

## Introduzione alla rappresentazione della conoscenza

ovvero...  
Come costruire agenti basati su conoscenza e dotati di capacità di ragionamento

Maria Simi, 2010/2011

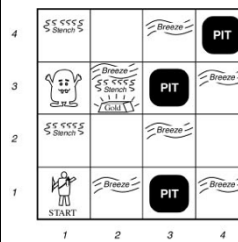
## Che cosa abbiamo fatto fin'ora ...

- Abbiamo trattato:
  - agenti con stato e con obiettivo, più razionali rispetto ad agenti reattivi
  - stati e azioni descrivibili in maniera semplice
  - enfasi sul processo di ricerca
- Vogliamo adesso migliorare le capacità razionali dei nostri agenti dotandoli di rappresentazioni di mondi più complessi, non descrivibili semplicemente
- Agenti *basati su conoscenza*, con conoscenza espressa in maniera esplicita e dichiarativa (non cablata)

## Perché?

- Il mondo è tipicamente complesso: ci serve una rappresentazione parziale e incompleta di una astrazione del mondo utile agli scopi dell'agente
- Per ambienti parzialmente osservabili ci servono linguaggi di rappresentazione della conoscenza più espressivi e capacità inferenziali
- La maggior parte dei problemi di I.A. sono "knowledge intensive" tanto che Sistemi Basati sulla Conoscenza è quasi sinonimo di sistemi di I.A.

## Il mondo del Wumpus: un esempio



- Misura delle prestazioni:
  - +1000 se trova l'oro, torna in [1,1] e esce;
  - 1000 se muore;
  - 1 per ogni azione;
  - 10 se usa la freccia.
- Percezioni:
  - puzza nelle caselle adiacenti al Wumpus;
  - brezza nelle caselle adiacenti alle buche;
  - luccichio nelle caselle con l'oro;
  - bump se sbatte in un muro;
  - urlo se il Wumpus viene ucciso.
  - L'agente non percepisce la sua locazione.
- Azioni:
  - avanti
  - a destra di 90°, a sinistra di 90°
  - afferra un oggetto
  - scaglia la freccia (solo una)
  - Esce
- Ambienti generati a caso ([1,1] safe)

## Il mondo del Wumpus: uno scenario

1,4	2,4 P?	3,4	4,4
1,3 W!	2,3  S G B	3,3 p?	4,3
1,2 S V OK	2,2 V OK	3,2	4,2
1,1 V OK	2,1 B V OK	3,1 p?	4,1

(b)

- L'agente si sposta in [2,2] e poi in [2,3]
- Qui percepisce un luccichio, afferra l'oro e torna sui suoi passi, percorrendo caselle OK

## Agente basato su conoscenza

- Un agente basato su conoscenza mantiene una base di conoscenza (KB): un insieme di enunciati espressi in un linguaggio di rappresentazione
- Interagisce con la KB mediante una interfaccia funzionale Tell-Ask:
  - Tell: per aggiungere nuovi fatti a KB
  - Ask: per interrogare la KB
  - ... forse Retract
- Le risposte  $\alpha$  devono essere tali che  $\alpha$  segue logicamente da KB (è conseguenza logica di KB)

## Il problema da risolvere

- *Il problema*: data una base di conoscenza KB, contenente una rappresentazione dei fatti che si ritengono veri, vorrei sapere se un certo fatto  $\alpha$  è vero di conseguenza

KB  $\models \alpha$  (conseguenza logica)

## Programma di un agente B.C.

**Function** Agente-KB (*percezione*) **returns** un'azione  
**persistent**: KB, una base di conoscenza  
 $t$ , un contatore, inizialmente a 0, che indica il tempo  
TELL(KB, Costruisci-Formula-Percezione(*percezione*,  $t$ ))  
*azione*  $\leftarrow$  ASK(KB, Costruisci-Query-Azione( $t$ ))  
TELL(KB, Make-Action-Sentence(*azione*,  $t$ ))  
 $t \leftarrow t + 1$   
**return** *azione*

## Agente basato su conoscenza

- Approccio dichiarativo vs approccio procedurale
- La differenza principale è che la KB racchiude tutta la conoscenza necessaria a decidere l'azione da compiere in forma *dichiarativa*
- L'alternativa (*approccio procedurale*) è scrivere un programma in cui il processo decisionale è cablato, una volta per tutte.
- Più flessibile: più semplice acquisire conoscenza incrementalmente e modificare il comportamento con l'esperienza

## Base di conoscenza ...

- *Base di conoscenza*: una rappresentazione esplicita, parziale e compatta, in un linguaggio simbolico, che contiene:
  - fatti di tipo specifico (Es. *Socrate è un uomo*)
  - fatti di tipo generale (Es. *Tutti gli uomini sono mortali*)
- Quello che caratterizza una B.C. è la *capacità di inferire* nuovi fatti da quelli memorizzati esplicitamente (Es. *Socrate è mortale*)

... e base di dati

## ... e base di dati

- Nelle basi di dati solo fatti specifici e positivi
- Le basi di dati assumono una conoscenza completa del mondo (Closed World Assumption)
- Nessuna capacità inferenziale (vincoli di integrità solo per il controllo, non per la generazione)

## Il *trade-off* fondamentale della R.C.

- Il problema 'fondamentale' in R.C. è trovare il giusto compromesso tra:
  - *Espressività* del linguaggio di rappresentazione;
  - *Complessità* del meccanismo inferenziale
- Vogliamo meccanismi *espressivi*, ma anche *efficienti*.
- Questi due obiettivi sono in contrasto e si tratta di mediare tra queste due esigenze

## Espressività come imprecisione

- Cosa vuol dire *espressivo*? ... e perchè l'espressività è in contrasto con l'efficienza?
- Un linguaggio più espressivo ci consente di essere vaghi, imprecisi, di esprimere conoscenze parziali, di omettere dettagli che non si conoscono ...
- L'espressività determina non tanto quello che può essere detto ma quello che può essere *lasciato non detto*

## Espressività come imprecisione : esempi

Nelle B.D. quello che possiamo esprimere sono solo fatti specifici e positivi:

1.  $Moglie(Rossi, Paola)$

Con linguaggi più espressivi ...

2.  $\exists x Moglie(Rossi, x)$       *Rossi ha una moglie*
3.  $\neg Operaio(Rossi)$       *Rossi non è un operaio*
4.  $Moglie(Rossi, Anna) \vee Moglie(Rossi, Paola)$   
*Rossi ha una moglie; si chiama Anna o Paola*
5.  $\forall y \exists x Moglie(y, x) \Rightarrow Coniugato(y)$   
*Coloro che hanno una moglie sono coniugati*

## Espressività e complessità inferenziale

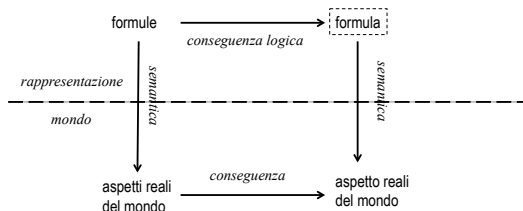
- Nelle basi di dati nessuna deduzione è possibile, solo *recupero*. Si assume una descrizione *completa* del mondo.
- Dai fatti
  - $\exists x Moglie(Rossi, x)$
  - $\forall y \exists x Moglie(y, x) \Rightarrow Coniugato(y)$
 è possibile dedurre  $Coniugato(Rossi)$
- Dai fatti
  - $Moglie(Rossi, Anna) \vee Moglie(Rossi, Paola)$
  - $\forall y \exists x Moglie(y, x) \Rightarrow Coniugato(y)$
 è possibile dedurre  $Coniugato(Rossi)$  ma è più complicato (richiede un ragionamento per casi)

## Formalismi per la R.C.

Un formalismo per la rappresentazione della conoscenza ha tre componenti:

1. Una *sintassi*: un linguaggio composto da un vocabolario e regole per la formazione delle frasi (*enunciati*)
2. Una *semantica*: che stabilisce una corrispondenza tra le frasi e fatti del mondo; se un agente ha un enunciato  $\alpha$  nella sua KB, crede che il fatto corrispondente sia vero nel mondo
3. Un *meccanismo inferenziale* (codificato o meno tramite regole di inferenza come nella logica) che ci consente di inferire nuovi fatti.

## Rappresentazione e mondo



## Grounding (radicamento)

- Come sappiamo che la KB è vera nel mondo reale? Come l'agente forma le sue credenze?
- Attraverso i sensori si crea una connessione con il mondo; le credenze sono il risultato di percezioni.
- Non solo: le regole sono il risultato di un processo di apprendimento, che può essere fallibile (es. ragionamento induttivo).

## Logica come linguaggio per la R.C.

- I linguaggi logici, calcolo proposizionale (Prop) e logica dei predicati (FOL), sono adatti per la rappresentazione della conoscenza?
- Complessità computazionale del problema  $KB \models \alpha$ 
  - In PROP il problema è decidibile, ma intrattabile (NP)
  - FOL è un linguaggio espressivo, con una semantica ben definita, ma ha un problema: il meccanismo inferenziale non è decidibile
  - In FOL il problema  $KB \models \alpha$  è semidecidibile

## Linguaggi per la R.C. : efficienza

1. Superamento del FOL verso linguaggi ad inferenza limitata: *contrazioni* del FOL alla ricerca di proprietà computazionali migliori (es. i linguaggi di *programmazione logica*, le logiche *descrittive*)
2. Linguaggi di rappresentazione che propongono meccanismi di strutturazione della conoscenza per guadagnare efficienza su forme particolari di inferenza (es. *reti semantiche* e connettività, *frame* e aggregazione, *ereditarietà*). FOL per la semantica

## Limiti in espressività del FOL

Molti linguaggi della R.C. sono *estensioni* [di sottoinsiemi] del FOL per superare limiti di espressività nel ragionamento di "senso comune"

Ne possiamo citare tre importanti:

- Atteggiamenti proposizionali
- Ragionamento incerto
- Ragionamento non monotòno

## Atteggiamenti proposizionali

- Atteggiamenti epistemici
  - conoscenze, credenze (convinzioni o opinioni)
- Atteggiamenti motivazionali
  - desideri, obiettivi, intenzioni, ...
- L'oggetto del discorso sono le proposizioni
  - Bel(P) operatori e logiche modali
  - Bel('P') reificazione o meta-livello

## Ragionamento incerto

- Nella logica classica le proposizioni sono vere o false (*assunzione epistemologica*)
- Il superamento della dicotomia T|F può avvenire in modi diversi:
  - logiche a più valori (vero, falso, non so)
  - ragionamento probabilistico (vero con probabilità  $p$ )
  - vero con grado di fiducia  $c$
  - logica fuzzy (proprietà sfumate, es. 'alto' in misura  $m$ )

## Ragionamento non monotòno

Nella logica classica vale la proprietà di *monotonia*:

*Monotonia*: Se  $KB \models \alpha$  allora  $KB \cup \{\beta\} \models \alpha$

- Il ragionamento di senso comune è spesso *non monotòno*: si fanno inferenze tentative, anche in mancanza di informazioni complete.
- Esempio 1: ragionamento *default*  
*Gli uccelli tipicamente volano. Tweety è un uccello. Quindi Tweety vola.*
- Esempio 2: *assunzione di mondo chiuso*  
Se un fatto non è presente nella KB si assume che non sia vero (come nelle basi di dati). Quando si aggiunge un nuovo fatto può invalidare le vecchie conclusioni.

## Assunzioni ontologiche

- Ogni linguaggio per la R. C. fa assunzioni diverse su come è fatto il mondo (ontologico  $\cong$  che riguarda ciò che esiste):
- Nel calcolo proposizionale il mondo è visto come popolato di *fatti veri* o *falsi* (le proposizioni).
- Il calcolo dei predicati fa una *assunzione ontologica* più sofisticata: il mondo è fatto di *oggetti*, che hanno *proprietà* e tra cui sussistono *relazioni*.
- Logiche specializzate assumono *ontologie* più ricche:
  - gli stati e le azioni nel *calcolo di situazioni*
  - il tempo nelle *logiche temporali*
  - Concetti o categorie nelle *logiche descrittive*