

Rappresentazione della Conoscenza

Sistemi multi-agente

Sommario

- ◇ Introduzione [Wooldridge, Ferber - MAS, cap. 1]
- ◇ Interazione tra agenti [Ferber - MAS, cap. 2]
- ◇ Applicazioni
- ◇ Comunicazione tra agenti [Wooldridge - MAS, cap 6]

Sistemi multi-agente

Sistemi composti da multiple entità in grado di interagire tra di loro.

Entità → difficile da definire con precisione Caratteristiche fondamentali:

- Rilevante grado di autonomia
- Capacità di interagire con l'ambiente
- Capacità di comprendere l'ambiente circostante
- ...

Evoluzione in informatica

Caratterizzata da:

- Ubiquità
- Interconnessione
- Intelligenza
- Delega
- Programmazione ad alto livello (Human-Orientation)

Tutte queste componenti favoriscono lo sviluppo di MAS

Motivazioni

- ◇ I MAS sono sistemi complessi per problemi **difficili**
 - computazionalmente
 - articolati (richiedono competenze specifiche)
 - distribuiti fisicamente

- ◇ Socialità condizione necessaria per l'intelligenza

Motivazioni II

- ◇ Agenti come paradigma computazionale
 - interazione → componente fondamentale dell'ingegneria del software
 - MAS catturano le interazioni e possono aiutare a studiarle, formalizzarle e modellarle.
 - Agenti evoluzione Object Oriented Programming: **Agent Oriented Programming**

- ◇ Componente Sociale:
 - Capire come evolve un sistema sociale
 - progetto EOS: studiare la comparsa di complessità sociale nell'era paleolitica

Interazione tra agenti

- meccanismi per il funzionamento di sistemi complessi
- forme di interazione per migliorare le prestazioni
- tecniche di autoorganizzazione
- costruzione di modelli computazionali per l'interazione tra agenti

Kenetics: la scienza e la tecnologia di organizzazioni artificiali

MAS

$\langle E, O, A, R, Op, L \rangle$

- E ambiente
- O oggetti
- A agenti ($\subseteq O$)
- R relazioni tra agenti
- Op operazioni relative agli oggetti
- L leggi di evoluzione dell'ambiente

Il ruolo dell'ambiente

- agenti **situati** (comunicano solo attraverso modifiche del'ambiente, **stigmergy**)
- agenti **comunicanti** (non interagiscono con l'ambiente, **agenti puramente computazionali**)
- agenti **comunicanti** e **situati**

MAS e organizzazione

- top-down: si definiscono gli obiettivi e si studia la forma di interazione migliore
- bottom-up: si definiscono gli agenti e le modalità di interazione e si vede come evolve il sistema

- interazione tra pochi agenti (micro-sociale)
- creazione di gruppi (livello di gruppo)
- popolazioni di agenti (livello di società)

Relazione tra MAS ed altre discipline

- ◇ **Sistemi Distribuiti (SD)**: oggetto di studio per molti anni

- ◇ MAS sono sistemi distribuiti. MAS aggiungono qualcosa ai sistemi distribuiti ?
 - Sincronizzazione tra entità **dinamica** e non statica
 - Entità con **interessi propri**: PDA che prenota un viaggio aereo

MAS vs AI

$MAS \subset AI$ oppure $AI \subset MAS$

- *Agenti* \neq *AI*
- **Interconnessione** tra agenti non trattata in AI

MAS vs Economia e Teoria dei Giochi

- ◇ Teoria dei Giochi molto vicina ad AI ed Informatica
 - Teoria dei giochi analizza **cosa** accade quando due (o piú) agenti interagiscono, ma non **come** avviene l'interazione
 - Assunzioni molto forti (capacità di ragionamento, conoscenza e sincronizzazione)

- ◇ Scienze Sociali
 - MAS sono uno strumento per le scienze sociali

Applicazioni

- risoluzione di problemi distribuita (planning, scheduling)
- acquisizione informazioni (cooperative perception, information retrieval)
- e-commerce
- simulazioni di società di agenti (Biologia, Scienze sociali, EOS)
- costruzione di mondi sintetici (Rescue)
- sistemi multi-robot

Gestione dei Processi Produttivi

- ◇ risoluzione di problemi distribuita
- ◇ ADEPT (Jennings et al.)
 - società modellata come agenti che negoziano e forniscono servizi
 - fornire preventivi per installazione di reti di telecomunicazione (BT)
 - basato su CORBA, GRATE* ed ARCHON

Agenti per l'assistenza agli umani

- ◇ risoluzione di problemi distribuita
- ◇ ELVES PDA autonomi (Tambe et al.)
 - Coordinare un gruppo di persone per organizzare un meeting
 - Basato su Adjustable Autonomy(AA) e Markov Decision Process(MDP)
 - Realizzato in Java su palmari

Sistemi Multi Robot

- ◇ Comunità molto divisa dai MAS sino ad ora
 - FIRE: space exploration
 - SDR, CENTIBOTS: surveillance
 - RESCUE: victims search and rescue in post disaster scenario

Interazione

Situazione determinata dal comportamento di un gruppo di agenti.

Fattori critici:

- obiettivi (compatibili/incompatibili)
- risorse (conflitti)
- capacità (skills)

Tipi di Interazione

- indipendenza (C,S,S)
 - collaborazione semplice (C,S,I)
 - ostruzione (C,I,S)
 - collaborazione coordinata (C,I,I)
 - pura competizione individuale (I,S,S)
 - pura competizione collettiva (I,S,I)
 - conflitti individuali sulle risorse (I,I,S)
 - conflitti collettivi sulle risorse (I,I,I)
 - ragionamento non-monotono
-
- ritrattazioni
 - spiegazioni
 - ragionamento non-monotono

Forme di cooperazione

- intenzionale (obiettivo comune)
- reattiva(osservata)
- aumento delle capacità di sopravvivenza
- aumento delle prestazioni
- risoluzione di conflitti

Caratteristiche del sistema cooperativo

- comunicazione
- specializzazione
- numerosità
- architettura (reattiva/intenzionale)

Forme di cooperazione

- collaborazione (distribuzione dei compiti)
 - centralizzata
 - distribuita
 - negoziazione
- coordinamento
 - sincronizzazione delle azioni
 - uso delle risorse
- allocazione emergente

Modellazione degli altri agenti

- obiettivi/ruoli
- capacità/skills
- conoscenze

Ragionamento sulla conoscenza

Un re che voleva conoscere chi fosse il più saggio tra tre pretendenti alla mano della figlia, pose un cappello bianco sulla testa di ciascuno di essi. I pretendenti potevano vedere il cappello degli altri ma non il proprio. Il re disse:

“Almeno uno dei tre cappelli è bianco”

e chiese al primo pretendente (P1):

“Qual’è il colore del tuo cappello?”

P1: “Non so”

poi chiese al secondo pretendente (P2):

“Qual’è il colore del tuo cappello?”

P2: “Non so”

poi chiese al terzo pretendente (P3):

“Qual’è il colore del tuo cappello?”

P3: “Bianco”

e P3 sposò la figlia del re

Comunicare tra agenti

- ◇ Caratteristica fondamentale per sistemi distribuiti e quindi anche per i MAS
- ◇ permette di sincronizzare diversi processi
- ◇ comunicazione negli agenti differisce dalla comunicazione tra processi:

$$O_2.exec(O_1.store(D)) \neq A1tellsA2aboutD$$

- ◇ Differenza fondamentale: autonomia degli agenti

Speech Acts (Austin)

- ◇ Le comunicazioni **sono** azioni e **cambiano** lo stato del mondo (Austin)
- ◇ Categorie per gli Speech Acts (Performative verbs):
 - Request
 - Inform
 - Promise
- ◇ Aspetti degli Speech Acts
 - Locutionary (fai questo!)
 - Illocutionary (A mi ha detto di fare questo)
 - Perlocutionary (A mi ha fatto fare questo)

Speech Acts (Austin) II

- ◇ Condizioni di successo per uno Speech acts
 - Deve esistere un a procedura da seguire
 - La procedura deve essere seguita completamente e senza errori
 - Gli agenti devono eseguire la procedura in maniera collaborativa e “sincera” .

Speech Acts (Searle)

◇ Condizioni per far sì che uno speech act abbia effetto (succes conditions)

- Condizioni normali di I/O
- Precondizioni
- Condizioni di sincerità

◇ diversa sistematizzazione degli speech act

- Rappresentative (Es inform)
- Direttive (Es request)
- Commissive (Es promise)
- Espressive (Es thanking)
- Dichiarative (Es declaring war)

Integrare Speech Acts e Planning

- ◇ Formalizzazione degli Speech Act per poterli integrare nella pianificazione come le altre azioni ordinarie
- ◇ [Cohen-Penaults] hanno utilizzato notazione STRIP-like
→ precondizioni e postcondizioni rappresentate utilizzando logiche modali.

Es. $Request(S, H, \alpha)$ e $CauseToWant(A_1, A_2, \alpha)$ $Request(S, H, \alpha)$

- Pre: $(S\text{Bel}(H\text{CanDo}\alpha) \wedge S\text{Bel}(H\text{Bel}(H\text{CanDo}\alpha))) (S\text{Bel}(S\text{WantReqInst}))$
- Post: $(H\text{Bel}(S\text{Bel}(S\text{Want}\alpha)))$

Integrare Speech Acts e Planning II

CauseToWant(A_1, A_2, α)

- Pre: ($A_1 \text{Bel}(A_2 \text{Bel}(S \text{Want} \alpha))$)
- Post: ($A_1 \text{Bel}(A_1 \text{Want} \alpha)$)

◇ Ugualmente *Inform*(S, H, ϕ) *Convince*(A_1, A_2, ϕ)

Linguaggi di comunicazione fra agenti

◇ Knowledge Sharing Effort:

Progetto per Sviluppare protocolli per lo scambio di informazioni tra sistemi di informazione autonomi.

- KIF (Knowledge Information Format) Linguaggio per rappresentare la conoscenza
- KQML (Knowledge Query Manipulation Language) Linguaggio per incapsulare messaggi tra agenti astratto rispetto al contenuto del messaggio

KIF

- Basato su logica del primo ordine e Lisp
- Può specificare:
 - proprietà di oggetti in un dominio
 - relazioni tra oggetti in un dominio
 - proprietà generiche del dominio

KQML

Linguaggio basato sui messaggi Un messaggio KQML é un oggetto con:

- Performative specifica il tipo di messaggio
- Lista di coppie Attr/Val

Ask-One(

:content(contenuto del messaggio)

:Receiver(ricevente)

:Language(Linguaggio di specifica dei messaggi)

:Ontology(ontologia)

)

Problemi rilevati per KQML

- Performative non ben definite
- Semantica del linguaggio non ben definita
- Primitive per invio messaggi non ben definite
- Non sono presenti performative commissive

◇ lo standard FIPA

FIPA

- ◇ Federation for Intelligent Physical Agents
 - Molto simile al KQML
 - Performative standardizzate
 - Semantica definita in SL (Linguaggio formale)

- ◇ Semantica definita con:
 - **Feasibility conditions:** Vincoli che devono essere rispettati affinché l'agente possa spedire un messaggio. I vincoli sono ottenuti traducendo il Messaggio FIPA in una formula SL
 - **Rational Effect:** (Possibili) effetti dell'invio del messaggio

FIPA II

◇ Tutte le performative sono rappresentate tramite **inform** e **request**

definizione di **Inform**:

$(i, Inform(j, \phi))$

Feasibility Conditions: $B_i\phi \wedge \neg B_i(Bif_j\phi \vee Uif_j\phi)$

Rational effects: $B_j\phi$

Test di conformità semantica: Dato un agente A ed un Linguaggio di comunicazione ACL dire se A rispetta la semantica di ACL ogni volta che comunica.

Ontologie

- Terminologia utilizzata per descrivere un dominio
- Specifica formale di un DataBase
- KIF é un linguaggio per definire ontologie
- XML e DAML sono linguaggi di specifica per ontologie

Linguaggi per il Coordinamento

◇ Modello Blackboard

- Agenti coinvolti nella risoluzione di un problema accedono ad una struttura dati condivisa
- LINDA collezione di costrutti linguistici per definire un modello BlackBoard