PCAD
Programmazione Concorrente
Algoritmi Distribuiti

Arnaud Sangnier
arnaud.sangnier@unige.it

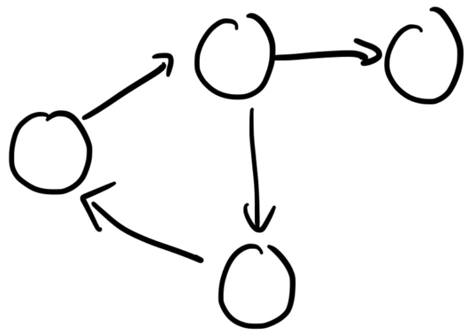
Verifica di sistemi

Model-checking

- **Scopo:**
  - Definire dei modelli matematici per rappresentare il comportamento dei sistemi **=> Modelli**
  - Definire dei linguaggi matematici per descrivere il comportamento atteso dei sistemi **=> Specifica**
  - Trovare degli algoritmi di verifica per dire se un modello satisfa la sua specifica **=> Algoritmo di model-checking**

Principio del model-checking

Un sistema soddisfa una specifica ?



model

$\overset{?}{\models}$

Algoritmo di model-checking

φ

Formula

Alcuni commenti

- I modelli possono essere diversi secondo le caratteristiche del sistema che uno vuole verificare
- I linguaggi di specifica dipendono anche delle proprietà
- Non c'è sempre un algoritmo di verifica.
  - Alcuni problemi sono i**ndecidibile**
  - Ad esempio il problema del halting per le macchine di Türing è **indecidibile**

Un modello semplice

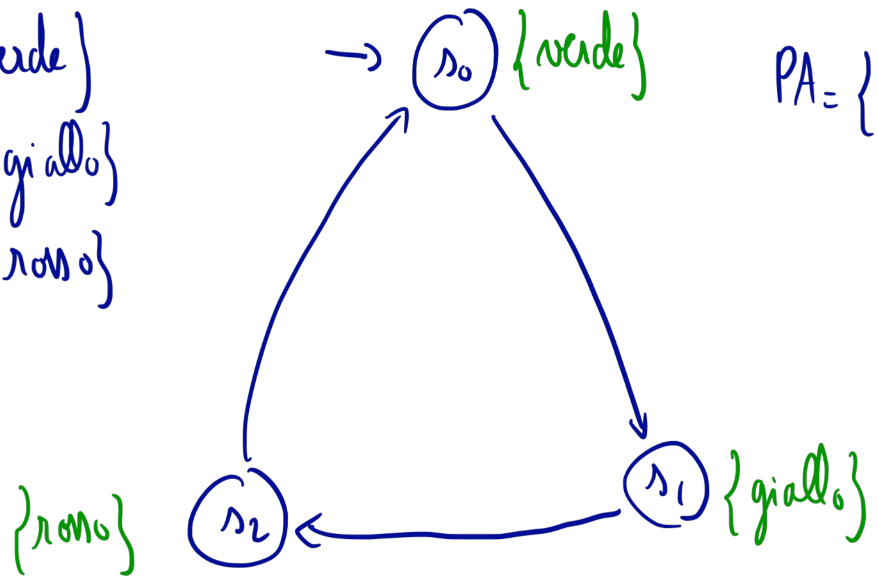
- Le strutture di Kripke sono dei grafi orientati semplice dove:
  - i vertici rappresentano gli stati di un sistema (li chiamiamo **stati**)
  - gli archi rappresentano gli cambi di stati (li chiamiamo **transizione**)
  - ogni stato è etichettato con un insieme di proprietà che sono le proprietà vere nello stato (chiamiamo queste proprietà, **proposte atomiche**)
- Definizione: Una **struttura di Kripke** KS è un tuple $(S, \rightarrow, s_{in}, PA, L)$ dove:
  - S è un insieme di stato
  - $\rightarrow \subseteq S \times S$ è la relazione di transizione
  - $s_{in} \in S$ è lo stato iniziale
  - PA è l'insieme di proposte atomiche
  - $L : S \mapsto 2^{PA}$ è la funzione di etichettatura

Osservazione sulle strutture di Kripke

- 2^{PA} è l'insieme dei sottoinsiemi di PA
- Per ogni stato $s \in S$, abbiamo $L(s) \subseteq PA$. I.e. $L(s)$ è l'insieme delle proposte atomiche vere in questo stato. Questo insieme può essere vuoto!
- Due stati diversi s e s', possono avere le stesse etichette. I.e possiamo avere $s \neq s'$ e $L(s)=L(s')$
- Al posto di $(s,s') \in \rightarrow$, scriveremo $s \rightarrow s'$.

Struttura di Kripke - Esempio 1

$L(s_0) = \{verde\}$
$L(s_1) = \{giallo\}$
$L(s_2) = \{rosso\}$

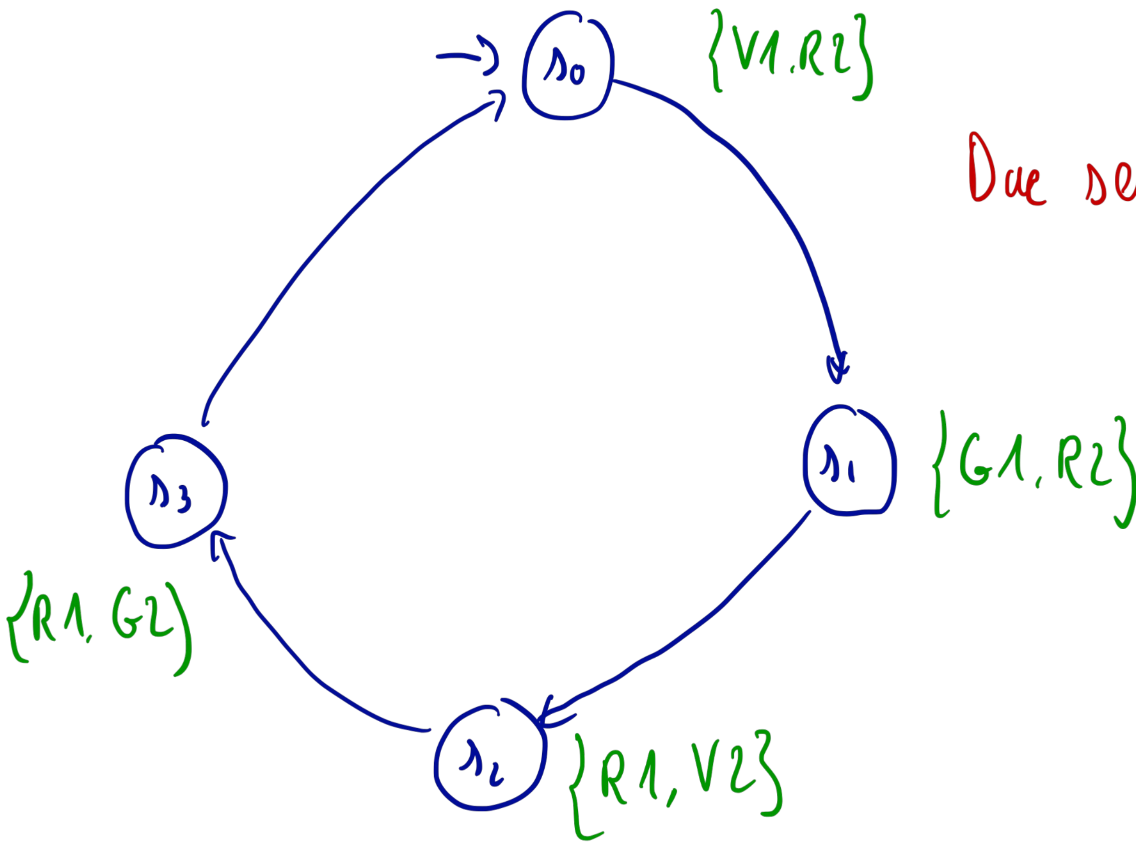


$PA = \{verde, giallo, rosso\}$

$S = \{s_0, s_1, s_2\}$ $\quad \rightarrow = \{(s_0, s_1), (s_1, s_2), (s_2, s_0)\}$

$s_{in} = s_0$

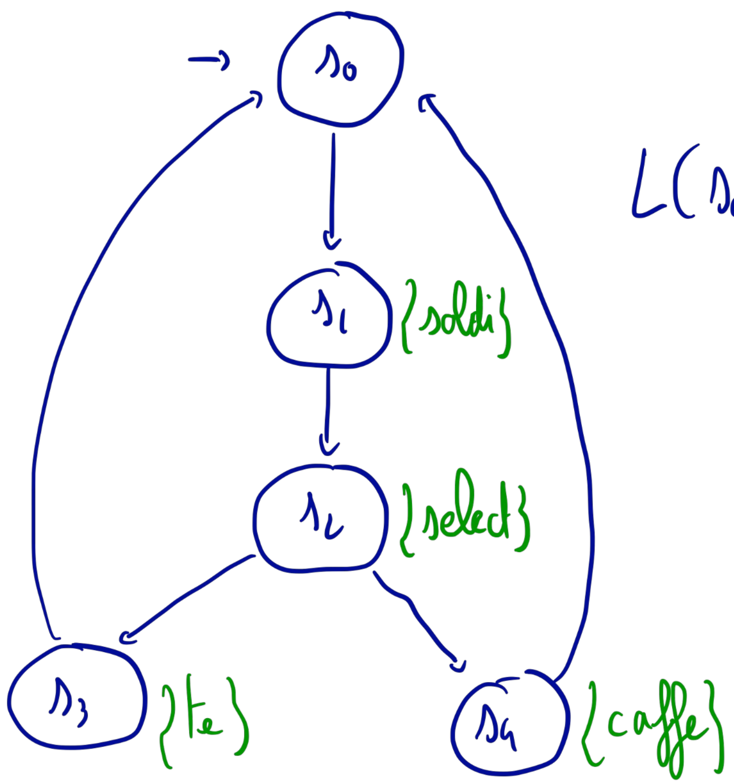
Struttura di Kripke - Esempio 2



Due semafori

V : Verde
R : Rosso
G : Giallo

Struttura di Kripke - Esempio 3



$L(s_0) = \{\} = \emptyset$

Alcune definizione supplementare

- Un **cammino finito** è una sequenza finita di stati $s_0\ s_1 \ldots\ s_n$ tale che per tutti i in {0,...,n-1}, abbiamo $s_i \rightarrow s_{i+1}$
- Un **cammino infinito** è una sequenza infinita di stati $s_0\ s_1 \ldots$ tale che per tutti $i \geq 0$, abbiamo $s_i \rightarrow s_{i+1}$
- Un **esecuzione** è un cammino infinito $s_0\ s_1 \ldots$ tale che $s_0 = s_{in}$
- Usiamo **Exec(KS)** per rappresentare l'insieme delle esecuzioni della struttura di Kripke KS

Tracce di una struttura

- Consideriamo una struttura di Kripke KS=$(S,\rightarrow, s_{in}, PA, L)$
- Quello che vogliamo osservare non è tanto la sequenza degli stati, ma la sequenza delle insiemi d'etichette
- Piuttosto di considerare le esecuzioni, guardiamo le sequenze di proposte atomiche associate
- Una **traccia** della struttura KS è una sequenza infinita d'insiemi P_0 P_1 P_2 ... tale che per tutti $i \geq 0$, abbiamo $P_i=L(s_i)$ e s_0 s_1 s_2 ... è una esecuzione di KS
- Scriviamo Trace(KS) l'insieme delle **tracce** di KS
- Una traccia è dunque una sequenza infinita di sotto-insiemi di PA
- Per una esecuzione $\varepsilon=s_0\ s_1\ s_2$ ..., scriviamo trace(ε) la traccia $L(s_0)$ $L(s_1)$ $L(s_2)$...
- Abbiamo quindi Trace(KS)=$\{$trace(ε) $\mid$ $\varepsilon \in$ Exec(KS) $\}$

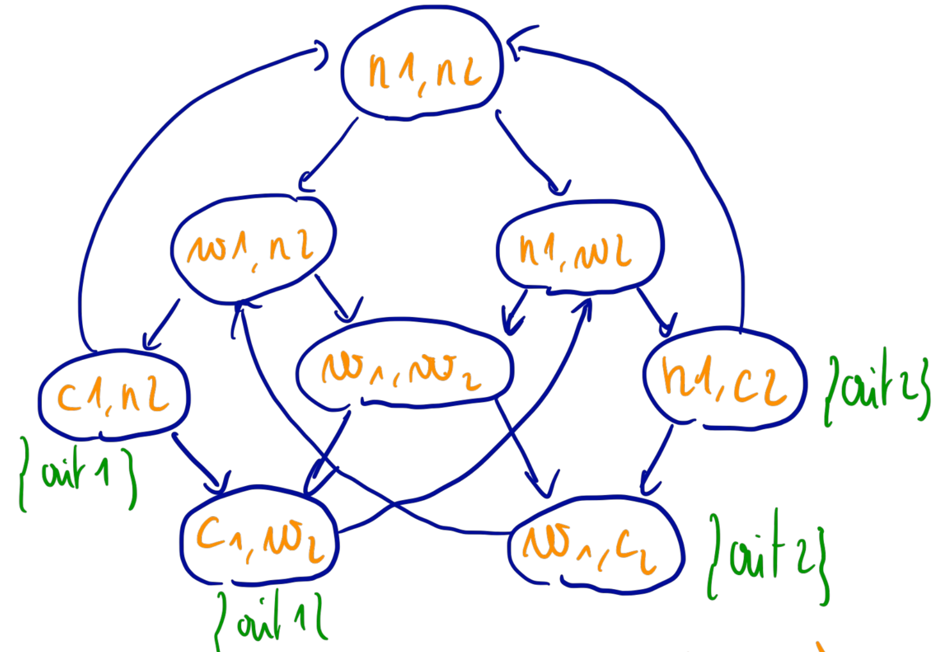
Esempio

Due processi vogliono accedere a una regione critica

n_i: proc i non fa nulla

w_i: proc i aspetta la seg. crit.

c_i: proc i è in seg. crit.



exec $\varepsilon = (n1,n2) \to (w_1,n2) \to (c1,n2) \to (n1,n2) \to (w1,n2) \to$
$(w_1,w_2) \to (c_1,w_2) \to (n1,w2) \to (n1,c2) \to \ldots$

trace(ε): $\emptyset$ $\emptyset$ {crit1} $\emptyset$ $\emptyset$ $\emptyset$ {crit1} $\emptyset$ {crit2} ...

Notazione

Per un insieme E:

- **2^E** $=\{E' \mid E' \subseteq E\}$
- **E^***: l'insieme delle sequenze finite $e_0\ e_1\ \dots\ e_n$ tale che per tutti $i \in \{0,\dots,n\}$, abbiamo $e_i \in E$
- **E^ω**: l'insieme delle sequenze infinite $e_0\ e_1\ \dots$ tale che per tutti $i \geq 0$, abbiamo $e_i \in E$

Proprietà temporale lineare

Sia una struttura di Kripke $KS=(S,\rightarrow, s_{in}, PA, L)$. Abbiamo:

- $\mathrm{Trace}(KS) \subseteq (2^{PA})^{\omega}$
- Una **proprietà temporale lineare** su PA è un sotto-insieme di $(2^{PA})^{\omega}$ i.e. un insieme di sequenze infinite d'insieme di proposte atomiche
- Sia P una proprietà temporale lineare, diciamo che **KS soddisfa P**, scritto

  $KS \models P$, se e solo se, abbiamo $\mathrm{Trace}(KS) \subseteq P$.

Una struttura soddisfa P, se e solo se, tutte le sue tracce sono in P.

Esempio

- Due semafori di strada con due colori rosso e verde
- Consideriamo PA={V1,R1,V2,R2}
- Prendiamo la proprietà temporale lineare seguente:
  - Il primo fuoco è infinitamente spesso verde
  - PT ={P_0 P_1 P_2 ... ϵ $(2^{PA})^{\omega}$ | il numero di i tale che V1 ϵ P_i è infinito}
- Le sequenze seguente sono in PT:

  1) {V1,R2} {R2} Ø {V1,R2} {R2} Ø ... {V1,R2} {R2} Ø ...
  2) Ø {V1} Ø {V1}Ø {V1}... Ø {V1}....
  3) {V1,V2} {V1,V2}...{V1,V2}...

- La sequenza seguente non è in PT:

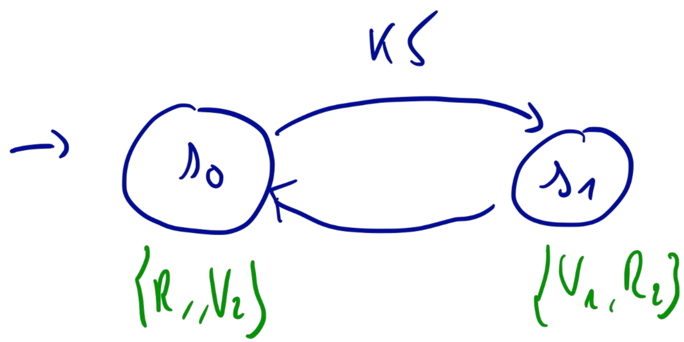
  1) {V1} {V1} Ø Ø Ø ... Ø ...

Esempio

- Prendiamo la proprietà temporale lineare seguente:
  - I due semafori non sono mai verdi allo stesso istante
  - $PT' = \{P_0\ P_1\ P_2 \ldots \in (2^{PA})^{\omega} \mid$ non esiste i tale che $V1 \in P_i$ e $V2 \in P_i\ \}$
- Le sequenze seguente sono in PT':

1) {V1} {V1} ... {V1}....
2) Ø {R1,R2} Ø {R1,R2} Ø {R1,R2} .... Ø {R1,R2} ...

Esempio



Abbiamo $Trace(KS) \subseteq PT$ e $Trace(KS) \subseteq PT'$

$KS \models PT$ e $KS \models PT'$

Proprietà di safety

- Le **proprietà di safety** sono delle proprietà temporale lineare particolare
- Verificano la proposta seguente:
  - se una traccia non è nella proprietà allora esiste un prefisso tale che ogni tracce con lo stesso prefisso non è nella proprietà
- Formalmente, una proprietà temporale lineare Psafe su PA è una proprietà di safety se per ogni sequenza $\sigma \in (2^{PA})^{\omega} \backslash Psafe$, esiste un prefisso π di σ tale che:

$$Psafe \cap \{ \sigma' \in (2^{PA})^{\omega} \mid \pi \text{ è prefisso di } \sigma' \} = \varnothing$$

Proprietà di safety

- **Esempio 1:**
  - Si PA={CS1,CS2} e Perr è la proprietà dicendo che due processi non sono mai in sezione critiche allo stesso momento
  - Perr=$\{P_0\ P_1\ P_2 \ldots \in (2^{PA})^\omega \mid \nexists\, i.\ \{CS1,CS2\} \in P_i \}$
  - Perr è una proprietà di safety
  - Se $\sigma = P_0\ P_1\ P_2 \ldots \notin$ Perr allora esiste i tale che $\{CS1,CS2\} \in P_i$ e si prendiamo $\sigma' = P_0\ P_1\ P_2 \ldots.\ P_i\ \sigma'' \ldots$ con $\sigma'' \in (2^{PA})^\omega$ abbiamo $\sigma' \notin (2^{PA})^\omega \backslash$Perr
- **Esempio 2:**
  - PA={verde,rosso, giallo} e Pgr dice che ogni volta che il semaforo è rosso, era giallo nello stato precedente
  - Pgr=$\{P_0\ P_1\ P_2 \ldots \in (2^{PA})^\omega \mid \forall i.\ rosso \in P_i \Rightarrow i>0$ e $giallo \in P_{i-1} \}$
  - Pgr è una proprietà di safety

Proprietà di safety

- **Esempio 3:**
  - Esempio di proprietà temporale lineare che non è una proprietà di safety
  - PA={verde,rosso, giallo}
  - Pv : il semaforo è giorno un giorno
  - Pv=$\{P_0\ P_1\ P_2 \ldots \in (2^{PA})^\omega \mid \exists i. \{verde\} \in P_i \}$
  - Prendiamo $\sigma = \emptyset\ \emptyset\ \emptyset\ \emptyset \ldots.$
  - $\sigma \notin$ Pv, ma per ogni prefisso $\pi = \emptyset \ldots \emptyset$ di σ, abbiamo $\sigma' = \pi$ {verde} $\emptyset\ \emptyset\ \emptyset\ \emptyset \ldots.$ in Pv

Proprietà di liveness

- **Intuizione:**
  - Proprietà di safety: qualcosa di non voluto non succede mai
  - Proprietà di liveness: qualcosa di voluto accade un giorno
- Una proprietà temporale lineare è Pliv è una **proprietà di liveness** se per ogni sequenza finita $\pi \in (2^{PA})^*$ esiste $\sigma \in (2^{PA})^\omega$ tale che $\sigma' = \pi\sigma$ è in Pliv
- Ogni sequenza finita può dunque essere estesa per formare una sequenza di Pliv

Esempio

- Un sistema concorrente per la sezione critica con due processi
- PA={wait1,cs1,wait2,cs2}
- Prendiamo le proprietà:
  1) Ogni processo entrerà un giorno in sezione critica
  2) Ogni processo entrerà infinitamente spesso in sezione critica
  3) Ogni processo in attesa entrerà un giorno in sezione critica
- Possono essere scritte cosi:
  - P1=$\{P_0\ P_1\ P_2\ \ldots \in (2^{PA})^{\omega} \mid \exists\, i. \exists\, j.\ cs1 \in P_i$ e $cs2 \in P_j\}$
  - P2=$\{P_0\ P_1\ P_2\ \ldots \in (2^{PA})^{\omega} \mid$ esiste un numero infinito di i tale che $cs1 \in P_i$ e un numero infinito di j tale che $cs2 \in P_j\}$
  - P3=$\{P_0\ P_1\ P_2\ \ldots \in (2^{PA})^{\omega} \mid \forall i.\ wait1 \in P_i \Rightarrow \exists\, k.\ k \geq i$ e $cs1 \in P_k$ e $\forall j.\ wait2 \in P_j \Rightarrow \exists\, l.\ l \geq j$ e $cs2 \in P_l\ \}$

P1, P2 e P3 sono delle proprietà di liveness

Connessione fra safety e liveness

Teorema:
Se P è una proprietà di liveness e di safety allora $P=(2^{PA})^{\omega}$

Prova:

- Sia P una proprietà di liveness e di safety. Allora per ogni sequenza finita $\pi \in (2^{PA})^{*}$ esiste $\sigma \in (2^{PA})^{\omega}$ tale che $\sigma'=\pi\sigma$ è in P. Ma come P è una proprietà di safety, abbiamo necessariamente $(2^{PA})^{\omega} \setminus P= \emptyset$.Dunque $P=(2^{PA})^{\omega}$

Esistono delle proprietà che non sono ne di safety ne di liveness. Ad esempio:

- I due processi non sono mai insieme in sezione critica e un giorno uno dei due arriva in sezione critica

Forma delle proprietà

Teorema:
Se P è una proprietà temporale lineare allora $P = P_{safe} \cap P_{live}$ dove

Psafe è un proprietà di safety e Plive una proprietà di liveness.

Prova:

- Definiamo $Psafe = \{\sigma \in (2^{PA})^{\omega} \mid \forall \pi \in (2^{PA})^{*}.$ se π è prefisso di σ allora $\exists \sigma' \in (2^{PA})^{\omega}$ tale che $\pi\sigma' \in P\}$
- $Plive = P \cup ((2^{PA})^{\omega} \setminus Psafe)$
- Per provare che $P = Psafe \cap Plive$, notare che $P \subseteq Psafe$