreveld M., Overmars M., Schwarzkopf O.: *Computational Geometry: Algorithms and Applications.* Springer-Verlag, Heidelberg 2000.

de Weerdt M. M., Bos A., Tonino J. F. M., Witteveen C.: *A Resource Logic for Multi–agent Plan Merging.* Annals of Mathematics and Artificial Intelligence, 37(1–2), 2003, s. 93–130.

Demel J.: *Grafy a jejich aplikace.* Academia, Praha 2002.

DesJardins M. E., Wolverton M. J.: *Coordinating a Distributed Planning System.* AI Magazine, 20(4), 1999, s. 45–53.

Dudek G., Freedman P., Hadjres S.: *Using Local Information in a Non-local Way for Mapping Graph-like Worlds.* In: Proceedings of IJCAI'93 Conference. Chambery, France, 1993, s. 1639–1647.

Dudek G., Jenkin M., Milios E., Wilkes D.: *A Taxonomy for Multi–agent Robotics.* Autonomous Robots, Vol. 3, 1996, s. 375–397.

Elfes A.: *Occupancy Grids: A Stochastic Spatial Representation for Active Robot Perception.* In: Proceedings of the Sixth Conference on Uncertainty in AI. Morgan Kaufmann Publishers, 1990, s. 60–70.

Ephrati E., Rosenschein J. S.: *A Heuristic Technique for Multiagent Planning.* Annals of Mathematics and Artificial Intelligence, 20(1–4), 1997, s. 13–67.

Erol K., Hendler J., Nau D. S.: *HTN planning: Complexity and Expressivity.* In: Proc. of AAAI–94, volume 2, AAAI Press/MIT Press, 1994, s. 1123–1128.

Excelente–Toledo C. B., Bourne R. A., Jennings N. R.: *Reasoning about Commitments and Penalties for Coordination between Autonomous Agents.* In: Proc. of the Fifth International Conference on Autonomous Agents, ACM Press, 2001, s. 131–138.

Fitzgibbon A. W. , Zisserman A.: *Automatic Camera Recovery for Closed or Open Image Sequences.* In Proc. of the European Conference on Computer Vision, Springer-Verlag, 1998.

Fletcher M., McFarlane D., Lucas A., Brusey J., Jarvis J.: *The Cambridge Packing Cell – A Holonic Enterprise Demonstrator.* In: Proc. 3rd CEEMAS Conf., LNAI No. 2691, Springer-Verlag, 2003, s. 533–543.

Fox D., Burgard W., Thrun S.: *Markov Localization for Mobile Robots in Dynamic Environments.* Journal of Artificial Intelligence Research, Vol. 11, No. 3, 1999a, s. 391–427.

Fox D., Burgard W., Dellaert F., Thrun S.: *Monte Carlo Localization: Efficient Position Estimation for Mobile Robots.* In: Proc. AAAI/IAAI, 1999b, s. 343–349.

Gatt E.: *Integrating Planning and Reacting in a Heterogenous Asynchronous Architecture for Controlling Real-World Mobile Robots.* In: Proc. of the Tenth National Conference on Artificial Intelligence, AAAI Press, 1992, s. 809–815.

Haehnel D., Burgard W., Thrun S.: *Learning Compact 3d Models of Indoor and Outdoor* Environments with a Mobile Robot. Robotics and Autonomous Systems, 44(1), 2003, s. 15–27.

Haehnel D., Fox D., Burgard W., Thrun S.: *A Higly Efficient FastSLAM Algorithm for Generating Cyclic Maps of Large-Scale Enviroments from Raw Laser Range Measurements.* In: Proc. of the IEEE/RSJ, International Conference on Inteligent Robots and Systems, IROS 2003.

Havel I. M.: *Robotika – Úvod do teorie kognitivních robotů.* SNTL – Nakladatelství technické literatury, Praha 1980.

Hlaváč V., Šonka M.: *Počítačové vidění.* Grada, Praha 1992.

Horling B., Lesser V.: *A Survey of Multi-Agent Organizational Paradigms.* The Knowledge Engineering Review, Cambridge University Press, Vol. 19 (4), 2005, s. 281–316.

Horswill I.: *Polly: A Vision–based Artificial Agent.* In: Proceedings of the Eleventh National Conference on Artificial Intelligence, AAI Press, 1993, s. 824-834.

Hunsberger L., Grosz B. J.: *A Combinatorial Auction for Collaborative Planning.* In: Proc. of ICMAS '00, IEEE Computer Society, 2000, s. 151–158.

Chazelle B.: *Approximation and Decomposition of Shapes.* Advances in Robotics 1: Algorithmic and Geometric Aspects of Robotics, (J. T. Schwartz, C. K. Yap, eds), Lawrence Erlbaum Associates, 1987, s. 145–185.

Jensfelt P., Kristensen S.: *Active Global Localisation for a Mobile Robot Using Multiple Hypothesis Tracking.* In: Proc. of the IJCAI–99 Workshop on Reasoning with Uncertainty in Robot Navigation, 1999, s. 13–22.

Jung D., Zelinsky A.: *Grounded Symbolic Communication between Heterogeneous Cooperating Robots.* Autonomous Robots, Vol. 8 (3), 2000, s. 269–292.

Jung D., Zelinsky A.: *Whisker–based Mobile Robot Navigation.* In: Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vol. 2, 1996, s. 497–504.

Ko J., Steward B., Fox D., Konolige K., Limketkai B.: *A Practical, Decision-theoretic Approach to Multi-robot Mapping and Exploration.* In: Proc. of th IEEE/RSJ, International Conference on Inteligent Robots and Systems, IROS 2003, s. 3232–3238.

Kolp M., Giorgini P., Mylopoulos J.: *Multi-Agent Architectures as Organizational Structures.* Autonomous Agents and Multi-Agent Systems, Vol. 13 (1), 2006, s. 3–25.

Komenda A.: *Použití multi–agentního systému pro nekooperativní vyhýbání letadel.* Diplomová práce, ČVUT FEL, Praha 2007.

Kulich M.: *Lokalizace a tvorba modelu prostředí v inteligentní robotice.* Disertační práce. ČVUT FEL, Praha 2003.

Latombe J. C.: *Robot Motion Planning.* Kluwer Academic Publishers, Boston, MA, 1991.

LaValle S. M: *Planning Algorithms.* Cambridge University Press, New York, 2006.

Lefèvre P., Prüss A., Zimmer U. R.: *Alice – Topographic Exploration, Cartography and Adaptive Navigation on a Simple Mobile Robot.* In: Proc. of TSRPC '94. Leeuwenhorst, Nizozemí, 1994.

Lemaire T., Berger C., Jung I., Lacroix S.: *Simultaneous Localisation and Mapping: A Stereo Vision Based Approach.* In: Proc. of the International Conference on Intelligent Robots and Systems, IROS 2006, Peking, Čína, 2006.

Lu F., Milios E.: *Globally Consistent Range Scan Alignment for Environment Mapping.* Autonomous Robots, Vol. 4, 1997, s. 333–349.

Lu F., Milios E.: *Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans.* In: Proc. of CVPR'94, 1994, s. 935–938.

Maes P., Brooks R. A.: *Learning to Coordinate Behaviors.* In: Proc. of the 8th National Conference on Artificial Intelligence, AAI Press, 1990, s. 796-804.

Mahadevan S., Connell J.: *Automatic Programming of Behaviour-Based Robots Using Reinforcement Learning.* In: Proc. of the 9th National Conference on Artificial Intelligence, Vol. 2, AAAI Press/MIT Press, Cambridge MA, 1991.

Malý K.: *Vytvoření symbolického modelu světa pro řízení robota.* Disertační práce, ČVUT FEL, Praha 2005.

Mařík V., Pěchouček M., Štěpánková O.: *Social Knowledge in Multi-Agent Systems.* In: Multi-Agent Systems and Applications, LNAI 2086, Springer-Verlag, Heidelberg, 2001a, s. 211–245.

Mařík V., Štěpánková O., Lažanský J. a kol: *Umělá inteligence (1).* Academia, Praha 1993.

Mařík V., Štěpánková O., Lažanský J. a kol.: *Umělá Inteligence (2).* Academia, Praha 1997.

Mařík V., Štěpánková O., Lažanský J. a kol.: *Umělá Inteligence (3).* Academia, Praha 2001b.

Mařík V., Štěpánková O., Lažanský J. a kol.: *Umělá Inteligence (4).* Academia, Praha 2003.

Mařík V., McFarlane D.: *Industrial Adoption of Agent-Based Technologies.* IEEE Intelligent Systems, Vol. 20, No. 1, 2005, s. 27–35.

Mataric M. J.: *Integration of Representation into Goal-Driven Behavior-Based Robots.* IEEE Transactions on Robotics and Automation, 8 (3), 1992, s. 304–312.

May S., Werner B., Surmann H., Pervoelz K.: *3D Time–of–flight Cameras for Mobile Robotics.* In: Proc. of the International Conference on Intelligent Robots and Systems, IROS 2006, Peking, Čína, 2006.

Moening C., Dogston N.A.: *A New Point Cloud Simplification Algorithm.* In: Proc. of the 3rd International Conference on Visualization, Imaging, and Image Processing VIIP 2003, Benalmadena, Španělsko, 2003.

Moravec H., Elfes A.: *High Resolution Map from Wide-angle Sonar.* In: Proc. of the IEEE International Conference on Robotics and Automation, 1985, s. 116–121.

Murphy R. R., Mali A.: *Lessons Learned in Integrating Sensing into Autonomous Mobile Robot Architectures.* Journal of Experimental and Theoretical Artificial Intelligence, 9(2), 1996, s. 191–202.

Nemzow U.: *„Meaning" through Clustering by Self–Organization of Spatial and Temporal Information.* In: Computation for Metaphors, Analogy and Agents, Lecture Notes in Artificial Intelligence, Springer-Verlag Heidelberg, Londýn, New York 1999, s. 1562.

Nilsson N. J.: *A Mobile Automaton: An Application of AI Techniques.* In Proc. of the First International Joint Conference on Artificial Intelligence, Morgan Kaufmann Publ., San Francisco 1969, s. 509–520.

Ortiz C. L., Vincent R, Morisset B.: *Task Inference and Distributed Task Management in the Centibots Robotic System.* In: Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems, ACM, 2005, s. 860–867.

Pěchouček M., Mařík V., Bárta J.: *A Knowledge–Based Approach to Coalition Formation.* IEEE Intelligent Systems, Vol. 17 (3), 2002, s. 17–25.

Přeučil L., Štepán P.: *Intelligent Self Guided Vehicles: Common Methods, Requirements and Optimal Design.* École Nationale Supérieure des Télécommunications. Paříž, 1996.

Riley P., Veloso M. M.: *Coach Planning with Opponent Models for Distributed Execution.* Autonomous Agents and Multi-Agent Systems, 13/3, 2006, s. 293–325.

Sandholm T.: *Algorithm for Optimal Winner Determination in Combinatorial Auctions.* Artificial Intelligence, 135(1–2), 2002, s. 1–54.

Shaffer G.: *Two-Dimensional Mapping of Expansive Unknown Areas.* PhD thesis. Robotics Institute, Carnegie Mellon University. Pittsburgh, PA, 1995.

Smith R. G.: *The Contract Net Protocol: High Level Communication and Control in a Distributed Problem Solver.* IEEE Transactions on Computers, C–29(12), 1980, s. 1104–1113.

Smith R.G., Davis R.: *Negotiation as a Metaphor for Distributed Problem Solving.* Artificial Intelligence, Vol. 20, 1983, s. 63–109.

Suomela J., Saarinen J., Halme A., Harmo P.: *Online Interactive Building of Presence.* In: Proc. of the 4th International Conference on Field and Service Robotics FSR 2003, University of Tsukuba, Japonsko, 2003, s. 195–200.

Šišlák D., Pěchouček M., Rehák M., Tožička J., Benda P.: *Solving Inaccessibility in Multi-agent Systems by Mobile Middle-agents.* Multiagent and Grid Systems, Vol. 1 (2), 2005, s. 73–87.

Šišlák D., Rehák M., Pěchouček M., Rollo M., Pavlíček D.: *A-globe: Agent Development Platform with Inaccessibility and Mobility Support.* In: Software Agent-Based Applications, Platforms and Development Kits (Unland R., Klusch M., Calisti M., eds), Birkhauser Verlag, 2005, s. 21–46.

Tambe M.: *Towards Flexible Teamwork.* Journal of Artificial Intelligence Research, Vol. 7, 1997, s. 83–124.

Thrun S., Bucken A.: *Integrating Grid-based and Topological Maps for Mobile Robot Navigation.* In: AAAI/IAAI, Vol. 2, 1996, s. 944–950.

Thrun S., Burgard W., Fox D.: *A Real-Time Algorithm for Mobile Robot Mapping with Applications to Multi–Robot and 3D Mapping.* In: Proc. of the International Conference on Robotics and Automation, vol. 1, San Francisco 2000, s. 321–328.

Thrun S., Fox D., Burgard W., Dellaert F.: *Robust Monte Carlo Localization for Mobile Robots.* Artificial Intelligence, 128(1–2), 2001, s. 99–141.

Vrba P.: MAST: *Manufacturing Agent Simulation Tool.* In: Proc. of the IEEE Conf. on Emerging Technologies and Factory Automation ETFA´03, Vol. 1, Lisbon, Portugalsko, 2003, s. 282–287.

Vrba P., Mařík V.: *Simulation in Agent-based Manufacturing Control Systems.* In: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Havajské o., 2005, s. 1718–1723.

Vrba P., Macůrek F. Mařík, V.: *Using Radio Frequency Identification in Agent-Based Control Systems for Industrial Applications.* In: Proc. of the 12th IFAC Symposium on Information Control Problems in Manufacturing INCOM´06, Vol. 1, Saint-Etienne, Francie, 2006, s. 459–464.

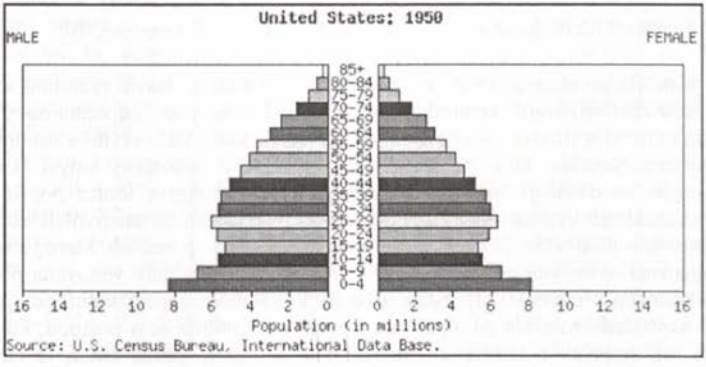
Weiss G., Puttkamer E.: *A Map Based on Laserscans without Geometric Interpretation.* In: Intelligent Autonomous systems (Rembold U. A kol., eds) , IOS Press, 1995, s. 403–407.

[1] http://www.eurobot.org
[2] http://www.fira.org
[3] http://labe.felk.cvut.cz/~pelote

# 21 Asistivní technologie

*„Tajemství toho, jak zůstat mladý, je žít slušně, jíst pomalu, neprozrazovat svůj věk a šikovně používat technologie UI."*
Martha E. Pollack (AI Magazine, 2005)

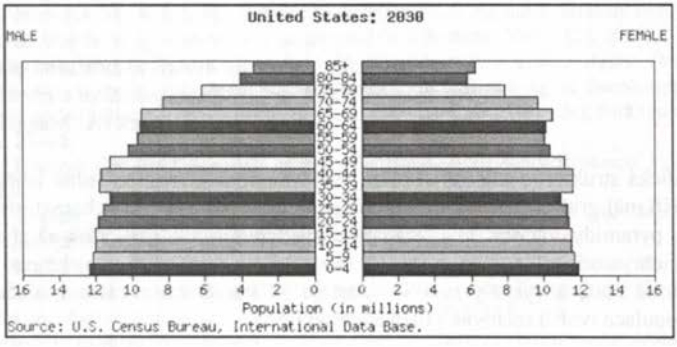
Demografická struktura společnosti se za posledních padesát let zásadně změnila. Ještě v roce 1950 měl graf znázorňující věkovou strukturu obyvatel USA charakteristický tvar populační pyramidy, viz obr. 21.1. Statistické modely pro rok 2030 však už předpovídají zcela jiný obrys odpovídajícího grafu, viz obr. 21.2. Procento lidí ve věku nad 65 let již dnes výrazně roste a vyspělý svět se vzdaluje od tradičního modelu společnosti, kdy většinu populace tvořili relativně mladí a zdraví lidé.




Obr. 21.1 Populační pyramida USA pro rok 1950.

Přestože řada starších lidí zůstává zdravá, produktivní a čilá, musíme počítat s tím, že výskyt různých zdravotních problémů se v tomto významném segmentu společnosti zásadně zvyšuje. Velmi často se zde setkáváme nejen s větší či menší ztrátou hybnosti, ale také s kognitivními poruchami. Není tajemstvím, že 50 % lidí starších 85 let trpí Alzheimerovou chorobou (Herbert, 2003), která je doprovázena nejprve náhlými výpadky paměti, později dokonce ztrátou schopnosti samostatně provádět i tak jednoduché aktivity, jako je stravování či osobní hygiena. Komplex fyzických a kognitivních problémů souvisejících s vyšším věkem vede k tomu, že postižené osoby se neobejdou bez pomoci svého okolí. Populační předpovědi však upozorňují na zásadní rozpor, který

musí naše společnost v tomto kontextu řešit. Spočívá v tom, že počet lidí, kteří potřebují při zajišťování běžných denních aktivit každodenní pomoc svých bližních, prudce stoupá, zatímco počet těch, kteří mají sílu a schopnost takovou pomoc poskytovat, zůstává stejný. Můžeme se však pokusit využít moderní techniku, pro kterou tato situace představuje novou náročnou i smělou výzvu. Samozřejmě při tom stále platí, že člověk jako pečovatel a laskavý pomocník je zcela nezastupitelný. Ale právě pro něj je třeba hledat nové prostředky, které mu budou schopny náročnou úlohu pomocníka nebo pečovatele usnadnit.




Obr. 21.2 Předpokládaný tvar populační pyramidy USA pro rok 2030.

Tato kapitola je věnována stručné informaci o aktuálním stavu výzkumu v oblasti tvorby a testování asistivních technologií, které dají člověku jako individuu nový rozměr tím, že mu umožní realizovat složitější úlohy než ty, na které stačí svými vlastními přirozenými silami. Nástroje, které se snaží nahradit některý poškozený smysl či ne plně funkční orgán, se označují jako *kompenzační pomůcky*. Nejprve krátce popíšeme cíle některých aktuálních výzkumných projektů zaměřených na tvorbu asistivních technologií a povšimneme si především postupů používaných pro tvorbu pomůcek, které jsou určeny pro kompenzaci některých poruch kognitivních funkcí. Dále bude věnována pozornost hledání vhodných prostředků pro bezdotykovou komunikaci s počítačem, což je řešení nezbytné například pro zcela paralyzované pacienty. Zmíníme se o postupu, který staví na výsledcích smělých a náročných mezioborových studií, jehož cílem je navrhnout a realizovat vznik přímého rozhraní propojujícího lidský mozek a počítač. Pak představíme původní nástroj I4Control®, který byl vytvořen na FEL ČVUT a se kterým lze ovládat počítač bez použití rukou, pouhým pohybem oka. V závěru upozorníme na otevřené aplikační možnosti pro UI v rámci vývoje asistivních technologií.

## 21.1 Co jsou asistivní technologie?

Tímto termínem jsou společně označovány:

- kompenzační či rehabilitační nástroje a pomůcky, které pomáhají člověku překonat různé typy omezení, která jej odlišují od průměrné populace,

- metodika používaná při výběru vhodných pomůcek pro konkrétního uživatele i
- postupy, které uživateli pomohou se s těmito nástroji co nejlépe „sžít" a využívat je.

Jako hlavní cíl si kladou překonávat a bourat bariéry, které řadě lidí s různými zdravotními, psychickými či sociálními problémy brání v tom, aby využívali běžné nástroje či služby (Cook, 2004), a snaží se vytvořit společenské prostředí otevřené všem. V první řadě nabízejí postiženým různé pomůcky, které jim umožní realizovat ty úkoly, které pro ně byly jinak velmi obtížné nebo dokonce nezvladatelné. Současně i upozorňují na to, že při návrhu nástrojů či architektonických prvků je třeba respektovat různorodost lidské společnosti a dbát na potřeby všech tak, aby nové výrobky či prostředí lidi nerozdělovaly, ale naopak spojovaly. V této souvislosti se často hovoří o designu orientovaném na uživatele, vznikají metodiky pro testování kvality různých výrobků z hlediska vhodnosti pro konkrétní skupinu uživatelů, např. (WAI).

Člověka mohou znevýhodňovat tři základní typy psychosomatických problémů, pro které je třeba stále hledat nová řešení. Vedle dobře rozpoznatelných poruch, které ovlivňují mobilitu a senzorický aparát, se stále častěji setkáváme i s poruchami kognitivními, které souvisejí se způsobem, jakým náš mozek zachycuje, uchovává a vybavuje si informace.

### 21.1.1 Technologie a kognitivní poruchy

*Kognitivní poruchy* nepostihují jen paměť, ale podle stupně závažnosti mohou mít řadu různých projevů od mírných kognitivních poruch až po demenci. Mohou ovlivňovat fatické funkce (schopnost formulovat myšlenky do slov a vět a takto komunikovat s okolím), vést k apraxii (poruchám složitých pohybových návyků) či měnit schopnost poznávat lidi či předměty, orientovat se v prostoru, chápat souvislosti atd. Stárnutí je obecně spojeno s poklesem kognitivního výkonu (Konrád, 2005), a proto se v návaznosti na demografický vývoj pozornost asistivních technologií začíná zaměřovat i na řešení problémů souvisejících s kognitivními poruchami.

Vzhledem k tomu, že kognitivní poruchy mají vliv na celkové chování člověka, nestačí na jejich překonání jednoduché řešení rutinního charakteru, ale musí jít o komplexní přístup. Vhodné řešení by mohly reprezentovat například systémy, které budou vystupovat v roli diskrétního společníka, který bude svého klienta nenápadně „hlídat", tedy sledovat, zda uživatel učinil vše, co patří k jeho denní rutině z hlediska zdravotní péče a taktně opomenuté kroky ve vhodný okamžik připomínat. Velmi pravděpodobně by mohl vhodný typ podpory pro realizaci kognitivních aktivit představovat racionální řešení kognitivního přetížení zaviněného přemírou informací, kterými je člověk v současné době průběžně zahlcován a které musí filtrovat. Silné kognitivní přetížení je v současné době charakteristické pro velkou řadu zaměstnání. To je možná i důvod, proč roztržitost už dávno není považována za typickou vlastnost profesorů, ale pomalu je přijímána jako norma.

### 21.1.2 Kompenzační pomůcky

Kompenzační pomůcky mají pomoci jednotlivcům se zdravotním postižením vykonávat každodenní aktivity, pomoci při komunikaci, studiu či práci tak, aby získali vyšší nezávislost a kvalitu života. Tyto pomůcky mohou zlepšit výkon fyzických i mentálních funkcí, překonat zdravotní poruchu, nemoc, preventivně zabránit zhoršování stavu nebo i nahradit ztracenou končetinu. Podpůrné technologie jsou vždy spjaté s konkrétním uži-

vatelem a jeho potřebami. Uveďme některé příklady kompenzačních pomůcek, u nichž je zřejmá vazba na různé metody umělé inteligence.

- **Kognitivní poruchy**. Do této skupiny se řadí vedle již zmíněných poruch paměti a učení, také poruchy řeči. Ovšem porucha řeči může mít i čistě „mechanickou" příčinu, kterou lze obejít použitím vhodného technického řešení. Již existuji elektrické hrtany, které umožňují hlasový výstup pomocí vibračního mechanismu umístěného na krku. Podpůrné komunikační systémy velikosti kalkulačky a napájené monočlánky disponují syntetizovaným hlasovým výstupem sledujícím napsaný text. Tato zařízení používají techniky popsané v příslušných kapitolách 15 a 14 této knihy; (Psutka a kol., 2007), (Hajičová a kol., 1997).
- **Poruchy mobility**. Ty jsou naopak způsobeny například zraněním míchy, mozku či infarktem. Při těchto handicapech jsou elektrické skútry či vozíčky nenahraditelnou pomůckou pro osoby se ztracenou nebo omezenou mobilitou. Vozíčkářům se nabízejí různé doplňkové výbavy, které zvyšují jejich nezávislost a přístup k pracovním i domácím aktivitám. Do této kategorie patří i různí servisní roboti (Šolc, 2007) stejně jako speciální dedikovaná rozhraní pro ovládání počítače, kterým se budeme podrobněji věnovat v dalším textu.
- **Senzorické poruchy**. Jsou jimi především vada zraku či sluchu. Na trhu jsou dostupné například „hovořící" hodinky, teploměr, váha či kalkulačka pomáhající slepým či slabozrakým v jejich každodenních aktivitách. Pomocí ručního skeneru si může člověk se zrakovou vadou uložit určitý text do osobního počítače a specializovaná aplikace mu s použitím zvukového výstupu tuto informaci zprostředkuje. Pomůckou pro slabozraké jsou publikace se zvětšeným písmem, programy na zvětšení textu a grafiky na monitoru nebo zvětšovací kamerové zařízení, popř. i speciální lupy. V současné době se již provádějí pokusy s „umělou sítnicí", která umožní i slepému člověku rozpoznávat tvary okolních předmětů (Dowling, 2005), (Horning, 2005). Pro osoby s poškozeným sluchem se nabízejí světelná oznamovací zařízení, která se připojují k telefonům, budíkům, vchodovým zvonkům atd. Dále jsou k dispozici i miniaturní sluchadla, která jsou upravována jednotlivým pacientům na míru podle jejich konkrétní sluchové ztráty, či různé jiné zesilovací prvky.

### 21.1.3 Může pomoci umělá inteligence?

Naštěstí sny o technologii v roli osobního asistenta nebo chytré kompenzační pomůcky už dnes nemají jen spekulativní charakter, ale stávají se předmětem seriózního vědeckého zkoumání. Výsledkem jsou nové a nové pomůcky, které jsou určeny ke kompenzaci různých fyzických či kognitivních změn a mají pomoci člověku se s nimi vyrovnat. Nejde již jen o pouhá zařízení, která pomáhají pacienta zdvihat a přepravovat, ale vzniká i řada kompenzačních pomůcek, které se chovají „chytře", neboť předvídají potřeby či záměry svých vlastníků díky tomu, že jsou v nich zabudovány různé techniky původně vzniklé jako součást umělé inteligence (Pollack, 2005). Stáváme se svědky významné změny ve způsobu transferu technologií, neboť jejich odběratelem už není jen průmyslová výroba či zdravotnictví. Novým jevem posledních desetiletí je to, že technická zařízení se stávají dokonce i nedílnou součástí člověka jako individua a hovoří se o vzniku kyborgů (K. Warwick v kap. 22). Vedou se i diskuse o tom, jak tento vývoj změní tvář lidské společnosti (Hughes, 2004), (Warwick, 1999).

Moderní zařízení typu kardiostimulátor či kochleární implantát jsou zcela nenápadným vhojeným doplňkem těla svého vlastníka, který mu umožňuje plnohodnotné samostatné působení a výrazně jej obohacuje (Chorost, 2005). Tajemství toho, že tato složitá zařízení správně slouží svému uživateli nejen v laboratorních podmínkách, ale i v běžném dynamicky se měnícím životním prostředí, spočívá v tom, že nabízejí řadu různých režimů, jejichž volbu či úpravu provádí systém samostatně na základě vlastního hodnocení aktuální situace a potřeb svého klienta. Zde se používají techniky rozpoznávání, strojového učení podobně jako u kognitivních pomůcek (Rudary, 2004) i metody (Müller, 2001) pro podporu rozhodování, tedy typické nástroje umělé inteligence.

## 21.2 Technologie v roli osobního asistenta nebo společníka

Jaké jsou úkoly osobního asistenta, ve kterých by mohla technologie pomoci? Pro osoby s nepravidelnými výpadky paměti nebo s poruchami kognitivních funkcí, které nejsou zcela zásadního charakteru, je důležité například to, aby jim někdo včas přátelsky připomněl, co je třeba udělat. Je možné rozlišit tři typy aktivit, které mohou zajistit současné asistivní technologie (Pollack, 2005):

Úlohou *jisticího systému* (assurance s.) je sledovat, zda jeho klient je v pořádku a vykonává standardně své běžné a nutné denní aktivity. Pokud se tak neděje, má ihned informovat zodpovědnou osobu, která může obratem zasáhnout.

*Kompenzační systém* (compensation s.) slouží svému klientovi jako přátelský individualizovaný průvodce denními aktivitami: připomíná, co se má udělat – popř. i jak, pokud na to klient zapomene.

Úkolem *hodnoticího systému* (assesment s.) je průběžně pozorovat chování svého klienta a na základě získaných údajů odhadnout jeho současný stav.

Základním postupem, který musí všechny tři uvedené typy systémů realizovat, je uvažování s použitím rozsáhlých znalostí o prostředí, například na základě nějaké posloupnosti akcí musí systém usoudit, co byl původní záměr jejich klienta a co je důvod toho, že záměru nebylo dosaženo. Jde tedy o kombinaci rozpoznávání aktivit na základě průběžného monitorování akcí klienta, distribuovaného sběru a vyhodnocování dat, plánování a logické úvahy. Nepochybně se zde využívají postupy a metody, se kterými se setkáme v každé učebnici umělé inteligence.

Právě návrhu postupů pro *rozpoznávání aktivit* je nyní věnována značná pozornost. Například systém PROACT (Philipose a kol., 2005) předpokládá, že klient má obléknutou speciální rukavici vybavenou čtečkou RFID. Tato rukavice identifikuje označené předměty v domácnosti, kterých se klient dotýká, a z těchto údajů se snaží rozpoznat běžnou činnost (vaření čaje, čistění zubů, ...), která probíhá, nebo má proběhnout. Vzhledem k tomu, že mezi nutné součásti nějaké činnosti se mohou vmístit i další kroky, které je třeba odfiltrovat, snaží se tento systém pomocí Bayesovy sítě odhadnout pravděpodobnost, s jakou vede získaná posloupnost kroků k jednotlivým možným činnostem.

Velmi ambiciózní projekt tvorby kompenzačního SW systému představuje Autominder (Pollack a kol., 2003), který má za úkol navrhnout denní plán činnosti pro svého klienta (např. diabetika) tak, aby bral v úvahu všechna předem daná omezení (podávání léků a jídla ve správnou dobu, čas oblíbených TV pořadů, ...), a taktně dohlížet na jeho

dodržení prostřednictvím inteligentního generátoru upomínek (IRG – intelligent reminder generator). Pokud rozpozná změnu v posloupnosti plánovaných činností (Philipose, 2004), systém provede náležitou modifikaci tak, aby původní záměr byl splněn. I zde se používají Bayesovy odvozovací techniky pro tvorbu modelu chování klienta, standardní metody práce s omezujícími podmínkami při tvorbě a úpravě plánu a pravidlový systém pro tvorbu IRG. Soubor pravidel pro IRG nemůže zůstat stále stejný, ale musí se přizpůsobovat aktuálnímu psychickému stavu klienta. K tomu používá posilované učení (Rudary a kol., 2004). Při jednom z experimentů byl systémem Autominder vybaven robot Pearl (Pineau, 2003) vyrobený na Carnegie Mellon University. Užitečnost této kombinace se osvědčila i v reálném prostředí domu s pečovatelskou službou (Pollack, 2003).

Osoby s mírnými motorickými obtížemi mají sníženou možnost vycházet, a proto potřebují zajistit různé donáškové služby. To samo o sobě nestačí, neboť snížená hybnost zvyšuje nebezpečí, že i doma dojde k pádu. A na takovou nepředvídatelnou situaci je třeba obratem reagovat (Nussbaum, 2004). Zde by mohl pomoci *jisticí systém.* Pokud však mají pohybové obtíže chronický charakter, je jejich důsledkem přerušení těch existujících sociálních kontaktů, které se odehrávají mimo domov a dotyčný člověk může začít trpět sociální izolací. I pro tento případ je vhodné hledat nějaké technologií podporované řešení. Pro mnoho mladých lidí by bylo řešením dostatečně kvalitní připojení na Internet, to však není ta pravá cesta pro starší populaci, která často nemá chuť se učit zacházet s přístroji, které považuje za složité (Busuoli, 2007). Hledání vhodného řešení je předmětem aktuálního výzkumu, např. Projektu OLDES [5] či (Cook, 2004).

## 21.3 Podpora pro komunikaci člověka s počítačem

Termín komunikace charakterizuje způsob sdělování a výměny informací za použití různých komunikačních médií. V posledních deseti létech je kladena velká váha na komunikaci člověka s technickým systémem. Vstupní informace mohou být počítači předávány některým z vhodných dorozumívacích prostředků – periferií, za kterou můžeme označit například myš, světelné pero, tiskárnu či děrné štítky. Vyvíjejí se speciální podpůrné technologie či kompenzační pomůcky pro ovládání osobního počítače handicapovanými (Fejtová, 2006). V řadě případů stačí mírná úprava používaných periferií (např. speciální myš či trackbally, řádek s Braillovým písmem), jindy je vhodné se opřít o stávající možnosti hlasového vstupu či výstupu. Většina obecně dostupných řešení spoléhá na funkci rukou. Ovšem jen v Evropě je víc než půl milionů lidí, kteří v důsledku úrazu či onemocnění neovládají ruce tak, aby je mohli používat k běžným denním úkonům (COGAIN [1]). Jedná se zejména o motorická postižení, která znemožňují myš samotnou uchopit, natož pak s ní jemně manipulovat.

Při návrhu vhodného řešení pro konkrétního uživatele je třeba vycházet z jeho potřeb a omezení. Představme si člověka, který nemá kognitivní potíže, ale je zcela paralyzován – ztratil možnost ovládat svaly celého těla. Tyto problémy se promítají i do způsobu, jak artikuluje, popř. mu zcela brání mluvit. Může jít o pacienta s ALS (amyotrophic lateral sclerosis) nebo o osobu po úrazu. Můžeme jim nabídnout nějaké řešení jak se dorozumět s okolím? Musíme vycházet z toho, jaké prostředky má takový pacient proto, aby projevil svou vůli. Svaly oka obvykle velmi dlouho odolávají nemoci a schopnost sledovat zrakem předmět zájmu zůstává i u paralyzovaných pacientů jednou z viditelných známek

jejich bdělosti. Pacient bez kognitivních potíží samozřejmě přemýšlí a ani jeho svalová dystrofie nezastaví běh jeho myšlenek. Bylo by možné řídit počítač prostřednictvím myšlenek nebo pohybů oka?

### 21.3.1 Počítač řízený myšlenkou

V posledních deseti létech se v několika výzkumných centrech v Evropě, USA i v Japonsku intenzivně zkoumají reálné možnosti vytvořit přímé neinvazivní rozhraní mezi mozkem uživatele a počítačem (Wolpaw a kol., 2002), označované jako BCI (Brain-Computer Interface). Při tomto typu ovládání počítače prostřednictvím myšlenky (Dornhege a kol., 2003) se jako řídicí signál používá EEG, které se snímá klasickým způsobem prostřednictvím elektrod připevněných k hlavě uživatele. Podmínkou pro úspěch tohoto přístupu je dosažení schopnosti jednoznačně rozlišit prostřednictvím signálu EEG alespoň dva mentální stavy uživatele, jejichž navození může uživatel ovládat svou vůlí. Tento úkol může být velice časově náročný a v mnoha případech předpokládá, že se člověk – uživatel systému – sám musí naučit navodit ty stavy, které je používaný systém BCI schopný rozpoznat. Vzájemné přizpůsobování mezi uživatelem a systémem BCI pak trvá i řadu týdnů. Jedním z hlavních problémů zde je vysoká interpersonální variabilita snímaných signálů EEG, které mají výrazně individuální charakter.

Velmi zajímavou aplikaci metod umělé inteligence představuje vývoj „Berlínského BCI" (Blankerz a kol., 2007), jehož motto zní „ať se stroj učí sám". Uživatel je nejprve instruován, aby si intenzivně interpretoval jako nějaké dva typy konkrétních pohybů, například napřažení pravé a levé ruky, které budou představovat řídicí signál pro počítač. Odpovídající snímané signály EEG jsou prezentovány jako pozitivní příklady dvou cílových tříd, zatímco ostatní situace jsou chápany jako negativní příklady. Počítač se z takto získaných příkladů sám naučí rozpoznávat pokyny uživatele za pomoci nejmodernějších metod z oblasti strojového učení (Müller a kol., 2003). Tímto způsobem se výrazně zkrátí doba potřebná proto, aby se uživatel naučil systém BCI ovládat. Díky strojovému učení se tato doba redukuje z týdnů na hodiny.

O robustnosti navrženého řešení se mohli přesvědčit návštěvníci výstavy CeBIT 2006, kde tvůrci předváděli, jak lze jeho prostřednictvím vytvářet text s využitím „mentálního psacího stroje", který je obdobou virtuální SW klávesnice popsané v dalším odstavci. Uživatelé dosáhli rychlosti až 7,6 písmen za minutu. Velmi zajímavé jsou i experimenty, při nichž se využívá rozhraní BBI pro rehabilitaci úchopu u kvadruplegického pacienta (Müller a kol., 2003).

### 21.3.2 Počítač řízený pohybem oka

Využití oka jako řídicího elementu pro rozhraní k počítači člověka láká už dávno: první pokusy s tvorbou textu, při nichž bylo použito oko jako aktuátor, se objevily už před čtvrt stoletím (Majaranta, Raiha, 2002). Má-li být pohyb oka použit jako řídicí prvek, je nutné mít k dispozici kvalitní metody snímání očních pohybů. Jejich aktuální přehled poskytuje například článek (Fejtová, 2006). Uveďme jako příklad dva charakteristické systémy a jejich vlastnosti.

Na Boston College v projektu nazvaném EagleEyes [3] využívají elektrookulografii (EOG) k přesouvání polohy kurzoru na monitoru počítače. Uživateli jsou nalepeny na obličej elektrody, z jejichž signálu se vypočítává poloha oka a uživatel může pomocí pohybů očí ovlivňovat pohyb kurzoru ve speciálních aplikacích.

Druhým příkladem je projekt Dasher [4], který probíhá na univerzitě v Cambridgi. Je zaměřen na vývoj speciálního nástroje, pomocí kterého mohou handicapovaní uživatelé psát text. Potřebný EOG signál není měřen prostřednictvím dotykových elektrod, ale infračervenou kamerou, která detekuje polohu oka na základě kontrastu mezi světlem pohlceným zorničkou a světlem odraženým od zbytku oka. Zdroj infračerveného světla představují dva infračervené zářiče, které jsou umístěny po stranách monitoru počítače, zatímco infračervená kamera je přichycena uprostřed spodního okraje monitoru. O dalších zajímavých řešeních se lze dozvědět na stránkách projektu COGAIN [2]

Obecně se sledování polohy oka kamerou označuje jako *videookulografie* (VOG). Pokud je kamera umístěna samostatně, je třeba udržet oko v jejím zorném poli a zároveň odlišit rotaci oka od translačních pohybů hlavy. Pokud je kamera upevněna přímo na hlavě subjektu, je měřena relativní poloha očí vůči hlavě a k určení absolutního směru pohledu je třeba sledovat i polohu hlavy. Sledovací algoritmy jsou v současné době založeny na digitálním zpracování obrazu (Pajdla, 2007). Je dosahováno rozlišení desetin stupně a vzorkovací frekvence až stovek hertzů.

### *21.3.2.1 Princip systému I4Control®*

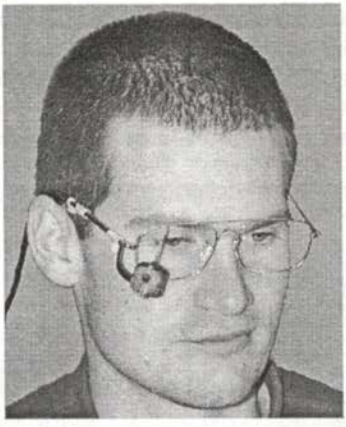
Systém I4Control® byl primárně navržen a vytvořen jako kompenzační pomůcka určená pro handicapované počítačové uživatele. Byl zkonstruován jako nový druh počítačové periferie tak, aby fyzicky znevýhodnění uživatelé mohli plně ovládat svůj počítač pomocí pohybů očí, případně hlavy (zde je obratně využíván okulocefalický reflex).

Základem zařízení je malá kamera připevněná k brýlové obroučce snímající z bezprostřední blízkosti obraz uživatelova oka (obr. 21.3, též barev. příl.), který je následně vyhodnocován s cílem zjistit směr výchylky zorničky[1] od původní rovnovážné polohy. Díky tomuto umístění získává systém I4Control® tyto výhody:

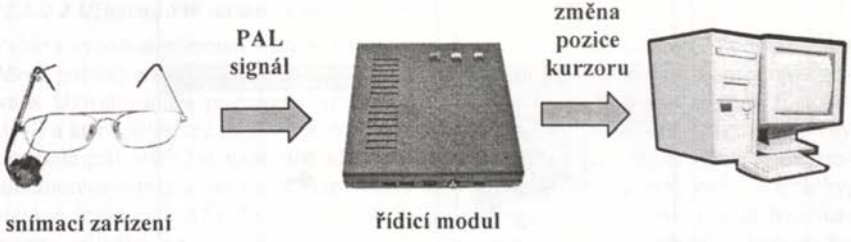
- nízké nároky na technické parametry kamery, což znamená podstatné snížení celkových nákladů;
- snížení nároků na výpočetní algoritmy (kamera snímá pouze oko a není nutná implementace algoritmů na rozpoznání oka ve složité scéně) – důsledkem je další pokles nákladů;
- uživatel může volně hýbat hlavou (není nutná žádná její fixace);
- uživatel není vázán pouze ke konkrétnímu místu – může se v případě potřeby volně pohybovat.

Aby bylo zařízení nezávislé na okolních světelných podmínkách, je kamera doplněna o infračervené osvětlení (pro lidské oko neviditelné) snímané scény, které je automaticky řízeno podle okolních světelných podmínek. Výstupem kamery je běžný analogový signál PAL, který je přiveden do externí řídicí jednotky. Tam je videosignál předzpracován a je v něm detekována pozice zorničky, která je prostřednictvím standardního rozhraní USB zasílána do běžného osobního počítače (obr. 21.4). Směr pohybu zorničky pak určuje směr pohybu počítačového kurzoru. Uživatel si může pohyb očí usnadnit, pokud hýbá hlavou, neboť při otočení hlavy oči „automaticky" putují na opačnou stranu.

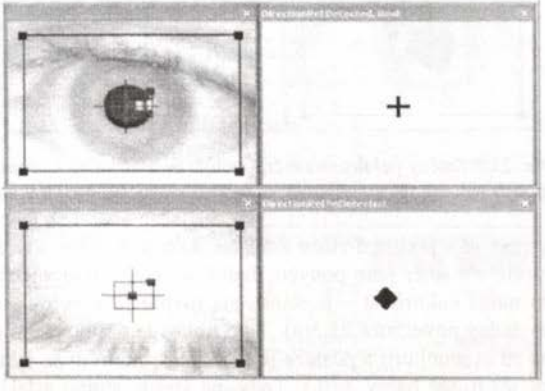
[1] Zornička, nebo-li také panenka, pupila či zřítelnice, je otvor v duhovce, který soustřeďuje světlo na sítnici a prostřednictvím zornice vniká světlo odražené od pozorovaného předmětu do oka. Zornice se může zvětšit nebo zmenšit a tak reguluje množství procházejícího světla.



Obr. 21.3 Systém I4Control®.

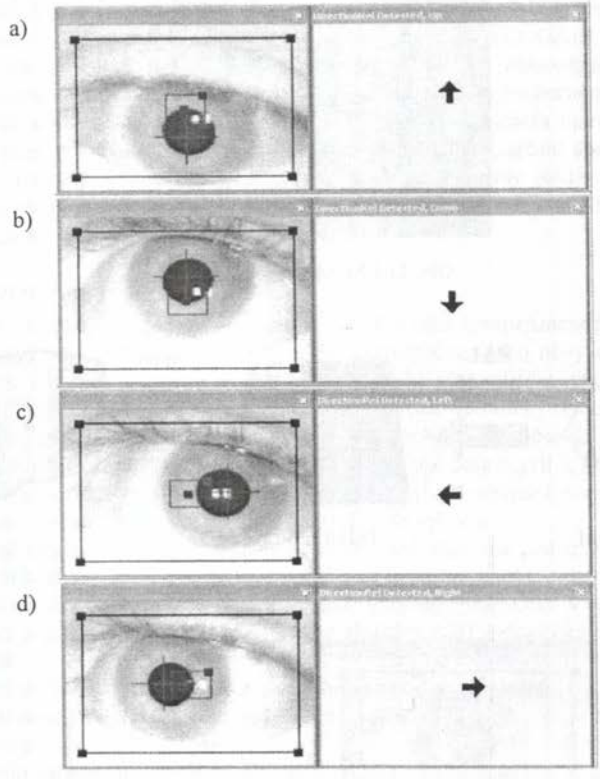



Obr. 21.4. Základní blokové schéma systému I4Control®.



Obr. 21.5 Definice akcí.
(a) systém nevysílá žádný řídicí signál (b) vyslání signálu (např. klik).

Systém I4Control® ve své podstatě emuluje (nahrazuje) počítačovou myš. Motoricky handicapovaným uživatelům tak umožňuje ovládat jakoukoliv aplikaci, která je řízena myší. Klik (stisk tlačítka myši) či dvojitý klik je nahrazen mrknutím, obr. 21.5(b), též barev. příl., tj. zavřením oka na jednu sekundu pro klik a dvě sekundy pro double klik (časovou konstantu je možné nastavit podle konkrétních potřeb uživatele).




Obr. 21.6 Změny polohy oka určují pohyb počítačového kurzoru:
(a) pohyb nahoru, (b) pohyb dolů, (c) pohyb vlevo, (d) pohyb vpravo.

Počítačový kurzor je v podstatě řízen změnou polohy zornice, která neurčuje absolutní polohu kurzoru, ale směr jeho pohybu. Jedná se tedy o inkrementální metodu. Na začátku je systém nutné kalibrovat – je stanovena rovnovážná poloha určující zónu, ve které není vysílán žádný povel, obr 21.5(a). Tato oblast je definována nejlépe při uživatelově pohledu na střed monitoru a posléze je směr pohybu určován jako směr výchylky z této pozice (obr. 21.6, též barev. příl.). Tedy, na kterou stranu a jak dlouho se bude kurzor pohybovat, určuje směr a doba výchylky z této rovnovážné polohy. Jakmile se detekovaná zornice vrátí zpět do rovnovážné polohy, přestane systém I4Control® vysílat

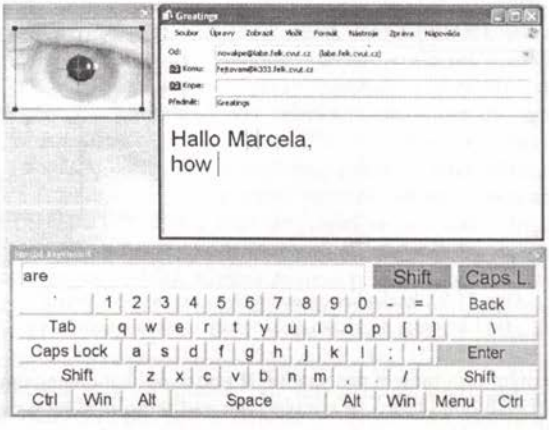
povely pro pohyb. Rovnovážná pozice je určena zeleným čtvercem. Naopak modrý obdélník vymezuje oblast zpracovávaného videa. Nastavitelnost této oblasti umožní nejen rychlejší a tedy i přesnější zpracování obrazu z kamery, ale vyřeší také problémy týkající se odlišného umístění očí vůči nosu u různých uživatelů. Kalibrace se pro každého uživatele provádí pouze jednou a její hodnoty jsou uloženy do zařízení pro další použití. Po kalibraci se tedy pro příští práci se systémem výchozí fáze „zúží" pouze na nasazení brýlí, spuštění systému a ovládání běžného osobního počítače.

**Vlastnosti systému I4Control®**. Mezi základní vlastnosti systému I4Control® patří:

- univerzálnost – snadná napojitelnost na řízené zařízení,
- mobilnost – zařízení je malé, lehké, kompaktní a snadno přenositelné,
- snadné ovládání – použití systému je intuitivní,
- jednoduchý způsob instalace – kalibraci systému je nutné provést pouze při prvním použití zařízení,
- žádné omezení v pohybu – uživatel se může pohybovat bez jakéhokoli omezení,
- nízká cena.

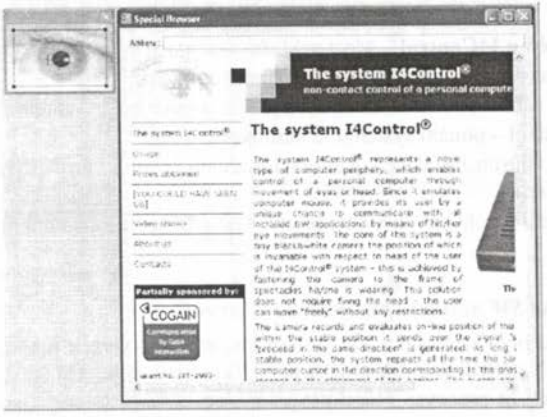
### *21.3.2.2 Užitečné SW nástavby pro systém I4Control®*

Velkou výhodou plynoucí z faktu, že systém I4Control® navenek funguje podobně jako běžná počítačová myš, je, že pro práci s počítačem není zapotřebí žádný speciální software. Uživatel může pracovat s těmi programy, na které je zvyklý (jsou-li ovladatelné myší) a které potřebuje ke své činnosti. Vnímavého čtenáře jistě napadne otázka, jak tedy uživatelé píší text? Jak například sdělují své myšlenky či pocity svým kamarádům prostřednictvím e-mailu, jak napíší dopis nebo do svého prohlížeče adresu webové stránky, která je právě zajímá? Odpověď je zdánlivě jednoduchá. Pro psaní textu může handicapovaný uživatel použít softwarovou klávesnici, která je součástí běžného operačního systému (např. Microsoft Windows od verze 2000). Je to obyčejný počítačový program,

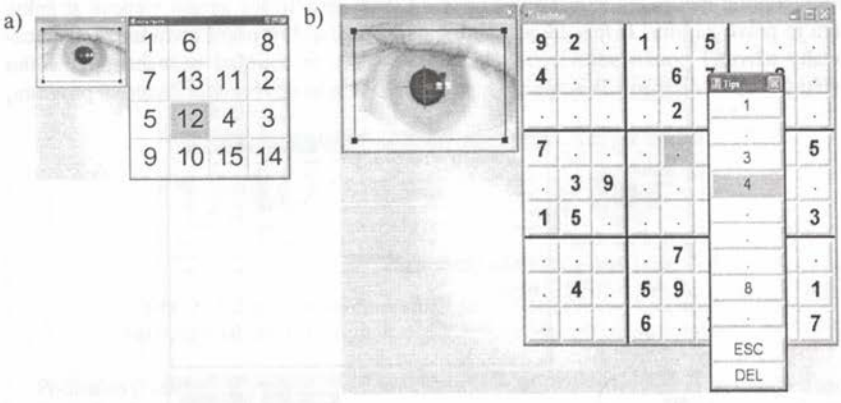



Obr. 21.7 Psaní textu se systémem I4Control®.

který vypadá jako normální klávesnice, ale jednotlivá tlačítka reprezentující písmenka či jiné klávesy se aktivují ne stiskem prstu, ale klepnutím levým tlačítkem myši. Tato aplikace je ovšem pro začínajícího uživatele náročná z důvodu nutnosti jemné manipulace s počítačovým kurzorem. Náš tým tedy vyvinul speciální klávesnici (obr. 21.7), kdy uživatel přímo vybírá celé tlačítko. Tento nástroj je ještě doplněn i o predikci textu, což umožní snížení počtu použitých kláves k napsání daného textu.




Obr. 21.8 Nadstavba nad webový prohlížeč.




Obr. 21.9 Ukázka her upravených pro systém I4Control:
(a) Lloydova patnáctka, (b) logický hlavolam sudoku.

Dalším upraveným programem pro snadnější používání handicapovanými uživateli je Microsoft Internet Explorer. Jedná se o nadstavbu nad tento rozšířený webový prohlížeč (obr. 21.8, též barev. příl.), která automaticky projde požadovanou webovou stránku a vyhledá všechny dostupné odkazy. Aktivní odkaz je potom zvýrazněn změnou velkých

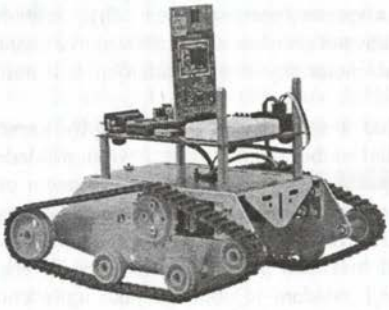
a malých písmen v textu odkazu. Přepínání mezi jednotlivými odkazy je prováděno pohledem vlevo (posun o jeden odkaz zpět) a vpravo (posun o jeden odkaz dopředu). Naopak pohybem oka nahoru a dolů se může uživatel pohybovat po celé webové stránce, tzv. skrolovat. Tato úprava umožní i velmi motoricky znevýhodněným uživatelům získávat informace na Internetu.

Aby se uživatelé naučili jednoduše, rychle a spolehlivě ovládat pohyby kurzoru zábavnou formou, vytvořil náš kolektiv speciální soubor několika her. Prvním příkladem je tzv. Lloydova patnáctka (obr. 21.9(a)), jejímž cílem je seřadit čísla přesunem přes volné políčko. Ovládání je jednoduché. Pohledem vybereme číslo, které chceme posunout. Přesun čísla do vedlejší pozice zajistí zavření oka reprezentující klik. V naší kolekci nechybí v současnosti nejpopulárnější hlavolam světa. Sudoku (obr. 21.9(b)), jako logická hra, zaujala mnoho uživatelů. Její ovládání je obdobné jako u předchozí patnáctky. Zavřením oka hráč vyvolá dialog, ze kterého lze vybírat možná čísla nebo již vložené číslo vymazat.

### *21.3.2.3 I4Control® a ovládání prostředí a jeho předmětů*

Pro ukázku použitelnosti principu zařízení v průmyslu jsme řízení počítačového kurzoru nahradili ovládáním i jiných zařízení. Našeho prvního okem (popř. hlavou) řízeného robota jsme sestavili ze stavebnice LEGO®. Robot Gertík (obr. 21.10, též barev. příl.) je zcela ovladatelný buď pohyby oka, nebo hlavy. Dívá-li se řidič nahoru, robot jede vpřed, při pohledu dolů, robot couvá. Uživatel ovládá směr pohybu robota stejným způsobem – dívá-li se doleva, robot se otáčí vlevo, dívá-li se doprava, otáčí se vpravo. Odezvou na zavření oka je otočení radarové věže. Robot se zastaví, jakmile se uživatel podívá přímo vpřed (detekovaná zornička se vrátí zpět do rovnovážné polohy). Okem může tedy operátor ovládat minimálně 5 nezávislých kanálů, kterým budou přiřazeny různé polohy oka detekovatelné našim zařízením (vlevo, vpravo, nahoru, dolů, zavření oka). Počet ovládaných kanálů může být dále zvýšen například zavedením vhodných časových konstant (tj. pokud je uživatelovo oko zavřené na 3 sekundy spustí se ...). Další možností je použití některého z pěti základních kanálů pro přepínání funkcionality ostatních či pro výběr odpovídajících příkazů z dodatečného „menu". Pokud bude ovládaný systém řízen prostřednictvím počítače nebo bude obsahovat vizuální vstup pro operátora (např. obrazovku nebo displej) lze aplikaci doplnit o prvky využívané pro rychlé ovládání programů handicapovanými uživateli. Například, aby nebyl uživatel nucen ovládat pohyb kurzoru ve vysokém rozlišení obrazovky, je řízení doplněno o virtuální rastr překrývající původní plochu obrazovky a vlastní pohyb je uskutečňován pouze výběrem oblasti z tohoto rastru buď inkrementálně (přírůstkově od předchozí vybrané oblasti), nebo absolutně (pohledem na konkrétní oblast). Příslušnou akci skrývající se pod takto virtuálně vytvořenou oblastí lze aktivovat buď přímým podnětem (např. zavření oka na stanovený časový okamžik), nebo automaticky (oblast zůstane zvolena po určitou dobu – tzv. dwell-time).

Jako další technické zařízení ovládané pomocí pohybů očí jsme zvolili jednoduchý manipulátor (obr. 21.11, též barev. příl.), který jsme tentokrát nazvali Robík. Způsob ovládání je totožný jako v předchozích ukázkách. To jest otáčení základny ovládá operátor pohybem oka vlevo a vpravo (řídí servomechanismus pro pohyb základny). Stejný princip je využíván i pro pohyb ramena (pohybem oka nahoru a dolů). Poslední servomechanismus pro otevírání a uzavírání kleštiny je řízen uzavřením (popř. otevřením) oka.



Obr. 21.10 Robot Gertík.

Obr. 21.11 Robot Robík.

### *21.3.2.4 Problémy s ovládáním invalidního vozíku pohledem*

Úspěch I4Control® při řízení hračkového vozítka Gertík přirozeně láká k pokusům využít podobný postup i ve spojení s kolečkovým křeslem. Tento nápad se zdá být na první pohled poměrně jednoduchý, ale tak je tomu jen z hlediska navigace. To samo o sobě nestačí, neboť je třeba vzít v úvahu i hledisko bezpečnosti samotného uživatele křesla. Vždyť pokud řídicí pokyny pro ovládání křesla zprostředkovává pohled či oko, může dojít k řadě „nedorozumění" vzniklých nevhodnou interpretací některých signálů způsobených samovolným pohybem oka. Například je nutné počítat s tím, že se uživatel:

- náhle, možná neúmyslně, podívá jinam (například ve směru nenadálého zvukového podnětu),
- dostane do špatně osvětleného prostředí (kdy systému chybí kontrolní signál zprostředkovaný okem),
- setká s něčím (prach, ostré světlo, ...), co ho podráždí natolik, že mimoděk zavře oči.

Za těchto podmínek je zřejmé, že kolečkové křeslo nemůže být ovládáno jen uživatelem samotným. Je možné požadovat pro každý pokyn ke změně směru cesty zprostředkovaný okem ještě i potvrzení z nějakého nezávislého zdroje, např. s využitím signálu EEG (Tanaka, 2005). Jiným řešením je vybavit křeslo jistým stupněm autonomie a dát uživateli na výběr prostřednictvím očních pohybů jen omezenou nabídku zásahů do plánu samostatné jízdy vozíku. Lze si představit řadu různých kombinací těchto alternativ, které se budou lišit mírou autonomie poskytnuté křeslu. Při řešení bude nutné vždy plně respektovat individuální potřeby a schopnosti uživatele vozíku. Pro konstrukci a konfiguraci tak individualizovaných řešení se jistě neobejdeme bez využití nástrojů umělé inteligence.

## 21.4 Závěr

Vývoj asistivních technologií představuje velmi zajímavou interdisciplinární oblast, která nabízí stimulující prostředí pro společné uplatnění rozličných výsledků získaných v oboru umělé inteligence. Klíčové pro rozvoj pomůcek při kognitivních poruchách jsou výsledky týkající se problematiky plánování či přeplánování pro jednoho i více agentů,

uvažování s neúplnou informací, distribuovaný sběr dat, rozpoznávání i dobývání znalostí (Choudhury, 2006). Při návrhu kompenzačních pomůcek se již dnes uplatňují metody strojového učení i vnímání, zpracování přirozeného jazyka i robotiky. O zájmu o tuto aplikační problematiku svědčí například to, že v rámci konference AAAI 2002 bylo zorganizováno samostatné setkání *Workshop on Automation as Caregiver: The Role of Intelligent Technology in Elder Care* a tématu byla věnována zvaná přednáška Marthy Pollackové na IJCAI 2007. Čerstvé informace o novinkách v této oblasti lze nalézt na webových stránkách CAST [6].

### Literatura


Blankertz B., Dornhege G., Krauledat M., Curie G., Müller K.-R.: *The non-invaseve Berlin Brai-Computer Interface: Fast acquisition of effective performace in untrained subject.* NeuroImage, Elsevier Inc., 2007.

Busuoli M. a kol.: *Entertainment and Ambient: a new OLDES' view.* The HCI Internetional 2007 Conference Proceedings, LNCS 4564, Springer, 2007.

Cook A., Hussey S.: *Assistive Technologies - Principles and Practice.* 2nd Edition. Mosby 2002.

Cook D., Das S.: *Smart Environments: Technology, Protocols and Application.* Wiley-Interscience, listopad, 2004.

Dornhege a kol.: *Combining features for BCI, in Advances in Neural.* Inf. Proc. systems 9NIPS 02), Becker S., Thrun S., Obermayer K. (eds), vol. 15, 2003, s. 1115–1122.

Dowling J.: *Artificial human vision.* Expert Review of Medical Devices. 2, 2005, s. 73–85.

Hajičová E.: *Zpracování přirozeného jazyka.* Umělá inteligence (2), Academia, Praha 1997, s. 244–256.

Hajičová a kol.: *Jazyk a umělá inteligence. Kudy a kam.* Umělá inteligence (5), Academia, Praha 2007, s. 272–283.

Hebert a kol.: *Alzheimer disease in the U.S. population.* Prevalence estimates using the 2000 census. Archives of Neurology 60 (8), s.1119–1122

Hornig R. a kol.: *A Method and technical Equipment for an Acute Human Trial to Evaluate Retinal Implant Technology.* Journal of Neural Engineering, 2(1), 2005, s. 129–134.

Hughes J.: *CITIZEN CYBORG:Why Democratic Societies Must Respond to the Redesig-ned Human of the Future.* Westview Press, 2004.

Chorost M.: *Rebuilt: How Becoming Part Computer Made Me More Human.* Houghton Mifflin Company, 2005.

Choudhury T., Philipose M., Wyatt D., Lester J.: *Towards activity Databases: Using sensors and Statistical Models to Summarize People's Lives.* In IEEE Data Engineering Bulletin, březen, 2006.

Fejtová M., Fejt J.: *Oko jako počítačová periferie. Lékař a technika.* 2006, roč. 36, č. 1, s. 45–50.

Konrád J.: *Kognitivní poruchy ve stáří.* Psychiatrie pro praxi 5/2005, s. 229–232.

Majaranta P., Raiha K.-J.: *Twenty years of eye typing: systems and design issues.* In Proceedings of ETRA '02, ACM Press, Cambridge, s. 15–22.

Müller G. R., Pfurtscheller J., Gerner H. J., Rupp R.: *"Thought" – control of functional electrical stimulation to restore hand grasp in a patient with tetraplegia.* Neurosci Lett, vol. 351, 2003, s. 33–36.

Müller G. R. a kol.: *An introduction to kernel-based learning algorithms.* IEEE Neural Net-works, vol.12, no. 2. květen 2001, s. 181–201.

Nussbaum G., Miesenberger K.: *The Assistive Home – More than Just Another Approach to Independent Living?* Lecture Notes in Computer Science, Volume 3118, Springer-Verlag Berlín, Heidleberg, 2004, s. 891–897.

Pajdla T.: *Geometrie v počítačovém vidění – principy a aplikace rekonstrukce prostorové scény z obrazů.* Umělá inteligence (5), Academia, Praha 2007, s. 328–347.
Pineau J. a kol.: *Towards Robotic Assistants in Nursing Homes: Challenges and Results.* Robotics and Autonomous Systems, 42, 2004, s. 50–57.
Philipose M. a kol.: Inferring ADLs (Activities of Daily Living) from interactions with objects. IEEE Pervasive Computing, Vol. 3, no. 4, 2004, s. 50–57.
Pollack M. E. a kol.: *Autominder: An intellgent cognitive orthotics system for people with mamory impairment.* Robotics and autonomous Systems. 44, 2003, s. 273–282.
Pollack M. E.: *Intellgent Technolgy for an Aging Population: The Use of AI to Assist Elders with Cognitive Impairment.* AI Magazine 26 (2), 2005, s. 9–24.
Pollack M. E.: *„Autominder: A Case Study of Assistive Technology for Elders with Cognitive Impairment".* Generations: The Journal of the American Society on Aging, 30 (2), 2006, s.67–69.
Psutka a kol.: Porozumění mluvené řeči, Umělá inteligence (5), Academia, Praha 2007, s. 284–327.
Rudary M., Singh S. B., Pollack M.: *Adaptive cognitive orthotics: Combining reinforcement learning and constraint-base temporal reasoning.* Proceedings of the 21st Int. Conference on Nachine Learning. 2004, s. 719–726.
Šolc F. a kol.: *Servisní roboti.* Umělá inteligence (5), Academia, Praha 2007, s. 405–430.
Tanaka K., Matsunaga K., Wang H. O.: *Electroencephalogram – Based Control of an Electric Wheelchair.* IEEE Transactions on Robotics, Vol. 21, No. 4, IEEE, 2005, s. 762–766.
Warwick K.: *Úsvit robotů, soumrak lidstva.* Vesmír, Praha 1999.
Warwick K.: *Evoluce – poslední hra pro lidstvo.* Umělá inteligence (5), Academia, Praha 2007, s. 515–522.
Wolpaw J. R. a kol.: *Brain-computer interfaces fro communication and control.* Clin. Neurophysiology, vol.13, 2002, s. 767–791.

[1] COGAIN Brochure, 2006. http://www.cogain.org/downloads/
[2] http://www.cogain.org/eyetrackers/
[3] EagleEyes Project, www.eagleeyes.org
[4] The Dasher Project, http://www.inference.phy.cam.ac.uk/dasher/
[5] Project OLDES – Older People's e-Services at Home, www.oldes.eu
[6] Centrum for Aging Services Technology, www.agingtech.org
[7] Web Accessibility Initiative (WAI), www.w3.org/WAI

# 22 Evoluce: Poslední hra pro lidstvo

## 22.1 Úvod

Až doposud procházelo lidstvo procesem evoluce na Zemi jako dominantní druh. Zřejmě je také jediným živočišným druhem, který aktivně zkoumá a je též schopen obývat všechny oblasti naší planety – což se určitě týká všech šesti kontinentů. Ale evoluce se nezadržitelně žene kupředu. Nyní je druh *Homo sapiens* dominantním druhem. Může však nastat čas, kdy toto místo zaujme jiný druh nebo jiná rasa, které se podaří v současnosti dominantní druh překonat.

Vzhledem k milionům let trvajícímu vývoji naší Země by bylo skutečně pošetilé se domnívat, že by se evoluce mohla náhle zastavit. Proto je tedy namístě položit si otázku, co vlastně dovedlo lidstvo na současnou dominantní pozici, kterou jistě nemáme zaručenu žádnou úmluvou. Evoluce rozhodně neuznává žádné smlouvy. Vyznává pouze kult nejsilnějšího jedince nebo jedince, který si našel nejvhodnější prostředí k životu. Nicméně vyznává i jedince, kteří si jakýmkoliv způsobem zajistili vítězství v nelítostném souboji, ať již čestným, nebo nečestným způsobem.

Lidé disponují určitými fyzickými schopnostmi. Pokud však srovnáme rychlost, sílu a atletické dovednosti lidí s některými druhy hmyzu, lidé zřetelně zaostávají. A to ani nemusíme srovnávat ta zvířata, která nám jsou geneticky bližší. V porovnání s mnohými dalšími druhy máme v těchto disciplínách obrovské nedostatky. Neumíme létat, nedokážeme plavat v hlubinách oceánu. Pokud však budeme srovnávat inteligenci, vítězíme. Díky naší inteligenci (obecně) jsme dokázali prozkoumat a obývat skoro každý kout naší Země. Dokázali jsme se dokonce dostat i na jiné planety.

Pokud je tedy inteligence klíčovou vlastností, měli bychom se na ni více zaměřit. Položme si například otázku, jaká je pravděpodobnost, že by se u nějakého druhu mohla vyvinout inteligence, která by předčila lidskou? A pokud by k tomu došlo, jak dlouho by trvalo, než by se druh *Homo sapiens* stal nějakým poddruhem, nebo dokonce vymizel, jako tomu bylo v případě neandrtálského člověka?

Je velmi obtížné, ne však nemožné, si představit, že by jiná biologická forma života na Zemi předčila inteligenci druhu *Homo sapiens*. Jedná se však o krajní, velmi nepravděpodobný případ. Musel by proti nám zasáhnout nějaký druh katastrofy (například globální oteplování) nebo radikální změna životních podmínek. Pokud by se však na dějišti objevil zcela nový, nám podobný druh, velké změny by na sebe jistě nenechaly dlouho čekat.

Pokud se nám protivník podobá, ale není zas až tak schopný, sotva nás může ohrozit. Situace se však stává vážnou, pokud je protivník odlišný a má schopnosti, které jsou

mimo naše chápání a kterými může v případném střetu soupeře snadno překonat. Přesně taková situace by mohla nastat v případě strojové inteligence.

Slýcháme spoustu neuvěřitelných a patetických humanocentrických argumentů (Penrose, 1994), které podstrkují rádoby významní filosofové, aby „dokázali", jak je strojová inteligence primitivní ve srovnání s tou lidskou. A jak je nemožné, aby stroje měly nějaké vědomí, natož vědomí, jaké mají lidé. Z toho pak vyvozují (Penrose, 1994) (netuším, kde se tato úvaha vzala), že stroje budou navždy našimi otroky.

Příklady takových úvah lze nalézt v některých interpretacích tak známých problémů, jakými jsou například Turingův test nebo problém čínského pokoje (Searle, 1997), (Preston, Bishop, 2004). Ve skutečnosti nejsou názory těchto poklidně stárnoucích pánů ničím víc než dětskou ukolébavkou pro ty, kteří se navzdory realitě přejí cítit bezpečně, nebo pro ty, kteří věří ve šťastné konce v hollywoodském stylu (Preston, Bishop, 2004).

Vezměme si například následující úvahu: Konstrukce ptačího těla umožňuje ptákům létat. Lidské tělo však má odlišnou konstrukci. Můžeme tedy učinit závěr, že člověk nebude nikdy létat? V dnešní době zní takový závěr poněkud hloupě – a přece přesně k takové situaci kdysi došlo. Lord Kelvin v roce 1895 prohlásil, že stroje těžší vzduchu nikdy nemohou létat. O pět let později Simon Newcomb, uznávaný americký astronom, tento výrok matematicky dokázal. Přesto se pan Orville Wright v roce 1903 vznesl do vzduchu u města Kitty Hawk v Severní Karolíně (Heppenheimer, 2003). Je však nepochopitelné, že Newcomb a jeho následovníci trvali na svých názorech ještě několik dalších let!

Musí nám být jasné, že pokud budou mít stroje v budoucnu určitou formu inteligence, která předčí lidskou inteligenci, budeme jistě čelit závažnému problému: Kdo bude činit závažná rozhodnutí týkající se budoucího života na Zemi? (Warwick, 1997), (Joy, 2000). Dokud však nebudeme realisticky schopni určit, co to inteligence opravdu je (Warwick, 2001), nebudeme schopni rozpoznat, kdy takový čas nastane. Již samotné kulturní rozdíly mezi lidmi způsobují, že jakékoli srovnání inteligence jedinců a skupin je do značné míry subjektivní. V případě strojů je důležitá jejich výsledná inteligence jako celek a její dopad na lidstvo. Taková inteligence však závisí na tom, jakým způsobem stroje chápou a vnímají okolní svět.

Faktem je, že přizpůsobivost a potenciální síla strojové inteligence daleko přesahuje lidskou (Warwick, 2001). Ovšem pouze v oblasti výpočtů a ukládání dat. Vzhledem k lidské touze po pokroku není možné tento rozvoj zastavit. Musíme tedy hledat způsob, jak omezit pravomoci strojů. Určitě je zde možnost rozšíření lidské inteligence tak, že ji přímo propojíme se strojovou inteligencí. Pokud zůstanou člověk a stroj odděleni, každý z nich bude čelit svým vlastním problémům. Pokud však vytvoříme partnerství a propojíme je dohromady, můžeme využít z obou to nejlepší. Ano, hovoříme zde o technologickém vývoji lidstva pomocí přímého propojení mozku člověka a stroje (pokud to tak lze nazvat). Tento proces bývá označován jako vytváření Kyborgů (Cyborgs) (Clynes, Kline, 1960).

Kyborg je kybernetickým organismem, zčásti člověk, zčásti stroj, který je vytvořen přímým propojením člověka a techniky. Pod tím si však lze představit příliš mnoho – počínaje člověkem, který nosí brýle až k člověku s umělou kyčlí. V této oblasti je největší pozornost věnována nejvyšším formám této technologie – lidem, kteří mají svůj mozek a nervový systém přímo propojen s počítačem.

## 22.2 Vytváření rozhraní k nervovému systému

Od šedesátých let dvacátého století jsou zkoumány možnosti vytvoření elektronického rozhraní k nervovému systému člověka. Nedávné práce (Warwick a kol., 2003) v tomto oboru otevřely možnost technologiím neurální protetiky, které nejen obnovují, ale dokáží navíc vylepšit lidské schopnosti (například přidáním dalších smyslů nebo umožněním komunikace pomocí myšlenky). Výzkum byl původně motivován snahou zmírnit následky poškození míchy nebo amputace, podrobněji viz kap. 21.

Jedním ze způsobů interakce s lidským nervovým systémem je přímé měření neurální aktivity pomocí elektrod, které jsou implantovány do lidského mozku (Donoghue, 2002). Pomocí takovýchto invazivních metod je možné sledovat určité mozkové aktivity. Zmíněnou technologii je možné použít u pacientů s tzv. locked-in syndromem (kteří jsou při plném vědomí, kognitivně nejsou nijak narušeni, nemohou se však pohybovat ani hovořit) (Kennedy, 2000) a umožnit jim například ovládání kurzoru počítače pouhou myšlenkou.

Velice slibnou se ukazuje technologie vytváření rozhraní k periferním nervovému systému, protože v mnoha případech nervového poškození se pouze přeruší přenos elektrických signálů ze senzorických receptorů do mozku (což způsobí narušení vnímání) nebo z mozku do motorických jednotek (což zapříčiní ztrátu pohybové funkce). Smyslové vnímání je možné obnovit přemostěním takového přerušení propojením jiných nervových vláken pomocí polí elektrod MEA (MicroElectrode Array). Pole MEA umožňují také externě ovládat pohyb končetiny (Yu a kol., 2001), (Poboroniuc, 2002), a to pomocí řízení v uzavřené smyčce (Warwick a kol., 2003).

Mnohem zajímavěji začnou znít úvahy, pokud namísto o terapii začneme uvažovat o rozšířeních člověka (Warwick, 2003). Lidé mají velmi omezené mentální schopnosti (pouze trojrozměrné vnímání, pouhých pět smyslů). A propojení nervového systému člověka s počítačem nabízí mnoho nových možností. V první řadě se nabízí vylepšení matematických schopností a rozšíření paměti. Napadnou nás však jistě další možnosti, které by byly vhodnými argumenty pro vytvoření takových Kyborgů: vícerozměrné uvažování, rozšířené možnosti vnímání a vylepšená komunikace (prostřednictvím myšlenkových signálů samotných).

A jak daleko již lidstvo pokročilo na této cestě? Dovolte mi, abych se s vámi podělil o výsledky svých vlastních experimentů.

## 22.3 Pokus s implantátem v roce 1998

Koncem roku 1998 bylo do mé levé paže chirurgickým zákrokem voperováno zařízení RFID (Radio Frequency ID). Ze zdravotního hlediska to nebylo nijak nutné, spíše jsem chtěl vyzkoušet dostupné možnosti. V době experimentu byly k dispozici čipové karty stejné technologie, ale ještě nikomu nebylo toto zařízení voperováno.

Budova Katedry kybernetiky Reading University byla zařízena tak, aby mohl být dlouhodobě monitorován pohyb několika osob z katedry. Tito lidé však neměli implantáty, pouze čipové karty. Budova byla upravena tak, aby bylo možné implantát na různých místech rozpoznat. Jakmile jsem vstoupil do budovy, implantát byl zaktivován a zaslal jednoznačný zakódovaný signál, který oznamoval můj vstup. Odpovědí při vstupu do chodby mi byl hlasitý pozdrav „Hello Profesor Warwick" (Vítejte, profesore

Warwicku) a rozsvícení vchodových světel. Můj pohyb v ostatních částech budovy bylo možné neustále monitorovat. Když jsem se například přiblížil ke své laboratoři, automaticky se mi otevřely dveře. Počítač okamžitě zobrazil mou výchozí webovou stránku a zobrazil počet nových e-mailů. Do mapy budovy byly neustále zanášeny informace o mé pozici a času, kdy jsem vstoupil do jednotlivých místností a jak dlouho jsem v nich setrval.

Monitorování a sledování pohybu osob pomocí takovéto technologie implantátů nabízí široké možnosti. Tuto kapitolu píši v roce 2005, kdy je stejný typ implantátu komerčně dostupný v mnoha nočních klubech v Evropě (např. Baja Beach Club, [1]) – již nepotřebujete kreditní kartu, která může být snadno ukradena nebo ztracena. Při každém zakoupení nápoje se přečte implantovaný čip. Tím dostáváme další překvapivý důvod, proč bychom si měli takový implantát pořídit – módní trend!

Implantát, který jsem měl v roce 1998, je nyní vystaven v Science Museum v Londýně. Jeho délka je přibližně 2,5 cm. Implantáty používané v evropských nočních klubech jsou znatelně menší. Překvapivým zjištěním bylo, že jsem si rychle zvykl na fakt, že implantát je mou nedílnou součástí. Podobnou zkušenost mají zřejmě lidé, kteří mají implantován kardiostimulátor nebo kochleární implantát – lidské tělo úspěšně přijme novou technologii jak fyzicky, tak psychicky. Spíše než o jednoduché zařízení RFID (nebo kochleární implantát) se jedná o rozhraní nervového systému a mozku. To nás přivádí k zajímavé otázce – kde končí tělo lidské a začíná tělo technologické?

Samozřejmě není možné na základě subjektivního experimentu činit velkolepé a obecné závěry. Mohu však popsat dosažené cíle a nechat ostatní, aby si vytvořili vlastní závěry. Některé předložené případy si jistě zaslouží pozornost, zejména ty, které se týkají víry v sebe sama a sebeuvědomování: Kým vlastně jsem? Co je to „já"? Jaká je moje identita? Z technického hlediska je pak jistě zajímavé to, že výsledky pokusu posunuly výzkum o další krok kupředu.

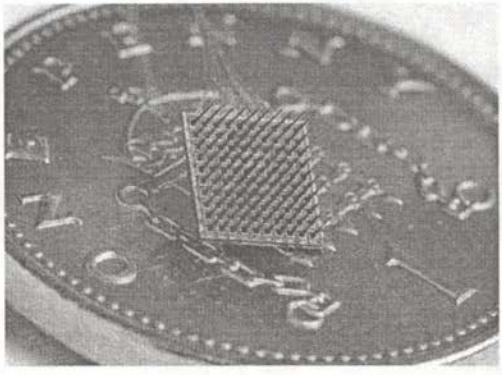
## 22.4 Pokus s implantátem v roce 2002

Rozeznáváme dva různé typy rozhraní, které lze použít pro styk s periferními nervy: extraneurální a intraneurální. Extraneurální (též manžetové elektrody) pevně obepínají nervová vlákna a umožňují získat složený akční potenciál (Compound Action Potential), což je záznam součtu všech signálů uvnitř těchto vláken (Loeb, Peck, 1996), (Slot a kol., 1997) v rozsáhlé oblasti nervového kmenu. Umožňují také jistou formu nedokonale selektivní neurální stimulace (Sweeney, Mortimer, 1990), (Naples, Mortimer, 1996). Nevýhodou je však velké množství vedlejších účinků.

Mnohem přínosnější by však bylo neurální rozhraní, které by zajistilo vysoce selektivní záznam a stimulaci jednotlivých nervových signálů. Takováto charakteristika se spíše hodí pro intraneurální elektrody (Kovacs a kol., 1994), (Lefurge a kol., 1991). Některé typy polí elektrod MEA (viz obrázek 22.1, též barev. příl.) obsahují mnoho elektrod, které se najednou zavedou do svazku smíšeného periferního nervu. Velikost hrotů je srovnatelná s rozměry neuronu, jeden hrot se často dotýká více neuronů. Tím je zajištěn přímý přístup k nervovým vláknům z různých smyslových orgánů a k nervovým vláknům vedoucím k jednotlivým motorickým jednotkám. Pokud známe úmysl pacienta, lze ze získaných signálů vysledovat, které ze získaných signálů tvoří část neurálního

spojení. Takové zařízení umožňuje vytvořit vícekanálové neurální rozhraní. Přesně takové pole elektrod MEA mi bylo při pokusu s implantátem v roce 2002 neurochirurgicky voperováno do vláken středového nervu (*n. medianus*) levé ruky. Stejně jako při předchozím pokusu neexistoval pro tento zásah žádný zdravotní důvod. Jednalo se čistě o pokus na přání.

Je vhodné uvést, že existují i jiné typy polí MEA, které mohou být použity k vytvoření rozhraní mezi nervovým systémem a technikou. Dostupné je poměrně velké množství různých leptaných polí elektrod, které bývají umístěny z vnější strany nervového vlákna, čímž se spíše řadí k manžetovým elektrodám. Získané signály jsou také



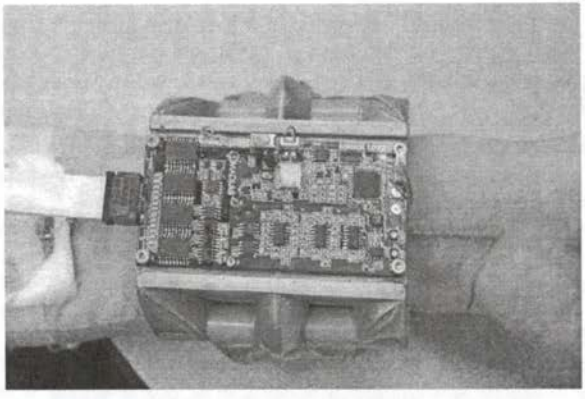
Obr. 22.1 Pole elektrod MEA o rozměru 4 × 4 mm, které obsahuje 100 elektrod. Pro srovnání je umístěno na minci jedné pence (Velká Británie).

podobné těm, které bychom obdrželi prostřednictvím manžetové elektrody – lze získat pouze složené signály. A proto jsme pro naše účely tento typ elektrod nezvolili. Pole elektrod MEA, které bylo použito v našem případě, se skládá z pole jehličkovitých elektrod, které jsou do svazku nervových vláken zasunuty přímo, což je vhodnější než jejich umístění paralelně s vlákny nebo v těsné blízkosti vláken.

Zvláště v současné době, kdy technologie dosáhla takové úrovně, že je možné vytvářet spolehlivá a použitelná neurální rozhraní o rozměrech několika mikrometrů, se rozšiřují možnosti aplikace implantované neurální protetiky (například v případě postižení Parkinsonovou nemocí). Při našem experimentu jsme úzce spolupracovali s ošetřovatelským centrem Radcliffe Infirmary a centrem National Spinal Injuries Centre nemocnice Stoke Manderville Hospital ve městě Aylesbury ve Velké Británii. Dílčím cílem tohoto experimentu bylo zhodnocení přínosu takového implantátu ke zlepšení situace pacientů s poraněním páteře.

Vytvořené neurální rozhraní umožnilo dvousměrný tok informací. Stimulace elektrickým proudem takové úrovně, kterou bylo možné vnímat, nabídla prostředek pro odesílání informací do mého nervového systému. Z aktivity nervového svazku v oblasti elektrod, v němž bylo snímáno asi pět neuronů, se podařilo dekódovat řídicí signály mozku vedoucí k motorickým jednotkám. Mezi polem elektrod MEA a vzdáleným počítačem bylo vyvinuto obousměrné komunikační rádiové rozhraní (Warwick a kol.,

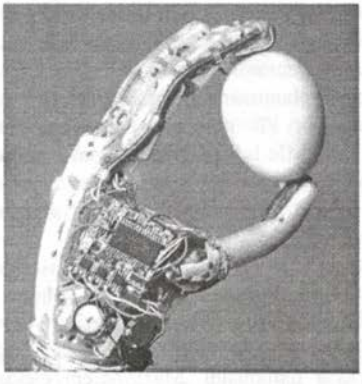
2003), viz obrázek 22.2, též barev. příl., s dosahem kolem 10 metrů, to dovolovalo jistou volnost pohybu. Tímto způsobem bylo možné posílat signály z mého nervového systému do počítače a naopak z počítače mohla být informace přenesena do mého nervového systému.



Obr. 22.2 Mobilní rozhraní připojené k mému zápěstí.

## 22.5 Pokusná fáze

S voperovaným implantátem byla provedena celá řada pokusů (Warwick a kol., 2004), z nichž některé byly spíše zaměřeny na možnosti rehabilitace, zatímco další byly zaměřeny na různé způsoby vylepšení a rozšíření člověka. Když jsem například pohnul prstem, přenesly se nervové signály z mého nervového systému do počítače a odtud do robotické ruky (viz obr. 22.3). Senzory v konečcích prstů robotické ruky poté převedly



Obr. 22.3 Inteligentní antropomorfní protéza ruky.

informaci o snímaném tlaku na signály, které byly poslány zpět do mého nervového systému. V testech, při kterých jsem měl zakryté oči, jsem byl nejen schopen prostřednictvím svých nervových signálů pohybovat robotickou rukou, ale dokázal jsem také přesně rozeznat, s jakou silou robot uchopil daný předmět.

V jedné fázi byl experiment proveden přes Internet. Zatímco já jsem byl fyzicky přítomen na University of Columbia v New Yorku, robotická ruka byla umístěna na Reading University ve Velké Británii. Výsledkem tohoto pokusu je zjištění, že pokud je nervový systém člověka přímo propojen s Internetem, stává se Internet de facto rozšířením nervového systému. Tělo takto vzniklého Kyborga velmi rychle zapomene na lidská omezení a dosáhne tak daleko, jak jen to Internet dovolí. V mém případě byl mozek schopen přímo ovládat robotickou ruku na jiném kontinentě – mé kybernetické tělo tak dosahovalo až za Atlantský oceán.

V dalším experimentu, který byl proveden s popsaným implantátem, byla zkoumána možnost vytvořit další smysl, pomocí kterého by bylo možné vnímat signály z ultrazvukových senzorů. Při tomto experimentu jsem získal přesnou obdobu cítění netopýra, díky které jsem byl schopen určit vzdálenost předmětů i se zavřenýma očima. Výsledek ukazuje, že je možné rozšířit smyslové vnímání jedince mnoha rozličnými způsoby, například infračervenými nebo rentgenovými senzory, jejichž výstup by byl připojen na lidský nervový systém, a tím i k lidskému mozku. Z této studie jasně plyne skutečnost, že lidský mozek je schopen vyrovnat se s novými pocity tohoto typu.

Asi nejvíce vzrušující část pokusů byla provedena s mou ženou Irenou, která má ve svém nervovém systému také voperované elektrody, kdy se mezi našimi nervovými systémy přenášely motorické nervové signály. Když jsem pohnul prstem, příslušný nervový signál se objevil i v nervovém systému mé ženy. Tímto způsobem jsme vytvořili jakýsi způsob telegrafické komunikace mezi nervovými systémy dvou lidských bytostí propojených přes počítačovou síť (Warwick a kol., 2004). Dalším krokem tohoto výzkumu bude nepochybně nalezení stejného způsobu komunikace mezi dvěma jedinci, jejichž mozky budou zapojeny podobným způsobem – komunikaci prostřednictvím myšlenky. Je až vzrušující přemýšlet o tom, jak daleko můžeme tímto směrem zajít – bude to například znamenat zánik jazyka takového, jak jej známe dosud? Budeme se muset učit novému jazyku, jazyku myšlenek?

## 22.6 Některé praktické postřehy

Je důležité si uvědomit, že interakce elektronických signálů s lidským mozkem může zapříčinit to, že mozek bude výrazně ovlivněn, jak se také stalo v případě mého voperovaného implantátu. Příkladem mohou být úspěšné stimulační implantáty, které pouhým elektronickým způsobem potlačují třes, který způsobuje Parkinsonova choroba. Takové stimulační implantáty již byly voperovány mnoha lidem a při aktivaci je jejich účinek stejně dramatický jako okamžitý – mozek začne fungovat zcela jinak.

V případech, kdy se implantáty místo k pouhé obnově nefunkčních částí těla použijí k rozšíření funkcí lidského mozku, není v otázce implantátu v současné době zcela jasno. Zatímco externí nástroje, např. vojenské zařízení pro noční vidění, slouží jako krátkodobé pomůcky, dlouhodoběji působící implantáty mají mnohem přímější dopad, protože

změní schopnosti jedince, jeho chápání, sebevědomí a uvědomování si především sebe samého. Člověk s implantátem již není stejný jako předtím – stal se Kyborgem.

Spojením mentálních funkcí člověka a stroje dochází k vytvoření hybridní osobnosti. Pokud se nervový systém člověka spojí přímo s technologií, nezmění to pouze jeho povahu (pokud to tak stále můžeme nazývat), ale zároveň vyvstanou nové otázky týkající se nového významu slova „já" a také autonomie daného jedince. Kým vlastně jste, je-li váš mozek a nervový systém částečně lidský a částečně strojový? Člověk vybavený zařízením pro noční vidění zůstává stále svou podstatou člověkem. Osobnost a identita člověka s nervovým systémem, který je propojen s počítačovou sítí, je diskutabilní. Pokud není vědomí jedince založeno pouze na mozku lidském, ale spíše na kombinaci lidského a strojového mozku, nutně to ovlivní jeho rozhodování, morálku, charakter a dokonce i jeho pojetí etiky (Warwick, 2003). Můžeme předpokládat, že takový Kyborg se bude na obyčejné lidi dívat s opovržením a bude je jistě považovat za takové, jací mu připadají, tedy intelektuálně podřazené. Jeho jednání bude ovlivněno podmínkami funkčnosti jeho rozšíření, například funkčností síťového spojení, čímž se takový Kyborg stává závislým. Navíc může dojít k situaci, kdy by byl ovládán externě.

Je zřejmé, že pokud porovnáme schopnosti strojů se schopnostmi lidskými, najdeme značné rozdíly jak po stránce fyzické, tak i mentální. Co se týče inteligence, je zřejmé, že ta strojová má ve srovnání s lidskou mnoho různých výhod. Tyto výhody se nejvíce projeví při přímém spojení lidského mozku s počítačovou sítí. Ukazují možnosti, jak může být intelekt člověka vylepšen a jsou hlavní motivací a důvodem pro spojení lidí a strojů.

## 22.7 Vylepšení ve stylu Kyborga

Výhody strojové inteligence, například vícerozměrné výpočty, množství vstupů a rozšířené schopnosti snímání, jsou hlavním důvodem, proč lidé využívají počítače. Rychlé a vysoce přesné matematické schopnosti, vysoká rychlost výpočtů, internetová znalostní báze a přesná dlouhodobá paměť, to jsou přednosti, které mohou být kombinovány s výhodami lidského mozku, kterými jsou pružnost uvažování, schopnost abstrakce a práce s nejednoznačnými informacemi. To by však mohl být pouze začátek.

Lidský mozek se vyznačuje omezenými smyslovými schopnostmi, přinejmenším ve světle našeho současného chápání jeho funkce. Lidé mají pouze pět smyslů o kterých víme, zatímco strojové vnímání světa nabízí mnohem širší možnosti vnímání okolí. V experimentu z roku 2002 jsme zjistili, že člověka můžeme vybavit přinejmenším ultrazvukovým smyslem – ostatní smysly budou jistě následovat. Lidé jsou také omezeni svým pouhým třírozměrným chápáním, vnímáním a vizualizací okolního světa. Oproti tomu počítače jsou schopné poměrně snadno se zabývat stovkami dimenzí a pojmovými vztahy mezi těmito dimenzemi.

Pravděpodobně největší výhodou strojové inteligence je jejich komunikace. Lidský způsob komunikace využívá velkého počtu různých fyzikálních principů, na jejichž konci figuruje přenos elektrochemických signálů. Takový způsob je poměrně nedokonalý, velmi pomalý, nepřesný a omezený vzdáleností, protože zahrnuje převod mechanických signálů – zvukových vln oběma směry. Základním rysem lidského jazyka je sériový typ komunikace a omezené porozumění významu myšlenek, což je příčinou značného množ-

ství chyb při přenosu informací mezi lidskými mozky. Naproti tomu stroje mohou komunikovat s mnoha dalšími stroji najednou a po celém světě. A to pouze s malým množstvím chyb či zcela bezchybně.

Obecně lze říci, že spojení lidského mozku a počítačové sítě pomocí implantátu přináší jedinci s implantátem zřejmé výhody strojové inteligence. Možnost získání pouhé jedné či dvou zmíněných schopností by mohl lidi zlákat, aby si takové vylepšení pořídili.

## 22.8 Závěr

Zdá se, že v blízké budoucnosti bude lidstvo čelit výzvě evoluce ve formě strojové inteligence – životní formě, o které ještě mnoho nevíme a které se dostatečně neobáváme jako potenciálního protivníka. Naštěstí se ukazuje, že alternativní strategie, tj. vytvoření oboustranně výhodného partnerství, je reálná a v budoucnosti přinese další pokrok a vývoj.

V této kapitole jsem se pokusil zdůraznit závažnost potenciální hrozby a současně prostřednictvím mých probíhajících experimentů poukázat, jaké mají lidé k dispozici technologické možnosti pro zlepšení svého mozku. A nastínit vzrušující, realistickou, avšak zatím nejistou budoucnost.

Nezadržitelně se blíží doba, kdy budeme muset rozhodnout – být nahrazeni stroji, které jsme sami původně vytvořili, nebo zůstat konkurenceschopní díky vylepšení vlastní osoby. Je ovšem jisté, že budoucnosti bude dominovat jiný druh než přírodní druh *Homo sapiens*. Jinou možností je, že dominantní rasou se stanou stroje a lidé budou žít jako odpadlíci, budou podřadným druhem.

Podle mého názoru je mnohem lepší alternativou chopit se příležitosti a vstoupit do budoucnosti jako Kyborgové se schopnostmi daleko převyšujícími ty lidské. To je rozhodně můj směr, kterým se chci ubírat. A co vy?

### Literatura

Clynes M. E., Kline N. S.: *Cyborgs and Space*. In: Astronautics, no. 26/27, Sept. 1960, s. 74–75.

Donoghue J. P.: *Connecting cortex to machines: recent advances in brain interfaces*. Nature Neuroscience, Vol. 5 Supplement, November 2002, s. 1085–1088.

Heppenheimer T. A.: *First Flight*. John Wiley & Sons, 2003.

Joy, W.: *Why the future doesn't need us*. Wired, Vol. 8, Issue. 4. 2000.

Kennedy P., Bakay R., Moore M., Adams K., Goldwaith J.: *Direct control of a computer from the human central nervous system*. IEEE Transactions on Rehabilitation Engineering, Vol. 8, 2000, s. 198–202.

Kovacs G. T., Storment C. W., Halks Miller M., Belczynski C. R., Della Santina C. C., Lewis E. R., Maluf N. I.: *Silicon-substrate microelecrode arrays for parallel recording of neural activity in peripheral and cranial nerves*. IEEE Transactions on Biomedical Engineering, 41, 1994, s. 567–577.

Lefurge T., Goodall E., Horch, K. W., Stensaas L., Schoenberg A. A.: *Chronically implanted intrafascicular recording electrodes*. Ann Biomedical Engineering, 19, 1991 s. 197–207.

Loeb G. E., Peck R. A.: *Cuff Electrodes for Cronic Stimulation and Recording of Peripheral Nerve Activity*. J Neurosci Methods, 64, 1996, s. 95–103.

Naples G., Mortimer J.: *A Spiral Nerve Cuff Electrode for Peripheral Nerve Stimulation*. IEEE Trans. Biomed. Eng., Vol. 35, 1996, s. 905–916.

Penrose R.: *Shadows of the Mind*. Oxford University Press. 1994.

Poboroniuc M. S., Fuhr T., Riener R., Donaldson N.: *Closed-Loop Control for FES-Supported Standing Up and Sitting Down*. Proc. 7th Conf. of the IFESS, Ljubljana, Slovinsko, 2002, s. 307–309.

Preston J., Bishop J. M. (eds): *Views into the Chinese Room*. Oxford University Press, 2004.

Searle J. R.: *The Mystery of Consciousness*. New York Review Book, 1997.

Slot P. J., Selmar P., Rasmussen A., Sinkjaer T.: *Effect of long-term implanted nerve cuff electrodes on the electrophysiological properties of the human sensory nerves*. Artif-Organs, 21, 1997, s. 207–209.

Sweeney J., Mortimer J.: *A Nerve Cuff Technique for Selective Excitation of Peripheral Nerve Trunk Regions*. IEEE Trans. Biomed. Eng., Vol. 37, 1990, s. 706–715.

Warwick K.: *March of the Machines*. Century, 1997.

Warwick K. QI: *The Quest for Intelligence*. Piatkus, 2001.

Warwick K.: *Cyborg morals, Cyborg values, Cyborg ethics*. Ethics and Information Technology, Vol. 5(3), 2003, s.131–137.

Warwick K., Gasson M., Hutt B., Goodhew I., Kyberd P., Andrews B., Teddy P., Shad A.: *The Application of Implant Technology for Cybernetic Systems*. Archives of Neurology, Vol. 60, No. 10, 2003, s. 1369–1373.

Warwick K., Gasson M., Hutt B., Goodhew I., Kyberd P., Schulzrinne H., Wu X.: *Thought Communication and Control: A First Step using Radiotelegraphy*. IEE Proceedings on Communications, Vol.151, No. 3, 2004, s. 185–189.

Yu N. Y., Chen J. J., Ju M. S.: *Closed-Loop Control of Quadriceps/Hamstring activation for FES-Induced Standing-Up Movement of Paraplegics*. Journal of Musculo-skeletal Research, Vol. 5(3), 2001.

[1] http://www.bajabeach.es

# Summary

*Sir, you only become a real reader after you have found out that what you have been looking for cannot be found in books.*

The target audience of this volume is much wider than that addressed by the former four books in this series on artificial intelligence. The first two volumes offer introduction into the domain, the third one treats various aspects of interaction among agents (including their environment) and the fourth one considers problems related to mostly uncertain knowledge, which has to be extracted, manipulated and applied. While the previous four volumes are intended for technically minded students or experts the current volume tries to attract attention of an interdisciplinary community by offering not only deeper philosophical context but also real-life examples of AI solutions improving quality of life or enhancing human capabilities. We believe that individuals as well as entire human society can highly benefit from wise applications of AI technologies. The crucial task is to recognize and seize appropriate opportunities in various fields including not only industry and medicine but also applications with social dimension. The best way how to achieve this is to open dialog among experts from very diverse domains. This book tries to contribute towards this goal. We hope that some of the presented results explained as clearly as possible without excessive technicalities can inspire novel approaches and solutions to problems appearing elsewhere or identify new challenges (in-step with the comment of the librarian on the cartoon).

Our choice of contributions for this publication has been inspired by the Second Czech-Argentinean Biennale entitled "E-Golems" organized in Prague as a part of the

16th World IFAC Congress, July 3 – 5, 2005. The biennale discussed the roots of man-machine collaboration in historical horizon starting from the medieval vision of the Golem up-to the current interpretation of Golem as a robotic system equipped by tools and features resulting from research in artificial intelligence, cybernetics, virtual reality, artificial life, nanotechnologies, and Internet services. Internet and grid computing can be conceived as an electronic or virtual version of Golem or E-Golem for short – this concept gave the name to the considered scientific event. Several speakers of the biennale were invited to contribute to this volume with true international dimension – the authors of four chapters are from Argentina and Great Britain.

The block of first five chapters treats philosophical problems related to our potency of creating ever more complex robots or even "virtual beings" with increasing degree of independence. This feature draws such entities closer to human beings and consequently it invites the authors to reconsider number of fundamental concepts including mind, emergence as well as computational limits of cognitive systems.

Mathematics is the binding theme of the chapters 6 to 8 in the second block. There are explained advantages of the constraint solving paradigm, useful mathematical tools for modeling complex cognitive processes and even raised provocative questions like "Can beauty be reduced to a mathematical expression?".

The third block (chapters 9 to 13) studies knowledge, the way it is elicited and handled. It attends to its role in the process of engineering design in general as well as in the case of design of intelligent machines. Special attention is devoted to nature inspired knowledge extraction algorithms, to reinforcement learning as well as to evaluation of empirical results characterizing the discovery process itself.

The fourth block (chapters 14 to 17) summarizes the state of the art in the domains necessary for exchange of information between a robot and its environment or its human partners. It covers processing of natural language as well as handling of visual or tactile information.

The last block (chapters 18 to 22) reviews number of real-life applications of current technologies. Three chapters are devoted to individual or cooperating robots, the way they can play football, substitute for a human in a hostile environment, ensure production or partially support a nurse. Separate chapter is devoted to assistive tools offering alternative means of communication for challenged persons. The final chapter is discussing possibilities of creation of Cyborgs, e.g. by technologically evolving humans by linking the human brain directly with a machine brain network. There are reported results and experiences obtained during the first bold experiments in this direction.

Vladimír Mařík
Olga Štěpánková
Jiří Lažanský

# Česko-anglický slovník

| | |
|---|---|
| **A** | |
| abstraktní objekt | abstract object |
| agent | agent |
| akustický model | acoustic model |
| algoritmus EM | expectation-maximization algorithm |
| algoritmus GAME | GAME algorithm |
| aliance | alliance |
| anamorfóza | anamorphosis |
| android | android |
| animat | animat |
| anotovaný korpus textů | annotated corpus of texts |
| anticipace | anticipation |
| anticipační chování | anticipated behaviour |
| antropomorfní úchopná hlavice | anthropomorphous grasp device |
| architektura virtuálního člověka | virtual human architecture |
| asistivní technologie | assistive technologies |
| atomická akce | atomic action |
| automatické rozpoznávání mluvené řeči | automatic speech recognition |
| autonomní chování | autonomous behavior |
| autonomní robot | autonomous robot |
| **B** | |
| BCI | Brain-Computer Interface |
| biomechanika | biomechanics |
| bořič symetrií | symmetry breaking constraint |
| **C** | |
| cena cíle | target cost |
| cena konkatenace | concatenation cost |
| cílem řízené chování | goal oriented behaviour |
| **Č** | |
| časoprostor Malamentův–Hogartův | Malemment–Hogart spacetime |
| **D** | |
| dekódování | decoding |
| dekonflikce | deconfliction |
| dekonflikční algoritmy | deconfliction algorithms |

| | |
|---|---|
| dialog s iniciativou systému | system-initiative dialogue |
| dialog s iniciativou uživatele | user-initiative dialogue |
| dialog se smíšenou iniciativou | mixed-initiative dialogue |
| dialogová akce | dialogue action |
| dialogový agent | dialogue agent |
| dialogový akt | dialogue act |
| dialogový manažer | dialogue manager |
| dialogový systém | dialogue system |
| difuzí omezené shlukování | diffusion limited aggregation |
| dlouhodobá paměť | long-term memory |
| dynamická teorie množství | dynamical set theory |

**E**

| | |
|---|---|
| emergence | emergence |
| emergentistická teze | emergentist thesis |
| emergentní efekt | emergent effect |
| enumerační strategie | enumeration |
| epipolární omezení | epipolar constraint |
| epizodická situace | episodic situation |
| Escherův design | Escher design |
| evoluce | evolution |
| evoluce umělá | artificial evolution |
| evoluční interaktivní výpočetní systém | evolutionary interactive computing system |
| explicitní komunikace | explicit communication |
| explicitní znalosti | explicit knowledge |
| expresivní robot | expressive robots |

**F**

| | |
|---|---|
| fázový přechod | phase transition |
| filtrace domén | domain filtering |
| fraktální umění | fractal art |
| fundamentální agnosticismus | fundamental agnosticism |
| funkční inteligence | functional intelligence |

**G**

| | |
|---|---|
| generátor prozodie | prosody generator |
| genetický algoritmus | genetic algorithm |
| geometrické mapy | geometric maps |
| globální podmínka | global constraint |
| Golem | Golem |
| gradientní optimalizace | gradient optimization |

**H**

| | |
|---|---|
| Hadamardova definice stability | Hadamard definition of stability |
| heuristika výběru hodnoty | value ordering heuristics |
| heuristika výběru proměnné | variable ordering heuristics |
| hierarchické reaktivní plánování | hierarchical reactive planning |
| hodnoticí systém | assesment system |
| holarchie | holarchy |
| Homo sapiens | Homo sapiens |

| | |
|---|---|
| hraní | play |
| hranová konzistence | arc consistency |
| hybridní tým | hybrid team |
| **CH** | |
| chování | behaviour |
| Churchova–Turingova teze | Church–Turing Thesis |
| chytrý objekt | smart object |
| **I** | |
| implantát | implant |
| implicitní komunikace | implicit communication |
| implicitní podmínka | implied constraint |
| implicitní znalosti | implicit knowledge, tacit knowledge |
| induktivní modelování | inductive modeling |
| inovativní design | innovative design |
| inteligence při navrhování | intelligence in design |
| inteligentní agent | intelligent agent |
| inteligentní entita | intelligent entity |
| inteligentní robot | intelligent robot |
| inteligentní softwarový agent | intelligent software agent |
| inteligentní stroj | intelligent machine |
| inteligentní virtuální agent | intelligent virtual agent |
| inteligentní vozidlo | intelligent vehicle |
| interaktivní konečný transduktor | interactive finite transducer |
| interaktivní média | interactive media |
| interaktivní Turingův stroj | interactive Turing machine |
| interaktivní umění | interactive art |
| **J** | |
| jazykový model | language model |
| jednotková hranová konzistence | singleton arc consistency |
| jisticí systém | assurance system |
| **K** | |
| kauzalita | causality |
| kauzální doména | causal domain |
| kauzální paradox | causal paradox |
| knihovna plánů | plan library |
| koalice | coallition |
| koexistence člověk–stroj | co-existence man–machine |
| kognice | cognition |
| kognitivní poruchy | cognitive impairment (disorders) |
| kognitivní robot | cognitive robots |
| kognitivní systém | cognitive system |
| kognitivní věda | cognitive science |
| kolaborativní chování | collaborative behavior |
| kolaborativní robotika | collaborative robotics |
| komerční počítačová hra | computer game |
| komunikační inteligence | communication intelligence |

| | |
|---|---|
| komunikační nedostupnost | communication inavailability |
| komunikační topologie | communication topology |
| konceptuální design | conceptual design |
| konečný automat | finite state machine |
| konfigurační design | configuration design |
| konfigurační prostor | configuration space |
| konjugovaná funkce | conjugate function |
| konkatenační syntéza | concatenative synthesis |
| kontrola dopředu | forward checking |
| konzistence po cestě | path consistency |
| konzistenční algoritmus | consistency algorithm |
| kooperativní dekonflikce | cooperative deconfliction |
| kooperativní explorace | cooperative exploration |
| korespondence v obrazech | image correspondence |
| korpusově orientovaná syntéza řeči | corpus-based speech synthesis |
| kovariantní detektor | covariant detector |
| krátkodobá paměť | short-term memory |
| kreativní design | creative design |
| kroková strategie | step labelling |
| kybernetický organismus | cybernetic organism |
| kybernetika | cybernetics |
| kyborg | cyborg |

## L

| | |
|---|---|
| logické programování s podmínkami | constraint logic programming |
| lokalizace robotu | robot localization |

## M

| | |
|---|---|
| mapování prostředí | mapping of environment |
| mechanismus výběru akcí | action selection mechanism |
| mechatronika | mechatronics |
| melovské kepstrální koeficienty | mel cepstral coefficients (MFCC) |
| metaučení | meta-learning |
| metoda maximální věrohodnosti | maximum likelihood estimation (MLE) |
| metoda pohledu dopředu | look ahead |
| metoda pohledu zpět | look back |
| metoda větví a mezí | branch and bound |
| migrace agentů | agent migration |
| miniinvazivní chirurgie | minimal invazive surgery |
| mocninné rozložení | power-law distribution |
| modalita situace | modalities of situations |
| model dírkové kamery | pinohole camera model |
| model kauzální domény | causal domain model |
| modelování zdrojů znalostí | knowledge source modeling |
| modely prostředí | models of environment |
| modul fonetické transkripce | grapheme-to-phoneme module (GTP) |
| mozaika | mosaic |
| mřížka obsazenosti | occupancy grid |
| multiagentní platforma A-globe | multi-agent platform A-globe |
| multiagentní systémy | multiagent systems |
| multirobotické systémy | multirobotic systems |

**N**

| | |
|---|---|
| nalezení přijatelného řešení | satisficing |
| navigační graf | way-point graph |
| navigační mřížka | navigation mesh |
| navigační pravidlo | steering rule |
| neformální znalosti | informal knowledge |
| nekooperativní dekonflikce | non-cooperative deconfliction |
| nekývající jeřáb | anti-sway crane |
| nerozhodnutelný problém | undecidable problem |
| neuniformní Turingův stroj | non-uniform Turing machine |
| neuniformní výpočty | non-uniform computations |
| neurální protetika | neural orthotics |
| neuronová síť | neural network |
| *n*-gramový jazykový model | *n*-gram language model |
| nová média | new media |

**O**

| | |
|---|---|
| obzor | horizon |
| obzorný řez | horizon segment |
| Ockhamova břitva | Ockham's razor |
| olbřímí úkol | supertask |
| omezující podmínka | constraint |
| optimalizace inspirovaná přírodou | nature inspired optimization |
| optimalizační problém se splňováním podmínek | constrained optimisation problem |

**P**

| | |
|---|---|
| panoramatický obraz | panoramic image |
| parametrický design | parametric design |
| páteřní ontologie | backbone ontology |
| perceptivní lineární analýza | perceptual linear predictive analysis (PLP) |
| plánování cesty | path-planning |
| počítačové vidění | computer vision |
| počítačový experiment | computer experiment |
| podoba s člověkem | resemblance to man |
| pole mikroelektrod | microelectrode array |
| polomnožina | semiset |
| polygonální modely prostředí | polygonal models of environment |
| populační pyramida | population pyramid |
| porozumění přirozenému jazyku | natural language understanding |
| poruchy mobility | mobility disorders |
| posilovací funkce | reinforcement function |
| posilované učení | reinforced learning |
| posthumanismus | posthumanism |
| Pražský závislostní korpus | Prague Dependency Treebank |
| prediktivní lineární analýza | linear predictive coding (LPC) |
| prezence | presence |
| princip prvního neúspěchu | fail first |
| princip prvního úspěchu | succeed first |
| problém dobře strukturovaný | tame problem, well-structured problem |

| | |
|---|---|
| problém splnitelnosti | satisfiability problem |
| problém splňování podmínek | constraint satisfaction problem |
| problém zastavení | halting problem |
| procedurální paměť | procedural memory |
| prohledávací strom | search tree |
| prohledávání s diskrepancemi | discrepancy search |
| prohledávání s navracením | backtracking |
| prozodie | speech prosody |
| průzkumný robot | survey robot |
| předcházení kolizím | collision avoidance |
| přehledná třída | distinctive class |
| přeposílání zpráv | relaying |
| přirozená indukce | natural induction |
| přirozené nekonečno | natural infinity |
| pseudonáhodná řada | pseudorandom sequence |

**R**

| | |
|---|---|
| reaktivní agent | reactive agent |
| reaktivní plánování | reactive planning |
| rekonfigurovatelnost systému | system reconfigurability |
| rekonstrukce tvaru | shape reconstruction |
| relativistický Turingův stroj | relativistic Turing machine |
| robot | robot |
| robotická chirurgie | robotic surgery |
| robotické soutěže | robotics competitions |
| robotické vozidlo | robotic vehicle |
| robotický fotbal | robot soccer |
| robotický simulátor | robotic simulator |
| rozhodovací pravidlo | decision rule |
| rozhodovací strom | decision tree |
| rozpoznávání řeči | speech recognition |
| rozšířená Churchova–Turingova teze | extended Church–Turing Thesis |
| runtime rozhraní | runtime interface |
| rutinní design | routine design |

**Ř**

| | |
|---|---|
| řez na přirozených číslech | segment of natural numbers |

**S**

| | |
|---|---|
| sémantická reprezentace | semantic representation |
| sémantická síť | semantic net |
| sémantický rámec | semantic frames |
| senzorické mapy | senzoric maps |
| senzorické poruchy | senzoric impairment |
| servisní robotika | service robotics |
| simulační nástroje | simulation tools |
| simulační systém | simulation system |
| simulátor světa | virtual world simulator |
| síťové rozhraní | network interface |
| skrytý Markovův model | hidden Markov model (HMM) |
| slabé metody | weak methods |

| | |
|---|---|
| sledování polohy | position tracking |
| societní teorie mysli | society theory of mind |
| soutěže záchranných robotů | rescue robot competitions |
| společenství robotů | society of robots |
| společná prezence | common presence |
| stabilní funkce | stable function |
| stabilní posloupnost | stable sequence |
| statistický strojový překlad | statistical machine translation |
| stav dialogu | dialogue state |
| stavová komunikace | state communication |
| strategie půlení domény | domain splitting |
| symbolické mapy | symbolic maps |
| symetrie | symetry |
| synergie | synergy |
| syntaktická analýza | parsing |
| syntéza řeči | speech synthesis |
| syntéza řeči z textu | text-to-speech synthesis (TTS) |
| systém ekogramatik | ecogrammar system |
| systém formálních gramatik | formal grammar system |

## Š

| | |
|---|---|
| špatně strukturovaný problém | wicked problem, ill-structured problem |

## T

| | |
|---|---|
| tagování | tagging |
| taktilní informace | tactile information |
| taktilní obraz | tactile image |
| taktilní otisk | tactile pattern |
| taktilní senzor | tactile sensor |
| taktilní snímač | tactile transducer |
| telematický systém | telematic system |
| teleprezence | telepresence |
| teorie afordancí | affordance theory |
| topologické mapy | topologic maps |
| třída | class |
| Turingův stroj | Turing machine |
| Turingův stroj s orákulem | oracle Turing machine |
| Turingův stroj s rádcem | Turing machine with advice |
| Turingův test | Turing test |

## U

| | |
|---|---|
| údolí odpornosti | uncanny valley |
| umělá inteligence | artificial intelligence |
| umělá mysl | artificial mind |
| umělá stvoření | artificial creatures |
| umělý život | artificial life, alife |

## V

| | |
|---|---|
| vedení dialogu | dialogue management |
| virtuální člověk | virtual human |

| | |
|---|---|
| virtuální drama | virtual drama |
| virtuální průhledový displej | virtual head-up display |
| virtuální vypravěčství | virtual storytelling |
| Viterbiův algoritmus | Viterbi algorithm |
| vizuální teleprezence | visual telepresence |
| vyhýbání se překážkám | obstacle avoidance |
| výpočetní inteligence | computational intelligence |
| výpočetní teorie proudění | computational fluid dynamics |
| výuková hra | serious game |
| vzdálené povědomí | remote awareness |

**Z**

| | |
|---|---|
| zdroj znalostí | knowledge source |
| zlatý řez | golden section |
| znalostní model | acquantaince model |
| znalostní přístup | knowledge-based approach |
| zobecněná emergentistická teze | generalised emergentist thesis |
| zpracování přirozeného jazyka | natural language processing |

# Anglicko-český slovník

**A**

| | |
|---|---|
| abstract object | abstraktní objekt |
| acoustic model | akustický model |
| acquantaince model | znalostní model |
| action selection mechanism | mechanismus výběru akcí |
| affordance theory | teorie afordancí |
| agent | agent |
| agent migration | migrace agentů |
| alife | umělý život |
| alliance | aliance |
| anamorphosis | anamorfóza |
| android | android |
| animat | animat |
| annotated corpus of texts | anotovaný korpus textů |
| anthropomorphous grasp device | antropomorfní úchopná hlavice |
| anticipated behaviour | anticipační chování |
| anticipation | anticipace |
| anti-sway crane | nekývající jeřáb |
| arc consistency | hranová konzistence |
| artificial creatures | umělá stvoření |
| artificial evolution | evoluce umělá |
| artificial intelligence | umělá inteligence |
| artificial life | umělý život |
| artificial mind | umělá mysl |
| assesment system | hodnoticí systém |
| assistive technologies | asistivní technologie |
| assurance system | jisticí systém |
| atomic action | atomická akce |
| automatic speech recognition | automatické rozpoznávání mluvené řeči |
| autonomous behavior | autonomní chování |
| autonomous robot | autonomní robot |

**B**

| | |
|---|---|
| backbone ontology | páteřní ontologie |
| backtracking | prohledávání s navracením |
| behaviour | chování |
| biomechanics | biomechanika |
| Brain-Computer Interface | BCI |
| branch and bound | metoda větví a mezí |

C

| | |
|---|---|
| causal domain | kauzální doména |
| causal domain model | model kauzální domény |
| causal paradox | kauzální paradox |
| causality | kauzalita |
| Church–Turing Thesis | Churchova–Turingova teze |
| class | třída |
| coallition | koalice |
| co-existence man–machine | koexistence člověk–stroj |
| cognition | kognice |
| cognitive impairment (disorders) | kognitivní poruchy |
| cognitive robots | kognitivní robot |
| cognitive science | kognitivní věda |
| cognitive system | kognitivní systém |
| collaborative behavior | kolaborativní chování |
| collaborative robotics | kolaborativní robotika |
| collision avoidance | předcházení kolizím |
| common presence | společná prezence |
| communication inavailability | komunikační nedostupnost |
| communication intelligence | komunikační inteligence |
| communication topology | komunikační topologie |
| computational fluid dynamics | výpočetní teorie proudění |
| computational intelligence | výpočetní inteligence |
| computer experiment | počítačový experiment |
| computer game | komerční počítačová hra |
| computer vision | počítačové vidění |
| concatenation cost | cena konkatenace |
| concatenative synthesis | konkatenační syntéza |
| conceptual design | konceptuální design |
| configuration design | konfigurační design |
| configuration space | konfigurační prostor |
| conjugate function | konjugovaná funkce |
| consistency algorithm | konzistenční algoritmus |
| constrained optimisation problem | optimalizační problém se splňováním podmínek |
| constraint | omezující podmínka |
| constraint logic programming | logické programování s podmínkami |
| constraint satisfaction problem | problém splňování podmínek |
| cooperative deconfliction | kooperativní dekonflikce |
| cooperative exploration | kooperativní explorace |
| corpus-based speech synthesis | korpusově orientovaná syntéza řeči |
| covariant detector | kovariantní detektor |
| creative design | kreativní design |
| cybernetic organism | kybernetický organismus |
| cybernetics | kybernetika |
| cyborg | kyborg |

D

| | |
|---|---|
| decision rule | rozhodovací pravidlo |
| decision tree | rozhodovací strom |
| decoding | dekódování |

| | |
|---|---|
| deconfliction | dekonflikce |
| deconfliction algorithms | dekonflikční algoritmy |
| dialogue act | dialogový akt |
| dialogue action | dialogová akce |
| dialogue agent | dialogový agent |
| dialogue management | vedení dialogu |
| dialogue manager | dialogový manažer |
| dialogue state | stav dialogu |
| dialogue system | dialogový systém |
| diffusion limited aggregation | difuzí omezené shlukování |
| discrepancy search | prohledávání s diskrepancemi |
| distinctive class | přehledná třída |
| domain filtering | filtrace domén |
| domain splitting | strategie půlení domény |
| dynamical set theory | dynamická teorie množství |

**E**

| | |
|---|---|
| ecogrammar system | systém ekogramatik |
| emergence | emergence |
| emergent effect | emergentní efekt |
| emergentist thesis | emergentistická teze |
| enumeration | enumerační strategie |
| epipolar constraint | epipolární omezení |
| episodic situation | epizodická situace |
| Escher design | Escherův design |
| evolution | evoluce |
| evolutionary interactive computing system | evoluční interaktivní výpočetní systém |
| expectation-maximization algorithm | algoritmus EM |
| explicit communication | explicitní komunikace |
| explicit knowledge | explicitní znalosti |
| expressive robots | expresivní robot |
| extended Church–Turing Thesis | rozšířená Churchova–Turingova teze |

**F**

| | |
|---|---|
| fail first | princip prvního neúspěchu |
| finite state machine | konečný automat |
| formal grammar system | systém formálních gramatik |
| forward checking | kontrola dopředu |
| fractal art | fraktální umění |
| functional intelligence | funkční inteligence |
| fundamental agnosticism | fundamentální agnosticismus |

**G**

| | |
|---|---|
| generalised emergentist thesis | zobecněná emergentistická teze |
| genetic algorithm | genetický algoritmus |
| geometric maps | geometrické mapy |
| global constraint | globální podmínka |
| goal oriented behaviour | cílem řízené chování |
| golden section | zlatý řez |
| gradient optimization | gradientní optimalizace |
| grapheme-to-phoneme module (GTP) | modul fonetické transkripce |

**H**

| | |
|---|---|
| Hadamard definition of stability | Hadamardova definice stability |
| halting problem | problém zastavení |
| hidden Markov model (HMM) | skrytý Markovův model |
| hierarchical reactive planning | hierarchické reaktivní plánování |
| holarchy | holarchie |
| Homo sapiens | Homo sapiens |
| horizon | obzor |
| horizon segment | obzorný řez |
| hybrid team | hybridní tým |

**I**

| | |
|---|---|
| ill-structured problem | špatně strukturovaný problém |
| image correspondence | korespondence v obrazech |
| implant | implantát |
| implicit communication | implicitní komunikace |
| implicit knowledge, tacit knowledge | implicitní znalosti |
| implied constraint | implicitní podmínka |
| inductive modeling | induktivní modelování |
| informal knowledge | neformální znalosti |
| innovative design | inovativní design |
| intelligence in design | inteligence při navrhování |
| intelligent agent | inteligentní agent |
| intelligent entity | inteligentní entita |
| intelligent machine | inteligentní stroj |
| intelligent robot | inteligentní robot |
| intelligent software agent | inteligentní softwarový agent |
| intelligent vehicle | inteligentní vozidlo |
| intelligent virtual agent | inteligentní virtuální agent |
| interactive art | interaktivní umění |
| interactive finite transducer | interaktivní konečný transduktor |
| interactive media | interaktivní média |
| interactive Turing machine | interaktivní Turingův stroj |

**K**

| | |
|---|---|
| knowledge source | zdroj znalostí |
| knowledge source modeling | modelování zdrojů znalostí |
| knowledge-based approach | znalostní přístup |

**L**

| | |
|---|---|
| language model | jazykový model |
| linear predictive coding (LPC) | prediktivní lineární analýza |
| long-term memory | dlouhodobá paměť |
| look ahead | metoda pohledu dopředu |
| look back | metoda pohledu zpět |

**M**

| | |
|---|---|
| Malemment–Hogart spacetime | časoprostor Malamentův–Hogartův |
| mapping of environment | mapování prostředí |
| maximum likelihood estimation (MLE) | metoda maximální věrohodnosti |

| | |
|---|---|
| mechatronics | mechatronika |
| mel cepstral coefficients (MFCC) | melovské kepstrální koeficienty |
| meta-learning | metaučení |
| microelectrode array | pole mikroelektrod |
| minimal invazive surgery | miniinvazivní chirurgie |
| mixed-initiative dialogue | dialog se smíšenou iniciativou |
| mobility disorders | poruchy mobility |
| modalities of situations | modalita situace |
| models of environment | modely prostředí |
| mosaic | mozaika |
| multi-agent platform A-globe | multiagentní platforma A-globe |
| multi-agent systems | multiagentní systémy |
| multirobotic systems | multirobotické systémy |

**N**

| | |
|---|---|
| natural induction | přirozená indukce |
| natural infinity | přirozené nekonečno |
| natural language processing | zpracování přirozeného jazyka |
| natural language understanding | porozumění přirozenému jazyku |
| nature inspired optimization | optimalizace inspirovaná přírodou |
| navigation mesh | navigační mřížka |
| network interface | síťové rozhraní |
| neural network | neuronová síť |
| neural orthotics | neurální protetika |
| new media | nová média |
| *n*-gram language model | *n*-gramový jazykový model |
| non-cooperative deconfliction | nekooperativní dekonflikce |
| non-uniform computations | neuniformní výpočty |
| non-uniform Turing machine | neuniformní Turingův stroj |

**O**

| | |
|---|---|
| obstacle avoidance | vyhýbání se překážkám |
| occupancy grids | obsazenosti mřížky |
| Ockham's razor | Ockhamova břitva |
| oracle Turing machine | Turingův stroj s orákulem |

**P**

| | |
|---|---|
| panoramic image | panoramatický obraz |
| parametric design | parametrický design |
| parsing | syntaktická analýza |
| path consistency | konzistence po cestě |
| path-planning | plánování cesty |
| perceptual linear predictive analysis (PLP) | perceptivní lineární analýza |
| phase transition | fázový přechod |
| pinohole camera model | model dírkové kamery |
| plan library | knihovna plánů |
| play | hraní |
| polygonal models of environment | polygonální modely prostředí |
| population pyramid | populační pyramida |
| position tracking | sledování polohy |
| posthumanism | posthumanismus |

| | |
|---|---|
| power-law distribution | mocninné rozložení |
| Prague Dependency Treebank | Pražský závislostní korpus |
| presence | prezence |
| procedural memory | procedurální paměť |
| prosody generator | generátor prozodie |
| pseudorandom sequence | pseudonáhodná řada |

**R**

| | |
|---|---|
| reactive agent | reaktivní agent |
| reactive planning | reaktivní plánování |
| reinforced learning | posilované učení |
| reinforcement function | posilovací funkce |
| relativistic Turing machine | relativistický Turingův stroj |
| relaying | přeposílání zpráv |
| remote awareness | vzdálené povědomí |
| rescue robot competitions | soutěže záchranných robotů |
| resemblance to man | podoba s člověkem |
| robot | robot |
| robot localization | lokalizace robotu |
| robot soccer | robotický fotbal |
| robotic simulator | robotický simulátor |
| robotic surgery | robotická chirurgie |
| robotic vehicle | robotické vozidlo |
| robotics competitions | robotické soutěže |
| routine design | rutinní design |
| runtime interface | runtime rozhraní |

**S**

| | |
|---|---|
| satisfiability problem | problém splnitelnosti |
| satisficing | nalezení přijatelného řešení |
| search tree | prohledávací strom |
| segment of natural numbers | řez na přirozených číslech |
| semantic frames | sémantický rámec |
| semantic net | sémantická síť |
| semantic representation | sémantická reprezentace |
| semiset | polomnožina |
| senzoric impairment | senzorické poruchy |
| senzoric maps | senzorické mapy |
| serious game | výuková hra |
| service robotics | servisní robotika |
| shape reconstruction | rekonstrukce tvaru |
| short-term memory | krátkodobá paměť |
| simulation system | simulační systém |
| simulation tools | simulační nástroje |
| singleton arc consistency | jednotková hranová konzistence |
| smart object | chytrý objekt |
| society of robots | společenství robotů |
| society theory of mind | societní teorie mysli |
| speech prosody | prozodie |
| speech recognition | rozpoznávání řeči |
| speech synthesis | syntéza řeči |

| | |
|---|---|
| stable function | stabilní funkce |
| stable sequence | stabilní posloupnost |
| state communication | stavová komunikace |
| statistical machine translation | statistický strojový překlad |
| steering rule | navigační pravidlo |
| step labelling | kroková strategie |
| succeed first | princip prvního úspěchu |
| supertask | olbřímí úkol |
| survey robot | průzkumný robot |
| symbolic maps | symbolické mapy |
| symetry | symetrie |
| symmetry breaking constraint | bořič symetrií |
| synergy | synergie |
| system reconfigurability | rekonfigurovatelnost systému |
| system-initiative dialogue | dialog s iniciativou systému |

**T**

| | |
|---|---|
| tactile image | taktilní obraz |
| tactile information | taktilní informace |
| tactile pattern | taktilní otisk |
| tactile sensor | taktilní senzor |
| tactile transducer | taktilní snímač |
| tagging | tagování |
| tame problem, well-structured problem | problém dobře strukturovaný |
| target cost | cena cíle |
| telematic system | telematický systém |
| telepresence | teleprezence |
| text-to-speech synthesis | syntéza řeči z textu |
| topologic maps | topologické mapy |
| Turing machine | Turingův stroj |
| Turing machine with advice | Turingův stroj s rádcem |
| Turing test | Turingův test |

**U**

| | |
|---|---|
| uncanny valley | údolí odpornosti |
| undecidable problem | nerozhodnutelný problém |
| user-initiative dialogue | dialog s iniciativou uživatele |

**V**

| | |
|---|---|
| value ordering heuristics | heuristika výběru hodnoty |
| variable ordering heuristics | heuristika výběru proměnné |
| virtual drama | virtuální drama |
| virtual head-up display | virtuální průhledový displej |
| virtual human | virtuální člověk |
| virtual human architecture | architektura virtuálního člověka |
| virtual storytelling | virtuální vypravěčství |
| virtual world simulator | simulátor světa |
| visual telepresence | vizuální teleprezence |
| Viterbi algorithm | Viterbiův algoritmus |

**W**

| | |
|---|---|
| way-point graph | navigační graf |
| weak methods | slabé metody |
| wicked problem | špatně strukturovaný problém |

# Rejstřík


A
agent 60, 135
– dialogový 323
– inteligentní 387
– – softwarový 135
– – virtuální 135
–, migrace 450
– reaktivní 383, 517
agnosticismus fundamentální 111
akce atomická 115
akt dialogový 313
algoritmus A* 200
– dekonfliktní 486
– EM 294
– GAME 237
– genetický 238
– GMDH COMBI 235
– GMDH MIA 236
– konzistenční 152
– SLAM 437
– Viterbiův 294
aliance 452
analýza lineární perceptivní 288
– – prediktivní 287
– syntaktická 297
anamorfóza 181
android 50
animat 129
anticipace 387
automat konečný 120
automobil stabilizovaný 229

B
BCI 503
biomechanika 368
bořič symetrií 165
břitva Ockhamova 257

C
cena cíle 309
cena konkatenace 309
Clockwork 214

Č
časoprostor Malamentův–Hogartův 82
člověk virtuální 113
– –, architektura 115

D
definice stability Hadamardova 186
dekódování 296
dekonflikce 437
– kooperativní 486
– nekooperativní 486
design Escherův 179
– inovativní 201
– konceptuální 203
– konfigurační 201
– kreativní 201
– parametrický 201
– rutinní 201
detektor kovariantní 341
dialog, akce 311
– s iniciativou smíšenou 316
– – systému 315
– – uživatele 317
–, stav 311
–, vedení 311
displej virtuální průhledový 415
doména kauzální 94
– –, model 101
–, filtrace 150
drama virtuální 114

E
efekt emergentní 92
elektrookulografie 506
emergence 91
entita inteligentní 433
evoluce 385

evoluce umělá 401
experiment počítačový 187
explorace kooperativní 470

F
fotbal robotický 425
funkce konjugovaná 98
– posilovací 264
– stabilní 99

G
Golem 50
GPS 66
graf navigační 123

H
heuristika výběru hodnoty 159
– – proměnné 159
holarchie 452
Homo sapiens 515
hra počítačová komerční 128
– výuková 125
hraní 30

CH
chirurgie miniinvazivní 407
– robotická 407
chování 376
– anticipační 387
– autonomní 443
– kolaborativní 443
– řízené cílem 387

I
IBIS 206
implantát 517
indexy sémantické 215
indukce přirozená 102
informace taktilní 351
inteligence funkční 226
– komunikační 226
– při navrhování 231
– umělá 32
– výpočetní 247

J
jeřáb nekývající 227

K
kamera dírková, model 332
kauzalita 94
knihovna plánů 115
koalice 452
koeficienty melovské kepstrální 289
koexistence člověk–stroj 398
kognice 75
komunikace explicitní 448
– implicitní 446
– stavová 447
koncepty, rozpracování 204
kontrola dopředu 157
konzistence hranová 151
– – jednotková 163
– po cestě 155
korespondence v obrazech 331
korpus textů, anotovaný 273
kybernetika 67
kyborg 60, 514

L
learning by doing 210
lokalizace 454
L-systém 59

M
MALACH 280
manažer dialogový 311
mapování prostředí 454
mapy geometrické 457
– senzorické 456
– symbolické 459
– topologické 458
média interaktivní 63
– nová 63
mechanismus výběru akcí 117
mechatronika 220
metaučení 260
metoda maximální věrohodnosti 294
– pohledu dopředu 157
– – zpět 156
– větví a mezí 160
metody gradientní 200
– slabé 200
modalita situace 37
model akustický 286
– Frenchův 202
– jazykový 286
– – *n*-gramový 295
– skrytý Markovův 290
– prostředí 454
– – polygonální 457
– znalostní 443
modelování induktivní 234
modul fonetické transkripce 301

mozaika 175
mřížka navigační 123
– obsazenosti 456
mysl umělá 113
myšlení neverbální 212
– verbální 212

N
nástroje simulační 465
nedostupnost komunikační 449
nekonečno přirozené 98

O
objekt abstraktní 97
– chytrý 132
obraz panoramatický 330
– taktilní 351
obzor 96
omezení epipolární 335
ontologie páteřní 218
optimalizace gradientní 242
– přírodou inspirovaná 242
organismus kybernetický 514
otisk taktilní 351

P
paměť dlouhodobá 115
– krátkodobá 115
– procedurální 115
paradox kauzální 104
plánování cesty 123
– reaktivní 115, 118
– – hierarchické 119
platforma multiagentní A-globe 468
podmínka globální 153
– implicitní 165
– omezující 148
podoba s člověkem 34
pole elektrod MEA 515
polomnožina 97
pomůcky kompenzační 499
porozumění přirozenému jazyku 272
poruchy kognitivní 500
– mobility 500
– senzorické 500
posloupnost stabilní 99
posthumanismus 72
povědomí vzdálené 451
pravidlo navigační 123
– rozhodovací 117
Pražský závislostní korpus 273
prezence 439
prezence společná 439
princip prvního neúspěchu 158
– – úspěchu 158
problém nerozhodnutelný 77
– optimalizační, splňování podmínek 148
– splnitelnosti 251
– splňování podmínek 147
– strukturovaný dobře 198
– – špatně 198
– zastavení 84
programování logické, s podmínkami 171
prohledávací strom 156
– – s diskrepancemi 159
– – s navracením 155
prostor konfigurační 462
prozodie 304
–, generátor 301
předcházení kolizím 437
přechod fázový 252
překlad strojový statistický 276
přeposílání zpráv 450
přístup znalostní 285
pyramida populační 497

R
rámec sémantický 297
RANSAC 343
rekonfigurovatelnost systému 445
rekonstrukce tvaru 328
reprezentace sémantická 297
robot 32, 47, 431
– autonomní 270
– Emil 53
– expresivní 32
– inteligentní 379
– kognitivní 32
– Pearl 507
– průzkumný 411
robotika kolaborativní 432
– servisní 405
rozložení mocninné 256
R. U. R. 45

Ř
řada pseudonáhodná 187
řeč, rozpoznávání 284
–, – automatické 281
–, syntéza 299
–, – z textu 299
řez na přirozených číslech 99
– obzorný 99
– zlatý 174

S
satisficing 199
senzor taktilní 348
shlukování omezené difuzí 189
simulátor robotický Gazebo 466
– – Stage 466
– světa 113
síť neuronová 67
– sémantická 297
síťové rozhraní Player 466
situace epizodická 37
sledování polohy 455
SNARC 67
snímač taktilní 348
soutěže robotů 417
společenství robotů 38
strategie enumerační 167
– kroková 167
– půlení domény 168
stroj inteligentní 220
– Turingův 54, 77
– – interaktivní 80
– – neuniformní 84
– – relativistický 83
– – s orákulem 83
– – s rádcem 84
strom rozhodovací 119
stvoření umělá 374
symetrie 177
synergie 220
syntéza konkatenační 305
– řeči korpusově orientovaná 309
systém dialogový 284
– ekogramatik 69
– gramatik formálních 69
– hodnoticí 501
– jisticí 501
– kognitivní 76
– kompenzační 501
– multiagentní 433
– multirobotický 441
– simulační MAST 453
– telematický PELOTE 474
– výpočetní interaktivní evoluční 88

T
tagování 302
technologie asistivní 501
teleprezence 406, 440
teleprezence vizuální 412
teorie afordancí 132
– množství dynamická 96
– mysli societní 67
– proudění výpočetní 189
test Turingův 55
teze emergentistická 92
– – zobecněná 95
– Churchova–Turingova 77
– – rozšířená 87
topologie komunikační 445
transduktor interaktivní konečný 79
třída 97
– přehledná 99
tým hybridní 439

U
učení posilované 263
údolí odpornosti 136
úchopná hlavice antropomorfní 361
úkol olbřímí 81
umění fraktální 182
– interaktivní 63

V
věda kognitivní 65
vidění počítačové 328
videookulografie 504
vozidlo inteligentní 378
– robotické 418
vyhýbání se překážkám 267
výpočty neuniformní 83
vypravěčství virtuální 129

Z
zdroj znalostí 311
– –, modelování 311
znalosti explicitní 207
–, externalizace 209
– implicitní 207
–, internalizace 209
–, kombinace 209
– neformální 207
–, socializace 209
zpracování přirozeného jazyka 300

Ž
život umělý 59, 383

# Umělá inteligence (5)

prof. Ing. Vladimír Mařík, DrSc., prof. RNDr. Olga Štěpánková, CSc.,
doc. Ing. Jiří Lažanský, CSc.,
doc. RNDr. Roman Barták, Ph.D., Dr. Diego Bendersky, Mgr. Cyril Brom, RNDr. Jan Cuřín, Ph.D., prof. MUDr. I. Čapov, CSc., MUDr. J. Doležel, Ing. Marcela Fejtová, doc. RNDr. Jan Hajič, Dr., prof. PhDr. Eva Hajičová, DrSc., doc. Ing. Ivan M. Havel, CSc., Ph.D., Mgr. Petr Hoffmann, Mgr. Jana Horáková, Ph.D., Ing. Jakub Hrabec, prof. Dr. Pablo Miguel Jacovkis, prof. RNDr. Jozef Kelemen, DrSc., Ing. Pavel Kordík, Ph.D., Ing. Tomáš Krajník, Mgr. Ondřej Kučera, RNDr. Miroslav Kulich, Ph.D., Dr. José Sellés-Martínez, Ing. Roman Mázl, doc. Ing. Pavel Nahodil, CSc. Ing. Tomáš Neužil, Ing. Petr Novák, Ing. Tomáš Pajdla Ph.D., Ing. Jiří Pavlíček, doc. Dr. Ing. Michal Pěchouček, MSc., Ing. František Pikeš, Ing. Libor Přeučil, CSc., prof. Ing. Josef Psutka, CSc., Ing. Milan Rollo, Ing. Jan Romportl, Dr. Juan Miguel Santos, Mgr. Vít Šisler, Ing. David Šišlák, doc. Ing. Miroslav Šnorek, CSc., prof. Ing. František Šolc, CSc., prof. Ing. Michael Valášek, DrSc., MUDr. Lenka Veverková, Ph.D., Mgr. Barbora Vidová Hladká, Ph.D., MUDr. Petr Vlček, Ph.D., doc. Ing. Jaromír Volf, DrSc., Ing. Pavel Vrba, Ph.D., prof. Dr. Kevin Warwick, prof. RNDr. Jiří Wiedermann, DrSc., doc. Ing. Zdeněk Zdráhal, CSc., doc. Ing. Luděk Žalud, Ph.D., Ing. Filip Železný, Ph.D.

Vydalo Nakladatelství Academia,
Středisko společných činností AVČR, v.v.i.
Vodičkova 40, 110 00 Praha 1

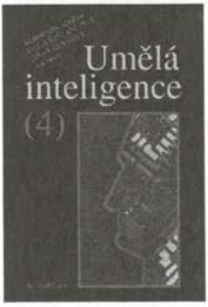
Vazbu navrhla Jolana Malátová, pro pátý díl upravil Oleg Man
Redaktoři publikace Aleš Baďura, Václav Havlíček
Technické zpracování, sazba, grafická úprava a zlom Václav Havlíček
Barevnou přílohu graficky upravil Oleg Man
Tisk SERIFA, s. r. o., Jinonická 80, Praha 5

Vydání 1., Praha 2007

Ediční číslo 10 397
ISBN 978-80-200-1470-2
ISBN 80-200-0502-1 (soubor)
ISBN 80-200-0496-3 (1. díl)
ISBN 80-200-0504-8 (2. díl)
ISBN 80-200-0472-6 (3. díl)
ISBN 80-200-1044-0 (4. díl)

Knihy Nakladatelství Academia můžete zakoupit též na
www.academiaknihy.cz
www.academiabooks.com

## ACADEMIA vám dále nabízí:




Vladimír Mařík, Olga Štěpánková,
Jiří Lažanský a kolektiv

**UMĚLÁ INTELIGENCE (1) – (4)**

Umělá inteligence (UI) je hraniční obor, ve kterém se setkávají počítačové vědy a technika s psychologií a filosofií. Předchozí čtyři svazky této publikace se soustřeďují na problematiku UI z technického hlediska, které si klade za cíl rozšířit třídu úloh řešitelných pomocí počítače směrem k úlohám, které se doposud považovaly za doménu lidí (od hraní tak náročných her, jako jsou šachy, přes překlad textu v přirozeném jazyce až ke stanovení a srozumitelnému zdůvodnění diagnózy složitého případu). Společným rysem různorodých problémů řešených v rámci UI je práce se znalostmi charakterizujícími studovanou oblast.

Proto je první kniha věnována obecným metodám reprezentace, využití a získávání znalostí, všímá si i kombinatorické náročnosti některých předkládaných metod (stavový prostor a plánování) a upozorňuje na význam heuristického přístupu. Vedle kapitol o dnes již klasických metodách reprezentace znalostí, jako jsou matematická logika a sémantické sítě, zde nalezneme i přehled formalismů, které se pokouší zachytit přirozený úsudek (kvalitativní modelování). Schopnost rozpoznávání a učení patří mezi nejvlastnější charakteristiky živých organismů. Problematice návrhu algoritmu pro simulaci těchto schopností s využitím klasické počítačové architektury i pomocí neuronových sítích jsou věnovány samostatné kapitoly.

Cílem druhého dílu je poskytnout čtenáři přehled o řadě nástrojů i technik, které byly vyvinuty pro řešení speciálních aplikačních úloh. Samozřejmě není možné začít jinak než kapitolami o expertních systémech, jejich základních rysech, nástrojích pro jejich vytváření a postupech pro zacházení s neurčitostí. Samostatná kapitola je věnována znalostnímu inženýrství a praktickým problémům získávání a využívání znalostí. Kapitoly o počítačovém vidění a zpracování přirozeného jazyka či zvukových signálů naznačují, jaké problémy je třeba řešit, aby počítačové systémy získaly schopnost těžit z tak významných zdrojů informací, jakými jsou zrakové a sluchové vjemy pro člověka. Úlohy, se kterými se obor umělá inteligence setkává, jsou mnohdy tak složité, že je nelze zvládnout klasickými „centralizovanými" metodami. Zajímavé alternativní přístupy nabízejí distribuované systémy UI. Programovací nástroje pro UI mají velmi specifické nároky. Pro řešení praktických plánovacích úloh je atraktivní použití logického programování s omezujícími podmínkami vysvětlené v kapitole o jazycích pro UI.

Úvodní kapitola „Přirozené a umělé myšlení" třetí knihy klade řadu provokativních otázek o podstatě mysli i vědomí a hledá principiální možnosti a meze UI. Hlavním tématem knihy jsou netradiční přístupy k vytváření SW systémů inspirované moderní biologií. Kapitola „Umělý život" názorně ukazuje, že počítačové programy s vhodně

definovanými pravidly vzájemné interakce vykazuji při dlouhodobém vzájemném působení vlastnosti připomínající chování živých systémů. Navazující kapitola „Evoluční výpočetní techniky" vysvětluje, jak a za jakých podmínek lze princip evoluce využít pro hledání suboptimálního řešení kombinatoricky náročných úloh. Kapitola „Reaktivní agenti" se zabývá studiem společenství jednoduchých agentů situovaných v reálném fyzikálním světě a ukazuje, že jejich interakce může vést ke vzniku nového překvapivého a nepředvídaného chování celého společenství. Kapitola o multiagentních systémech vysvětluje ty principy architektury intencionálních agentů, které umožňují agentům vytvářet, udržovat a využívat vlastní model světa potřebný pro rozhodování. Jeden z často používaných matematických modelů účelového rozhodování za neurčitosti vysvětluje kapitola „Základní modely teorie rozhodování a teorie her". Závěrečná kapitola „Teorie složitosti a úlohy UI" ilustruje na běžných úlohách UI nejpodstatnější pojmy teorie složitosti a zmiňuje se o některých jejích zásadních důsledcích, např. pro strojové učení. V závěru jsou popsány některé praktické výsledky dosažené metodami UI v tak významných aplikačních oborech, jakými jsou například medicína, přírodní vědy a technika.

Svazek Umělá inteligence (4) analyzuje pojem informace a dále rozvíjí témata související s otázkami reprezentace informací či znalostí, nabízí důkladný pohled na problematiku zacházení s neurčitostí a nepřesností v datech, pravidlech i výsledcích inteligentního zpracování dat.

Úvodní čtyři kapitoly se zabývají prací s neurčitými a nejistými informacemi. Nabízejí obecný rámec pro formalizaci neurčitosti, kdy výchozím pojmem je možnost. Hledají způsob jak s výhodou využít pokusy s nejistým výsledkem pro řešení složitých úloh. Analyzují teoretické otázky kalkulu pro práci s neurčitými údaji a vše je vhodně doplněno praktickým pohledem, který shrnuje současný stav těchto aplikací a naznačuje bariéry, s nimiž je třeba se vyrovnat.

Druhá část svazku tvořená kapitolami 5–9 je věnována oblasti často společně označované jako „softcomputing" vzhledem k tomu, že její výchozí inspiraci tvoří živá příroda. Tato část se věnuje především problematice volby selekčních operátorů, představuje speciální evoluční techniku schopnou provádět globální optimalizaci a dále přináší nejzajímavější výsledky z oblasti neuronových sítí, a to jak úvod do problematiky, tak teoretický rámec.

Třetí část knihy je zaměřena na praktické otázky práce s daty a znalostmi. Zde se čtenář seznámí s nejčastějším způsobem skladování dat – databázemi a datovými sklady, a též metodikou získávání znalostí z rozsáhlých dat. Dozví se o využívání znalostí ve vysoce distribuovaných systémech pro průmyslovou výrobu.

Publikace je určena studentům, doktorandům i všem zájemcům o obor s bakalářským vzděláním.

*1. díl – 264 str., 225Kč, ISBN 80-200-0496-3*
*2. díl – 374 str., 285 Kč, 2. vyd., ISBN 80-200-0504-8*
*3. díl – 329 str., 225 Kč, ISBN 80-200-0472-6*
*4. díl – 476 str., 185 Kč, ISBN 80-200-1044-0*

Josef Psutka, Jindřich Matoušek
Luděk Müller, Vlasta Radová

## MLUVÍME S POČÍTAČEM ČESKY

Od vydání knižní publikace „Komunikace s počítačem mluvenou řečí" uplynulo již více než deset let a vědní disciplína zabývající se problematikou analýzy, syntézy a rozpoznávání mluvené řeči, tedy obecně problematikou řečových technologií, zaznamenala zřetelný pokrok. To je důvodem přípravy a vydání nové, zcela přepracované a výrazně rozšířené publikace s názvem „Mluvíme s počítačem česky", kterou právě držíte v rukou.

Kniha vznikla opět na autorském pracovišti Katedry kybernetiky, Fakulty aplikovaných věd, Západočeské univerzity v Plzni, nicméně na její přípravě se již podílel širší autorský kolektiv, a to i z ciziny. Záměrem bylo využít širokého know-how domovského pracoviště ke zmapování pokroků a nejnovějších trendů v oblasti rozvoje řečových technologií. Kniha odráží rozsáhlé znalosti a zkušenosti, které získali autoři i jejich spolupracovníci intenzivní mezinárodní spoluprací podpořenou řešením mnoha zahraničních a tuzemských projektů, ale též podílem na četných aplikacích, ve kterých zajišťovali, společně se spolupracujícími firmami, transfer těchto poznatků do společenské a průmyslové praxe.

Publikace „Mluvíme s počítačem česky" podává ucelený pohled na současné řešení podstatných problémů, které se objevují v systémech hlasové komunikace s počítačem a klade specifický důraz na české národní prostředí. Kniha je určena zejména studentům, doktorandům a absolventům vysokých škol technického a přírodovědného zaměření, ale i všem ostatním zájemcům, kteří chtějí porozumět pokročilým technikám analýzy, syntézy a rozpoznávání mluvené řeči a osvojit si základní metody návrhu a konstrukce hlasových dialogových systémů. S knihou by se měli seznámit i inženýři, počítačoví odborníci a návrháři systémů, aby mohli pružně využít atraktivní technologie hlasové komunikace s počítačem v nových projektech a aplikacích.

*752 str., obr. v textu, 495 Kč*
*ISBN 80-200-1309-1*

Milan Meloun, Jiří Militký

## KOMPEDIUM STATISTICKÉHO ZPRACOVÁNÍ DAT

(2. přepracované a rozšířené vydání)

Nakladatelství Academia připravilo nové vydání publikace, jež je zaměřena na statistickou analýzu experimentálních dat. V každé kapitole je uveden postup analýzy dat, vzorový příklad s komentovaným výstupem z počítače a vzorem protokolu. Úlohy jsou podle zaměření dále tříděny do několika oddílů, jimiž jsou: úlohy z laboratoří technické kontroly kvality (chemická data), technické kontroly strojírenských závodů (fyzikální data), hygienických stanic, zdravotních ústavů a laboratoří klinické chemie (biochemická data), výrobců léčiv (farmakologická data), potravinářské, zemědělské a vodohospodářské inspekce (potravinářská a zemědělská data), kontroly životního prostředí a energetiky (enviromentální data), hutní analýzy a zpracování rud (hutní a mineralogická data) a konečně managementu a ekonomie (ekonomická data). Vzorový software a data k úlohám jsou na doprovodném kompaktním disku, a to ve formátu Excel, ASCII atd.

Obsahem úloh je teorie chyb a nejistot laboratorních měření, průzkumová analýza jednorozměrných dat, klasické, robustní a adaptivní odhady polohy, rozptýlení a tvaru, testování statistických hypotéz, analýza vícerozměrných dat – metoda hlavních komponent, klasifikace objektů a shluková analýza, diskriminační analýza a kanonická korelace – analýza rozptylu, prosté a vícerozměrné lineární regresní modely, regresní diagnostika, lineární a nelineární kalibrace, korelace a korelační modely, nelineární regresní modely, interpolace, aproximace a vyhlazování, regulační diagramy počítačové kontroly a řízení jakosti.

Publikace je vítanou pomůckou pro studenty vysokých škol technického, přírodovědného, ale také ekonomického směru. Je dále určena pracovníkům výzkumných ústavů, podnikových a kontrolních laboratoří a zkušeben ke kontrole kvality.

*984 str., obr. v textu, CD s daty k příkladům, 750 Kč*
*ISBN 80-200-1396-2*

Milan Mareš

## SLOVA, KTERÁ SE HODÍ
### ANEB JAK SI POPOVÍDAT O MATEMATICE, KYBERNETICE A INFORMATICE

Kniha je určena čtenářům, kteří se nechtějí učit matematiku, zajímá je ale, k čemu je lidem dobrá, jakou mezeru v našich znalostech chtěli zaplnit ti, kdo ji vytvářeli a vytvářejí a jakými cestami při tom ve svém uvažování postupovali.

Text je rozdělen do dvou částí, z nichž první je věnována klasické matematice. Jsou v ní kapitoly o slavných matematických problémech, o tom, jak matematické vzdělání přispívá k celkové kultuře, o slavných amatérech i o ženách v matematice. Další kapitoly vyprávějí o pojmech, které už jistě slyšeli všichni – například číslo, náhoda, matematický model, logika, nekonečno, křivý prostor. Ve druhé části jsou kapitoly věnovány pojmům a událostem, které přinesla kybernetika a informatika – vznik a pravěk počítačů i počítačových sítí, umělá inteligence, počítačové modelování.

Poměrně hodně prostoru je věnováno historii matematického vidění světa a lidem, kteří k matematice nebo informatice přispěli něčím významným, měli zajímavé a někdy i napínavé osudy a hlavně v matematickém a informatickém světě našli smysl života.

*352 str., obrazová příloha, 249 Kč*
*ISBN 80-200-1445-4*

---

Knihy si můžete objednat na adrese: ACADEMIA – expedice,
Rozvojová 135, 165 02 Praha 6
tel.: 296 780 510; e-mail: expedice@academia.cz
www.academia.cz

---

# ACADEMIA připravuje:

Nakladatelství Academia vám představuje novou edici GERSTNER, v níž připravuje publikace na vysoké odborné úrovni, podávané způsobem přístupným nejen pro naprosté začátečníky v oboru, ale též i studenty středních škol a gymnázií, a také zájemce o popisované jevy.

Jako první vychází kniha:

Martinus Veltman

## FAKTA A ZÁHADY VE FYZICE ELEMENTÁRNÍCH ČÁSTIC

Z anglického originálu
Facts and Mysteries in Elementary Particle Physics

přeložili RNDr. Aleš Cieplý, CSc., a RNDr. Jiří Rákosník, CSc.

Po více než dva tisíce let lidé usilují o poznání nejmenších částeček hmoty, základních stavebních kamenů, z nichž je stvořen náš svět. Kniha nositele Nobelovy ceny holandského fyzika M. Veltmana přibližuje čtenáři klikatou cestu, kterou moderní věda při odkrývání záhad nitra hmoty urazila – od objevu elektronu na konci 19. století přes formulování základů kvantové teorie až po objev kvarků a pochopení jejich interakcí.

Autor čtenáři předkládá nejnovější poznatky dosažené v oboru, který využívá prostředky stojící na hranici našich technických možností a který lze právem považovat za vrchol lidského bádání. Pozoruhodný výklad očitého svědka a přímého účastníka některých z největších objevů 20. století je veden na úrovni přístupné studentům vyšších ročníků gymnázií. Srozumitelné vysvětlení základů kvantové mechaniky a teorie relativity je východiskem k pochopení zákonů, kterými se řídí částicová fyzika a jejichž projevy jsou zkoumány v urychlovačích, srážečích a detektorech v ohromných laboratořích jako jsou CERN, SLAC nebo Fermilab.

Kniha není učebnicí fyziky, ale spíše poutavým příběhem o lidské touze po odhalení tajemství přírody, o dosažených úspěších i trpkých zklamáních. Svědčí o nich celá řada medailonů věnovaných lidem, kteří se na převratných objevech významnou měrou podíleli. Autorovy osobní postřehy a vzpomínky doplněné fotografiemi představují živý pohled na pozoruhodný svět fyziky elementárních částic a kus historie tohoto vědního oboru.

*288 str., celobarevné, 349 Kč*
*ISBN 978-80-200-1500-6*

## NAŠE KNIHKUPECTVÍ JSOU TU PRO VÁS

● **PRAHA**

**Knihkupectví Academia**
Václavské nám. 34, 110 00 Praha 1
tel.: 224 223 511–13
e-mail: knihy.vaclavskenam@academia.cz

**Knihkupectví Academia**
Národní tř. 7, 110 00 Praha 1
tel.: 224 240 547
e-mail: knihy.narodni@academia.cz

**Knihkupectví Academia**
Na Florenci 3, 110 00 Praha 1
tel.: 224 814 621
e-mail: knihy.naflorenci@academia.cz

● **BRNO**

**Knihkupectví Academia**
náměstí Svobody 13, 602 00 Brno
tel.: 542 217 954–56
e-mail: knihy.brno@academia.cz

● **OSTRAVA**

**Knihkupectví Academia**
Zámecká 2, 702 00 Ostrava
tel.: 596 114 578, 596 114 580
e-mail: knihy.ostrava@academia.cz

**ACADEMIA – sklad a expedice**
Rozvojová 135, 165 02 Praha 6
tel.: 296 780 510, e-mail: expedice@academia.cz

www.academia.cz

Pátý díl Umělé inteligence nese podtitul „Na cestě k Umělému (tělu, mozku či duši)“. Autoři jednotlivých kapitol se přístupnou formou zamýšlejí nad současným stavem i nad možnými směry vývoje teoretických disciplín i technologií potřebných pro vytváření systémů umělé inteligence. Jednotlivé kapitoly, mezi jejichž autory je i řada věhlasných zahraničních odborníků (např. K. Warwick), shrnují aktuální stav poznání nejen na úrovni filosofických úvah, ale představují i přírodou inspirovaná technická řešení, která otvírají nový pohled na základy učících se, kognitivních a robotických systémů včetně ukázek konkrétních existujících řešení. Publikace není zaměřena pouze na odbornou technickou veřejnost, ale je určena všem zájemcům o moderní informační technologie s humanitním či přírodovědným zaměřením. Kniha láká k zamyšlení a k diskusi.

www.academia.cz

ISBN 978-80-200-1470-2

9 788020 014702