

COMPITO DI APPLICAZIONI DI INTELLIGENZA ARTIFICIALE

12 aprile 2006 (Punteggio su 30/30; Tempo 2h)

Esercizio 1 (punti 8)

Dato il seguente training set S:

Ruote	Motore	Ripara
due	benzina	+
tre	diesel	-
quattro	benzina	-
tre	benzina	+
tre	diesel	-
?	benzina	-
quattro	benzina	+
tre	diesel	-
due	benzina	-
?	diesel	+
tre	diesel	-
quattro	benzina	+
due	diesel	-
quattro	diesel	+
quattro	diesel	+

- Si calcoli l'entropia del training set rispetto all'attributo Ripara (punti 1)
- Si calcoli il rapporto di guadagno dei due attributi rispetto a questi esempi di training (punti 4)
- si costruisca un albero decisionale ad un solo livello per il training set dato, indicando le etichette delle foglie (numero di esempi finiti nella foglia/numero di esempi finiti nella foglia non appartenenti alla classe della foglia). (punti 1,5)
- si classifichi l'istanza: (punti 1,5)

?	benzina
---	---------

Esercizio 2 (punti 8)

Un telefono cellulare è dotato di un microprocessore che può funzionare a tre diverse frequenze di clock (1, 2 e 3 MHz). Il cellulare deve eseguire la stessa sequenza di operazioni ogni TD millisecondi. Ciascuna delle operazioni da eseguire ha una durata base DB, che è la durata dell'operazione se eseguita a 1MHz (se eseguita a 2MHz la durata sarà DB/2, se eseguita a 3MHz sarà DB/3). Ciascuna delle operazioni consuma energia dalla batteria. Il consumo per un'operazione di durata D eseguita alla frequenza F è pari a $E = (2 * F - 1) * D$. Si scriva un predicato CLP

`cell(ListaDurateBase, TD, ListaFreq)`

che prende in ingresso

- ListaDurateBase: le durate base delle operazioni da svolgere,
- TD: il tempo entro cui tutte le operazioni devono essere svolte

e che fornisce in uscita la ListaFreq, che contiene le frequenze a cui devono essere svolte le varie operazioni. Il predicato deve calcolare per ogni operazione la frequenza a cui deve essere eseguita, in modo tale che tutte le operazioni vengano svolte entro il tempo TD, minimizzando l'energia consumata.

Esempio:

```
?- cell([24,30,18],F,60).
```

```
Found a solution with cost 84, F = [1, 1, 3]
```

Si supponga che il linguaggio CLP utilizzato contenga il vincolo `sumlist(L,S)`, che vincola la variabile `S` ad assumere come valore la somma dei valori nella lista `L`.

Esercizio 3 (punti 8)

Si consideri il seguente problema: un carrello è nella posizione 'a' ed è pieno di pietre. Inoltre, vi è della ghiaia nella posizione 'b' connessa ad 'a'.

Il goal che si vuole raggiungere è che le pietre siano tutte in 'a' e che la ghiaia sia caricata nel carrello nella posizione 'b', avendo a disposizione le seguenti azioni

Scaricamento carrello

```
unload(Wagon,Item)
```

```
PREC: full(Wagon,Item), in(Wagon,X)
```

```
EFFECT: empty(Wagon), in(Item,X)
```

Caricamento carrello

```
load(Wagon,Item)
```

```
PREC: empty(Wagon), in(Wagon,X), in(Item,X)
```

```
EFFECT: full(Wagon,Item), ¬in(Item,X)
```

Spostamento del carrello tra due posizioni connesse

```
move(Wagon,Loc1,Loc2)
```

```
PREC: in(Wagon,Loc1), connected(Loc1,Loc2)
```

```
EFFECT: in(Wagon,Loc2), ¬in(Wagon,Loc1),
```

Stato iniziale:

```
in(c,a), full(c,p), connected(a,b), in(g,b)
```

Stato goal: `in(p,a), in(c,b), full(c,g)`

Si mostrino i passi compiuti dall'algorithm POP per risolvere il problema. Si mostrino i causal link e le eventuali minacce.

Esercizio 4 (punti 6)

Si descrivano le tecniche di risoluzione delle minacce nel partial order planning.

SOLUZIONE

Esercizio 1:

a) $\text{info}(S) = -7/15 * \log_2 7/15 - 8/15 * \log_2 8/15 = 0.997$

b)

Per calcolare il guadagno dell'attributo Ruote non si usa l'entropia calcolata su tutto il training set ma solo sugli esempi che hanno Ruote noto (insieme F):

$$\text{info}(F) = -6/13 * \log_2 6/13 - 7/13 * \log_2 7/13 = 0.996$$

$$\text{info}_{\text{Ruote}}(F) = 3/13 * (-1/3 * \log_2 1/3 - 2/3 * \log_2 2/3) + 5/13 * (-1/5 * \log_2 1/5 - 4/5 * \log_2 4/5) + 5/13 * (-4/5 * \log_2 4/5 - 1/5 * \log_2 1/5) =$$

$$= 0.231 * 0.918 + 0.385 * 0.722 + 0.385 * 0.722 = 0.768$$

$$\text{gain}(\text{Ruote}) = 13/15 * (0.996 - 0.768) = 0.198$$

$$\text{splitinfo}(\text{Ruote}) = -3/15 * \log_2(3/15) - 5/15 * \log_2(5/15) - 5/15 * \log_2(5/15) - 2/15 * \log_2(2/15) = 1.909$$

$$\text{gainratio}(\text{Ruote}) = 0.198 / 1.909 = 0.104$$

$$\text{info}_{\text{Motore}}(S) = 7/15 * (-4/7 * \log_2 4/7 - 3/7 * \log_2 3/7) + 8/15 * (-3/8 * \log_2 3/8 - 5/8 * \log_2 5/8) =$$

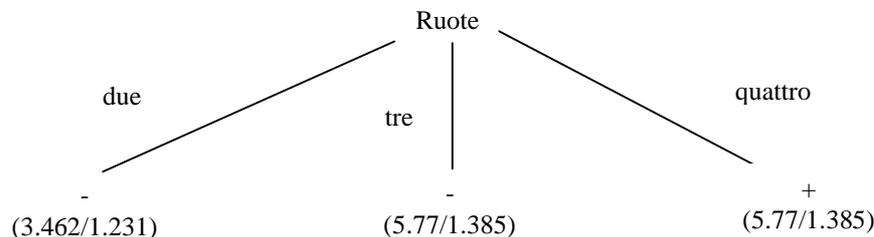
$$= 0.467 * 0.985 + 0.533 * 0.954 = 0.968$$

$$\text{gain}(\text{Motore}) = 0.997 - 0.968 = 0.029$$

$$\text{splitinfo}(\text{Motore}) = -7/15 * \log_2(7/15) - 8/15 * \log_2(8/15) = 0.997$$

$$\text{gainratio}(\text{Motore}) = 0.029 / 0.997 = 0.029$$

c) L'attributo scelto per la radice dell'albero è Ruote. Le istanze avente Ruote sconosciuto vengono divise in tre parti di pesi rispettivamente $3/13=0.231$, $5/13=0.385$ e $5/13=0.385$ che vengono mandate rispettivamente lungo i rami due, tre e quattro.



d) l'istanza viene divisa in tre parti, di peso rispettivamente $3.462/15=0.231$, $5.77/15=0.385$ e $5.77/15=0.385$. La prima parte viene mandata lungo il ramo due e viene classificata come - con probabilità $2.231/3.462 = 64.4\%$ e come + con probabilità $1-64.4\%=35.6\%$. La seconda parte viene mandata lungo il ramo tre e viene classificata come - con probabilità $4.385/5.77 = 76.0\%$ e come + con probabilità $1-76.0\%=24.0\%$. La terza parte viene mandata lungo il ramo quattro e viene classificata come + con probabilità $4.385/5.77=76.0\%$ e come - con probabilità $1-76.0\%=24.0\%$.

Quindi in totale la classificazione dell'istanza è

$$+ : 0.231 * 35.6\% + 0.385 * 24.0\% + 0.385 * 76.0\% = 46.7\%$$

$$- : 0.231 * 64.4\% + 0.385 * 76.0\% + 0.385 * 24.0\% = 53.4\%$$

Esercizio 2

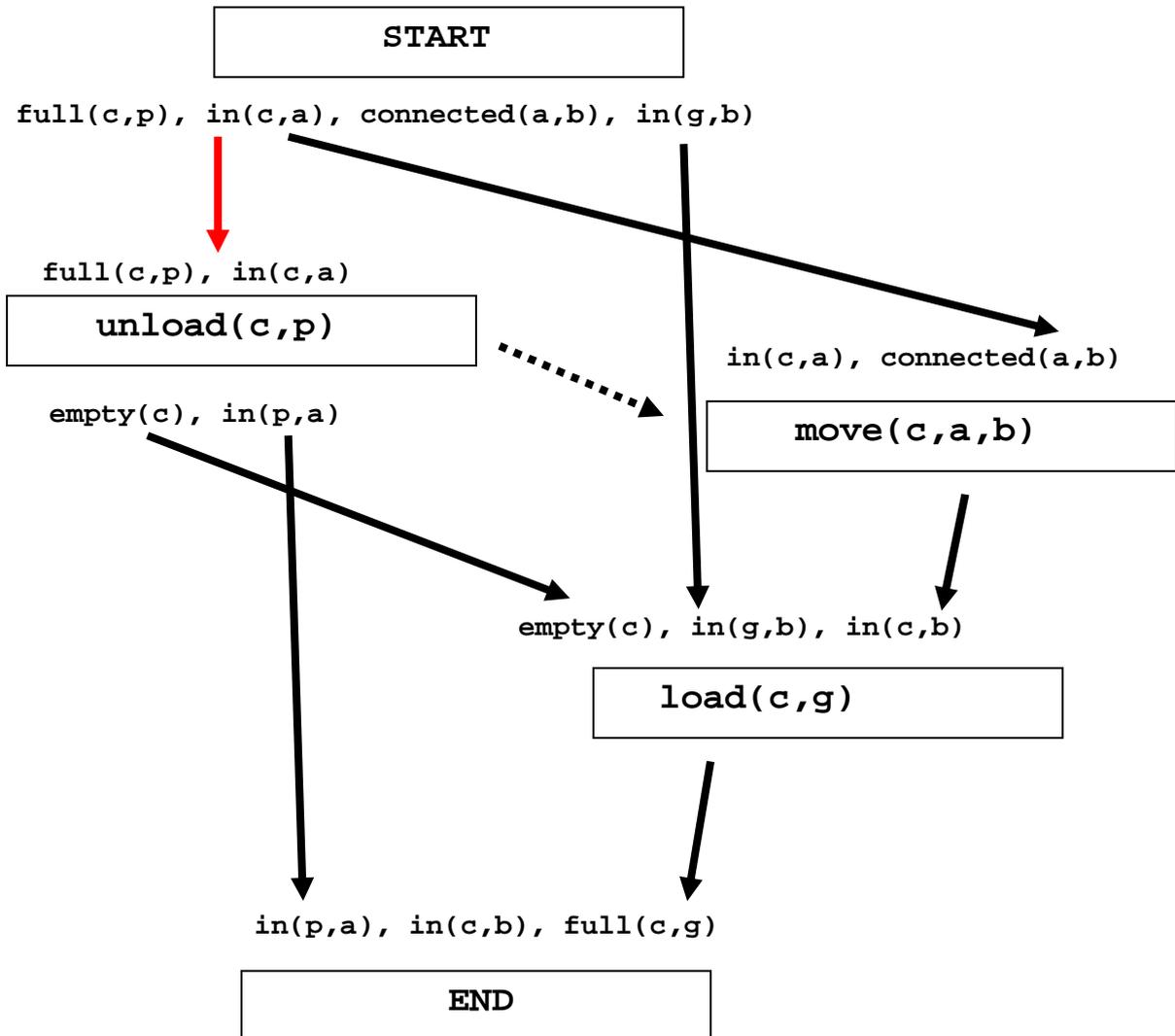
```
cell(DurBase, F, Dead) :-
    length(DurBase, N),
    length(Dur, N),
    length(F, N),
    F :: 1..3,
```

```
prodotto(F,Dur,DurBase),
energia(F,Dur,Energy),
sumlist(Energy,EnergyTot),
sumlist(Dur,DurTot), DurTot #<= Dead,
minimize(labeling(F),EnergyTot).
```

```
prodotto([],[],[]).
prodotto([B|LB],[D|LD],[P|LP]):-
    B*D #= P,
    prodotto(LB,LD,LP).
```

```
energia([],[],[]).
energia([F|LF],[D|LD],[E|LE]):-
    E #= (2*F-1)*D,
    energia(LF,LD,LE).
```

Esercizio 3



Questo piano contiene un threat: infatti il *causal link* in rosso ($in(c,a)$) è minacciato dagli effetti dell'azione $move(c,b,a)$ che come effetto contiene $not\ in(c,a)$.

In questo caso si può applicare la Demotion e far sì che $move(c,b,a)$ segua $unload(c,p)$ (vincolo di successione tratteggiato in figura).