Specifiche logiche: caso di studio

Leggere Sez. 5.6.2.5, 5.6.2.6
Ghezzi et al.

università di ferrara
DA SEICENTO ANNI GUARDIAMO AVANTI.

Caso di studio di specifiche logiche

- Predicati elementari
  - stati
  - eventi

  esprimono proprietà rilevanti rispetto allo stato e alle operazioni del sistema
- Regole: formule definite sui predicati elementari, che devono essere verificate

Predicati elementari

- Stati: condizione di durata non nulla nel tempo
- Es: standing(E, F, T_1, T_2)
- Eventi: condizioni che si verificano in un determinato istante
- Es. arrived (E, F, T)

Regole

- Costituite da:
  1. insieme di premesse,
  2. **implies**,
  3. conclusione
- Quantificazione implicita: universale per le variabili a sinistra di **implies**, esistenziale per quelle che compaiono solo a destra.

Eventi

arrival (E, F, T)

E in [1..n], F in [1..m], $T \geq t_0$, (t_0 initial time)

- Arrivo dell'ascensore al piano F nell'istante T (non necessariamente per fermarsi)

departure(E, F, D, T)

E in [1..n], F in [1..m], D in {up, down}, $T \geq t_0$

- Partenza dell'ascensore dal piano F, in direzione D, all'istante T.

Eventi

stop (E, F, T)

E in [1..n], F in [1.. m], $T \geq t_0$

- Arrivo e fermata al piano F, all'istante T.

new_list (E, L, T)

E in [1..n], L in $[1..\ m]^*$, $T \geq t_0$

- La lista delle fermate dell'ascensore diventa L al tempo T.

Eventi

call(F, D, T)

F in [1..m], D in {up, down}, $T \geq t_0$

- Chiamata dal piano F, per la direzione D, al tempo T

request(E, F, T)

E in [1..n], F in [1..m], $T \geq t_0$

- Richiesta interna per il piano F all'istante T

Stati

moving (E, F, D, T1, T2)

- Nell'intervallo [T1, T2[, l'ascensore si muove in direzione D e l'ultimo piano da cui è passato è F

standing (E, F, T1, T2)

- Nell'intervallo [T1, T2[, l'ascensore è fermo al piano F.

list (E, L, T1, T2)

- Nell'intervallo [T1, T2[, la lista dell'ascensore è L.

Regole su stati ed eventi

R$_1$:Quando E arriva al piano F, lo lascia subito se F non ha richiesto servizio e la lista non è vuota. Se il piano successivo da servire si trova più in alto di F, lo spostamento sarà verso l'alto; altrimenti, verso il basso.

arrival (E, F, T_a) and
list (E, L, T, T_a) and
first (L) > F
implies
departure (E, F, up, T_a)

Analogamente per il movimento verso il basso (per questa e altre regole)

Regole su stati ed eventi

R2: All'arrivo a F, E si ferma se F deve essere servito (cioè è il primo della lista)

arrival (E, F, T_a) and
list (E, L, T, T_a) and
first (L) = F
implies
stop (E, F,T_a)

R3: E si ferma ad F se la sua lista è vuota

arrival (E, F, T_a) and
list (E, empty, T, T_a)
implies
stop (E, F, T_a)

Regole su stati ed eventi

R4: Gli ascensori abbiano un tempo di servizio costante. Se la lista non è vuota alla fine dell'intervallo, l'ascensore lascia il piano immediatamente.

stop (E, F, T_a) and
list (E, L, T, T_a + Dt_s) and
first (L) > F
implies
departure (E, F, up, T_a + Dt_s)

R5: Se l'ascensore non ha piani da servire, si muove solo quando la sua lista non è più vuota.

stop (E, F, T_a) and list (E, L, T_p, T) and
T_p > T_a + Dt_s and list (E, empty, T_a + Dt_s, T_p) and
first (L) > F
implies
departure (E, F, up, T_p)

Regole su stati ed eventi

R6: Il tempo di spostamento da un piano all'altro sia fissato. L'arrivo a un piano avviene Dt dopo la partenza dal piano precedente

departure (E, F, up, T)
implies
arrival (E, F + 1, T + Dt)

R7: L'evento di fermata al piano F nell'istante T avvia uno stato di permanenza al piano di durata minima Dt_s

stop (E, F, T)
implies
standing (E, F, T, T + Dt_s)

Regole su stati ed eventi

R8: Al termine di una permanenza, l'ascensore rimane fermo se non ci sono altri piani da servire

stop (E, F, T_s) *and*
list $(E, empty, T_s + Dt_s, T)$
implies
standing (E, F, T_s, T)

R9: L'evento di partenza avvia uno stato di spostamento che dura almeno Dt

departure (E, F, D, T)
implies
moving $(E, F, D, T, T + Dt)$

Regole su stati ed eventi

R10: Se uno stato rimane costante nell'intervallo $[T_1, T_2[$, allora rimarrà costante anche in ogni intervallo $[T_3, T_4[$ incluso.

standing (E, F, T_1, T_2) *and*
$T_1 \leq T_3$ *and* $T_3 < T_4$ *and* $T_4 \leq T_2$
implies
standing (E, F, T_3, T_4)

Regole di controllo

- Definiscono l'introduzione di
  - eventi di tipo new_list
  - stati di tipo list
- Regole per la gestione delle richieste provenienti dall'interno.
- Ogni richiesta viene inserita nella lista, tenuta ordinata secondo la direzione di marcia.

Regole di controllo

R11: La richiesta per un piano F, a cui l'ascensore non è già fermo, causa l'inserimento di F nella lista L, ordinata.

request (E, F, T_R) *and*
not exists T_a (*standing* (E, F, T_a, T_R)) *and*
list (E, L, T_a, T_R) *and*
LF = insert_in_order(L, F, E)
implies
new_list (E, LF, T_R)

Regole di controllo

R12: All'arrivo al piano F, F viene tolto dalla lista, se è il primo elemento.

arrival(E, F, T_a) and list (E, L, T, T_a) and
F = first (L) and L_t = tail (L)
implies
new_list (E, L_t, T_a)

R13: La lista di un ascensore corrisponde a quella indicata dall'ultimo evento new_list.

new_list (E, L, T_1) and
not exists L_1, T_2 (new_list (E, L_1, T_2) and $L_1 \neq L$ and
$T_1 < T_2 < T_3$)
implies
list (E, L, T_1, T_3)

Verifica delle specifiche logiche

- Simulazione: deduzione delle conseguenze di formule ipotizzate vere. Esempio:
  standing (2, 3, 5, 7)
  list(2, empty, 5, 7)
  request(2, 8, 7)
  $\Rightarrow$ *new_list(2, {8}, 7)*

  (ed escludendo altri eventi)
  departure (2, up, 7 + Dt_s)
  *arrival (2, 8, 7 + Dt_s + Dt_a *(8-3))*

Verifica delle specifiche logiche

- Analisi: deduzione dalla specifica logica di proprietà espresse come formule logiche.
- Queste proprietà saranno ereditate da ogni implementazione valida della specifica.

Verifica di specifiche logiche

- Dimostratori di teoremi per simulazione e analisi
- In linea di principio, metodo potente, ma:
  - In generale, la dimostrazione di teoremi in logica del primo ordine è indecidibile.
  - Anche per sottoinsiemi della logica del primo ordine decidibili o semi-decidibili, la complessità può essere elevata.