Automi Ibridi

Carla Piazza[1]

[1]Dipartimento di Matematica ed Informatica
Università di Udine
carla.piazza@dimi.uniud.it

Indice del Corso (Dis)Ordinato

- Automi Ibridi: Sintassi e Semantica
- Sistemi a stati finiti (breve ripasso)
- Il problema della Raggiungibilità
- Risultati di Indecidibilità
- Classi notevoli di Automi Ibridi: timed, rectangular, o-minimal, ...
- Tecniche di Decisione: (Bi)Simulazione, Cylindric Algebraic Decomposition, Teoremi di Selezione, Semantiche approssimate
- ...e tanto altro:
  - Logiche temporali
  - Composizione di Automi
  - Il caso Stocastico
  - Stabilità, Osservabilità, Controllabilità
  - Strumenti Software
  - Applicazioni

Una tecnica generale

- dato un grafo $G = (V, E)$ e due insiemi $I, F \subseteq V$ vogliamo stabilire se $I \rightsquigarrow F$
- supponiamo di essere in grado di definire $G/\sim$, $\widetilde{I}$ e $\widetilde{F}$ tali che

$$\widetilde{I} \rightsquigarrow \widetilde{F} \qquad \textit{iff} \qquad I \rightsquigarrow F$$

- se G è finito, potrebbe essere conveniente lavorare su $G/\sim$
- se G è infinito e $G/\sim$ è finito, potrebbe essere necessario lavorare su $G/\sim$

Una tecnica generale

- dato un grafo $G = (V, E)$ e due insiemi $I, F \subseteq V$ vogliamo stabilire se $I \rightsquigarrow F$
- supponiamo di essere in grado di definire $G/\sim$, $\widetilde{I}$ e $\widetilde{F}$ tali che

$$\widetilde{I} \rightsquigarrow \widetilde{F} \qquad \textit{iff} \qquad I \rightsquigarrow F$$

- se G è finito, potrebbe essere conveniente lavorare su $G/\sim$
- se G è infinito e $G/\sim$ è finito, potrebbe essere necessario lavorare su $G/\sim$

Una tecnica generale

- dato un grafo $G = (V, E)$ e due insiemi $I, F \subseteq V$ vogliamo stabilire se $I \rightsquigarrow F$
- supponiamo di essere in grado di definire $G/\sim$, $\widetilde{I}$ e $\widetilde{F}$ tali che

$$\widetilde{I} \rightsquigarrow \widetilde{F} \qquad \textit{iff} \qquad I \rightsquigarrow F$$

- se G è finito, potrebbe essere conveniente lavorare su $G/\sim$
- se G è infinito e $G/\sim$ è finito, potrebbe essere necessario lavorare su $G/\sim$

Una tecnica generale

- dato un grafo $G = (V, E)$ e due insiemi $I, F \subseteq V$ vogliamo stabilire se $I \rightsquigarrow F$
- supponiamo di essere in grado di definire $G/\sim$, $\widetilde{I}$ e $\widetilde{F}$ tali che

$$\widetilde{I} \rightsquigarrow \widetilde{F} \qquad \textit{iff} \qquad I \rightsquigarrow F$$

- se G è finito, potrebbe essere conveniente lavorare su $G/\sim$
- se G è infinito e $G/\sim$ è finito, potrebbe essere necessario lavorare su $G/\sim$

Bisimulazione

Definition

Sia $G = (V, E, \ell)$, una bisimulazione su G è una relazione $R \subseteq V \times V$ tale che:

- se $u\ R\ v$, allora $\ell(u) = \ell(v)$
- se $u \to u'$ e $u\ R\ v$, allora $\exists v'$ tale che $v \to v'$ e $u'\ R\ v'$
- se $v \to v'$ e $u\ R\ v$, allora $\exists u'$ tale che $u \to u'$ e $u'\ R\ v'$

Theorem

Sia $\sim = \{(u, v) \mid \exists R \text{ bisimulazione } u\ R\ v\}$
$\sim$ *è la massima bisimulazione su* G *ed è una relazione di equivalenza*

Bisimulazione

Definition

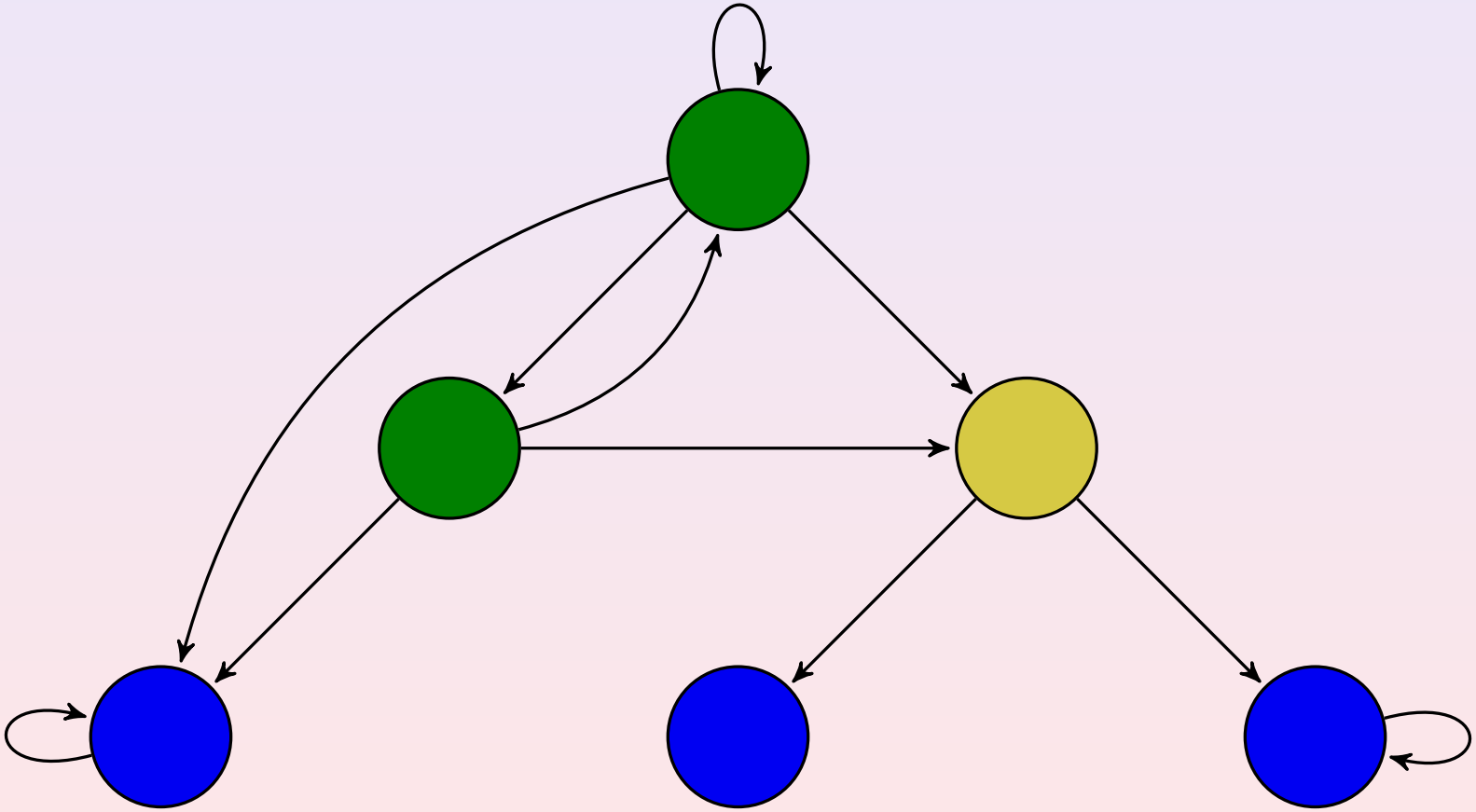
Sia $G = (V, E, \ell)$, una bisimulazione su G è una relazione $R \subseteq V \times V$ tale che:

- se $u\ R\ v$, allora $\ell(u) = \ell(v)$
- se $u \to u'$ e $u\ R\ v$, allora $\exists v'$ tale che $v \to v'$ e $u'\ R\ v'$
- se $v \to v'$ e $u\ R\ v$, allora $\exists u'$ tale che $u \to u'$ e $u'\ R\ v'$

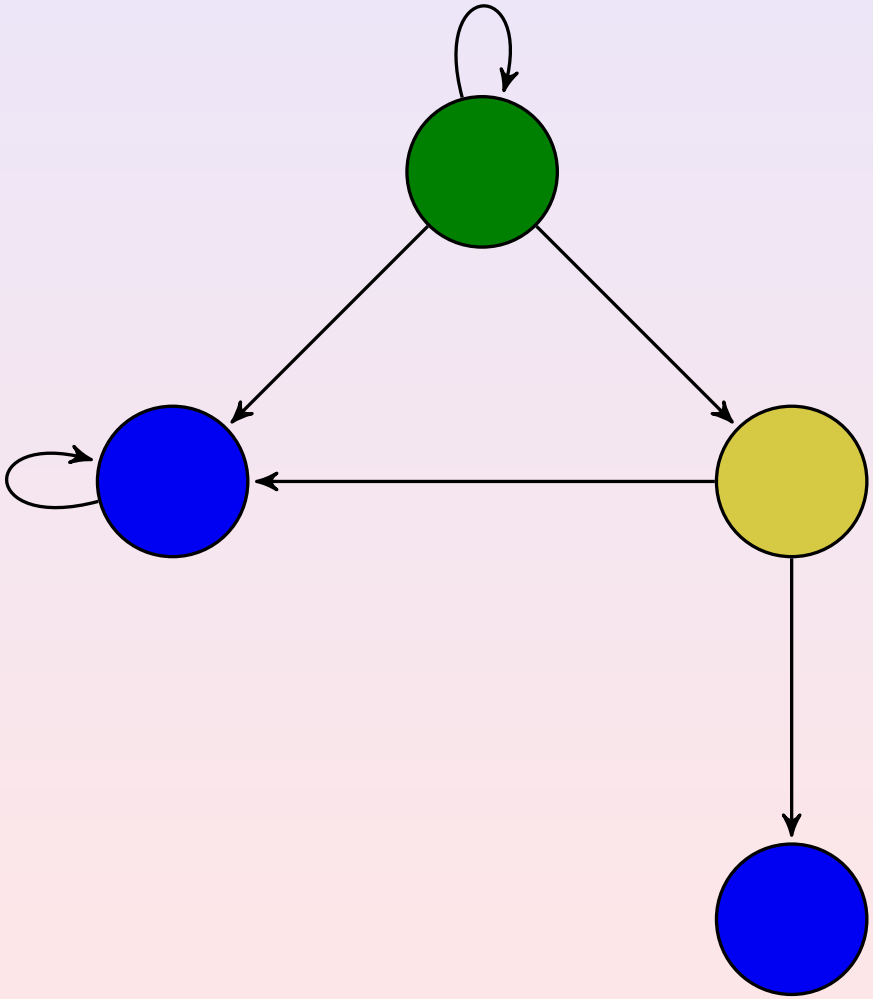
Theorem

Sia $\sim = \{(u, v) \mid \exists R$ *bisimulazione* $u\ R\ v\}$
$\sim$ *è la massima bisimulazione su* G *ed è una relazione di equivalenza*

Example



Example



Bisimulazione e Raggiungibilità

Sia $G = (V, E)$, $I, F \subseteq V$

considero $\ell : V \to \{0, 1\}$ tale che $\ell(u) = 1$ sse $u \in F$

Theorem

In G vale che $I \rightsquigarrow F$ sse in $G/\sim$ vale che $\cup_{i \in I} [i]_\sim \rightsquigarrow \cup_{f \in F} [f]_\sim$

Come calcolo $G/\sim$?

Bisimulazione e Raggiungibilità

Sia $G = (V, E)$, $I, F \subseteq V$

considero $\ell : V \to \{0, 1\}$ tale che $\ell(u) = 1$ sse $u \in F$

Theorem

In G vale che $I \rightsquigarrow F$ sse in $G/\sim$ vale che $\cup_{i \in I} [i]_\sim \rightsquigarrow \cup_{f \in F} [f]_\sim$

Come calcolo $G/\sim$?

Bisimulazione e Raggiungibilità

Sia $G = (V, E)$, $I, F \subseteq V$

considero $\ell : V \to \{0, 1\}$ tale che $\ell(u) = 1$ sse $u \in F$

Theorem

In G vale che $I \rightsquigarrow F$ sse in $G/\sim$ vale che $\cup_{i \in I} [i]_\sim \rightsquigarrow \cup_{f \in F} [f]_\sim$

Come calcolo $G/\sim$?

Bisimulazione e Raggiungibilità

Sia $G = (V, E)$, $I, F \subseteq V$

considero $\ell : V \to \{0, 1\}$ tale che $\ell(u) = 1$ sse $u \in F$

Theorem

In G vale che $I \rightsquigarrow F$ sse in $G/\sim$ vale che $\cup_{i \in I} [i]_\sim \rightsquigarrow \cup_{f \in F} [f]_\sim$

Come calcolo $G/\sim$?

Bisimulazione e Partizionamento Stabile

- parto da $V/\sim = \{V \setminus F, F\}$
- calcolo $E^{-1}(F)$
- "splitto" ogni classe X in $X_1 = X \cap E^{-1}(F)$ e $X_2 = X \setminus X_1$
- itero fino a che non raggiungo un punto fisso

- se G è finito la procedura termina dopo al più $|V|$ passi
- se G è infinito la procedura termina sse
  - sono in grado di calcolare gli split e
  - $V/\sim$ è finito

Bisimulazione e Partizionamento Stabile

- parto da $V/\sim = \{V \setminus F, F\}$
- calcolo $E^{-1}(F)$
- "splitto" ogni classe X in $X_1 = X \cap E^{-1}(F)$ e $X_2 = X \setminus X_1$
- itero fino a che non raggiungo un punto fisso

- se G è finito la procedura termina dopo al più $|V|$ passi
- se G è infinito la procedura termina sse
  - sono in grado di calcolare gli split e
  - $V/\sim$ è finito

Bisimulazione e Partizionamento Stabile

- parto da $V/\sim = \{V \setminus F, F\}$
- calcolo $E^{-1}(F)$
- "splitto" ogni classe X in $X_1 = X \cap E^{-1}(F)$ e $X_2 = X \setminus X_1$
- itero fino a che non raggiungo un punto fisso

- se G è finito la procedura termina dopo al più $|V|$ passi
- se G è infinito la procedura termina sse
  - sono in grado di calcolare gli split e
  - $V/\sim$ è finito

Bisimulazione e Partizionamento Stabile

- parto da $V/\sim = \{V \setminus F, F\}$
- calcolo $E^{-1}(F)$
- "splitto" ogni classe X in $X_1 = X \cap E^{-1}(F)$ e $X_2 = X \setminus X_1$
- itero fino a che non raggiungo un punto fisso

- se G è finito la procedura termina dopo al più $|V|$ passi
- se G è infinito la procedura termina sse
  - sono in grado di calcolare gli split e
  - $V/\sim$ è finito

Bisimulazione e Partizionamento Stabile

- parto da $V/\sim = \{V \setminus F, F\}$
- calcolo $E^{-1}(F)$
- "splitto" ogni classe X in $X_1 = X \cap E^{-1}(F)$ e $X_2 = X \setminus X_1$
- itero fino a che non raggiungo un punto fisso

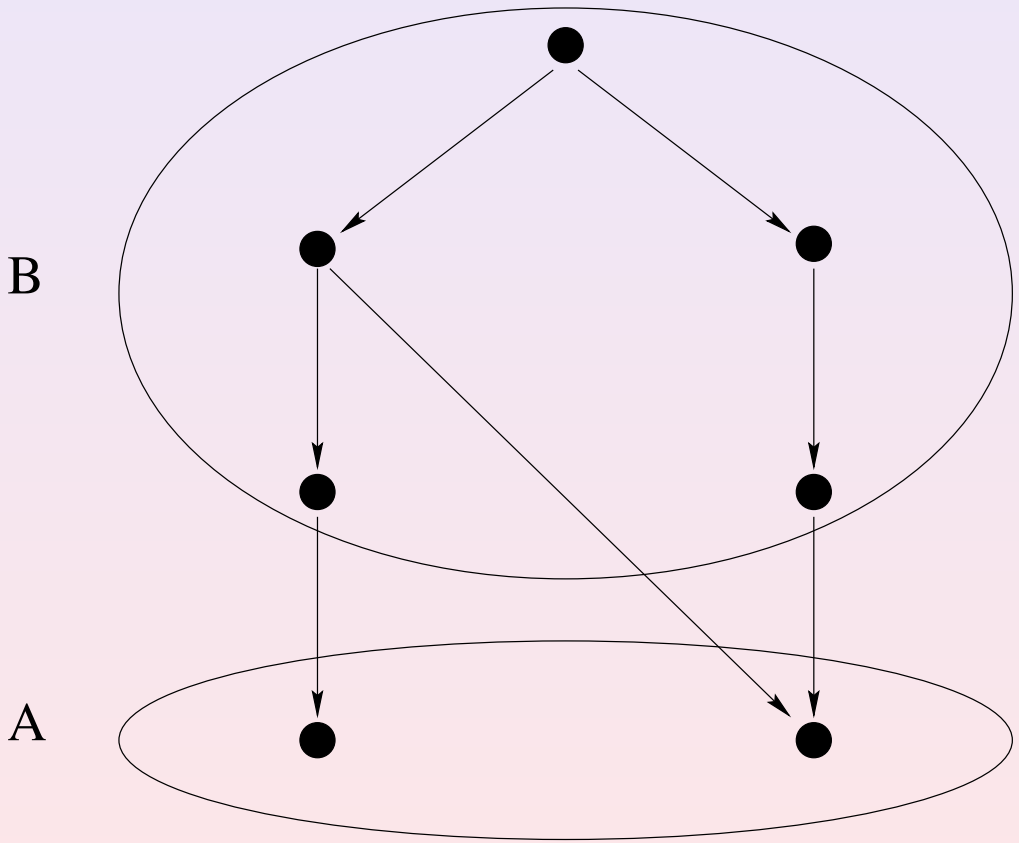
- se G è finito la procedura termina dopo al più $|V|$ passi
- se G è infinito la procedura termina sse
  - sono in grado di calcolare gli split e
  - $V/\sim$ è finito

Bisimulazione e Partizionamento Stabile

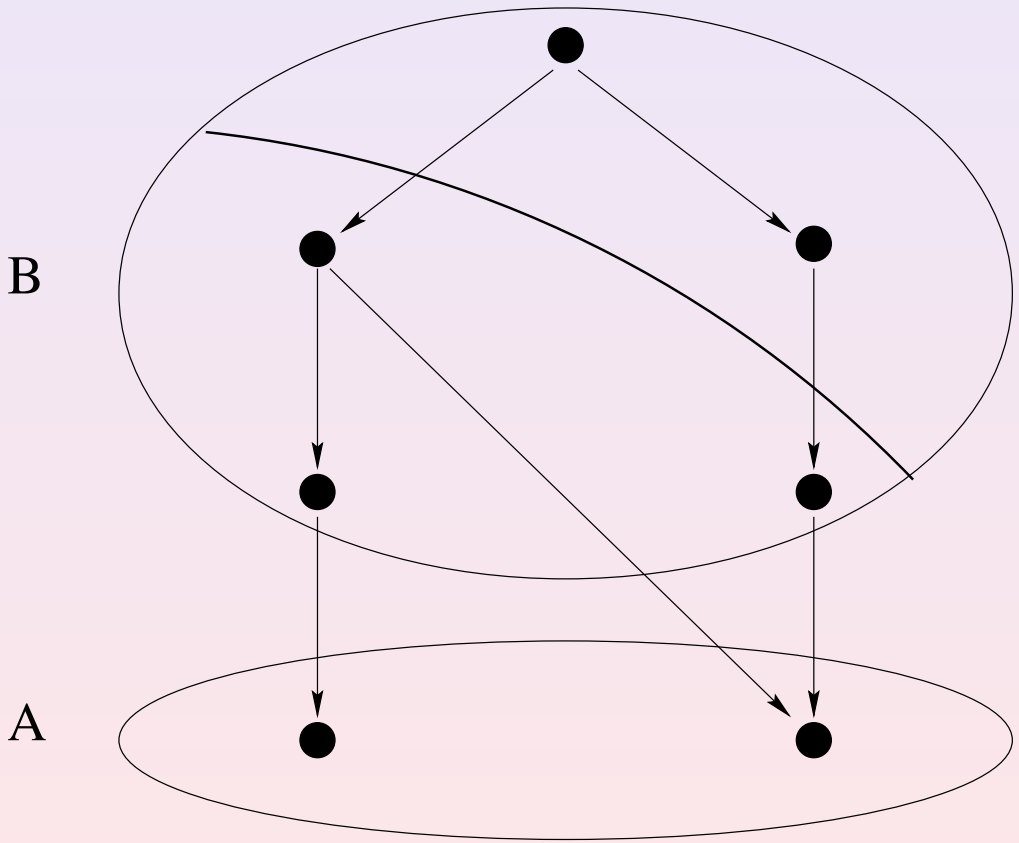
- parto da $V/\sim = \{V \setminus F, F\}$
- calcolo $E^{-1}(F)$
- "splitto" ogni classe X in $X_1 = X \cap E^{-1}(F)$ e $X_2 = X \setminus X_1$
- itero fino a che non raggiungo un punto fisso

- se G è finito la procedura termina dopo al più $|V|$ passi
- se G è infinito la procedura termina sse
  - sono in grado di calcolare gli split e
  - $V/\sim$ è finito

L'Operazione di Split



L'Operazione di Split



Ricordiamo alcune Definizioni

Consideriamo le possibili Dinamiche

Una variabile continua Z è detta:

- Clock se $Z' = Z + T$ in ogni locazione, ovvero $\dot{Z} = 1$
- Skewed Clock se $Z' = Z + gT$ con $g \notin \{0, 1\}$, ovvero $\dot{Z} = g$
- Memory Cell se $Z' = Z$ con $g \notin \{0, 1\}$, ovvero $\dot{Z} = 0$
- Stopwatch se $Z' = Z$ oppure $Z' = Z + T$, ovvero $\dot{Z} = 1$ oppure $\dot{Z} = 0$
- Two Slope se $Z' = Z + gT$ oppure $Z' = Z + hT$, ovvero $\dot{Z} = g$ oppure $\dot{Z} = h$, con $g, h > 0$
- Linear se per ogni locazione esistono a e b tali che $a \leq \dot{Z} \leq b$

Ricordiamo alcune Definizioni

Consideriamo le possibili Activation

- Weak Predicate ($\mathcal{W}$): congiunzione di vincoli del tipo $Z_i \leq n$ oppure $Z_i \geq n$
- $\mathcal{W}_=$: vincoli di $\mathcal{W}$ e vincoli del tipo $Z_i = Z_j$ (confronto tra variabili)
- $\mathcal{W}_{=2}$: vincoli di $\mathcal{W}_=$ in cui al posto di Z_i può comparire $2 * Z_i$
- $\mathcal{W}_{=+}$: vincoli di $\mathcal{W}_=$ in cui al posto di Z_i può comparire $Z_i + Z_j$

Ricordiamo alcune Definizioni

Consideriamo i possibili Reset

- Pass: $Z_i' := Z_i$ (identità)
- Reset: $Z_i' := 0$
- Switch: $Z_i' := Z_j$ con $i \neq j$

Indichiamo con:

- $\mathcal{R}$: reset Pass e Reset
- $\mathcal{V}$: reset Pass, Reset e Switch

Ricordiamo 2-rate Timed Automata

Theorem

Il problema della raggiungibilità per automi con:

- *variabili di tipo clock*
- *una variabile di tipo skewed clock*
- *activation ed in* $\mathcal{W}_{=}$
- *reset in* $\mathcal{R}$

è indecidibile

Alla base del risultato c'è la possibilità di confrontare variabili

Timed Automata

Definition (Timed Automata)

In un timed automaton H abbiamo:

- variabili di tipo clock
- activation e invarianti in $\mathcal{W}$
- reset in $\mathcal{R}$

Theorem (Alur e Dill)

Se H è un timed automaton, allora $H/\sim$ è finito e calcolabile

Corollary

Se H è un timed automaton, allora la raggiungibilità in H è decidibile

Timed Automata

Definition (Timed Automata)

In un timed automaton H abbiamo:

- variabili di tipo clock
- activation e invarianti in $\mathcal{W}$
- reset in $\mathcal{R}$

Theorem (Alur e Dill)

Se H è un timed automaton, allora $H/\sim$ è finito e calcolabile

Corollary

Se H è un timed automaton, allora la raggiungibilità in H è decidibile

Timed Automata

Definition (Timed Automata)

In un timed automaton H abbiamo:

- variabili di tipo clock
- activation e invarianti in $\mathcal{W}$
- reset in $\mathcal{R}$

Theorem (Alur e Dill)

Se H è un timed automaton, allora $H/\sim$ è finito e calcolabile

Corollary

Se H è un timed automaton, allora la raggiungibilità in H è decidibile

Intuitivamente

Siano $c_1, c_2, \ldots, c_k$ le più grandi costanti con cui vengono confrontate $Z_1, Z_2, \ldots, Z_k$

Considero $\sim \subseteq \mathbb{R}^k \times \mathbb{R}^k$ tale che $(p_1, \ldots, p_k) \sim (q_1, \ldots, q_k)$ sse:

- $\lfloor p_i \rfloor = \lfloor q_i \rfloor$ oppure $p_i, q_i \geq c_i$
- $fract(p_i) \leq fract(p_j)$ sse $fract(q_i) \leq fract(q_j)$
- $fract(p_i) = 0$ sse $fract(q_i) = 0$

Intuitivamente

Siano $c_1, c_2, \ldots, c_k$ le più grandi costanti con cui vengono confrontate $Z_1, Z_2, \ldots, Z_k$

Considero $\sim \subseteq \mathbb{R}^k \times \mathbb{R}^k$ tale che $(p_1, \ldots, p_k) \sim (q_1, \ldots, q_k)$ sse:

- $\lfloor p_i \rfloor = \lfloor q_i \rfloor$ oppure $p_i, q_i \geq c_i$
- $fract(p_i) \leq fract(p_j)$ sse $fract(q_i) \leq fract(q_j)$
- $fract(p_i) = 0$ sse $fract(q_i) = 0$

Intuitivamente

Siano $c_1, c_2, \ldots, c_k$ le più grandi costanti con cui vengono confrontate $Z_1, Z_2, \ldots, Z_k$

Considero $\sim \subseteq \mathbb{R}^k \times \mathbb{R}^k$ tale che $(p_1, \ldots, p_k) \sim (q_1, \ldots, q_k)$ sse:

- $\lfloor p_i \rfloor = \lfloor q_i \rfloor$ oppure $p_i, q_i \geq c_i$
- $fract(p_i) \leq fract(p_j)$ sse $fract(q_i) \leq fract(q_j)$
- $fract(p_i) = 0$ sse $fract(q_i) = 0$

Intuitivamente

Siano $c_1, c_2, \ldots, c_k$ le più grandi costanti con cui vengono confrontate $Z_1, Z_2, \ldots, Z_k$

Considero $\sim \subseteq \mathbb{R}^k \times \mathbb{R}^k$ tale che $(p_1, \ldots, p_k) \sim (q_1, \ldots, q_k)$ sse:

- $\lfloor p_i \rfloor = \lfloor q_i \rfloor$ oppure $p_i, q_i \geq c_i$
- $fract(p_i) \leq fract(p_j)$ sse $fract(q_i) \leq fract(q_j)$
- $fract(p_i) = 0$ sse $fract(q_i) = 0$

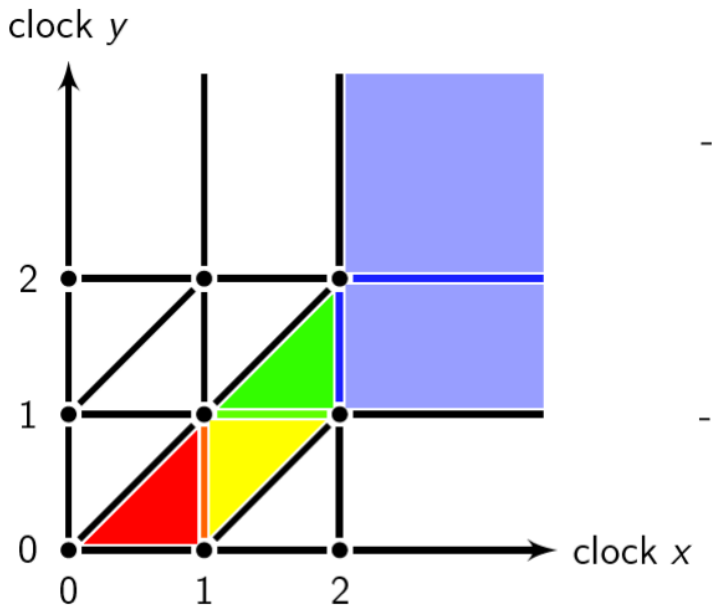
Intuitivamente

Siano $c_1, c_2, \ldots, c_k$ le più grandi costanti con cui vengono confrontate $Z_1, Z_2, \ldots, Z_k$

Considero $\sim \subseteq \mathbb{R}^k \times \mathbb{R}^k$ tale che $(p_1, \ldots, p_k) \sim (q_1, \ldots, q_k)$ sse:

- $\lfloor p_i \rfloor = \lfloor q_i \rfloor$ oppure $p_i, q_i \geq c_i$
- $fract(p_i) \leq fract(p_j)$ sse $fract(q_i) \leq fract(q_j)$
- $fract(p_i) = 0$ sse $fract(q_i) = 0$

Intuitivamente nel caso bidimensionale



Simple Multirate Automata

Definition (Simple Multirate Automata)

In un multirate automaton H abbiamo:

- variabili di tipo skewed clock
- activation e invarianti in $\mathcal{W}$
- reset in $\mathcal{R}$

Theorem

Il problema della raggiungibilità su simple multirate automata è decidibile

Hanno la bisimulazione finita

Bisimulazione Infinita: Rectangular Automata

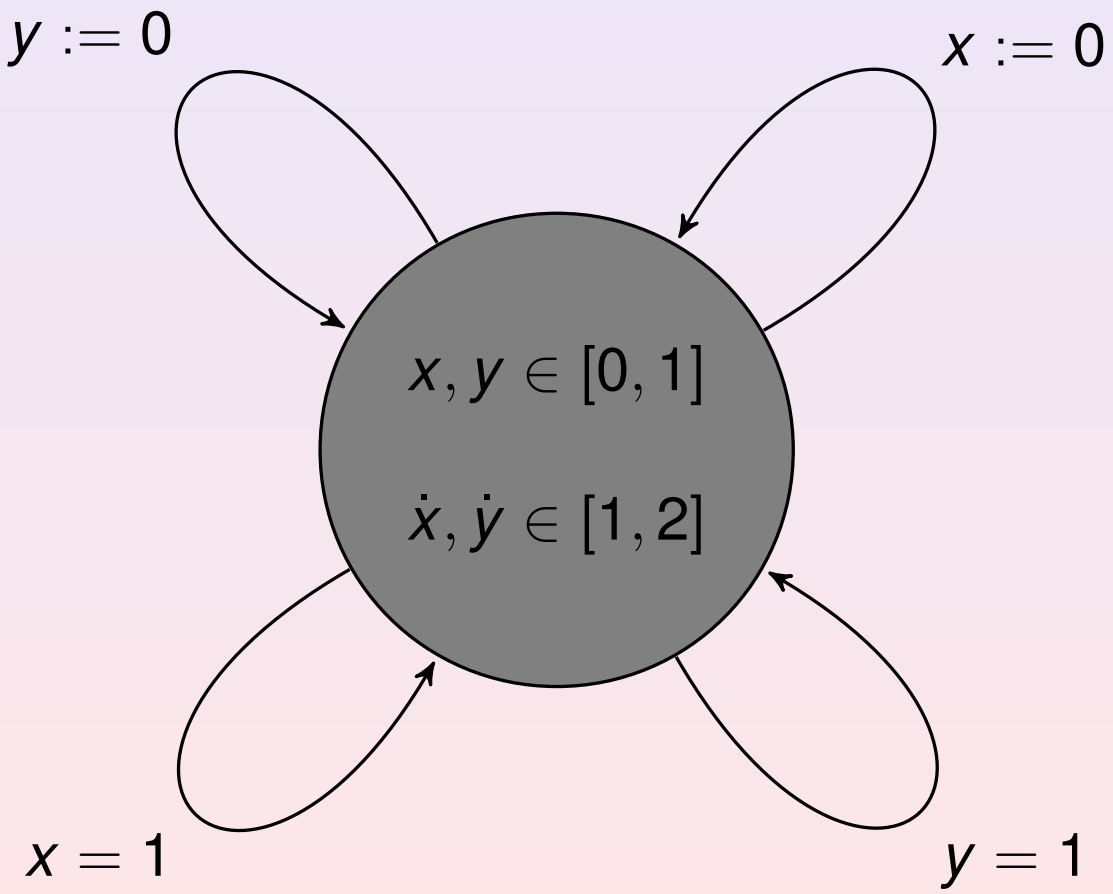
Definition (Rectangular Automata)

Un rettangolo $R \subseteq \mathbb{R}^k$ è il prodotto cartesiano di punti e intervalli
In un rectangular automata gli invariant, flow, activation, reset sono rettangoli
In un inizialized rectangular automata se cambia un flow resetto la variabile

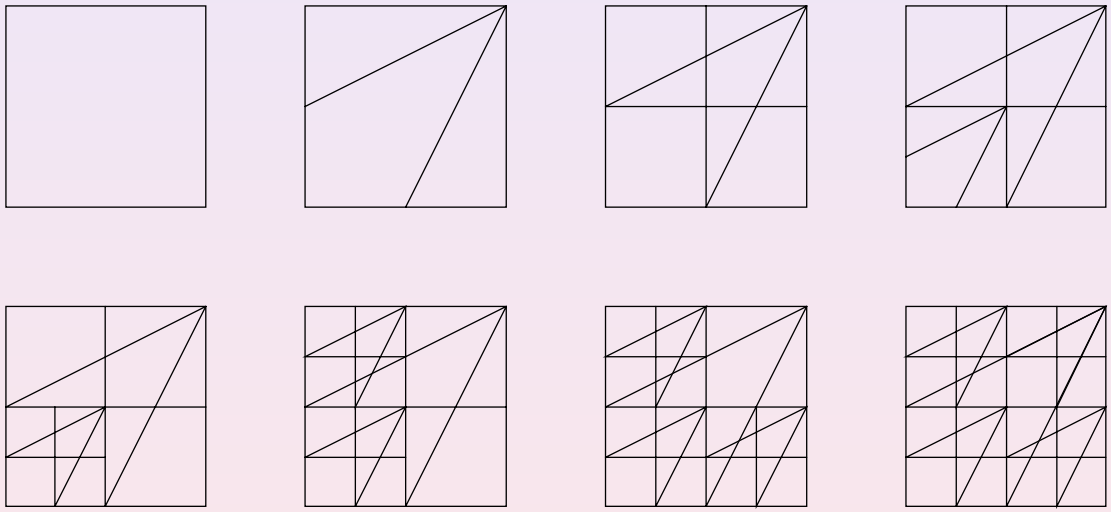
Bisimulazione Infinita

Esistono inizialized rectangular automata con bisimulazione infinita

Esempio



Esempio



Simulazione

Definition

Sia $G = (V, E, \ell)$, una simulazione su G è una relazione $S \subseteq V \times V$ tale che:

- se $u\ S\ v$, allora $\ell(u) = \ell(v)$
- se $u \to u'$ e $u\ S\ v$, allora $\exists v'$ tale che $v \to v'$ e $u'\ S\ v'$

Theorem

Sia $\preccurlyeq = \{(u, v) \mid \exists S \text{ simulazione } u\ S\ v\}$
$\preccurlyeq$ *è la massima simulazione su* G *ed è un preordine*
La relazione $\eqcirc \preccurlyeq \cap \preccurlyeq^{-1}$ *è una equivalenza e preserva la raggiungibilità*

Simulazione

Definition

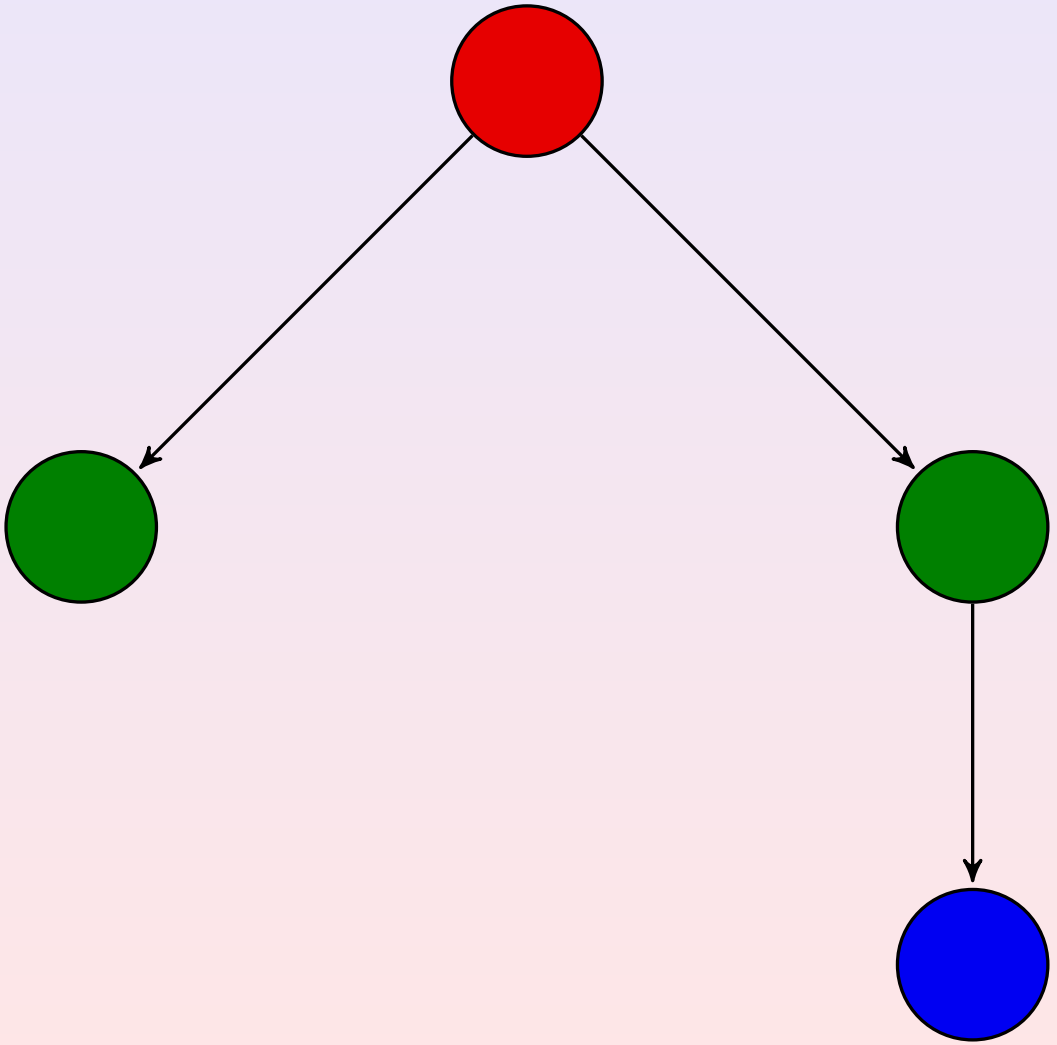
Sia $G = (V, E, \ell)$, una simulazione su G è una relazione $S \subseteq V \times V$ tale che:

- se $u\ S\ v$, allora $\ell(u) = \ell(v)$
- se $u \to u'$ e $u\ S\ v$, allora $\exists v'$ tale che $v \to v'$ e $u'\ S\ v'$

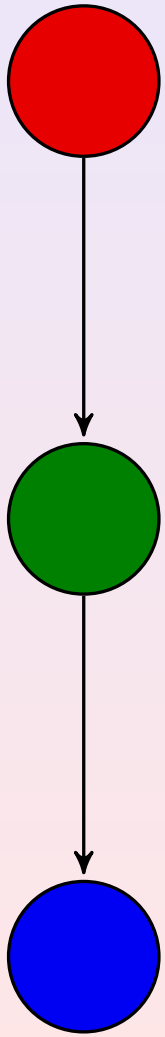
Theorem

Sia $\preccurlyeq = \{(u, v) \mid \exists S \text{ simulazione } u\ S\ v\}$
$\preccurlyeq$ *è la massima simulazione su* G *ed è un preordine*
La relazione $\simeq_{\preccurlyeq} \cap \preccurlyeq^{-1}$ *è una equivalenza e preserva la raggiungibilità*

Simili ma non Bisimili



(a)



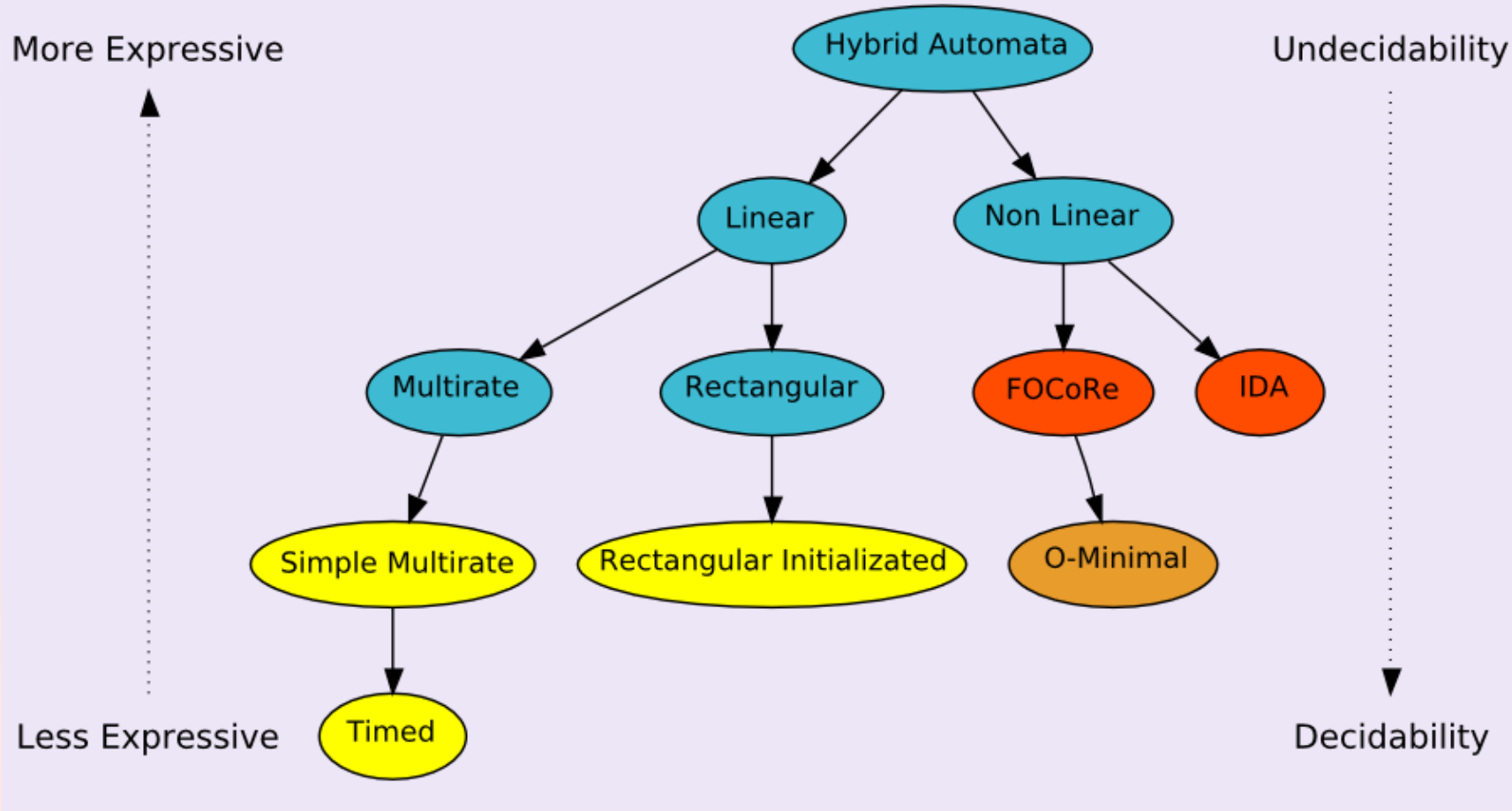
(b)

Simulazione e Inizialized Rectangular Automata

Theorem

Gli Inizialized Rectangular Automata hanno sempre la simulazione finita
Il problema della raggiungibilità su Inizialized Rectangular Automata è decidibile

Gerarchia di Classi Decidibili



Bibliografia

- The Algorithmic Analysis of Hybrid Systems. Alur, Courcobetis, Halbwachs, Henzinger, Ho, Nicolin, Olivero, Sifakis, Yovine. Theoretical Computer Science 138(1), 1995.
- What's Decidable about Hybrid Automata? Henzinger, Kopke, Puri, Varaiya. Journal of Computer and System Sciences 57(1), 1998.
- Hybrid Automata and Bisimulation Casagrande, 2010.
- The Theory of Rectangular Hybrid Automata Kopke. PhD thesis, 1996.