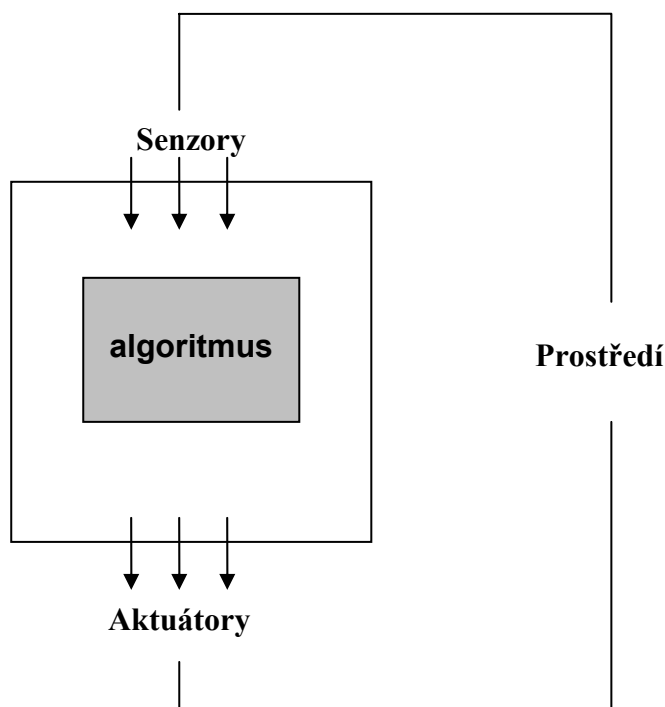


9 Komunikace a kooperace

Komunikace se jako jeden z principů objevuje v umělé inteligenci až v druhé polovině 80. let. V roce 1986 uveřejňuje M. Minsky knihu *The Society of Mind*, ve které podává představu mysli jako systému vzájemně interagujících agentů, kteří monitorují signály vysílané jinými agenty a vykonávají jednoduché akce. V téže době se začíná rovněž mluvit o „nové“ umělé inteligenci. Na rozdíl od „tradiční“ umělé inteligence, která se opírá o vnitřní (obvykle symbolický) model světa, je „nová“ umělá inteligence inteligencí bez reprezentace znalostí a bez uvažování. Předním představitelem tohoto směru je R. Brooks z MIT, podle kterého vzniká inteligence ze vzájemné interakce jednoduchých, takzvaných *reaktivních* agentů.

Odpovědí klasické umělé inteligence jsou multiagentní systémy, tvořené deliberativními agenty. Tyto agenty využívají znalosti pro usuzování a jsou schopny „vědomě“ kooperovat a komunikovat.

V této kapitole se tedy budeme zabývat agenty. Tímto názvem budeme označovat systémy, které jsou schopny „vnímat“ prostředí pomocí svých senzorů a provádět akce prostřednictvím svých aktuátorů (akčních členů) (Obr. 9.1). Jednotlivé agenty se liší v tom, jakým způsobem se přiřazují akce vstupním vjemům, tedy v tom, jak vypadá algoritmus agenta. Nás budou zajímat dva typy agentů: reaktivní agenty a deliberativní agenty.



Obr. 9.1 Schéma agenta

9.1 Reaktivní agenty

Def. 9.1: Reaktivní agent působící v prostředí E je dán čtveřicí $\{P, A, \text{vjem}, \text{akce}\}$

kde E je množina stavů prostředí

P je množina vjemů agenta

A je množina akcí

vjem: $E \rightarrow P$

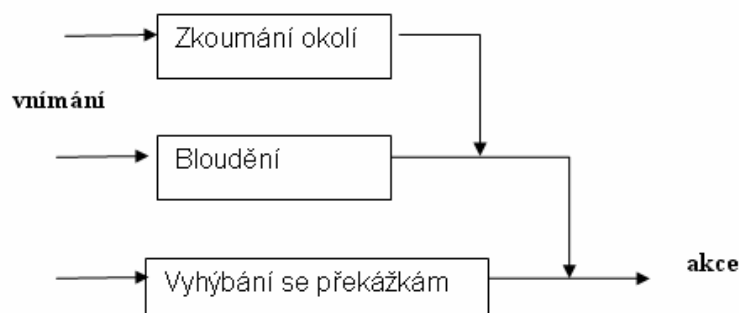
akce: $P \rightarrow A$

Základem reaktivního agenta je tzv. subsumpční architektura, která v protikladu s klasickou (symbolickou) umělou inteligencí vychází z následujících principů:

- tělesnost (embodiment) – agenty mají tělo se senzory a aktuátory
- situovanost (situatedness) – agenty jsou situovány v reálném světě, se kterým interagují, a který ovlivňuje jejich chování
- inteligence – jako výsledek střetu agenta s okolím a adekvátní reakce na podněty (stimulus \rightarrow response); tyto reakce nepotřebují centrální reprezentaci cílů a plánů
- emergence – inteligence systému vzniká samovolně jeho interakcí s prostředím a také vzájemnou interakcí komponent systému

Čelný představitel tohoto přístupu, R. Brooks, tyto principy formuloval v pracích „Intelligence bez reprezentace znalostí“ nebo „Intelligence bez usuzování“.

Subsumpční architektura znamená dekompozici agenta dle jeho aktivit do vrstev. Jednotlivé vrstvy spolu „soupeří“ o řízení agenta. Nejspodnější vrstva má přednost, pokud nepracuje, nastupuje další vrstva atd. Agent je tedy vytvářen zdola nahoru, postupně se tak rozšiřuje jeho funkcionalita. Jednotlivé vrstvy interagují na základě potlačení (suppression) na úrovni vstupů do jednotlivých vrstev agenta (vstup do nižší vrstvy je nahrazen signálem z vyšší vrstvy) a zabránění (inhibition) na úrovni výstupů z jednotlivých vrstev agenta (výstup z vyšší vrstvy zabráni výstupu z nižší vrstvy). Obr. 9.2 ukazuje subsumpční architekturu robota, který prozkoumává své okolí. Tato činnost má nejnižší prioritu, vyšší prioritu má bloudění a nejvyšší prioritu má vyhýbání se překážkám.



Obr. 9.2 Subsumpční architektura

Reaktivní agenty mohou simulovat chování jednoduchých organismů či živočichů. Tak např. H. Connell vytvořil jednoduchý robotický systém, který

- se pohybuje vzhůru jako reakce na gravitaci
- se pohybuje do tmy jako reakce na světlo

To odpovídá chování pobřežního slimáka, který se živí řasami rostoucími mezi skalami ve vodě i na vzduchu. Na dně se pohybuje do tmy skalních štěrbin a v nich se plazí směrem vzhůru. Je-li sluneční světlo příliš silné, zůstane pod hladinou, jinak vyleze a zůstane mezi skalami, kde hledá potravu (Kelemen, 1994).

Jiný příklad reaktivního agenta, tentokrát na mechanické bázi vidíme na Obr. 9.3 . Z hlediska subsumpční architektury je beruška tvořena dvěma vrstvami: zkoumání okolí (pohybem vpřed) a vyhýbání se překážkám (hraně stolu).



Obr. 9.3 Mechanická beruška

Jako poslední příklad si uveďme tzv. **Braitenbergova vozidla** (popis dle Kelemen, 2001). Jde o jednoduché autonomní systémy vybavené senzory citlivými na světlo a koly poháněnými každé vlastním motorem. Senzory jsou pak přímo propojeny s koly. Nejjednodušší vozidlo obsahuje jeden senzor a jedno kolo a řídí se těmito pravidly: růst intenzity světla vede k rychlejšímu pohybu, pokles intenzity světla vede k pomalejšímu pohybu, tma vede k zastavení. Složitější vozidlo má dva senzory a dvě kola, je tedy schopno zatáčet od zdroje světla případně ke zdroji světla podle toho jestli „pravý“ senzor připojíme k „pravému“ nebo „levému“ kolu.

Chování prvního vozidla tak napodobuje chování živočicha, který prchá před světlem do tmavého úkrytu, druhé vozidlo je opět schopno vyhýbat se resp. vyhledávat světlo, tentokrát ale sofistikovaněji. Více vozidel současně pak vykazuje navenek inteligentní chování: budou kličkovat od zdrojů světla, zrychlovat a zpomalovat, dokud nenajdou temný kout kde si odpočinou.

9.2 Umělý život

Na bázi reaktivity se rozvinula disciplína umělé inteligence nazývaná umělý život (artificial life). Cílem je modelovat resp. napodobit chování živých organismů, které můžeme chápat jako racionální. Základními tezemi umělého života jsou (dle Czontó, 2001):

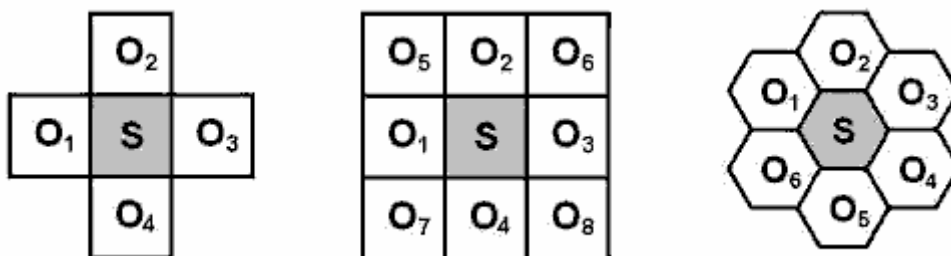
- podstatou života je informace, ne materiální forma,
- život vyžaduje jistou míru složitosti,
- informace má v živých organizmech dvojí podobu: genotyp slouží k rozmnožování a fenotyp slouží jako podklad pro vytváření struktury nového jedince,
- prostředkem vývoje směrem ke složitějším a dokonalejším strukturám (evoluce) je samoreprodukce, mutace a selekce,
- syntetický proces probíhá zdola nahoru od elementárních primitiv ke složitějším strukturám,
- pro dynamiku elementárních primitiv je charakteristická vysoká míra paralelismu,
- vzájemné lokální působení primitiv vyvolá na globální úrovni nové fenomény (emergence),
- jednou z podmínek emergence je nelineární chování elementárních primitiv (neplatí princip superpozice kdy chování celku lze odvodit „nasčítáním“ dílčích chování jednotlivých primitiv).

9.2.1 Celulární automat

Celulární automat (CA) je dynamický systém, diskretní v prostoru a čase. Je tvořený pravidelnou strukturou buněk v N -rozměrném prostoru (nejčastěji $N=2$, tzv. 2D CA, kde buňky tvoří čtvercovou mřížku). Každá buňka se může nacházet v jednom z K možných stavů (často $K=2$, tzn. živá nebo mrtvá buňka). Hodnoty stavů buněk v dalším kroku (čase $t+1$) se počítají paralelně na základě lokální přechodové funkce stejné pro všechny buňky):

$$S(t+1) = f(S(t), O_1(t), O_2(t), O_3(t), \dots)$$

kde O_i je okolí uvažované buňky. Obr. 9.4 ukazuje různá okolí buňky S ; zleva doprava to jsou neumannovské, úplné a šestiúhelníkové.



Obr. 9.4 Okolí buňky S

Pro celulární automat je tedy typický:

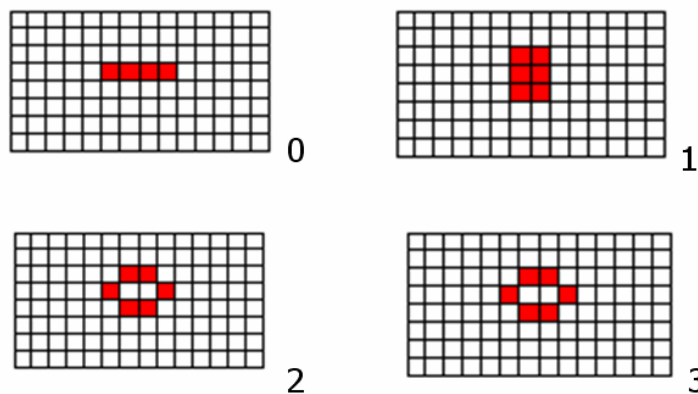
- **paralelismus** – výpočet nových hodnot stavů všech buněk probíhá současně
- **lokalita** – nový stav nějaké buňky závisí pouze na původním stavu buňky a na stavech buněk v okolí
- **homogenita** – pro všechny buňky existuje stejná přechodová funkce

9.2.1.1 Life

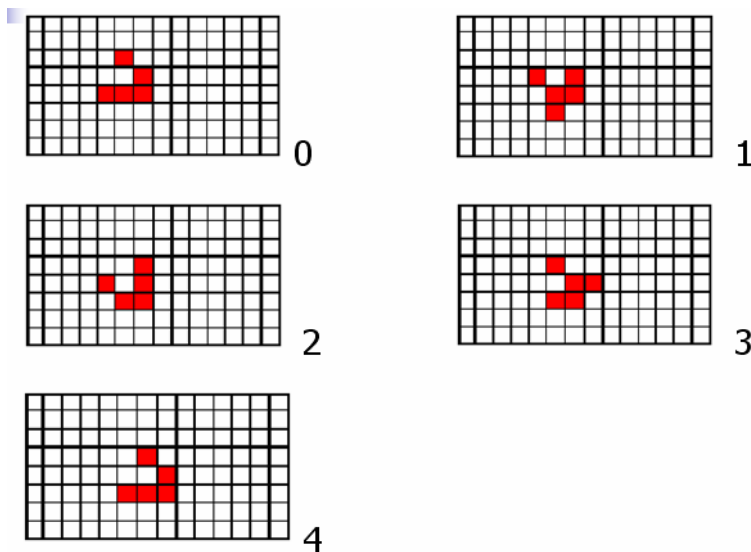
Life je příklad dvourozměrného celulárního automatu pracujícího s úplným okolím, přičemž každá buňka může být v jednom ze dvou stavů. Tento automat používá následující přechodová pravidla:

- **zrod** - v okolí prázdného políčka jsou právě tři buňky ("trojpohlavní" rozmnožování)
- **přežití** - v okolí buňky jsou dvě nebo tři další buňky
- **uhynutí** - v okolí buňky je 0, 1, 4, 5, 6, 7 nebo 8 dalších buněk

Tato sada pravidel poskytovala přijatelnou biologickou interpretaci (zánik osamocené buňky nebo buňky v přehuštěné populaci). Obr. 9.5 ukazuje, strukturu buněk, která se po 3 krocích ustálí, Obr. 9.6 pak ukazuje tzv. kluzák, neboli strukturu, která se pohybuje směrem „vpravo dolů“.

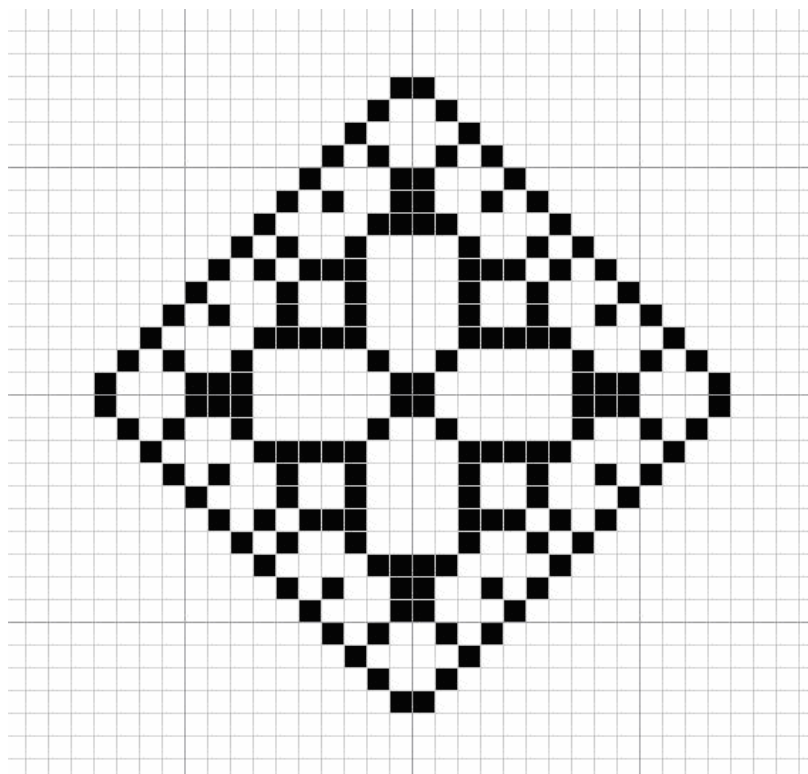


Obr. 9.5 Stabilní struktura ve hře Life



Obr. 9.6 Kluzák

Příkladem obrazce, u kterého by asi málokdo čekal, že vznikne samovolně ze čtyř buněk uprostřed, jen na základě jednoduchých pravidel, je tzv. Perský koberec (Obr. 9.7).

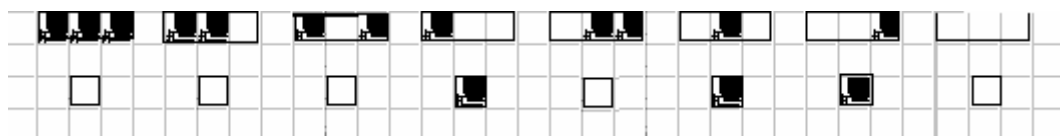


Obr. 9.7 Perský koberec

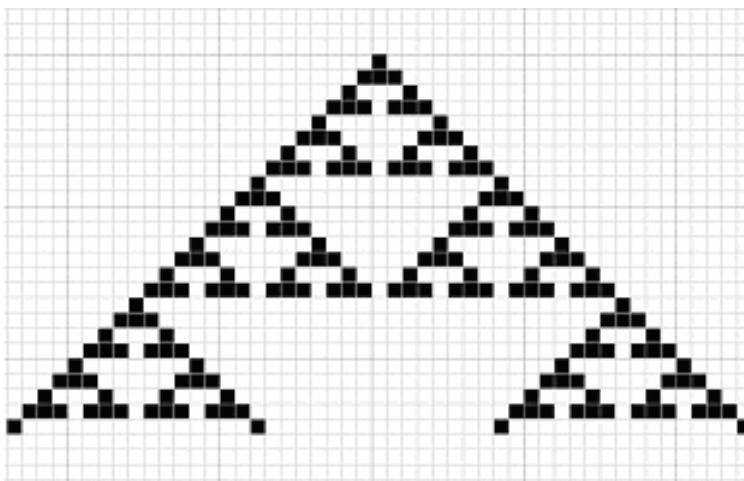
9.2.1.2 Wolframův jednorozměrný CA

V nejjednodušším případě je jednorozměrný CA tvořen buňkami, které se nacházejí v jednom ze dvou stavů (živá, mrtvá), přičemž pro výpočet hodnoty přechodové funkce nějaké buňky se obvykle uvažuje pouze tato buňka a její dva přímí sousedi (buňka napravo a buňka nalevo). Nová hodnota buňky tedy závisí na trojici (binárních) hodnot, tedy na osmi různých kombinacích živých a mrtvých buněk (2^3). Každé osmici můžeme přiřadit sadu pravidel, která říkají, kdy je nová hodnota uvažované buňky „živá“ a kdy „mrtvá“. Je tedy celkem k dispozici $2^8 = 256$ sad pravidel. Tyto sady se číslují dekadickým ekvivalentem binárně interpretovaného vektoru nových hodnot.

Výhodou jednorozměrných CA je poměrně malý počet možných pravidel a názorná reprezentace posloupnosti generací buněk v jednotlivých řádcích pod sebou. Obr. 9.8 ukazuje sadu 22, která (s jednou buňkou uprostřed řádku) generuje tzv. Sierpinského trojúhelníky (Obr. 9.9).



Obr. 9.8 Sada 22 = $0 \cdot 2^7 + 0 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0$



Obr. 9.9 Szierpinského trojúhelníky

9.2.2 Reynoldsův model shlukování ptáků

Dalším příkladem koncepce celulárních automatů jakožto společenství reaktivních agentů je simulace chování hejna ptáků, realizovaná C. Reynoldsem v roce 1987 (popis dle Czontó, 2001). Reynolds vytvořil počítačový model virtuálních jedinců nazvaných boidy. Pohyb boidů v prostoru se řídil třemi jednoduchými pravidly:

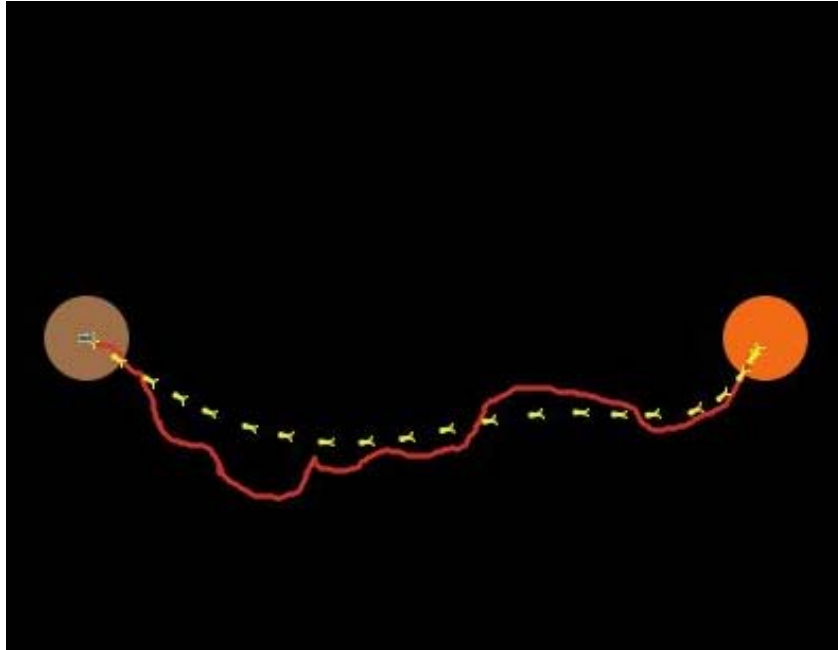
- **shromazďování** – jedinec směřuje k těžišti svých susedů,
- **sladění rychlosti** – jedinec přizpůsobuje svoji rychlost rychlosti susedů,
- **vyhýbání se kolizi** – jedinec opouští své místo ve chvíli, kdy se neúměrně přiblíží k některému ze svých susedů.

Virtuální hejno řídicí se těmito pravidly drží pohromadě, ať je trajektorie těžiště hejna jakkoli klikatá. V případě setkání s překážkou se hejno rozdělí a za překážkou zase spojí. Simulace byla tak věrná, že ornitologové přijali hypotézu, že chování živého hejna se řídí analogickými pravidly. Boidy z tohoto modelu byly použity v animacích filmu *Batman se vrací*.

9.2.3 Umělí mravenci

Jiným populárním příkladem společenství reaktivních agentů, které simuluje reálné živočichy je kolonie mravenců (ant colony). K prvním modelům patří Langtonovi virtuální mravenci (vanti) z roku 1986 (popis dle Czontó, 2001). Vant je organizmus tvaru V, který se pohybuje po mřížce ve směru svého zahrocení. Když se dostal na prázdné políčko mřížky, pokračoval v původním směru. Když se dostal na modré políčko, zahrnul vpravo a obarvil políčko na žlutě. Když se dostal na žluté políčko, zahrnul vlevo a obarvil políčko na modro. Tímto způsobem za sebou nechává feromonovou stopu. Uvedená pravidla produkovala výsledné chování podobné chování sociálního hmyzu. Vanty zpočátku náhodně kličkovaly, pak začaly interagovat až vytvořily spirálovou stopu.

Jiný příklad simulace chování mravenců (optimalizace cesty z mraveniště k potravě) vidíme na Obr. 9.10. Tuto (a další simulace založené na společenstvích reaktivních agentů) najdeme v systému NetLogo.



Obr. 9.10 Kolonie mravenců (systém NetLogo)

9.2.4 Hejno

Hejno (Swarm) je softwarové prostředí pro multiagentní simulaci adaptivních systémů vytvářených zdola nahoru, které bylo vyvinuto v Santa Fé Institutu v USA. Základní jednotkou v systému Swarm je agent interagující s prostředím a dalšími agenty prostřednictvím diskretních událostí (popis dle Czontó, 2001). V základním článku o hejnech (Mínar a kol., 1996) je jako příklad agentů uvedeno 15 kojotů, 50 zajíců a záhon s mrkví. Skupina agentů pak tvoří hejno. Součástí hejna je i rozvrh událostí pro jednotlivé agenty, např. kojoti žerou zajíce, zajíci žerou mrkev a schovávají se před kojoty. Systém umožňuje i vytváření hierarchických struktur: agentem může být hierarchicky nižší entita. Příkladem tedy může být hierarchie hejno zajíců → zajíc → ucho → buňky ucha.

9.3 Deliberativní agenty

Deliberativní (uvažující) agenty jsou odpovědí klasické umělé inteligence na reaktivní agenty. Základními charakteristikami deliberativních agentů jsou:

- **autonomnost** – proaktivní, cílově orientované chování umožňující samostatně řešit určité úlohy,
- **reaktivita** – reagování v souladu s vnímáním reálného času,
- **intencionalita** – schopnost mít na paměti dlouhodobé cíle,
- **sociální chování** – schopnost spolupracovat za účelem dosažení společně sdílených cílů.

Def. 9.2: Deliberativní agent je definován šesticí $\{P, A, D, \text{vjem}, \text{změna_stavu}, \text{akce}\}$, kde

E je množina stavů prostředí

P je množina vjemů agenta

D je množina vnitřních stavů agenta (stav je obvykle množina formulí v predikátové logice)

A je množina akcí

vjem: $E \rightarrow P$

změna_stavu: $P \times D \rightarrow D$

akce: $D \rightarrow A$

Agent pracuje tak, že se snaží aplikovat dedukční pravidla, použitelná na daný stav tak, aby dosáhl svého cíle.

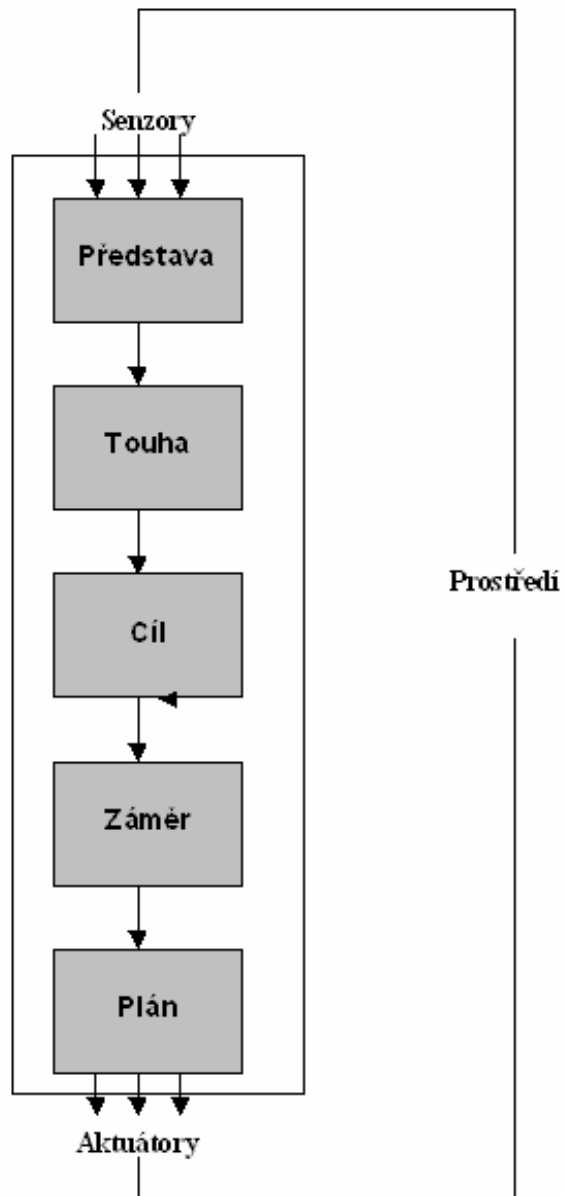
Architektura agenta je tvořena dvěma hlavními částmi:

- tělo (pro řešení úlohy)
- obal (pro integraci):
 - komunikační vrstva
 - model sociálního chování

Prostředí, ve kterém agent působí může být značně rozmanité. Tato prostředí je možno charakterizovat z následujících hledisek (Russel, Norvig, 2003):

- pozorovatelné plně vs. částečně – v plně pozorovatelném prostředí má agent prostřednictvím svých senzorů v libovolném čase přístup k úplnému stavu prostředí,
- deterministické vs. stochastické – v deterministickém prostředí je následující stav prostředí plně dán aktuálním stavem prostředí a akcí agenta,
- epizodické vs. sekvenční – v epizodickém prostředí je agentova zkušenost rozčleněna do atomických epizod; následující epizoda přitom nezávisí na akcích provedených v předcházejících epizodách,
- statické vs. dynamické – statické prostředí se nemění v průběhu doby, po kterou agent uvažuje,
- diskrétní vs. spojité – toto hledisko lze aplikovat na stav prostředí, čas, vjemy i akce,
- jednoagentové vs. multiagentní.

Ve zbývající části této kapitoly se budeme věnovat multiagentním prostředím, která mohou být kompetitivní nebo kooperativní.



Obr. 11 Schéma deliberativního agenta (Kubík, 2004)

9.3.1 Znalosti agenta

Znalosti agenta jsou v zásadě tří typů:

- problémově orientované (týkající se vlastní úlohy, kterou má agent řešit)
- znalosti o sobě samém
- sociální znalosti

Zatímco první typ znalostí známe z „běžných“ znalostních systémů, znalosti o sobě samém a znalosti sociální vnášejí do usuzování agentů novou kvalitu.

9.3.1.1 Znalosti o sobě samém

Znalosti o sobě samém (self knowledge) jsou v deliberativních agentech reprezentovány s využitím modální a temporální logiky. **Modální logika** umožňuje vyjádřit, že

- **nutně něco platí** (ve smyslu platí ve všech světech, ve kterých se agent může ocitnout na základě svého konání)

$$\Box \varphi$$

- **možná něco platí** (ve smyslu platí alespoň v jednom světě, ve kterých se agent může ocitnout na základě svého konání)

$$\Diamond \varphi$$

Platí přitom $\Box \varphi \Leftrightarrow \neg \Diamond (\neg \varphi)$

$$\Diamond \varphi \Leftrightarrow \neg \Box (\neg \varphi).$$

Temporální logika umožňuje pracovat s časovou sousledností událostí. Používají se operátory jako

- od teď dále do budoucnosti platí

$$past \varphi$$

- platí od teď, dokud

$$\varphi \text{ until } \psi$$

- eventuálně (někdy v budoucnosti) platí – lze vyjádřit i modalitou $\Diamond \varphi$

$$future \varphi$$

- vždy (vždy v budoucnosti) platí – lze vyjádřit i modalitou $\Box \varphi$

$$always \varphi$$

Pro usuzování agenta o sobě samém se jako jakýsi standard prosadila **teorie BDI**. Tato teorie využívá následující tři pojmy:

- **B** (beliefs, domněnky) - vyjadřují představy agenta o světě
- **D** (desires, přání) - vyjadřují touhy agenta dosáhnout cíle nebo určitého stavu světa
- **I** (intentions, záměry) – vyjadřují záměr agenta dosáhnout cíle

Na základě svých domněnek, přání a záměrů agent usuzuje o světě:

- $(Bel a \varphi)$ – agent a věří, že formule φ je pravdivá
- $(Des a \varphi)$ – agent a touží, aby formule φ byla pravdivá (pravdivost formule je cílem agenta)
- $(Int a \varphi)$ – agent a zamýšlí dosáhnout toho, aby formule φ byla pravdivá

Znalost agenta pak lze definovat jako

$$(Know a \varphi) \equiv \varphi \wedge (Bel a \varphi)$$

(agent a ví, že φ platí, právě když φ platí a on tomu věří).

Mezi operátory *Bel*, *Des* a *Int* platí následující vztahy:

- $(Des\ a\ \varphi) \Rightarrow (Bel\ a\ \varphi)$
(když agent touží, aby φ bylo pravdivé, tak věří, že φ je pravdivé)
- $(Int\ a\ \varphi) \Rightarrow (Des\ a\ \varphi)$
(když agent zamýšlí, aby φ bylo pravdivé, tak má φ jako svůj cíl)
- $(Des\ a\ \varphi) \Rightarrow (Bel\ a\ (Des\ a\ \varphi))$
(agent si uvědomuje po čem touží)
- $(Int\ a\ \varphi) \Rightarrow (Bel\ a\ (Int\ a\ \varphi))$
(agent si uvědomuje o co se snaží)

9.3.1.2 Sociální znalosti

Sociální znalosti jsou znalosti agenta o ostatních agentech v komunitě, o jejich znalostech, schopnostech, závazcích apod. Minimem je obdoba zlatých stránek: k dispozici jsou jméno, adresa a nabízené služby jednotlivých agentů. Sociální znalosti umožňují agentům:

- delegovat odpovědnost,
- dekomponovat úlohu na podúlohy,
- kontrahovat optimálně spolupracující agenty,
- formovat týmy a koalice,
- vyhledávat chybějící informace....

Sdílená znalost a společný záměr

Pro všechny primitivní (dále nedělitelné) akce je známa informace o tom, který agent je schopen ji vykonat. Formálně lze tuto skutečnost vyjádřit jako „akci α je schopen vykonat agent a “, ($Agt\ \alpha\ a$). Podobně lze definovat, že pro vykonání akce α je potřeba skupina agentů Θ ($Agt\ \alpha\ \Theta$). Pro sdílené znalosti, cíle, domněnky a záměry lze pak definovat operátory (M-Know $\Theta\ \varphi$), (M-Bel $\Theta\ \varphi$), (M-Des $\Theta\ \varphi$) a (M-Int $\Theta\ \varphi$), které vyjadřují, že φ je pro skupinu agentů Θ sdílená znalost, sdílená domněnka, sdílené přání a sdílený záměr (cíl).

Závazek

Když se agent rozhodne pro nějakou akci, tak se zaváže, že ji vykoná. Závazek má podobu příslibu (commitment), že se agent bude snažit zajistit pravdivost určité formule:

- blind commitment – agent se snaží dokud neuvěří, že φ je pravdivé
 $(Bel\ a\ \varphi)$
- single-minded commitment – agent se snaží dokud neuvěří, že φ je pravdivé, nebo dokud neuvěří, že φ nelze splnit
 $(Bel\ a\ \varphi) \vee (Bel\ a\ \Box\ \neg\varphi)$
- open-minded commitment – agent se snaží, pokud věří, že φ lze splnit
 $(Bel\ a\ \Diamond\varphi)$

Úmluva

Platnost závazku je vymezena **úmluvou**, která definuje podmínky, za nichž je závazek nutno splnit, nebo kdy lze od závazku ustoupit.

9.3.2 Interakce v multiagentních systémech

V souvislosti s interakcí v multiagentních systémech mluvíme o koordinaci, kooperaci a komunikaci. Dle (Štěpánková a kol., 1997) je cílem koordinace i kooperace vhodné rozdělení úlohy mezi jednotlivé agenty. Koordinace vnáší do jednání agentů řád, který zajišťuje, že v chování celého systému nevznikají rozpory. Za tím účelem navrhuje principy chování agentů, které se uplatňují v průběhu celého jejich životního cyklu. Metody kooperace se soustřeďují na návrh principů činnosti menší skupiny agentů vybraných pro realizaci dané úlohy. Komunikace je pak způsob dorozumívání mezi jednotlivými agenty i způsob vysílání zpráv.

Podobnost mezi multiagentními systémy a lidským kolektivem vede k hledání analogií se sociálním chováním lidí. Významnou roli zde mají tržní mechanismy a různé modely vyjednávání. Na vyjednávání se můžeme dívat ze tří pohledů (Štěpánková a kol., 1997):

- **lingvistický pohled** se soustředí na jazyk pro výměnu informací mezi agenty. Z tohoto pohledu obsahuje každá zpráva dva typy údajů. (1) Informace specifikující důvody komunikace jako např. návrh, žádost, dotaz, nabídka, odpověď, přijetí či odmítnutí návrhu (často v podobě elementárních konstruktů jazyka, které zprávu uvádějí). (2) Kontextová informace shrnuje údaje důležité pro cílové rozhodnutí (adresát, odesílatel, čas, odesílatelovy návrhy týkající se způsobu dosažení společného cíle)
- **rozhodovací pohled** používá agent, aby zhodnotil aktuální situaci ze svého hlediska a zvolil vhodnou taktiku. (1) Agent může hledat maximální možnou shodu mezi svým záměrem a plánem, který navrhuje partner, (2) Agent se snaží optimalizovat svůj užitek pro zamýšlený cíl, (3) Agent se rozhoduje podle svých preferencí mezi uvažovanými akcemi, (3) způsob vyjednávání závisí na „povaze“ agenta – v této souvislosti rozeznáváme agenty soutěživé a kooperativní.
- **procesní pohled** určuje chování agenta v průběhu celého procesu vyjednávání. Stanoví tedy vyjednávací strategie.

9.3.2.1 Koordinace

Cílem koordinace je především přidělování zdrojů a předávání mezivýsledků při řešení nějaké úlohy. Předpokladem koordinace je, že agenti dokáží uvažovat o důsledcích svého jednání i o důsledcích předpokládaných akcí ostatních agentů. Koordinace umožňuje řešení problému na základě decentralizace vykonávaných úkolů a případně decentralizace řízení. Ke koordinačním mechanismům patří:

- vzájemná dohoda rovnocenných agentů
- přímý dozor (existuje jeden agent pro centrální řízení)
- standardizace (koordinace na základě pravidel chování)

Jak už bylo zmíněno, řada koordinačních procesů je inspirována světem lidí. Typickým příkladem jsou aukce:

- anglická – licituje se „zezdola“, vyhraje nejvyšší nabídka
- holandská – licituje se „shora“, vyhraje první akceptovatelná nabídka
- obálková metoda – jedno kolo tajných nabídek, vyhraje nejvyšší
- Vickrey – jako obálková metoda, vítěz zaplatí druhou nejvyšší cenu

Bez specifikování rolí, vazeb a závislostí mezi agenty je dosažení vyššího stupně racionality celého společenství velice obtížné.

9.3.2.2 Kooperace

Cílem kooperace je zkvalitnění práce multiagentního systému. Toho může být dosaženo (Štěpánková a kol., 1997):

- zrychlením řešení využitím paralelního zpracování,
- rozšířením třídy řešitelných úloh vyplývajícím z výhod sdílení zdrojů,
- zvýšením počtu úspěšně vyřešených úloh díky tomu, že tutéž úlohu paralelně řeší více metod,
- snížením počtu kolizí mezi vzájemně soutěžícími agenty.

V multiagentních systémech se používají různé kooperační nástroje. Jako příklad si uvedme tabuli a kontraktační síť.

Tabule

Použití tabule je vhodné v systémech, kde je globální problém distribuován na několik samostatných výpočetních prvků řízených centrálním prvkem. Mechanismus tabule byl navržen pro heterogenní expertní systémy jako společná datová a řídicí struktura, přes kterou komunikují jednotlivé moduly systému podobně, jako spolu komunikují experti, kteří diskutují nějaký problém a na tabuli zapisují poznatky, o které se chtějí podělit s ostatními.

Kontraktační síť

Síť agentů s vymezenými rolemi pro distribuované řešení problémů. Síť tvoří agenty různých typů:

- řešitelské agenty – zdrojově omezené agenty specializující se na určitou oblast řešené úlohy
- manažerský agent (mediator, faciatiator, broker) – řídicí agent, který specifikuje jednotlivé úkoly a kontroluje jejich plnění.

Vyjednávání v kontraktační síti se řídí zákonem nabídky (služeb) a poptávky (po službách). Z hlediska chování v průběhu vyjednávání můžeme agentům přisoudit určité antropomorfní rysy:

- benevolentní agent se snaží vyhovět, kdykoliv je o něco požádán
- sobecký agent se snaží naplňovat své představy i na úkor ostatních

9.3.2.3 Komunikace

Komunikace představuje nutnou podmínku pro koordinaci a kooperaci. Způsob dorozumívání a vysílání zpráv výrazně určuje vlastnosti multiagentního systému. Základní typy komunikace se liší podle cíle, kam směřují zprávy. Dle (Štěpánková a kol., 1997) můžeme rozlišovat komunikaci **přímou**, kdy zprávy posíláme dalším agentům, a **nepřímou**, kdy se zprávy soustřeďují v předem dané struktuře (jakou je například tabule). U přímé komunikace rozeznáváme **adresné posílání zpráv**, kdy je zpráva předána přímo konkrétnímu agentovi a **všesměrové posílání zpráv**, kdy je zpráva poslána všem agentům najednou. Kombinací obou přístupů dostaneme **selektivní vysílání zpráv**, kdy agenti jsou rozděleni do skupin a zpráva je poslána všem agentům ve skupině.

U komunikace obecně (tedy jak mezi lidmi, tak mezi stroji-agenty) je důležité, aby jednotliví aktéři sdíleli informace na třech úrovních: syntaktické, sémantické a pragmatické. V multiagentních systémech je nejjednodušší zabezpečit první hledisko; agenti prostě používají jednotnou syntaxi. Pro zajištění porozumění obsahu zprávy jsou agenti vybaveni stejným ontologickým rámcem používaných znalostí. Pragmatická stránka komunikace se soustřeďuje na znalosti o tom, kde nějakého agenta nalézt a jak s ním komunikovat.

Jazyky navržené pro komunikaci mezi agenty (agent communication language – ACL) slouží pro vzájemné domlouvání, koordinaci a kooperaci. Na vlastním přenosu zpráv a informací se podílí různé úrovně (Mařík a kol., 2001):

- **fyzická úroveň** zahrnující přenos zprávy jako sekvence bitů (využívá se tři nejnižších vrstev ISO/OSI modelu: fyzické, linkové a síťové),
- **transportní úroveň** specifikující protokoly pro kódování zprávy: takovými protokoly jsou např. TCP/IP nebo HTTP (což odpovídá čtvrté, transportní vrstvě ISO/OSI),
- **úroveň komunikační architektury** zajišťující překlad zpráv do posloupnosti bytů; příkladem jsou různé komunikační platformy jako CORBA, Java RMI nebo UNIX RPC,
- **úroveň ACL** specifikující jak je ve zprávě identifikován odesílatel a adresát, jakým jazykem je zapsáno vlastní sdělení; příkladem jsou jazyka FIPA-ACL a KQML zmíněné v další části této podkapitoly,
- **úroveň obsahu zprávy** specifikující konkrétní reprezentaci přenášených znalostí; příkladem mohou být XML, KIF (Knowledge Interchange Format), ale i Lisp nebo Prolog či SQL.

Nejnámějšími příklady jazyků pro komunikaci mezi agenty jsou:

- KQML (Knowledge Query and Manipulation Language)
- FIPA-ACL (agent communication language) navržený organizací FIPA

Oba tyto jazyky jsou založeny na **teorii řečových aktů**, která vychází z lingvistické analýzy komunikace v přirozeném jazyce (Mařík a kol., 2001). Teorie řečového aktu vychází z představy, že v souvislosti s řečovou komunikací člověk nejenom vyslovuje nějaké tvrzení, ale současně koná i jisté akce. Řečové akty mohou být:

- oznamovací (dveře jsou zavřené)

- zavazující (za chvíli zavřu dveře)
- příkazovací (zavři ty dveře)
- expresivní (mrzí mne, že ty dveře jsou stále zavřené)
- deklarační (vyhlašuji válku)

V teorii řečových aktů lze každý takový akt rozdělit na tři dílčí prvky. Promluvový odpovídá formulaci promluvy (v jazycích ACL formulaci elementární zprávy), nepromluvový kategorizuje promluvu z hlediska řečníka (v jazycích ACL typ a kategorie zprávy) a mimopromluvový specifikuje zamýšlené změny vnitřního stavu posluchače (v jazycích ACL požadované změny vnitřních stavů přijímacího agenta).

Jazyk KQML

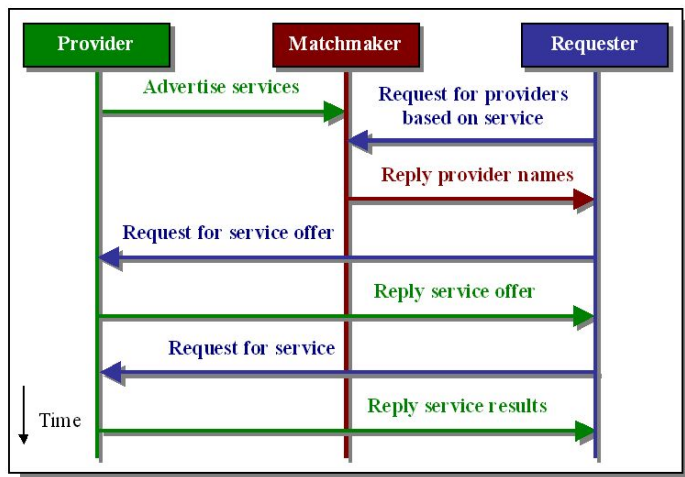
Jazyk KQML byl navržen v polovině 90. let jako pokus o standardizaci komunikačních a interakčních procesů v multiagentních systémech. Jazyk se zaměřuje na podporu pragmatických a sémantických aspektů komunikace mezi agenty. Je to tedy jazyk podporující činnost agentů při hledání agentů vhodných ke spolupráci, navazování spojení mezi agenty a výměně informací mezi agenty. Na syntaktické úrovni není KQML vázán na použití žádného konkrétního jazyka. Jazyk KQML je založen na dvou základních principech.

Prvním principem je definice relativně malého počtu tzv. performativů (komunikačních sloves) pro realizování předem specifikovaných řečových aktů. Příkladem performativů jsou

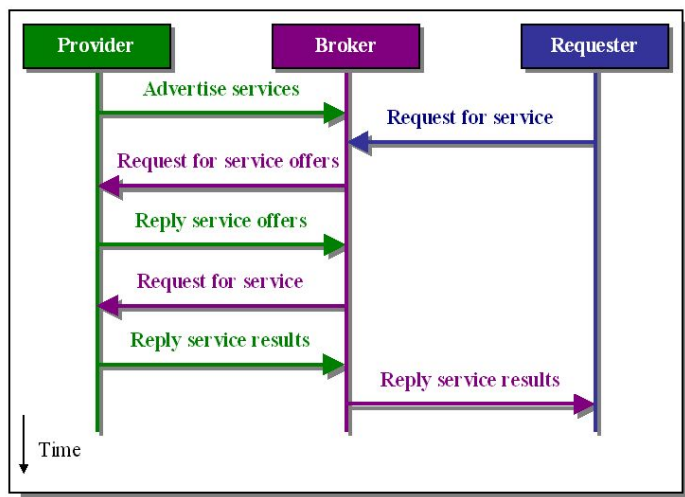
- ask_if (je pravda, že?)
- perform (proved' následující akci)
- tell (toto je pravda:)
- reply (moje odpověď je:)
- inform (sděluji, že)

Druhým principem je nepřímá komunikace s využitím tzv. **faciliátorů**. Zavádí se tedy speciální třída agentů. Faciliátorem může být:

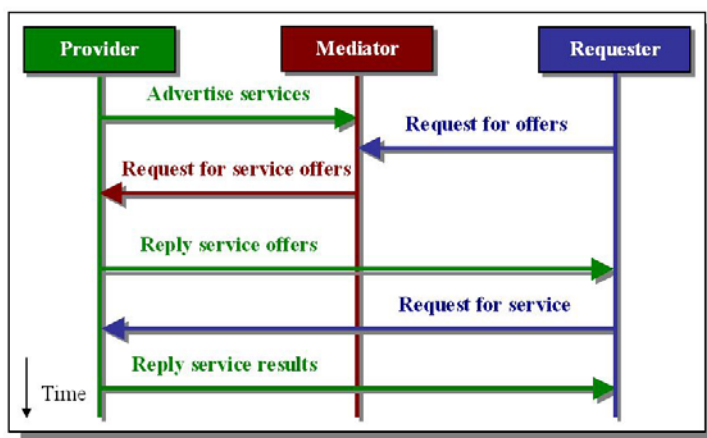
- **matchmaker** – agent, který pouze hledá vhodné agenty pro požadovanou službu. Agent, který poptává nějakou službu na základě informace od matchmakera přímo kontaktuje agenta, který službu poskytuje.
- **broker** – agent, který nejen hledá agenty pro požadovanou službu, ale službu přímo i objedná. Poptávajícímu agentovi pak předá rovnou výsledek poskytnuté služby.
- **mediátor** – agent, který podobně jako broker službu objedná. Agent, který službu poskytuje, pak ale službu nabídne a dále komunikuje přímo s poptávajícím agentem.



Obr. 9.12 Matchmaker (převzato z Mařík, Štěpánková, Pěchouček, 2001)



Obr. 9.13 Broker (převzato z Mařík, Štěpánková, Pěchouček, 2001)



Obr. 9.14 Mediator (převzato z Mařík, Štěpánková, Pěchouček, 2001)

Jazyk FIPA-ACL

Jazyk FIPA-ACL, navržený koncem 90. let, vychází z principů jazyka KQML. Snaží se ale lépe definovat sémantiku a komunikační protokoly (popis dle Mařík a kol., 2001). Základem jazyka jsou opět typizované druhy zpráv realizující řečové akty, nazývané tentokrát komunikativní akty. Komunikativní akty se konceptuálně neliší od performativů, jejich nabídka je ale odlišná. FIPA-ACL pracuje s 20 komunikativními akty rozdělenými do 5 skupin na akty pro:

- přenos informace (confirm, disconfirm, inform, inform-if, inform-ref),
- vyžádání informace (query-if, query-ref, subscribe),
- vyjednávání (accept-proposal, cfp, propose, reject-proposal),
- vykonání akce (agree, cancel, refuse, request-when, request-whenever),
- chybová hlášení (failure, not-understood).

Podstatných rozdílů je oproti KQML jen několik:

- sémantika jazyka je striktně formalizována s využitím modální logiky. Výhodou je existence precizního formálního popisu komunikace, nevýhodou je že agenti obvykle nedokáží v rámci modální logiky odvozovat,
- množina komunikativních aktů je uzavřená, nová komunikativní akty lze vytvářet pouze skládáním aktů existujících,
- neexistují performativy pro síťování a administraci přenosu zpráv.

9.3.3 Architektury multiagentních systémů

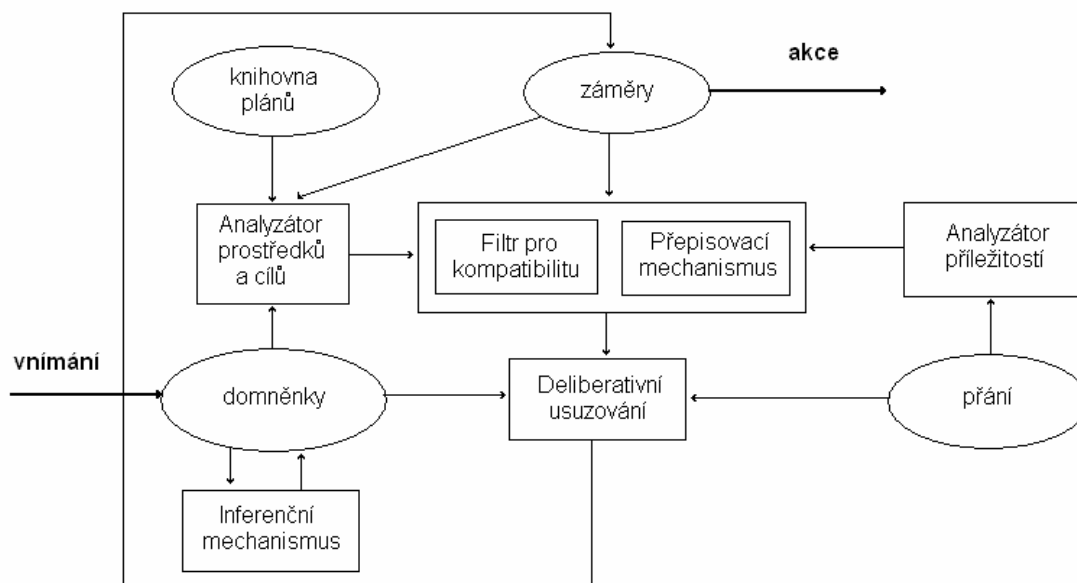
Na závěr této kapitoly se podíváme na různé typy architektur multiagentních systémů. Bude nás při tom zajímat, jak se do různých architektur promítnou výše uvedené teoretické úvahy.

9.3.3.1 Architektura z hlediska teorie BDI

IRMA (intelligent resource-bounded machine architecture) je standardní model architektury agenta podle teorie BDI. Tento model je zaměřen především na modelování záměrů agenta. Plánovací mechanismus je založen na analýze prostředků a cílů (tato metodika je zmíněna v první kapitole). IRMA využívá znalostní struktury jako jsou domněnky, cíle, knihovny plánů a strukturovaných záměrů (popis dle Kubík, 2004). Na obr 9.15 jsou tyto entity označeny elipsami. S těmito znalostními strukturami manipulují analyzátor prostředků a cílů, analyzátor příležitostí, inferenční mechanismus a různé filtrovací mechanismy (procesy jsou na obr 9.15 označeny obdélníky).

Agent má svá přání, která nemusí být nutně v souladu s jeho možnostmi (schopnostmi či zdroji). Na základě vnímání okolního prostředí vytváří a aktualizuje své domněnky o světě. Některá z přání se stanou cíli na základě výběru podle aktuálního stavu prostředí. Tento proces se nazývá deliberativní usuzování. Cíl, který hodlá agent splnit, se mění v záměr a plán, jak se má cíle dosáhnout. Plán je posuzován analyzátozem prostředků a cílů, který posoudí, zda má agent schopnosti, kompetence a zdroje ke splnění cíle. Analyzátor příležitostí navrhuje modifikace plánů v důsledku změn v prostředí. Vytvořený plán je posuzován filtrem kompatibility, který zjišťuje, je-li plán v souladu s aktuálním záměrem. Pokud ano, prochází

plán procesem deliberativního usuzování, jehož výsledkem je aktualizace záměru a závazek pro jeho splnění. Záměr se realizuje vykonáním akcí specifikovaných v plánu. Přepisovací mechanismus posuzuje nekompatibilní plány.



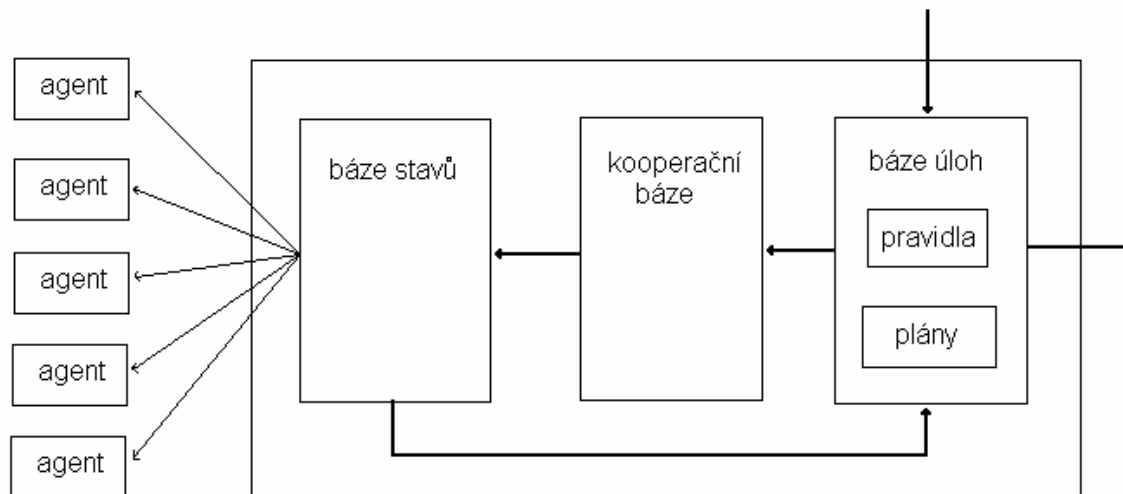
Obr. 9.15 Architektura IRMA (podle Kubík, 2004)

9.3.3.2 Architektura z hlediska sociálních znalostí

Architektura BDI se sice ujala jako referenční model racionálního chování agenta, neposkytuje ale prostředky k formalizaci a využívání znalostí o spolupracujících agentech, jejich cílech a plánech, ani se nezabývá vztahy nadřazenosti a podřízenosti mezi agenty či jejich povinnostmi a závazky. Tyto znalosti je možno modelovat s užitím tzv. sociálních modelů, ke kterým patří twin-bázový a tri-bázový model.

Tri-bázový model (Mařík a kol. 2000) je založen na reprezentaci a správě sociálních znalostí ve třech oddělených znalostních strukturách (popis dle Mařík a kol. 2001). V **kooperační bázi** jsou spravovány permanentní informace o všech spolupracujících agentech (fyzická adresa agenta, jazyk kterým hovoří, jeho předdefinované schopnosti apod.). Další ze struktur, **báze úloh**, obsahuje dvě sekce. Zatímco v **problémové sekci** se nachází obecné znalosti o tom, jak úlohu dekomponovat na podúlohy, **plánovací sekce** obsahuje již konkrétní instance pravidel s ohledem na aktuální zatížení a schopnosti spolupracujících agentů. Klíčovou znalostní strukturou tri-bázového modelu je **báze stavů**, která obsahuje neustále se měnící informace o (1) spolupracujících agentech, jejich zátěži, intencích, dostupnosti apod. (2) stavu rozpracovaných úloh. Schéma správy sociálních znalostí v tri-bázovém modelu je uvedeno na obr. 9.16. Tri-bázový model je použit např. v systému ProPlanT, systému pro plánování projektově orientované výroby (Mařík a kol. 2000).

Twin-bázový model (Cao a kol., 1996) je myšlenkovým předchůdcem tri-bázového modelu. Architektura tohoto modelu byla použita při implementaci systému ViSe² (virtuální sekretářky). Twin-bázový model je zjednodušením tri-bázového modelu v tom, že obsahuje pouze kooperační bázi a bázi úloh. Agent má všechny plány předem připravené, není třeba je vytvářet při požadavku na dekompozici. Každá změna v kooperační bázi má za následek přepočítání a přeskupení pořadí plánů v bázi úloh. Tento přístup je velmi efektivní pro relativně malý počet plánů v bázi úloh.



Obr. 9.16 Tri-bázový model (dle Mařík a kol. 2001)

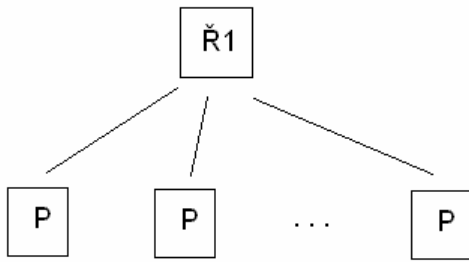
9.3.3.3 Architektura z hlediska komunikace

Architektura multiagentního systému může být:

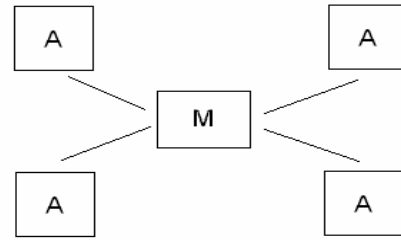
- centralizovaná – v multiagentním systému je jeden řídicí agent, kterému jsou ostatní podřízeni,
- hierarchická – v multiagentním systému se používají různé úrovně řízení,
- federovaná – v multiagentním systému se používá nepřímá komunikace přes prostředníka,
- decentralizovaná – v multiagentním systému jsou decentralizovány role i řízení.

Tyto architektury můžeme hodnotit z hlediska robustnosti (odolnosti vůči poruchám), škálovatelnosti (rozšiřitelnosti o další komponenty), implementačních nároků a nároků na komunikaci. Nás bude v této kapitole zajímat poslední hledisko.

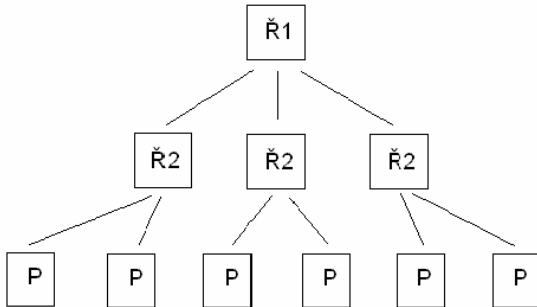
Nejmenší nároky na komunikaci má centralizovaná architektura, ta je ale nejobtížněji modifikovatelná. Nejsnáze se modifikuje decentralizovaná architektura, ta ale klade největší nároky na komunikaci.



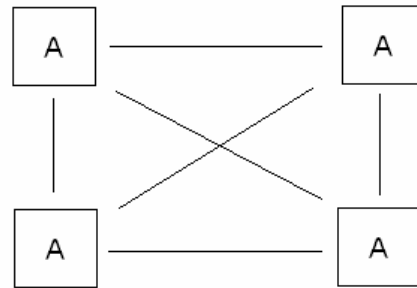
Obr. 9.17 Centralizovaná architektura



Obr. 9.19 Federovaná architektura



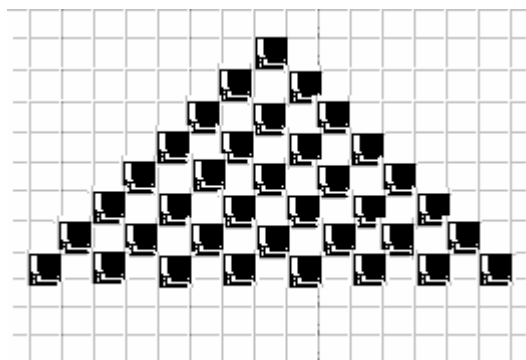
Obr. 9.18 Hierarchická architektura



Obr. 9.20 Decentralizovaná architektura

Cvičení:

- 1) Jaký obrazec vygeneruje v jednorozměrném CA sada pravidel 90 s jednou buňkou uprostřed řádku?
- 2) Jaký obrazec vygeneruje v jednorozměrném CA sada pravidel 18 s jednou buňkou uprostřed řádku?
- 3) Jaká sada pravidel vygeneruje v jednorozměrném CA obrazec uvedený na následujícím obrázku?



Literatura:

1. Csontó J.: Umělý život. V: (Mařík V., Štěpánková O., Lažanský, J. eds.): Umělá inteligence 3. Academia, Praha, 2001
2. Internet. Celulárne automaty, <http://alife.tuke.sk/>
3. Internet. Life, <http://alife.tuke.sk/>
4. Kelemen J.: Strojovia a agenty. Archa, Bratislava, 1994.
5. Kelemen J.: Reaktivní agenti. V: (Mařík V., Štěpánková O., Lažanský, J. eds.): Umělá inteligence 3. Academia, Praha, 2001
6. Kubík A.: Inteligentní agenty. Computer Press, Brno, 2004
7. Kvasnička V., Pospíchal J.: Informatika pre sociálne vedy. Skripta Univerzita Komenského, Bratislava, 2005.
8. Mařík V., Pěchouček M., Štěpánková O., Lažanský J.: ProPlanT: Multi-Agent Systém for Production Planning. *Applied Artificial Intelligence* vol. 14, No. 7, 2000, 727-762
9. Mařík V., Štěpánková O., Pěchouček M.: Znalosti a multiagentní systémy. Prezentace 2001
10. Mařík V., Pěchouček M., Štěpánková O.: Multiagentní systémy: Principy komunikace a základní formální architektury. V: (Mařík V., Štěpánková O., Lažanský, J. eds.): Umělá inteligence 3. Academia, Praha, 2001
11. Minar, N., R. Burkhart, C. Langton, and M. Askenazi.: The Swarm simulation system: A toolkit for building multi-agent simulations. Working Paper 96-06-042, Santa Fe Institute, Santa Fe., 1996
12. Russel S., Norvig P.: Artificial Intelligence, a Modern Approach. 2.vydání, Prentice Hall, 2003
13. Štěpánková O., Mařík V., Lhotská L.: Distribuovaná umělá inteligence. V: (Mařík V., Štěpánková O., Lažanský, J. eds.): Umělá inteligence 2. Academia, Praha, 1997