

# Lezione 4

## DataLog

14/03/2006

1

### 4.1. Dalla lezione scorsa

- Abbiamo visto l'algoritmo di ricerca di una prova DataLog Proporzionale top down (o goal oriented)
- Notare 2 tipi di non determinismo
  - don't care
  - don't know
- Vediamo un esempio

14/03/2006

2

1. soleggiato ← sereno.
2. soleggiato ← nubi\_passeggere.
3. fresco ← nubi\_passeggere.
4. fresco ← ventoso
5. benessere ← fresco ∧ soleggiato.
6. nubi\_passeggere.

**Prendo 5**

**benessere?**

14/03/2006

3

1. soleggiato ← sereno.
2. soleggiato ← nubi\_passeggere.
3. fresco ← nubi\_passeggere.
4. fresco ← ventoso
5. benessere ← fresco ∧ soleggiato.
6. nubi\_passeggere.

**Scelgo soleggiato?  
Prendo 2.  
(e se prendessi 1?)**

**fresco?                    soleggiato?**  
 \_\_\_\_\_ 5  
 benessere

14/03/2006

4

1. soleggiato ← sereno.
2. soleggiato ← nubi\_passeggere.
3. fresco ← nubi\_passeggere.
4. fresco ← ventoso
5. benessere ← fresco ∧ soleggiato.
6. nubi\_passeggere.

**Scelgo fresco?  
Prendo 3  
(se prendessi 4?)**

**nubi\_passeggere?**  
 \_\_\_\_\_ 2  
 soleggiato  
**fresco?** \_\_\_\_\_ 5  
 benessere

14/03/2006

5

1. soleggiato ← sereno.
2. soleggiato ← nubi\_passeggere.
3. fresco ← nubi\_passeggere.
4. fresco ← ventoso
5. benessere ← fresco ∧ soleggiato.
6. nubi\_passeggere.

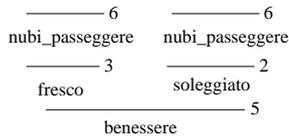
Andate avanti voi, si Arriva a

**nubi\_passeggere?    nubi\_passeggere?**  
 \_\_\_\_\_ 3                    \_\_\_\_\_ 2  
 fresco                    soleggiato  
 \_\_\_\_\_ 5  
 benessere

14/03/2006

6

1. soleggiato  $\leftarrow$  sereno.
2. soleggiato  $\leftarrow$  nubi\_passeggere.
3. fresco  $\leftarrow$  nubi\_passeggere.
4. fresco  $\leftarrow$  ventoso
5. benessere  $\leftarrow$  fresco  $\wedge$  soleggiato.
6. nubi\_passeggere.



14/03/2006

7

#### 4.2. Motore inferenziale: procedura Bottom-Up e modelli minimi

La procedura Bottom-Up genera dal basso tutti i fatti deducibili per **Modus Ponens**:

$$\frac{A \leftarrow B_1 \wedge \dots \wedge B_n \quad B_1, \dots, B_n}{A}$$

DEF. A **conseguenza immediata** di  $B_1, \dots, B_n$  e della la regola  $A \leftarrow B_1 \wedge \dots \wedge B_n$

14/03/2006

8

#### Conseguenza immediata e procedura bottom-up

- Operatore di conseguenza immediata  
Data KB e un'interpretazione  $I \subseteq \mathcal{P}$ :  
–  $T_{KB}(I) = I \cup \{H \mid H \text{ è un fatto di KB}$   
o una conseguenza immediata di I e di  
almeno una regola di KB}

#### PROCEDURA

$I := \{\};$   
while  $T_{KB}(I) \neq I$  do  $I := T_{KB}(I).$

14/03/2006

9

#### Esempio.

soleggiato  $\leftarrow$  sereno.  
soleggiato  $\leftarrow$  nubi\_passeggere.  
fresco  $\leftarrow$  nubi\_passeggere.  
fresco  $\leftarrow$  ventoso.  
benessere  $\leftarrow$  fresco  $\wedge$  soleggiato.  
ventoso.  
nubi\_passeggere

$TKB(\{\}) = I_0 = \{\text{ventoso, nubi\_passeggere}\}$

$TKB(I_0) = I_1 = \{\text{ventoso, nubi\_passeggere, soleggiato, fresco}\}$

$TKB(I_1) = I_2 = \{\text{ventoso, nubi\_passeggere, soleggiato, fresco, benessere}\}$

$TKB(I_2) = I_2 \quad \text{STOP}$

14/03/2006

10

#### Dimostrabilità bottom-up: proprietà

Il risultato della procedura si indica con  $KB^*$  :  
 $KB^* = I$  finale

**Theo. Validità:**  $A \in KB^* \rightarrow KB \models A$

**Theo. Completezza:**  $KB \models A \rightarrow A \in KB^*$

**Theo.** Esiste una prova di A, cioè  $KB \vdash A$ , sse  $A \in KB^*$

14/03/2006

11

- **Complessità:** lineare nella dimensione di KB  
– ogni clausola è usata al più una volta per aggiungere un atomo

- **Minimo punto fisso e minimo modello:**

**Teorema.**  $KB^*$  è un punto fisso di T:  $T(KB^*) = KB^*$

**Teorema.**  $KB^*$  è il minimo punto fisso

**Teorema.**  $KB^*$  è il modello minimo di KB.

- NOTA. La completezza segue dal fatto che  $KB^*$  è il modello minimo. Sapreste fare la dim.?
- NOTA: **ricordate l'assunzione del mondo chiuso?** L'assunzione del modello minimo ha a che fare? (ec)

14/03/2006

12

## Dove siamo

### PRIMO RRS

- DLP (DataLog Proporzionale)
  - Le prove
  - Algoritmo top-down (backward, goal oriented)
    - Non determinismo don't care e don't know
  - Algoritmo bottom-up (forward)
    - KB\* come minimo punto fisso e modello minimo
    - Assunzione del mondo chiuso
  - Validità e completezza dimostrate

14/03/2006

13

## 4.3. Estensione a DATALOG

- Tutte le proprietà viste e quanto detto sulle procedure bottom up e top down si estende pari pari a DATALOG

14/03/2006

14

### 4.3.1. DataLog: assunzioni

- **IR** (Individui e Relazioni): la conoscenza di un agente è descrivibile in termini di individui e relazioni
- **DK** (Definite Knowledge): la conoscenza di un agente è fornita da proposizioni definite e positive
  - Definite: non vaghe, escludo la disgiunzione
  - positive: escludo la negazione
- **SE** (Static Environment): l'ambiente non cambia (non c'è il tempo)
- **FD** (Finite Domain): l'universo di cui si parla è costituito da un insieme finito di individui.

14/03/2006

15

### Esercizio

- Ambito di problema:  
Un robot si muove in un ambiente formato da stanze in comunicazione fra loro e, tutte le sere, deve rimettere al loro posto gli oggetti che sono stati spostati durante il giorno. Conta gli spostamenti effettuati e li segna su file, che serve da archivio storico.
- Date le assunzioni di DataLog, individuare cosa si può modellare.
- Dare almeno un aspetto non modellabile usando DataLog, indicando quale delle assunzioni lo impedisce.

14/03/2006

16

### 4.3.2 DataLog: la sintassi

- **Costante**: formata da lettere o cifre, inizia con una lettera minuscola o è numerica; es: pippo33, il\_cane
- **Variabile**: formata da lettere, cifre, \_, inizia con maiuscola o \_  
es: X12, \_pippo
- **Simbolo di Predicato**: stessa sintassi delle costanti non numeriche
- **Termine**: costante o variabile
- **Atomo**:  $p(t_1, \dots, t_n)$ , con  $p$  predicato e  $t_1, \dots, t_n$  termini
  - esempio: padre(mario, X)
- **Body, Clausola definita, testa, KB**:
  - come nel proposizionale

14/03/2006

17

### Esempio

- **In un cortile** ci sono il **gatto felix**, il **cane fido** e un albero. Il gatto, quando **vede** il cane, **scappa sull' albero**.
- I colori suggeriscono l'astrazione:
  - Costanti: scritte **in rosso**
  - relazioni (in blu)
    - **vede** binaria
    - **in\_cortile, su\_albero, scappa, gatto, cane** unarie
- **Esempi di clausole**:  
 $in\_cortile(gatto)$ .  
 $scappa(X) \leftarrow gatto(X) \wedge vede(X, Y) \wedge cane(Y)$ . (ec)

14/03/2006

18

**Esercizio:** i seguenti colori suggeriscono un'astrazione diversa:

In un **cortile** ci sono il **gatto felix**, il **cane fido** e un **albero**.  
Il gatto, quando **vede** il cane, **scappa sull' albero**.

Dare l'alfabeto datalog corrispondente all'astrazione suggerita.

14/03/2006

19

### 4.3.3. DataLog: la semantica Interpretazioni

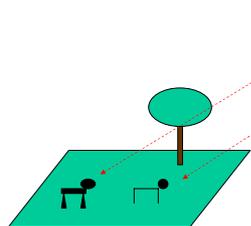
- Interpretazione  $I = (D, \Phi, \pi)$ 
  - D dominio di individui
  - $\Phi$  : simbolo di costante  $c \mapsto$  elemento  $c^\Phi \in D$
  - $\pi$  : simbolo di predicato n-ario  $p \mapsto$  (funzione  $p^\pi : D^n \rightarrow \{v, f\}$ )

14/03/2006

20

### Esempio

In un **cortile** ci sono il **gatto felix**, il **cane fido** e un albero.  
Il gatto, quando **vede** il cane, **scappa sull' albero**.



$fido^\Phi$   
 $felix^\Phi$

$gatto^\pi(felix^\Phi) = V$   
 $in\_cortile^\pi(fido^\Phi) = V$   
 $su\_albero^\pi(fido^\Phi) = F$   
ecc

14/03/2006

21

### 4.3.3.1. Interpretazioni (modelli) di Herbrandt

- Un'interpretazione di Herbrandt è un insieme di **atomi ground**
  - Un atomo è ground (o **chiuso**) se non contiene variabili
  - Un atomo con variabili è detto **aperto**
- Per poter rappresentare le interpretazioni alla Herbrandt in Datalog sono necessarie le ipotesi di **(ec)**
  - **nome unico** costanti diverse rappresentano individui diversi
  - **mondo chiuso per gli individui**: ogni individuo del dominio D è rappresentato da una costante
- Sotto queste ipotesi ha senso confondere, nella semantica interna, gli individui con le costanti che li denotano; è quanto avviene nelle interpretazioni di Herbrandt

14/03/2006

22

- **DEF. Interpretazione di Herbrandt per KB:**

$$H \subseteq \text{Atomi\_Ground}(KB)$$

- H rappresenta l'interpretazione  $I_H = (D, \Phi, \pi)$ :
  - D = insieme delle costanti
  - Per ogni costante c,  $c^\Phi = c$
  - Per ogni p n-ario,  $p^\pi(c_1, \dots, c_n) = V$  sse  $p(c_1, \dots, c_n) \in H$
- NOTA. I modelli di Herbrandt di una KB forniscono la semantica "interna", da distinguersi da quella "esterna"

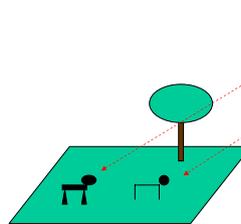
14/03/2006

23

### ESEMPIO

SEMANTICA ESTERNA

SEMANTICA INTERNA,  
Come modello di Herbrandt



$fido$   
 $felix$

$cane(fido)$   
 $gatto(felix)$   
 $in\_cortile(fido)$   
 $in\_cortile(felix)$   
 $vede(fido, felix)$

14/03/2006

24

### 4.3.3.2. Datalog: la semantica delle formule

- Diamo la semantica in versione semplificata, assumendo come interpretazioni quelle di Herbrandt
- Si veda il testo per la versione più generale, data usando la nozione di *assegnamento di valori alle variabili*
- *Quella più generale si usa quando vi siano individui non denotati da costanti.*

14/03/2006

25

### A) Semantica per le clausole ground

- Si usa la semantica già vista per il caso proposizionale, usando come simboli proposizionali gli atomi ground;
  - un atomo ground è vero in un'interpretazione di Herbrandt H sse appartiene ad H
  - i connettivi  $\leftarrow$   $\wedge$  si interpretano con le solite tavole di verità
- Per indicare che una clausola ground C è vera in un'interpretazione H scriveremo,
 
$$H \models C$$

14/03/2006

26

$H = \{ \text{cane(fido), gatto(felix), in_cortile(fido), in_cortile(felix), vede(fido,felix)} \}$  (ec)

$H \models \text{in_cortile(fido)} \leftarrow \text{vede(felix,fido)} ?$

$H \models \text{vede(felix,fido)} \leftarrow \text{vede(fido,felix)} ?$

14/03/2006

27

### B. Semantica per le clausole aperte

- Prendiamo la clausola:
 
$$\text{scappa}(X) \leftarrow \text{gatto}(X) \wedge \text{vede}(X,Y) \wedge \text{cane}(Y)$$
- È sottinteso che le variabili X,Y siano quantificate universalmente: per ogni X,Y ...
- Se il dominio D è {felix, fido}, la clausola è vera sse sono vere le sue istanze ground
 
$$\begin{aligned} \text{scappa(felix)} &\leftarrow \text{gatto(felix)} \wedge \text{vede(felix,felix)} \wedge \text{cane(felix)} \\ \text{scappa(felix)} &\leftarrow \text{gatto(felix)} \wedge \text{vede(felix,fido)} \wedge \text{cane(fido)} \\ \text{scappa(fido)} &\leftarrow \text{gatto(fido)} \wedge \text{vede(fido,felix)} \wedge \text{cane(felix)} \\ \text{scappa(fido)} &\leftarrow \text{gatto(fido)} \wedge \text{vede(fido,fido)} \wedge \text{cane(fido)} \end{aligned}$$

14/03/2006

28

### LA DEFINIZIONE FORMALE

- Istanza ground: si ottiene sostituendo uniformemente tutte le variabili con costanti
- **DEF.** Sia una clausola aperta;
 
$$H \models C(X_1, \dots, X_n)$$
 sse, per ogni istanza ground  $C(c_1, \dots, c_n)$ ,  $H \models C(c_1, \dots, c_n)$

14/03/2006

29

### 4.3.3.2. DataLog: i modelli di Herbrandt di una KB

- Una Base di Conoscenza è un insieme KB di clausole aperte, in un linguaggio con un dato insieme di costanti e simboli di predicato
- **DEF.** I modelli di Herbrandt di KB sono tutte e sole le interpretazioni di Herbrandt H tali che
 
$$H \models KB$$

14/03/2006

30

### Esercizio

$scappa(X) \leftarrow gatto(X) \wedge vede(X,Y) \wedge cane(Y).$

$gatto(felix).$

$cane(fido).$

- $\{gatto(felix)\}$  è modello?
- $\{cane(fido), gatto(felix)\}$  è modello?
- $\{cane(fido), gatto(felix), vede(felix,fido)\}$  è modello?
- $\{cane(fido), gatto(felix), vede(felix,fido), scappa(felix)\}$  è modello?
- $\{cane(fido), gatto(felix), vede(fido,felix)\}$  è modello?
- Qual è il modello minimo?