

## ASM – Analisi di Alcune Proprietà

## PN in sintesi

- PN
- Ideate per rappresentare sistemi complessi
  - Concorrenza
  - Sincronizzazione
- Composizione di sotto-reti
- Una transizione influenza solo una parte dello stato complessivo

## ASM in sintesi

- ASM
- Simulano l'esecuzione di pseudo-codice arbitrario su strutture dati
- Semplicità concettuale e facilità di utilizzo
- Metodologia di sviluppo gerarchica:
  - Ground model
  - Raffinamenti successivi
- Qualità dipendente dal problema, non dalla notazione delle ASM

## PN – ASM: Corrispondenze

- Si riconoscono alcune corrispondenze:
  - Posti e stati
  - Transizioni e coppie di condizione + update
  - Marcature: passaggio attraverso uno stato

## Liveness

- Qualcosa di **desiderato** prima o poi **dovrà accadere**
- Esempi
  - Raggiungibilità di uno stato (ad esempio lo stato finale)
  - Starvation freedom – prima o poi un processo dovrà essere attivato

## Safety

- Qualcosa di **indesiderato** non **deve mai accadere**
- Esempi
  - Violazione mutua esclusione
  - deadlock

## Strumenti dalla Logica Temporale

- La logica temporale fornisce strumenti adeguati per esprimere liveness/safety
- $\square$  (forever) usato per esprimere la safety ( $\neg\square$ )
- $\diamond$  (eventually) usato per liveness

## Liveness/Safety (1)

- Sono classi di proprietà
  - Non sono univocamente determinate
- Nei programmi sequenziali principali proprietà sono:
  - Correttezza parziale: il raggiungimento dello stato finale soddisfa le post-condizioni dell'esecuzione (safety)
  - Correttezza totale: correttezza parziale + terminazione (liveness)

## Liveness/Safety (2)

- Le ASM modellano programmi sequenziali
  - Caso particolare di DASM a singolo-agente
- Quindi possiamo definire univocamente safety / liveness per ASM

## Liveness/Safety (3)

- **Safety**: Una ASM di base è safe se la non soddisfacibilità della disgiunzione di tutte le sue condition implica sempre una desiderata post-condizione Q

$$\neg C \rightarrow \Box Q$$

## Liveness/Safety (4)

- **Liveness**: Una ASM di base è live se è possibile verificare la condizione di non soddisfacibilità di tutte le sue condition e quest'ultima implica una desiderata post-condizione Q

$$\Diamond \neg C \rightarrow Q$$

## Indecidibilità di Liveness/Safety (1)

- Liveness e safety sono proprietà indecidibili
- Teorema (Vessio2013):
  - Definire un algoritmo che data in input una ASM qualsiasi sia in grado di decidere se è safe o live è un problema indecidibile

## Indecidibilità di Liveness/Safety (2)

- Dimostrazione
  - Per decidere se la ASM è safe o live l'algoritmo deve verificare la condizione di non soddisfacibilità di tutte le sue condition.
  - Tale condizione è equivalente al raggiungimento di uno stato finale.
  - Le ASM sono Turing-equivalenti, quindi l'algoritmo è in grado di verificare la terminazione della MdT equivalente alla ASM.
  - L'algoritmo è in grado di risolvere l'Halting problem. . . è una contraddizione!

## Reachability

- Def.: Cammino
  - sequenza di stati collegati da regole governate da condizioni soddisfacibili
- Stato  $S_n$  raggiungibile dallo stato iniziale  $S_0$  se esiste un cammino che collega  $S_0$  con  $S_n$ 
  - $\exists P = \{S_0, \dots, S_n\}$
  - $\forall i \in \mathbb{N}, 0 \leq i < n, C_i$  soddisfatta,  $R_i$  eseguita

## Reversibility

- Se uno stato  $S_n$  è raggiungibile dallo stato iniziale  $S_0$ , allora  $S_0$  deve risultare raggiungibile da  $S_n$ 
  - Esistenza del cammino inverso
  - Cammino inverso  $\neq$  sequenza inversa

## Multimodality

- ASM multimodale se esistono esecuzioni alternative che portano la ASM in uno stesso stato
  - esecuzioni alternative = diverse successioni di coppie regola/condizione
  - $\exists P1 = \{S_0, \dots, S_n\}$ ,
  - $\exists P2 = \{S_0, \dots, S_n\}$ ,
  - $\exists Si \in P1$  e  $Sj \in P2 \mid Si, Sj \neq S_0, S_n \wedge Si \neq Sj$

# Complexity

- Strettamente legata alle capacità computazionali del sistema
- Def. Molteplicità: numero di percorsi distinti che collegano una coppia di stati
- Complessità è il massimo della molteplicità dello stato calcolata su ogni diverso stato da quello finale
  - $\max ( m(S, S_n) ), \forall S \neq S_n$