

Řešení složitých a rozsáhlých úloh

Zvýšení flexibility – jen zvýšením autonomnosti jednotlivých zdrojů znalostí v rámci modulární architektury

Volná integrace modulů prostřednictvím komunikace formou zasílání zpráv (“peer-to-peer”)

Redukce role centrálního prvku

Tak vznikají tzv. **multiagentní systémy**

Charakteristiky agenta

autonomnost - **agenti** jsou proaktivní, cílově-orientované moduly schopné samostatného řešení určitých úloh bez nezbytnosti komunikace s okolím, avšak schopní komunikace, event. koordinace činnosti či dokonce kooperace s jinými agenty v rámci určité komunity. **Agenti** mají možnost se dobrovolně hlásit ke komunitě, opouštět ji, poskytovat jí výsledky či výsledky jiných požadovat

reaktivita - **agenti** jsou aktivováni událostmi (event-driven), schopni reagovat v souladu s vnímáním reálného času

Charakteristiky agenta

intencionalita - schopnost mít na paměti dlouhodobé cíle, organizovat své chování tak, aby směřovalo k těmto cílům, formulovat vlastní plány a používat určitých typů vlastního usuzování

schopnost sociálního chování - schopnost agentů spolupracovat s cílem dosahovat společně sdílených cílů, schopnost udržovat si informaci o jiných agentech a uvažovat nad nimi, cílevědomě se sdružovat do koalic a týmů, z nichž lze očekávat vzájemný prospěch

Sociální znalosti

Znalosti agenta lze rozdělit na:

problémově-orientované znalosti (problem oriented knowledge) – “asociální” typ znalostí sloužící k lokálnímu, samostatnému řešení úloh (např. poskytování expertizy, hledání ve vlastní databázi agenta atd.)

znalosti o sobě samém (self knowledge) – znalosti o vlastním chování, vnitřním stavu, závazcích apod.

sociální znalosti (social knowledge) – znalosti o chování jiných agentů, o jejich schopnostech, zatížení, zkušenostech, závazcích, o jejich znalostech, záměrech a víře

Soustředíme se na oblast **softwarové integrace** a na problémy **agentifikace** – zaměříme se na roli sociálních znalostí při začleňování již existujícího softwarového modulu do multiagentní komunity

Organizace sociálních znalostí

Sociální znalosti mohou být organizovány:

centrálně

lokálně

kombinovaně

Základní otázky při lokální organizaci sociálních znalostí:

rozumný rozsah globálních znalostí

rozumný rozsah znalostí o kooperujících agentech

jak zabezpečit aktualizaci znalostí

Architektura agenta

Agent obvykle sestává z

obalu a

vlastního těla

pro úlohy **integrace**: tělo nemá informace o komunitě, obal je zodpovědný za plánování a realizaci sociálních interakcí v širším slova smyslu

obal tak sestává z

komunikační vrstvy

modelu sociálního chování
(acquaintance model)

Model sociálního
chování

Komunikační vrstva

tělo

Úloha sociálních znalostí

Sociální znalosti umožňují agentům:

delegovat odpovědnost,

dekomponovat úlohu na podúlohy,

kontrahovat optimálně spolupracující agenty,

formovat týmy a koalice,

vyhledávat chybějící informace....

Tyto úlohy jsou často řešeny s využitím centrálně organizovaných znalostí tedy pomocí **facilitátorů** a informačních **brokerů**, **mediátorů**, **matchmakerů** nebo **middle-agentů** – tyto **agenti** jsou obvykle součástí multiagentních platforem, díky centrálnímu uložení představují citlivý "bottleneck" celé komunity

Užití lokálně reprezentovaných sociálních znalostí zaručuje **vyšší stupeň autonomnosti** chování agentů a tím robustnost celé komunity, ale zejména **snižuje komunikační provoz** v rámci komunity

Typy sociálních znalostí

Znalosti v **obalu** (lokálně umístěné sociální znalosti):

permanentní

semi-permanentní

dočasné

Otázky:

jak detailních znalostí je potřeba?

jak od sebe tyto typy znalostí oddělit?

jak udržovat sociální znalosti trvale v aktualizované podobě?

Model sociálních znalostí

Tři způsoby komunikace:

centrální komunikační jednotka

všesměrové šíření zpráv (spojené např. s metodikou kontrahování agentů na bázi "contract-net-protocol")

komunikace s využitím lokálně umístěných sociálních modelů ("acquaintance-model-based contract-bidding strategies) – je cílená, opírá se o znalosti udržované průběžnou, nenáročnou komunikací v nekritických časových intervalech

Komunikace probíhá v ACL (Agent Communication Language) – Jak ho standardizovat??

Základní problémy

Principy komunikace mezi agenty, standardizace a interoperabilita

(Lokální) modely pro správu sociálních znalostí, jejich aplikace

Jazyky pro komunikaci mezi agenty

ACL – jazyk vyšší úrovně určený pro vzájemné domlouvání, koordinaci a kooperaci

Obvykle se opírají o tzv. **teorii řečového aktu** (**performativy** vyjadřující, že nějaká operace se opravdu vykonala nebo bude vykonána)

Sdílení informace na třech úrovních: **syntaktické**, **sémantické** a **pragmatické**

Syntaktická úroveň: problém jednoduchý, syntax lze snadno definovat

Hlavní problém: **sémantická úroveň** – tvorba **znalostních ontologií**

Pragmatická úroveň: s kým hovořit, kde dotyčného nalézt, jak iniciovat komunikaci mezi agenty – není složité

Jazyk KQML

KQML (Knowledge Query and Manipulation Language) – první pokus o formalizaci a standardizaci

zaměřen na pragmatické a sémantické aspekty komunikace

základní principy:

- zavedení **performativů** jako základních komunikačních aktů

- zavedení **facilitátorů**, tj. “centrálních” agentů poskytujících určité komunikační služby

Každá **zpráva** sestává ze jména performativu a jeho parametrů, vlastní obsah zprávy může být zapsán v libovolném jazyce a představuje obsah jednoho parametru

Jazyk KQML

Typická zpráva:

```
(ask-one  
  
:sender      agent A  
  
:receiver    server_ceník  
  
:language    PROLOG  
  
:ontology    SCP_seznam  
  
:content     "cena(unipetrol,CENA).")
```

Obsahem performativu může být další performativ jako požadovaná forma odpovědi, může dokonce docházet k vícenásobnému vnořování

Jazyk KQML

Tři typy performativů:

Performativy pro vedení diskuse: ask-if, ask-all, ask-one, tell, untell, deny, insert, uninsert, delete-one, delete-all, undelete, advertise, unadvertise, subscribe

Performativy pro zasahování do diskuse: error, sorry, standby, ready, next, rest, discard

Performativy pro síťování a podporu facilitátora: register, unregister, broadcast, forward, transport-address, broker-one, broker-all, recommend-one, recommend-all, recruit-one, recruit-all

Jazyk KQML

Jazyk KQML předpokládá, že většina dohadovacích procedur mezi agenty probíhá nepřímo, prostřednictvím facilitátora

Každý facilitátor zodpovídá za jednu doménu, veškerá komunikace vně domény jde výlučně přes příslušné facilitátory

Velmi často se používá kombinace performativů [advertise-subscribe](#)

Nejčastěji tři dohadovací mechanismy - odpovídající role agentů:

- matchmaker

- broker

- mediator

Jazyk FIPA-ACL

Sdružení **FIPA** (Foundation for Physical Intelligent Agents) navrhlo svůj vlastní ACL

Tento vychází z principů KQML: opět se používá performativů a jejich syntaxe je téměř stejná jako v KQML

Obsah performativů je však jiný, FIPA jich stanovila pouze **uzavřenou množinu** (20) a nové performativy mohou vzniknout jen přípustnou kombinací těch základních

Sémantika jazyka striktně formalizována pomocí výrazových prostředků **modální logiky** v rámci FIPA-SL:

agenti však nebývají vybaveni příslušnými vyvozovacími algoritmy

sémantika závisí na informacích typu belief, které nebývají přístupné jiným agentům a často se mění

Jazyk FIPA-ACL

Ve FIPA-ACL nejsou k dispozici performativy pro síťování a administraci zpráv, FIPA tento problém přenechává implementátorům

FIPA vydává standardy

a) **normativní**: FIPA-AA (Abstract Architecture), FIPA-ACL, FIPA-SL, FIPA-KIF (Knowledge Interchange Format), FIPA-RDF (Resource Description Framework), FIPA-AMT (Agent Message Transport)

b) **informativní**: Agent SW Integration, Personal Travel Assistance, Personal Assistant, Audio-visual Entertainment and Broadcasting, Network Management and Provisioning, AgentCities, úsilí v oblasti holonických výrobních systémů

FIPA

Jazyk FIPA-ACL

Open-source implementace:

JADE

FIPA-OS

APRIL

ZEUS

Protokoly spolupráce agentů

FIPA-ACL podporuje standardní protokoly spolupráce agentů:

Contract Net Protocol, anglická dražba, holandská dražba

obálková metoda

aukce typu Vickrey

Modely pro správu sociálních znalostí

Dva základní modely:

twin-base model (Cao, Hartwigsen)

tri-base model

Tri-base acquaintance model (3bA):

kooperační báze: permanentní/semipermanentní znalosti (adresa, jazyk, schopnosti agentů)

báze úloh: sestává ze sekce problémové (obecné znalosti pro dekompozice úloh) a ze sekce plánovací (konkrétní instance pravidel s ohledem na aktuální zatížení a schopnosti)

báze stavů: obsahuje stále se měnící informace o spolupracujících agentech a o stavu řešení úloh

Správa znalostí

Kooperační báze - permanentní znalosti

Báze úloh: problémová sekce je permanentní a sekce plánovací je spravována v průběhu přeplánování

Báze stavů je udržována buď formou:

periodických revizí – jedná se o pravidelnou výměnu informací v časových úsecích, kdy není komunikační infrastruktura příliš zatížena

nebo za pomoci performativů avdertise-subscribe (asynchronní údržba)

Vylepšování znalostí

Vylepšování znalostí na úrovni jednotlivých agentů – agent se sám učí, optimalizuje, reorganizuje svoji činnost, samostatně vyvozuje a mění i permanentní znalosti

Vylepšování na meta-úrovni – realizuje se za pomoci nezávislého agenta, tzv. meta-agenta, jenž pozoruje aktivity celé komunity nebo její části, zobecňuje posbíraná data a poté zobecňuje získané poznatky

Vylepšování znalostí je jednou z forem “sociálního uvažování”

Typy aktivit meta-agenta

Nenahrazuje žádnou z forem centrálního agenta (facilitátora, brokera, middle-agenta atd...), neboť bez něj komunita dále funguje, byť nezlepšuje celkovou efektivitu svého chování

meta-agent může být buď:

aktivním meta-agentem, jenž svými závěry, které zasílá formou zpráv, reviduje obsah sociálních modelů (konkrétně problémovou sekci báze úkolů) jednotlivých agentů

pasivním meta-agentem, jenž neovlivňuje životní cyklus dané komunity, pouze činnost komunity pozoruje a své závěry předkládá uživateli (ten pak uzavírá zpětnou vazbu zásahem do komunity)

Typy revizí z úrovně aktivního meta-agenta

Možné typy meta-agentem odhalených skutečností (a následných revizí):

agent odumřel

agent ztratil část svých schopností

nový agent

agent získal další, nové schopnosti

agent změnil míru své spolehlivosti

u agenta se změnila míra jeho zatížení

změnily se podmínky aplikovatelnosti některé části pravidel

agent obvykle kooperuje s jistou částí komunity (přirozená koalice)

Závěr

Přirozený trend v oblasti umělé inteligence: Dekompozice monolitních řešení do podoby distribuovaných a posléze multi-agentních systémů

V souvislosti s multi-agentními systémy - dva nové typy znalostí: znalosti sociální a znalosti ontologické

Tyto dva nové typy znalostí je nutno udržovat odděleně od znalostí pro (lokální) řešení úloh, obvykle v tzv. obalu agenta

Sociální znalosti...modely pro správu sociálních znalostí

Ontologické znalosti... sdílené znalostní ontologie v rámci komunikačních standardů

Závěr

Perspektivním modelem - 3bA model

oddělené spravování permanentních, semipermanentních a proměnných znalostí

silná redukce meziagentové komunikace

přirozené zavedení meta-uvažování (meta-agentů)

Z hlediska ACL - nejzávažnější problém = standardizace

Zatím KQML a FIPA

Usilovná snaha o tvorbu informativních standardů (formát zpráv, sdílené ontologie, doporučené služby)