Verifica – parte IIID

Rif. Ghezzi et al.
6.6

università di ferrara
DA SEICENTO ANNI GUARDIAMO AVANTI.

Logica modale

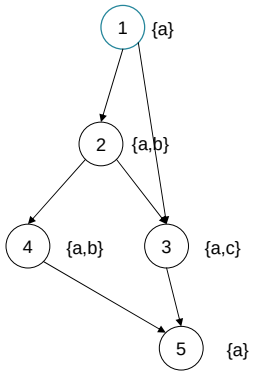
- In logica classica proposizionale, ogni formula ha un valore di verità fisso, definito dall'interpretazione.
- In molte applicazioni è interessante ammettere che i valori di verità possano cambiare al cambiare delle condizioni (tempo, azioni, possibilità)

Strutture di Kripke

- Sia A un insieme di proposizioni atomiche
- Una struttura di Kripke su A è definita da:
  - un insieme finito di stati S
  - un insieme I $\subseteq$ S di stati iniziali
  - una relazione R $\subseteq$ S x S (raggiungibilità fra stati: indica quali stati sono raggiungibili da un determinato stato)
  - una funzione V: S $\rightarrow 2^A$ (valutazione: definisce quali proposizioni sono vere in uno stato)

(se a$\in$ V(s), si scrive anche s$\models$a)

Esempio



- A = {a, b, c}
- S = {1,2,3,4,5}
- I = {1}
- R = {(1,2), (1,3), (2,4), (2,3), (3,5), (4,5)}
- Ad esempio, V(3) = {a,c}

Operatori modali: necessità e possibilità

- Dati una struttura di Kripke (S,I,R,V) su A, $s \in S$ e $a \in A$,
- $s \models \Box a$ se e solo se $\forall s' | (s,s') \in R \; s' \models a$
  - (box: a è vero in tutti gli stati raggiungibili da s)
- $s \models \Diamond a$ se e solo se $\exists s' | (s,s') \in R$ e $s' \models a$
  - (diamond: a è vero in almeno uno stato raggiungibile da s)

Model checking

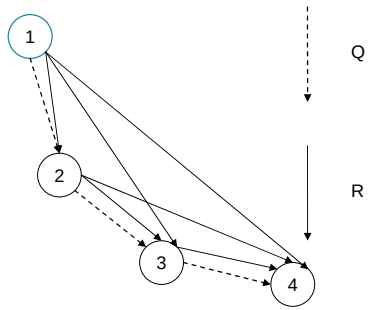
- Tecnica di dimostrazione di formule modali
- Data una struttura di Kripke determina se una formula modale è vera in un determinato stato.
- Con opportune limitazioni, il problema può essere decidibile, anche se complesso.

Logica temporale

- Stati diversi rappresentano lo stato del sistema in istanti (discreti) diversi
- Il valore di verità delle formule dipende dal tempo
- Lineare: c'è un solo istante successivo a ogni istante
- Branching: possono esserci più istanti successivi allo stesso istante (diverse evoluzioni possibili).

Logica temporale lineare (LTL)

- E' definita una relazione Q di stato all'istante successivo, tale che $\forall s \in S \; \exists! s' \in S \mid sQs'$
- R è la chiusura transitiva di Q (cioè sono raggiungibili da uno stato tutti quelli che rappresentano la sua evoluzione in istanti successivi)
- Adatta a rappresentare sistemi che hanno una sola evoluzione possibile.



Operatori LTL

- X: prossimo
  - $s \models X\ a \leftrightarrow \exists s' \mid sQs', s' \models a$

(a sarà vera al prossimo istante)

- F: futuro (◊)
  - $s \models F\ a \leftrightarrow \exists s' \mid sRs', s' \models a$

(a sarà vera per almeno un istante futuro)

- G: (sempre nel futuro) (□)
  - $s \models G\ a \leftrightarrow \forall s' \mid sRs'\ s' \models a$

(a sarà vera a tutti gli istanti futuri)

- U: (until)
  - $s \models a\ U\ b \leftrightarrow \exists s_1 \mid sRs_1\ ,\ s_1 \models b,\ \forall s' \mid (sRs',\ s'Rs_1)\ s' \models a$

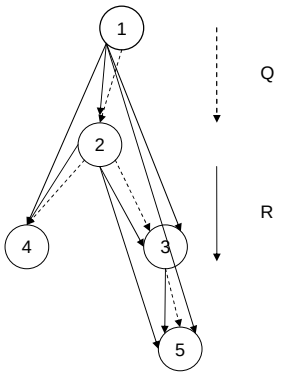
(b sarà vera a un istante futuro, e fino ad allora sarà vera a)

Proprietà esprimibili in LTL

- Safety: garantisce che una proposizione sia sempre vera da un certo istante in poi (spesso la negazione di una condizione non desiderabile)
  - Gq
  - p→Gq
- Liveness: garantisce che una proposizione sarà vera in almeno un istante nel futuro
  - Fq
  - p→Fq

Computational tree logic (CTL)

- La relazione Q di stato all'istante successivo non ha il vincolo di unicità (cioè da uno stato possono evolvere più stati all'istante successivo)
- R è la chiusura transitiva di Q



Operatori CTL

- Quantificano su
  - tempo
  - cammini
- AX
  - $s \models AX\ a \leftrightarrow \forall s' \mid sQs', s' \models a$

(a sarà sicuramente vera al prossimo istante)

- EX
  - $s \models EX\ a \leftrightarrow \exists s' \mid sQs', s' \models a$

(a può essere vera al prossimo istante)

Operatori CTL

- C(N) insieme dei cammini che partono dal nodo N. Es: C(2) = {(2,4), (2,3,5)}
- AU
  - $s \models A\ (a\ U\ b) \leftrightarrow \forall c \in C(s)\ \exists s_1 \in c |\ sRs_1\ ,\ s_1 \models b,\ \forall s' |(sRs', s'Rs_1)\ s' \models a$

  (b sarà sicuramente vera a un istante futuro, e fino ad allora sarà vera a)
- EU
  - $s \models E\ (a\ U\ b) \leftrightarrow \exists c \in C(s)\ \exists s_1 \in c |\ sRs_1\ ,\ s_1 \models b,\ \forall s' |(sRs', s'Rs_1)\ s' \models a$

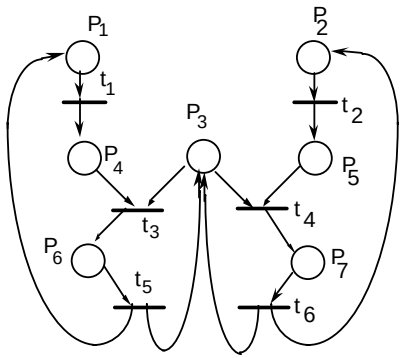
  (è possibile che b sia vera a un istante futuro, e che fino ad allora sia vera a)

Operatori CTL

- AFa
  - abbreviazione di A (true U a): a sarà sicuramente vera in un istante futuro
  - Liveness: AFa
- EFa
  - abbreviazione di E (true U a): è possibile che a sia vera in un istante futuro
  - Safety: ¬EFa

Esempio

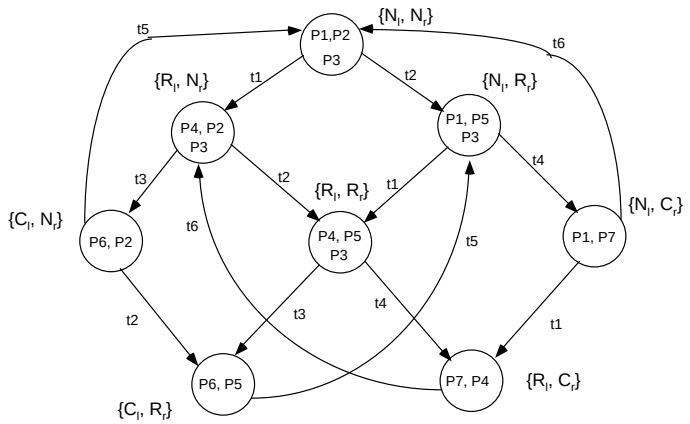
- Stato determinato dalla marcatura (es. (P1,P2,P3))
- Insieme finito di marcature possibili (quindi stati finiti)
- sQs' se s' si ottiene con lo scatto di una transizione abilitata in s



Proposizioni

- C_l, C_r significano che i processi l ed r sono nella regione critica (cioè hanno preso possesso della risorsa condivisa)
- N_l, N_r significano che i processi l ed r non sono nella regione critica (P1, P2)
- R_l, R_r significano che i rispettivi processi hanno richiesto la risorsa condivisa (P4, P5)

Rappresentazione grafica



Proprietà

- $EF(C_l)$ è vera in ogni stato: per il processo di sinistra è sempre possibile accedere alla regione critica (cioè avere accesso alla risorsa)
- $AF(C_l)$ è falsa in ogni stato escluso P_6: se il processo non è già in possesso della risorsa condivisa, potrebbe non accedervi mai.