

## Prodromi di Prolog

*Sistemi intelligenti distribuiti LS*  
2004/2005  
Prof. Andrea Omicini

## 1 – Cos'è la logica?

- *(Senza voler essere formali nella definizione)*
- È la scienza che studia i meccanismi del ragionamento umano, o razionale
  - senza voler indagare sulla coincidenza dei due termini...
- È una scienza che da sempre cerca di coniugare la ricchezza del pensiero e delle idee con la precisione dei sistemi formali
  - non a caso la sua storia intreccia filosofi e matematici
- *Libro interessante*
  - Piergiorgio Odifreddi. Le menzogne di Ulisse. L'avventura della logica da Parmenide ad Amartya Sen. Longanesi, 2004

## 2 – Perché ci interessa la logica?

- *Alcune motivazioni classiche*
  - capire e riprodurre il ragionamento umano
  - verificare la correttezza di un ragionamento
  - “meccanizzare” il processo razionale
    - nel senso di delegarlo alle macchine
  - più varie ragioni metafisiche che non citiamo
- *Ovviamente, a noi interessa soprattutto per la terza ragione*
  - anche se le nostre macchine sono “meccaniche” in senso lato

## 3 – Quale parte della logica ci interessa?

- *(Logica proposizionale)*
  - per introdurre con ordine le cose che ci servono
- *Logica del prim'ordine*
  - FOL (First-Order Logic)
  - è il nostro fuoco principale
- *(In minima parte, la logica del second'ordine)*
- *E comunque, ci servono solo alcune conoscenze ben selezionate che citeremo esplicitamente*

## 4 – Cosa ci serve della logica proposizionale?

- *Il concetto di proposizione*
  - es.,  $p$  è una proposizione
- *Il concetto di interpretazione*
  - $p$  può rappresentare il fatto che fuori piove, o che il prof. è presente
  - l'interpretazione assegna un valore di verità alla proposizione in un dato “mondo”
- *I connettivi logici, con le loro tabelle di verità*
  - $\wedge \vee \neg \Rightarrow$  (e derivati)
- *Le formule logiche*
  - ottenute combinando proposizioni con i connettivi, es.  $p \wedge q \Rightarrow r$
- *La valutazione di una formula via tabelle di verità*
  - che dicono se una formula è vera o falsa secondo una data interpretazione in un dato mondo

## 5 – Cosa dobbiamo sapere della FOL?

- Il dominio del discorso si rappresenta in maniera più articolata*
  - le proposizioni sono espresse tramite predicati, costanti, variabili, funzioni
  - essendoci le variabili, oltre ai connettivi logici ci sono i quantificatori
- L'interpretazione è più articolata*
  - pre-interpretazione
- Il concetto di teoria logica e di conseguenza logica*
  - $D \models G$

## 5a – Il dominio del discorso in FOL (I)

- costanti, funzioni, termini ground
  - se  $a$  è una costante,  $f$  una funzione di arità 1, allora  $f(a)$  è un termine ground
  - Universo di Herbrand: insieme di tutti i termini ground
    - rappresenta l'insieme di tutte le entità di cui si può parlare
- variabili, termini
  - rappresentano un termine non noto a priori
  - una funzione applicata a costanti, variabili e funzioni è un termine
    - un termine che non contiene variabili è ground

## 5a – Il dominio del discorso in FOL (II)

- predicati
  - un predicato del prim'ordine si applica a termini, e produce atomi (formule atomiche)
    - atomi ground
    - Base di Herbrand: insieme di tutte le formule atomiche esprimibili
      - rappresenta l'insieme delle "verità" possibili
- ai predicati che predicano su termini che contengono variabili si applicano i quantificatori
  - $\forall X p(X,a)$
  - $\exists Y q(b,c,Y)$
- rispetto alla logica proposizionale, FOL è più espressiva
  - e ovviamente ha meno proprietà...

## 5b – L'interpretazione è più articolata

- I simboli non sono interpretati in sé
  - se padre/1 è una funzione, non è detto che il suo significato sia la funzione che noi saremmo portati automaticamente ad assegnarle
    - quindi neppure 1 è il numero 1, a priori...
- Due "fasi" concettuali
  - pre-interpretazione
    - interpretazione degli enti del dominio del discorso
      - costanti, funzioni, termini ground: mapping sull'universo di Herbrand
  - interpretazione
    - interpretazione delle relazioni tra enti
      - predicati, termini: valori di verità per la Base di Herbrand

## 5c – Teoria logica e conseguenza logica

- Il valore di verità di una formula atomica  $A$  dipende dalla interpretazione
  - la verità di una formula non atomica  $F$  dipende dalla interpretazione e dalle tabelle di verità dei connettivi
  - un'interpretazione che rende vera  $F$  è un modello per  $F$
  - una teoria che non ammette modelli è insoddisfacibile
- $D \models G$ 
  - $D$  sia l'insieme delle formule considerate vere (o assiomi)
  - $G$  la formula di cui dobbiamo stabilire il valore di verità
  - $G$  deriva logicamente da  $D$  se  $G$  risulta vera secondo tutte le possibili interpretazioni che rendono vera  $D$ 
    - nozione model-theoretic

## 6 – Quale relazione tra logica e calcolo?

- FOL semi-decidibile (Mendelsson 1972)
  - Se  $D \models G$ , allora esiste un procedimento meccanico finito di derivazione ( $D \vdash G$ ,  $G$  è derivabile da  $D$ )
  - Se  $D \not\models G$ , allora tale procedimento non è detto che esista
- Esiste un frammento significativo della logica del prim'ordine che è computabile in senso lato
  - ossia, per una parte limitata ma ampia della FOL, dati gli assiomi, è possibile calcolare se una formula è vera o falsa
- Principio di Robinson
  - Se limitiamo la FOL alle clausole, allora esiste un procedimento di dimostrazione automatica dei teoremi
  - Correttezza e completezza:  $D \models G \Leftrightarrow D \vdash G$

## 6a – Cosa sono le clausole?

- Letterali:  $A, \neg A$
- Clausola: disgiunzione di letterali
  - $\equiv \forall X1, \dots, \forall Xk (A1 \vee \dots \vee An \vee \neg B1 \vee \dots \vee \neg Bm)$ 
    - $\equiv \neg(F \wedge F') = \neg F \vee \neg F'$
    - $\equiv \forall X1, \dots, \forall Xk (A1 \vee \dots \vee An \vee \neg(B1 \wedge \dots \wedge Bm))$ 
      - $\equiv F \leftarrow F' = F \vee \neg F'$
      - $\equiv \forall X1, \dots, \forall Xk ((A1 \vee \dots \vee An) \leftarrow (B1 \wedge \dots \wedge Bm))$ 
        - $\circ A1, \dots, An \leftarrow B1, \dots, Bm$
  - Clausola definita ( $n = 1$ ):  $A \leftarrow B1, \dots, Bm$
  - Clausola unitaria ( $m = 0$ ):  $A \leftarrow$
  - Goal definito ( $n = 0$ ):  $\leftarrow B1, \dots, Bm$
  - Clausola di Horn:  $n = 0$  oppure  $n = 1$

## 6b – Cos'è l'unificazione?

- Una sostituzione è un insieme di assegnamenti di termini a simboli di variabili
  - una sostituzione  $\sigma$  si applica a un termine  $t$  ( $t\sigma$ ) o a un atomo  $A$  ( $A\sigma$ ) sostituendo a ogni variabile in  $t$  o  $A$  il corrispondente termine in  $\sigma$
  - un renaming è una sostituzione che cambia i nomi delle variabili
  - (sostituzioni)  $\sigma$  è più generale di  $\vartheta$  se esiste una  $\lambda$  per cui  $\vartheta = \sigma\lambda$ .
- Due (o più) termini o atomi sono unificabili se esiste una sostituzione che applicata ad essi li rende uguali
  - tale sostituzione si chiama sostituzione unificatrice
- Il most general unifier (mgu) tra due termini è la sostituzione unificatrice più generale
  - che, se esiste, è unica a parte renaming

## 6c – Cosa è il Principio di Robinson? [Risoluzione per clausole generali]

- Clausole genitrici
  - $C1 = A1 \vee \dots \vee An$
  - $C2 = B1 \vee \dots \vee Bm$
- Se  $Ai = \sim p(t1, \dots, tk)$ ,  $Bj = p(t1', \dots, tk')$ , oppure  $Ai = p(t1, \dots, tk)$ ,  $Bj = \sim p(t1', \dots, tk')$
- e  $\vartheta = mgu(p(t1, \dots, tk), p(t1', \dots, tk'))$
- allora possiamo derivare da  $C1$  e  $C2$  la clausola (risolvente)  
 $C3 = [$   
     $A1 \vee \dots \vee Ai-1 \vee Ai+1 \vee \dots \vee An \vee$   
     $B1 \vee \dots \vee Bj-1 \vee Bj+1 \vee \dots \vee Bm ]\vartheta$

## 6d – Qual è il rapporto tra conseguenza logica e derivazione? (I)

- Se l'insieme di clausole  $H \cup \{\sim F\}$  è insoddisfacibile, mentre  $H$  è soddisfacibile, allora  $F$  segue logicamente da  $H$
- Se da un insieme di clausole posso derivare la contraddizione logica, esso è insoddisfacibile
- Se, applicando la risoluzione, arrivo al risolvente vuoto, questo rappresenta la contraddizione logica ( $\square$ ): da  
     $C1 = \sim p(t1, \dots, tk)$ ,  $C2 = p(t1', \dots, tk')$ , oppure  
     $C1 = p(t1, \dots, tk)$ ,  $C2 = \sim p(t1', \dots, tk')$ , con  
     $\vartheta = mgu(p(t1, \dots, tk), p(t1', \dots, tk'))$ , deriviamo  
     $C3 = \square$

## 6d – Qual è il rapporto tra conseguenza logica e derivazione? (II)

- Intuitivamente
  - derivo due atomi in contraddizione tra loro
  - derivo la contraddizione logica
  - non ci sono modelli per l'insieme di clausole  $H \cup \{\sim F\}$
  - l'insieme  $H \cup \{\sim F\}$  è insoddisfacibile
  - $F$  è una conseguenza logica di  $H$
- Collegamento tra derivazione
  - processo sintattico e automatizzabile
- e valore di verità
  - elemento di carattere semantico

## 6e – Cosa ci dice il Principio di Robinson?

- Ci dice che
  - limitandosi alle clausole generali
- esiste un metodo per la derivazione automatica di teoremi
  - arrivo a risultati semanticamente corretti con metodi sintattici e automatizzabili
- Quanto è riduttiva questa limitazione?
  - esiste un procedimento per ridurre ogni formula FOL alla forma clausale, quindi non limita granché...
- Cosa manca per arrivare a un linguaggio di programmazione logica?
  - un meccanismo implementabile efficientemente
  - una visione del rapporto tra dimostrazione e computazione

## 7 – Qual è il meccanismo che ci serve?

- Risoluzione Lineare per Clausole Definite con funzione di Selezione (Risoluzione SLD)
  - NOTA: completa strettamente per Clausole di Horn, usiamo quelle
- Sono dati una teoria logica  $P$  e una clausola goal  $G0$ 
  - ad ogni passo di risoluzione, dalla clausola goal ottenuta al passo precedente  $Gi$  (risolvente) e da una variante di una clausola appartenente a  $P$  si ricava, se possibile, un nuovo risolvente  $Gi+1$ 
    - selezionando un atomo  $Am$  dal risolvente  $Gi$
    - trovando almeno una clausola  $Ci$  in  $P$  la cui testa  $A$  unifici con  $Am$
    - determinando la sostituzione più generale  $\theta_i = mgu(A, Am)$
    - ottengo  $Gi+1$  sostituendo  $Am$  in  $Gi$  con  $B\theta_i$ , ove  $B$  è il corpo di  $Ci$
  - se  $Gi+1$  è vuoto, derivazione di successo:  $P \vdash G0$ , da cui  $P \models G0$

## 7a – La risoluzione SLD è implementabile?

- Corretta e completa per Clausole di Horn
- Due cose non abbiamo detto
  - come si seleziona l'atomo del risolvete, se ce ne sono più d'uno
    - regola di calcolo
  - come si sceglie la clausola unificante, se ce ne sono più d'una
    - strategia di ricerca
- Sono due fonti di non determinismo
  - e due possibili strade per linguaggi di programmazione logica
- Proprietà (Indipendenza dalla regola di calcolo)
  - Dato l'insieme di clausole  $P$ , l'insieme dei goal  $G_0$  per cui  $P \vdash G_0$  non dipende dalla regola di calcolo utilizzata dalla risoluzione SLD

## 7b – Cos'è un albero SLD?

- Rappresentazione grafica ad albero della risoluzione SLD
  - ogni nodo è un goal (risolvete)
  - la radice è il goal iniziale  $G_0$
  - ogni nodo ha un figlio per ogni clausola la cui testa unifica con  $A_m$ 
    - $A_m$  è l'atomo selezionato dalla regola di calcolo applicata al nodo
- Una strategia di ricerca corrisponde dunque a un modo di costruire / esplorare l'albero SLD
  - visite: in profondità, in ampiezza, ...

## 8 – Cosa manca per arrivare alla PL?

- Avere una procedura di calcolo non significa avere un linguaggio di programmazione
  - sicuramente non un linguaggio di programmazione general-purpose
- Interpretazione dichiarativa
  - calcolo come dimostrazione di teoremi
    - gli assiomi più la procedura di calcolo mi consentono di calcolare la verità di teoremi, e la sostituzione che li rende veri
  - non ci basta...
- Interpretazione procedurale (Kowalski 1974)
  - ci consente di leggere direttamente un insieme di clausole come un programma tradizionale

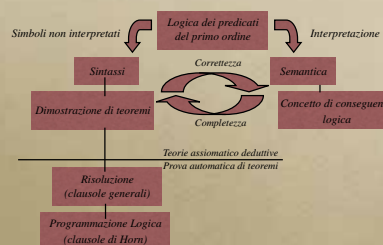
## 8a – Cos'è la interpretazione procedurale?

- Si può interpretare
  - un insieme di clausole come un insieme di definizioni di procedure
  - la testa di una clausola come la dichiarazione di una procedura
    - il predicato della testa di una clausola come il nome della procedura
  - il corpo di una clausola come il corpo di una procedura
  - l'insieme di tutte le clausole con lo stesso predicato nella testa come la definizione di una procedura non deterministica
  - una clausola vuota è la terminazione con successo
  - un goal come una chiamata di procedura
  - i termini nella testa della clausola come parametri formali
  - i termini nei goal come parametri attuali
  - la sostituzione calcolata come il risultato dell'invocazione del goal iniziale

## 8b – Cos'è la programmazione logica?

- È il paradigma di programmazione che
  - adotta le clausole di Horn come forma di programma
  - utilizza la derivazione di teoremi tramite risoluzione SLD come forma di calcolo
  - consente sia l'interpretazione dichiarativa sia quella procedurale dei suoi programmi e dei suoi processi di calcolo
- Un linguaggio logico è quindi un linguaggio di programmazione fondato sul paradigma della programmazione logica

## X – Schema riassuntivo



## 9 – Che fa il Prolog?

- Il linguaggio Prolog utilizza la risoluzione SLD con le seguenti scelte
  - regola di calcolo
    - regola left-most
      - dato un risolvete  $\leftarrow G1, G2, \dots, Gn$
      - viene sempre selezionato il letterale più a sinistra  $G1$
  - strategia di ricerca
    - in profondità (depth-first) con backtracking cronologico
    - seguendo l'ordine testuale delle clausole che li hanno generati
      - che quindi non è insieme di clausole...
- La strategia di ricerca adottata in Prolog è non completa

## 9a – Più precisamente...

- Dato un letterale  $G1$  da risolvere, viene selezionata la prima clausola (secondo l'ordine delle clausole nel programma  $P$ ) la cui testa è unificabile con  $G1$
- Nel caso vi siano più clausole la cui testa è unificabile con  $G1$ , la risoluzione di  $G1$  viene considerata come un punto di scelta (choice point) nella dimostrazione
- In caso di fallimento in un passo di dimostrazione, Prolog ritorna in backtracking all'ultimo punto di scelta in senso cronologico (il più recente), e seleziona la clausola successiva utilizzabile in quel punto per la dimostrazione

## 9b – Quali simboli in Prolog?

- Variabili come stringhe alfanumeriche con iniziale lettera maiuscola o underscore
  - underscore da solo variabile anonima
- Funzioni, costanti e predicati come stringhe alfanumeriche con iniziale lettera minuscola
  - mamma (francesca, giuseppe) è un termine o un atomo?
    - mamma simbolo di funzione o di predicato?
    - problema o vantaggio?
      - meta-interpreti: programmi come dati
- Clausole
  - A :- B1, B2, ..., Bn.

## 9c – Cos'è un programma Prolog?

- Una sequenza di clausole di Horn scritte con la sintassi Prolog
- Interpretazione dichiarativa di un programma Prolog
  - le clausole definiscono la logica del programma
  - il Prolog incorpora il meccanismo di controllo
    - il programmatore Prolog specifica la logica della soluzione, non l'algoritmo completo: il cosa, non il come
- Interpretazione procedurale di un programma Prolog
  - un programma Prolog può anche essere costruito secondo una logica classica
  - con in più l'unificazione, la reversibilità dei parametri, il backtracking e il non-determinismo