

Sistemi

Laurea Magistrale in Ingegneria e Scienze Informatiche

Anno Accademico 2009-2010

Docenti: Vincenzo Manca, Riccardo Muradore, Tiziano Villa

18 Febbraio 2010

Metodi di Specifica
18 Febbraio 2010

Nome e Cognome:

Corso di Laurea:

Matricola:

Posta elettronica:

problema	punti massimi	i tuoi punti
problema 1	6	
problema 2	4	
totale	10	

1. (a) Si considerino le due macchine a stati finiti seguenti:

Macchina M' :

- stati: s'_a, s'_b, s'_c con s'_a stato iniziale;
- transizione da s'_a a s'_b : $\bullet/0$,
transizione da s'_b a s'_c : $\bullet/0$,
transizione da s'_c a s'_b : $\bullet/1$,
transizione da s'_c a s'_a : $\bullet/1$.

Macchina M'' :

- stati: $s''_x, s''_y, s''_z, s''_u$ con s''_x stato iniziale;
- transizione da s''_x a s''_y : $\bullet/0$,
transizione da s''_y a s''_z : $\bullet/0$,
transizione da s''_y a s''_u : $\bullet/0$,
transizione da s''_z a s''_x : $\bullet/1$,
transizione da s''_u a s''_y : $\bullet/1$.

Si risponda in ordine alle seguenti domande (si indichi sempre il numerale romano in ogni risposta):

- i. Si disegnino i diagrammi di transizione delle due macchine.
- ii. Si classifichino le macchine rispetto al determinismo.

Traccia di risposta.

M' e' nondeterministica, ma non pseudo-nondeterministica.

M'' e' nondeterministica, ma non pseudo-nondeterministica.

- iii. Si trovi una simulazione di M' da parte di M'' , se esiste.

Traccia di risposta.

M' non e' simulata da M'' perche' nella relazione di simulazione dovrebbero esserci le seguenti coppie di stati: (s'_a, s''_x) , (s'_b, s''_y) , e inoltre almeno una delle due tra (s'_c, s''_z) e (s'_c, s''_u) . Ma nessuna delle due ultime coppie ci puo' essere, ad es. (s'_c, s''_z) implicherebbe (s'_b, s''_x) , (s'_c, s''_y) ma s''_y (che produce 0) non simula s'_c (che produce 1). Similmente si vede che da (s'_c, s''_u) si ha un'implicazione non soddisfacibile.

- iv. Si trovi una simulazione di M'' da parte di M' , se esiste.

Traccia di risposta.

M'' e' simulata da M' come mostrato dalla relazione

$$R_{M''-M'} = \{(s''_x, s'_a), (s''_y, s'_b), (s''_z, s'_c), (s''_u, s'_c)\}.$$

- v. Si trovi una bisimulazione tra le due macchine, se esiste.

Traccia di risposta.

Poiche' M' non e' simulata da M'' non ci puo' essere una bisimulazione tra le due macchine.

- vi. Si applichi a M' l'algoritmo di minimizzazione che ottiene $\min(M')$, una macchina equivalente a M' con un numero minimo di stati tra quelle bisimili a M' .

Traccia di risposta.

Si ottiene $\min(M') \equiv M'$. Cio' si ottiene applicando l'algoritmo di minimizzazione (a meno di bisimilarita'): prima si suppongono equivalenti i tre stati di M' , poi si separano s'_a e s'_b da s'_c per le uscite in contrasto, infine si separa s'_a da s'_b per gli stati futuri in blocchi diversi.

- vii. Si applichi a M'' l'algoritmo di minimizzazione che ottiene $\min(M'')$, una macchina equivalente a M'' con un numero minimo di stati tra quelle bisimili a M'' .

Traccia di risposta.

Si ottiene $\min(M'') \equiv M''$. Si ragioni come nel caso precedente.

- viii. Esiste una macchina equivalente a M' con meno stati di $\min(M')$?

Traccia di risposta.

No.

Si noti che in questa domanda e nella successiva non si chiede l'esistenza di una macchina bisimile a M'' con meno stati di $\min(M')$ perche' per definizione di $\min(M')$ essa non puo' esistere, bensì l'esistenza di una macchina equivalente a M' con meno stati di $\min(M')$. In classe non abbiamo presentato un algoritmo per determinare in modo sistematico la macchina minima (non necessariamente bisimile) equivalente a una data, percio' la risposta deve basarsi su un'analisi del caso specifico.

- ix. Esiste una macchina equivalente a M'' con meno stati di $\min(M'')$?

Traccia di risposta.

Si, e' la macchina M' .

- x. $\min(M')$ simula $\min(M'')$? $\min(M'')$ simula $\min(M')$? C'e' una bisimulazione tra $\min(M')$ e $\min(M'')$?

Traccia di risposta.

$\min(M')$ simula $\min(M'')$? Si (stessa domanda che M' simula M'').
 $\min(M'')$ simula $\min(M')$? No (stessa domanda che M'' simula

M').

C'è una bisimulazione tra $\min(M')$ e $\min(M'')$? No.

xi. Si commentino i risultati precedenti.

Traccia di risposta.

Le due macchine M' e M'' sono un esempio di macchine a stati finiti nondeterministiche equivalenti e ciascuna minimizzata (nella classe delle macchine bisimili), ma non isomorfe; in altri termini, esse mostrano che non esiste un'unica macchina a stati finita nondeterministica che realizza il sistema originale con il minimo numero di stati.

Si ricordi che se M'' è simulata da M' allora M'' raffina M' . In generale non vale il viceversa, cioè per M' e M'' nondeterministiche il fatto che M' raffini M'' non implica che ci sia una simulazione di M' da parte di M'' (ad es. nel nostro caso non c'è).

(b) Si consideri la macchina a stati finiti M seguente:

- stati: s_1, s_2 con s_1 stato iniziale;
- una variabile d'ingresso $I = \{vero, falso, \perp\}$ e una variabile d'uscita $U = \{vero, falso, \perp\}$;
- transizione da s_1 a s_1 : $vero/vero$,
transizione da s_1 a s_2 : $falso/falso$,
transizione da s_2 a s_2 : $falso/falso$,
transizione da s_2 a s_1 : $vero/vero$

Si chiuda la macchina ad anello, connettendo l'uscita all'ingresso. La composizione e' ben formata ? Se si, si costruisca la macchina composta, e si elenchino le prime 10 uscite (cioe' le uscite in corrispondenza ai primi 10 ticchettii dell'orologio esterno della composizione).

Traccia di soluzione.

In entrambi gli stati s_1 e s_2 non c'e' un unico punto fisso perche' ce ne sono due ! Infatti l'ingresso *vero* produce l'uscita *vero* e l'ingresso *falso* produce l'uscita *falso*, per cui esistono due punti fissi non muti $y(n)$ tali che $uscita(s_1, y(n)) = y(n)$, e percio' la composizione ad anello chiuso non e' ben formata.

Si noti che affiche la composizione sia ben formata si richiede che ci sia un unico punto fisso per ogni stato raggiungibile dallo stato iniziale.

2. Si consideri un impianto G con $\Sigma = \{a, b\}$, $\Sigma_{uc} = \{b\}$, $L(G) = \overline{a^*ba^*}$ (cioe' il linguaggio ottenuto dai prefissi delle stringhe dell'espressione regolare a^*ba^*), $L_m(G) = a^*ba^*$.

Si supponga che la specifica (il linguaggio generato desiderato) sia $K = \{a^kba^k, k \geq 0\} \subseteq L(G)$, cioe' si richiede che l'impianto controllato generi prefissi di stringhe con un numero uguale di a che precedono e seguono un unico b .

- (a) Il linguaggio K e' controllabile? Si enunci la definizione di controllabilita' di un linguaggio e la si applichi al caso.

Traccia di soluzione

Definizione Siano K e $M = \overline{M}$ linguaggi sull'alfabeto di eventi E , con $E_{uc} \subseteq E$. Si dice che K e' controllabile rispetto a M e E_{uc} , se per tutte le stringhe $s \in \overline{K}$ e per tutti gli eventi $\sigma \in E_{uc}$ si ha

$$s\sigma \in M \Rightarrow s\sigma \in \overline{K}.$$

[equivalente a $\overline{K}E_{uc} \cap M \subseteq \overline{K}$]

Si applichi la definizione al nostro esempio dove $M = L(G)$. Si consideri una stringa $s \in \overline{K} = K$,

- se $s = a^*$ e quindi $s\sigma = sb = a^*b \in L(G)$ allora $s\sigma = sb \in \overline{K}$, altrimenti
- se $s \neq a^*$ allora $s\sigma = sb \notin L(G)$ (cioe' se s deve essere della forma $s = a^*ba^*$ e quindi $sb = a^*ba^*b \notin L(G)$, poiche' le parole in $L(G)$ non possono contenere un secondo evento b dopo il primo).

Percio' non esiste una stringa $s \in \overline{K}$ tale che $s\sigma = sb \in L(G) \setminus \overline{K}$, cioe' K e' controllabile.

- (b) Si enunci il teorema di esistenza di un supervisore sotto controllabilita' limitata. Esiste un supervisore S tale che l'impianto controllato generi il linguaggio K ? Si mostri un tale supervisore S se esiste, e si descriva in breve la sua strategia di controllo.

Traccia di soluzione

Definizione Siano $G = (X, E, f, \gamma, x_0)$ un impianto, $E_{uc} \subseteq E$ gli eventi incontrollabili, $K \subseteq L(G)$, $K \neq \emptyset$ la specifica. Esiste un supervisore S tale che $L(S/G) = \overline{K}$ se e solo se

$$\overline{K}E_{uc} \cap L(G) \subseteq \overline{K}.$$

Un qualsiasi supervisore S con $L_m(S) = L(S) = \overline{K}$ induce un impianto controllato che genera il linguaggio K , cioè $L(S/G) = \overline{K}$.

In questo caso la strategia del supervisore è semplicemente quella di disabilitare l'evento a dopo ogni stringa del tipo $a^k b a^k$.

Si noti che il linguaggio della specifica non è regolare, quindi non c'è un automa a stati finiti che lo rappresenti. Il fatto che la specifica sia controllabile è indipendente dal fatto che sia realizzabile mediante un automa a stati finiti. In questo caso esiste una strategia di controllo - quella di disabilitare sempre a dopo che l'impianto ha prodotto una parola con un numero di a dopo b pari al numero di a prima di b -, ma non esiste un automa a stati finiti che la realizzi (in questo caso non esiste perché tale automa dovrebbe contare se sono uguali i numeri di eventi a prima e dopo l'evento b senza un limite prestabilito sul numero di eventi a).