Un diluvio di semantiche

The Linear Time - Branching Time Spectrum

Agnese Gini

17 maggio 2014

Processo

Comportamento di un sistema.

Teoria de processi

Studio di vari aspetti riguardanti i processi, come

- *modellazione*
- *verifica*.

Processo

Comportamento di un sistema.

Teoria de processi

Studio di vari aspetti riguardanti i processi, come

- *modellazione*
- *verifica*.

Processi in esame:

- *Sequenziali*
- *Concreti*
- *Finitely branching*

Labelled Transition System

Un LTS è una coppia $(\mathbb{P}, \rightarrow)$ con $\mathbb{P}$ una classe e $\rightarrow \subseteq \mathbb{P} \times Act \times \mathbb{P}$ tale che $\forall\ p \in \mathbb{P}$ e $a \in Act$ $\{q \in \mathbb{P} | (p, a, q) \in \rightarrow\}$ è un insieme.

Scriveremo $p \xrightarrow{a} q$ per $(p, a, q) \in \rightarrow$, dove i predicati binari $\xrightarrow{a}$ sono detti *action relations* .

Definizione

- *Generalized action relations* $p \xrightarrow{\sigma} q$ per $\sigma \in Act^*$ sono definite ricorsivamente:
  - $\forall p \; p \xrightarrow{\epsilon} p$.
  - $(p, a, q) \in \rightarrow$ con $a \in Act^* \Rightarrow p \xrightarrow{a} q$.
  - $p \xrightarrow{\sigma} q \xrightarrow{\rho} r \Rightarrow p \xrightarrow{\sigma\rho} r$.
- $q \in \mathbb{P}$ è *raggiungibile* da $p \in \mathbb{P}$ se $p \xrightarrow{\sigma} q$ per qualche $\sigma \in Act^*$.
- L'*insieme delle azioni iniziali* di un processo p è definito da :
  $I(p) := \{a \in Act | \exists q : p \xrightarrow{a} q\}$.
- Un processo è detto *finito* se l'insieme $\{(\sigma, q) \in (Act^* \times \mathbb{P}) | p \xrightarrow{\sigma} q\}$ è finito.
- p è *a immagine finita* se $\forall \sigma \in Act^*$ l'insieme $\{q \in \mathbb{P} | p \xrightarrow{\sigma} q\}$ è finito.
- p è *deterministico* se $p \xrightarrow{\sigma} q \wedge p \xrightarrow{\sigma} r \Rightarrow q = r$.
- p è *finitely branching* se $\forall q$ raggiungibile da p l'insieme
  $\{(a, r) \in (Act \times \mathbb{P}) | q \xrightarrow{a} r\}$ è finito.

Definizione

Un *process graph* su un alfabeto Act^* è un grafo diretto e radicato, i cui spigoli sono etichettati dagli elementi di Act. Si ha dunque che un grafo g è dato dalla tripla $(\mathtt{NODES}(g), \mathtt{ROOT}(g), \mathtt{EDGES}(g))$ dove

- $\mathtt{NODES}(g)$ è un insieme i cui elementi sono detti *nodi* o *stati*.
- $\mathtt{ROOT}(g) \in \mathtt{NODES}(g)$ è la *radice* o *stato iniziale*.
- $\mathtt{EDGES}(g) \subseteq \mathtt{NODES} \times Act \times \mathtt{NODES}$ è un insieme si triple (s, a, t) tali che $s, t \in \mathtt{NODES}(g)$ e $a \in Act$ che sono gli *spigoli* o *transizioni* di g.

Diremo *path* π (finito) in un process graph è una sequenza alternata di nodi e spigoli che inizia e finisce con un nodo e tale che ogni transizione va dal nodo che la precede a quello che la segue.
Esempio:

$$\pi = s_0(s_0, a_1, s_1)s_1(s_1, a_2, s_2)\cdots(s_{n-1}, a_n, s_n)s_n$$

$$\pi : s_0 \xrightarrow{a_1} s_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} s_n$$

Diremo dunque che π *inizia da* s_0 *e termina in* $end(\pi) = s_n$.

Diremo *path* π (finito) in un process graph è una sequenza alternata di nodi e spigoli che inizia e finisce con un nodo e tale che ogni transizione va dal nodo che la precede a quello che la segue.
Esempio:

$$\pi = s_0(s_0, a_1, s_1)s_1(s_1, a_2, s_2)\cdots(s_{n-1}, a_n, s_n)s_n$$

$$\pi : s_0 \xrightarrow{a_1} s_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} s_n$$

Diremo dunque che π *inizia da* s_0 *e termina in* $end(\pi) = s_n$.

Definizione

$\mathtt{PATHS}(g)$ *è l'insieme dei paths in* g *che partono dalla radice.*

Dominio

$\mathbb{G} = \{$ grafi di processi connessi su $Act\}$

Dominio

$\mathbb{G} = \{$ grafi di processi connessi su $Act\}$

Diremo inotre che due grafi $h, g \in \mathbb{G}$ sono *isomorfi*, $g \cong h$ se $\exists\ f :\ \mathtt{NODES}(g) \mapsto \mathtt{NODES}(h)$ tale che

- $f(\mathtt{ROOT}(g)) = \mathtt{ROOT}(h)$
- $(s, a, t) \in \mathtt{EDGES}(g) \ \Leftrightarrow\ (f(s), a, f(t)) \in \mathtt{EDGES}(h)$.

Definizione

Sia $(\mathbb{P}, \rightarrow)$ un LTS qualsiasi e $p \in \mathbb{P}$. Diremo *G(p) grafo canonico di p* il grafo dato da:

- $\mathtt{NODES}(G(p)) = \{q \in \mathbb{P} | \exists \sigma \in Act^* : p \xrightarrow{\sigma} q\}$,
- $\mathtt{ROOT}(G(p)) = p \in \mathtt{NODES}(G(p))$,
- $(q, a, r) \in \mathtt{EDGES}(G(p))$ *se* $q, r \in \mathtt{NODES}(G(p))$ *e* $q \xrightarrow{a} r$

È chiaro che $G(p) \in \mathbb{G}$ e dunque G induce una applicazione tra $\mathbb{P}$ e $\mathbb{G}$.

Definizione

Sia $(\mathbb{P}, \rightarrow)$ un LTS qualsiasi e $p \in \mathbb{P}$. Diremo *G(p) grafo canonico di p* il grafo dato da:

- $\mathtt{NODES}(G(p)) = \{q \in \mathbb{P} | \exists \sigma \in Act^* : p \xrightarrow{\sigma} q\}$,
- $\mathtt{ROOT}(G(p)) = p \in \mathtt{NODES}(G(p))$,
- $(q, a, r) \in \mathtt{EDGES}(G(p))$ *se* $q, r \in \mathtt{NODES}(G(p))$ *e* $q \xrightarrow{a} r$

È chiaro che $G(p) \in \mathbb{G}$ e dunque G induce una applicazione tra $\mathbb{P}$ e $\mathbb{G}$.

Proposizione

$\mathbb{P} \hookrightarrow \mathbb{G}$ è un'immersione e dunque ogni LTS su Act può essere rappresentato come una sottoclasse $G(\mathbb{P}) = \{G(p) \in \mathbb{G} | p \in \mathbb{P}\}$

Ogni semantica che esamineremo di baserà su

Osservabilità

$$\mathcal{O} : \mathbb{P} \to \mathcal{O}(\mathbb{P})$$

Ogni semantica che esamineremo di baserà su

Osservabilità

$$\mathcal{O} : \mathbb{P} \to \mathcal{O}(\mathbb{P})$$

Per ogni processo $\mathcal{O}(p)$ costituisce l'insieme dei comportamenti osservabili di p. E data una tale funzione si associa

- **Relazione di equivalenza** $=_{\mathcal{O}} \in \mathbb{P} \times \mathbb{P}$

$$p =_{\mathcal{O}} q \ \Leftrightarrow \mathcal{O}(p) = \mathcal{O}(q)$$

- **Preordine** $p \sqsubseteq_{\mathcal{O}} q \in \mathbb{P} \times \mathbb{P}$

$$p \sqsubseteq_{\mathcal{O}} q \ \Leftrightarrow \mathcal{O}(p) \subseteq \mathcal{O}(q)$$

Che da un ordine parziale sulle classi.

Proposizione

$$p \sqsubseteq_{\mathcal{O}} q \;\Leftrightarrow G(p) \sqsubseteq_{\mathcal{O}} G(q)$$

$$p =_{\mathcal{O}} q \;\Leftrightarrow G(p) =_{\mathcal{O}} G(q)$$

Proposizione

$$p \sqsubseteq_{\mathcal{O}} q \;\Leftrightarrow G(p) \sqsubseteq_{\mathcal{O}} G(q)$$

$$p =_{\mathcal{O}} q \;\Leftrightarrow G(p) =_{\mathcal{O}} G(q)$$

Possiamo dunque introdurre un operatore di confronto $\mathcal{N} \preceq_{\mathbb{P}} \mathcal{O}$, ossia se $p =_{\mathcal{N}} q \Rightarrow p =_{\mathcal{O}} q$.

Proposizione

$$p \sqsubseteq_{\mathcal{O}} q \;\Leftrightarrow G(p) \sqsubseteq_{\mathcal{O}} G(q)$$

$$p =_{\mathcal{O}} q \;\Leftrightarrow G(p) =_{\mathcal{O}} G(q)$$

Possiamo dunque introdurre un operatore di confronto $\mathcal{N} \preceq_{\mathbb{P}} \mathcal{O}$, ossia se $p =_{\mathcal{N}} q \Rightarrow p =_{\mathcal{O}} q$.

Obiettivo

Classificare tramite $\preceq_{\mathbb{P}}$ le possibili semantiche sugli LTS

Trace Semantics ($\mathcal{T}$)

Definizione

$\sigma \in Act^*$ è una *trace* di un processo p se esiste un processo q tale che $p \xrightarrow{\sigma} q$, diremo dunque $T(p)$ l'insieme delle traces di p.

Trace Semantics ($\mathcal{T}$)

Definizione

$\sigma \in Act^*$ è una *trace* di un processo p se esiste un processo q tale che $p \xrightarrow{\sigma} q$, diremo dunque $T(p)$ l'insieme delle traces di p.

Definizione

p e q sono *trace* equivalent, $p =_{\mathcal{T}} q$, se $T(p) = T(q)$.
L'equivalenza induce la *Trace semantics (T)*.

Testing Scenario
Due processi sono identificati se hanno lo stesso insieme di osservazioni, che in questo caso consistono semplicemente di una sequenza di azioni successive.

Caratterizzazione modale

L'insieme $\mathcal{L}_T$ di *Formule di trace* su *Act* è definito ricorsivamente da:

- $\top \in \mathcal{L}_T$,
- $\varphi \in \mathcal{L}_T \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_T$.

Caratterizzazione modale

L'insieme $\mathcal{L}_T$ di *Formule di trace* su *Act* è definito ricorsivamente da:

- $\top \in \mathcal{L}_T$,
- $\varphi \in \mathcal{L}_T \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_T$.

La *soddisfacibilità* $\models \subseteq \mathbb{P} \times \mathcal{L}_T$ è definita da:

•$p \models \top$ per ogni $p \in \mathbb{P}$,

•$p \models a\varphi$ se per qualche $q \in \mathbb{P} : p \xrightarrow{a} q$ e $q \models \varphi$.

Caratterizzazione modale

L'insieme $\mathcal{L}_T$ di *Formule di trace* su *Act* è definito ricorsivamente da:

- $\top \in \mathcal{L}_T$,
- $\varphi \in \mathcal{L}_T \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_T$.

La *soddisfacibilità* $\models \subseteq \mathbb{P} \times \mathcal{L}_T$ è definita da:

•$p \models \top$ per ogni $p \in \mathbb{P}$,

•$p \models a\varphi$ se per qualche $q \in \mathbb{P} : p \xrightarrow{a} q$ e $q \models \varphi$.

Proposizione

$p =_T q \Leftrightarrow \forall \varphi \in \mathcal{L}_T \ (p \models \varphi \Leftrightarrow q \models \varphi)$

Caratterizzazione Process Graph
Sia $g \in \mathbb{G}^{mr}$ e $\pi : s_0 \xrightarrow{a_1} s_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} s_n \ \in \mathtt{PATH}(g)$. Allora $T(\pi) := a_1 a_2 \cdots a_n \in Act^*$ è la *trace* di π.

Caratterizzazione Process Graph

Sia $g \in \mathbb{G}^{mr}$ e $\pi : s_0 \xrightarrow{a_1} s_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} s_n \in \mathtt{PATH}(g)$. Allora $T(\pi) := a_1 a_2 \cdots a_n \in Act^*$ è la *trace* di π.

Proposizione

$T(g) = \{T(\pi) | \pi \in \mathtt{PATHS}(g)\}$.

Modello Esplicito

Il *dominio delle tracce* $\mathbb{T}$ è l'insieme dei sottoinsiemi T di Act^* che soddisfa

T1 $\epsilon \in T$

T2 $\sigma\rho \in T \Rightarrow \sigma \in T$

Modello Esplicito

Il *dominio delle tracce* $\mathbb{T}$ è l'insieme dei sottoinsiemi T di Act^* che soddisfa

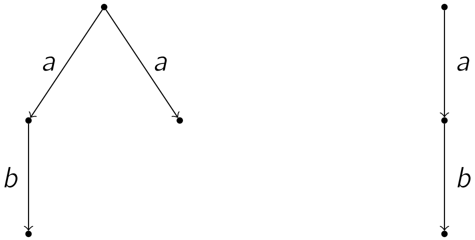
T1 $\epsilon \in T$

T2 $\sigma\rho \in T \Rightarrow \sigma \in T$

Proposizione

$T \in \mathbb{T} \Leftrightarrow \exists g \in \mathbb{G} : T(g) = T \Leftrightarrow \exists g \in \mathbb{G}^{mr} : T(g) = T$

Esempio



$$=_{\mathcal{T}}$$

Completed trace semantics $(\mathcal{CT})$

Definizione

$\sigma \in Act^*$ è una *complete trace* di un processo p se esiste un processo q tale che $p \xrightarrow{\sigma} q$ e $I(q) = \emptyset$, diremo dunque $CT(p)$ l'insieme delle complete traces di p.

Completed trace semantics $(\mathcal{CT})$

Definizione

$\sigma \in Act^*$ è una *complete trace* di un processo p se esiste un processo q tale che $p \xrightarrow{\sigma} q$ e $I(q) = \emptyset$, diremo dunque $CT(p)$ l'insieme delle complete traces di p.

Definizione

p e q sono *completed trace* equivalent, $p =_{\mathcal{CT}} q$, se $T(p) = T(q)$ e $CT(p) = CT(q)$.
L'equivalenza induce la *Completed trace semantics (CT)*.

Testing Scenario

Il processo è idenificato con una *black box* la quale si interfaccia col mondo esterno tramite un display che mostra l'azione è compiuta dal processo in quel momento. Il processo sceglie autonomente un path d'esecuzione che è coerente con la posizione nell'LTS $(\mathbb{P}, \rightarrow)$. Non appena non sono possibili più azioni il processo va in deadlock. L'osservatore guarda il display e registra la sequenza, si assume che l'osservazione possa interrompersi prima che il processo subisca una stasi (trace).

Caratterizzazione modale

L'insieme $\mathcal{L}_{CT}$ di *formule della completed trace* su *Act* è definito ricorsivamente da:

- $\top \in \mathcal{L}_{CT}$,
- $0 \in \mathcal{L}_{CT}$,
- $\varphi \in \mathcal{L}_{CT} \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_{CT}$.

Caratterizzazione modale

L'insieme $\mathcal{L}_{CT}$ di *formule della completed trace* su *Act* è definito ricorsivamente da:

- $\top \in \mathcal{L}_{CT}$,
- $0 \in \mathcal{L}_{CT}$,
- $\varphi \in \mathcal{L}_{CT} \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_{CT}$.

La *soddisfacibilità* $\models \subseteq \mathbb{P} \times \mathcal{L}_{CT}$ è definita da:

•$p \models \top$ per ogni $p \in \mathbb{P}$,

•$p \models 0$ se $I(p) = \varnothing$,

•$p \models a\varphi$ se per qualche $q \in \mathbb{P} : p \xrightarrow{a} q$ e $q \models \varphi$.

Esempio

$p = a_1 a_2 \cdots a_k \top$

$p = a_1 a_2 \cdots a_k 0$

Esempio

$p = a_1 a_2 \cdots a_k \top$

$p = a_1 a_2 \cdots a_k 0$

Proposizione

$p =_{\mathcal{CT}} q \Leftrightarrow \forall \varphi \in \mathcal{L}_{CT} \; (p \models \varphi \Leftrightarrow q \models \varphi)$

Caratterizzazione Process Graph

Sia $g \in \mathbb{G}^{mr}$ e $s \in \mathtt{NODES}(g)$.

$I(s) := \{a \in Act | \exists t : (s, a, t) \in \mathtt{EDGES}(g)\}$ è il *menu* di s.

Proposizione

$CT(g) = \{T(\pi) | \pi \in \mathtt{PATHS}(g) \wedge I(end(\pi)) = \varnothing\}$.

Modello Esplicito

Il *dominio delle tracce complete* $\mathbb{CT}$ è l'insieme delle coppie $(T, CT) \subseteq Act^* \times Act^*$ che soddisfa

$$T \in \mathbb{T} \wedge CT \subseteq T,$$

$$\sigma \in T - TC \Rightarrow \exists a \in Act : \sigma a \in T$$

Modello Esplicito

Il *dominio delle tracce complete* $\mathbb{CT}$ è l'insieme delle coppie $(T, CT) \subseteq Act^* \times Act^*$ che soddisfa

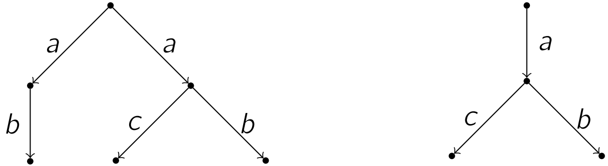
$$T \in \mathbb{T} \wedge CT \subseteq T,$$

$$\sigma \in T - TC \Rightarrow \exists a \in Act : \sigma a \in T$$

Proposizione

$(T, CT) \in \mathbb{CT} \Leftrightarrow \exists g \in \mathbb{G}^{mr} : T(g) = T \wedge CT(g) = T$

Esempio



$$=_{\mathcal{CT}}$$

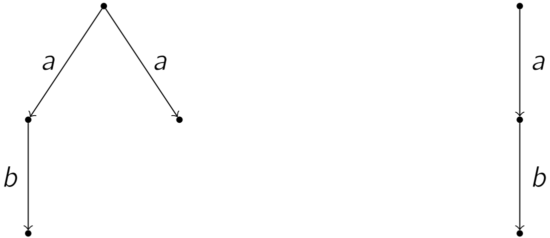
Classificazione

$$T \preceq CT$$

Classificazione

$$T \preceq CT$$

Esempio



$$=_{\mathcal{T}}$$
$$\neq_{\mathcal{CT}}$$

$$\begin{array}{c} CT \\ \downarrow \\ T \end{array}$$

Failure semantics ($\mathcal{F}$)

Definizione

$\langle \sigma, X \rangle \in Act^* \times \mathcal{P}(Act)$ è una *failure pair* di un processo p se esiste un processo q tale che $p \xrightarrow{\sigma} q$ e $I(q) \cap X = \emptyset$, diremo dunque $F(p)$ l'insieme delle failure pairs di p.
X è detto *refusal set*.

Failure semantics ($\mathcal{F}$)

Definizione

$\langle \sigma, X \rangle \in Act^* \times \mathcal{P}(Act)$ è una *failure pair* di un processo p se esiste un processo q tale che $p \xrightarrow{\sigma} q$ e $I(q) \cap X = \emptyset$, diremo dunque $F(p)$ l'insieme delle failure pairs di p.
X è detto *refusal set*.

Definizione

p e q sono *failures* equivalent, $p =_{\mathcal{F}} q$, se $F(p) = F(q)$.
L'equivalenza induce la *Failure semantics (F)*.

Nota: $T(p) = \{\sigma \in Act^* | \langle \sigma, \emptyset \rangle \in F(P)\}$

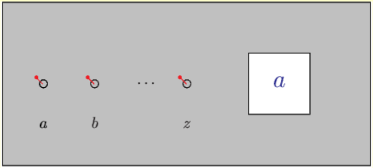
Definizione

Siamo $p \in \mathbb{P}$ e $\sigma \in T(p)$, $Cont_p(\sigma) = \{a \in Act | \sigma a \in T(p)\}$ è l'insieme delle *continuazioni di* σ.

Proposizione

Sia $p \in \mathbb{P}$, $\sigma \in T(p)$ *e* $X \subseteq Act$. *Allora* $\langle \sigma, X \rangle \in F(p) \Leftrightarrow \langle \sigma, X \cap Cont_p(\sigma) \rangle \in F(p)$.

Testing Scenario



In questo caso possiamo vedere un processo come una *black box* che ha sia un display, come per la CT, ma ha anche un interruttore per ogni elemento di *Act*. Gli interruttori, controllati dall'osservatore, segnalano quali azioni sono *libere* e quali *bloccate*. Il processo sceglie autonomente un path d'esecuzione che è coerente con la posizione nell'LTS $(\mathbb{P}, \rightarrow)$, ma con il vincolo che può intraprendere solo le azioni libere. Il processo termina con lo schero vuoto. L'osservatore guarda il display e registra la sequenza σ e le azioni libere rifiutate dall'esecuzione X.

Caratterizzazione modale

L'insieme $\mathcal{L}_F$ di *formule della failure* su *Act* è definito ricorsivamente da:

- $\top \in \mathcal{L}_F$,
- $\widetilde{X} \in \mathcal{L}_F$, per $X \subseteq Act$
- $\varphi \in \mathcal{L}_F \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_F$.

Caratterizzazione modale

L'insieme $\mathcal{L}_F$ di *formule della failure* su *Act* è definito ricorsivamente da:

- $\top \in \mathcal{L}_F$,
- $\widetilde{X} \in \mathcal{L}_F$, per $X \subseteq Act$
- $\varphi \in \mathcal{L}_F \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_F$.

La *soddisfacibilità* $\models \subseteq \mathbb{P} \times \mathcal{L}_F$ è definita da:

•$p \models \top$ per ogni $p \in \mathbb{P}$,

•$p \models \widetilde{X}$ se $I(p) \cap X = \emptyset$,

•$p \models a\varphi$ se per qualche $q \in \mathbb{P} : p \xrightarrow{a} q$ e $q \models \varphi$.

$\widetilde{X}$ rappresenta le osservazioni che il processo rifiuta di quelle in X, dunque si ha un blocco quando X è l'insieme delle azioni permesse.

Esempio

$p = a_1 a_2 \cdots a_k \top$
$p = a_1 a_2 \cdots a_k \widetilde{X}$

Esempio

$p = a_1 a_2 \cdots a_k \top$
$p = a_1 a_2 \cdots a_k \widetilde{X}$

Proposizione

$p =_{\mathcal{F}} q \Leftrightarrow \forall \varphi \in \mathcal{L}_F \; (p \models \varphi \Leftrightarrow q \models \varphi)$

Caratterizzazione Process Graph

Sia $g \in \mathbb{G}^{mr}$ e $\pi \in \mathtt{PATHS}(g)$.
$F(\pi) := \{\langle T(\pi), X\rangle | I(end(\pi)) \cap X = \varnothing\}$ è il failure set di π

Proposizione

$F(g) := \bigcup_{\pi \in \mathtt{PATHS}(g)} F(\pi)$.

Modello Esplicito

Il *dominio dei fallimenti* $\mathbb{F}$ è l'insieme dei sottoinsiemi $F \subseteq Act \times \mathcal{P}(Act)$ che soddisfa

F1 $\langle \epsilon, \varnothing \rangle \in F$

F2 $\langle \sigma\rho, \varnothing \rangle \in F \Leftrightarrow \langle \sigma, \varnothing \rangle \in F$

F3 $\langle \sigma, Y \rangle \in F \wedge Y \subseteq X \Leftrightarrow \langle \sigma, X \rangle \in F$

F4 $\langle \sigma, X \rangle \in F \wedge \forall a \in Y \, (\langle \sigma a, \varnothing \rangle \notin F) \Leftrightarrow \langle \sigma, X \cup Y \rangle \in F$

Modello Esplicito

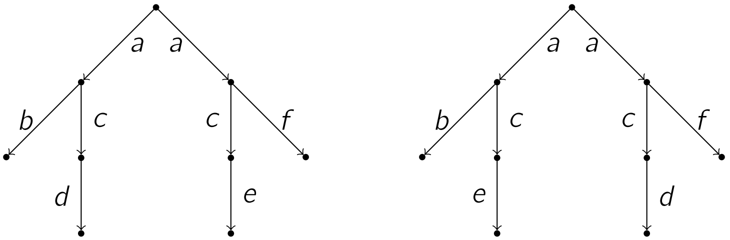
Il *dominio dei fallimenti* $\mathbb{F}$ è l'insieme dei sottoinsiemi $F \subseteq Act \times \mathcal{P}(Act)$ che soddisfa

- F1 $\langle \epsilon, \varnothing \rangle \in F$
- F2 $\langle \sigma\rho, \varnothing \rangle \in F \Leftrightarrow \langle \sigma, \varnothing \rangle \in F$
- F3 $\langle \sigma, Y \rangle \in F \wedge Y \subseteq X \Leftrightarrow \langle \sigma, X \rangle \in F$
- F4 $\langle \sigma, X \rangle \in F \wedge \forall a \in Y \, (\langle \sigma a, \varnothing \rangle \notin F) \Leftrightarrow \langle \sigma, X \cup Y \rangle \in F$

Proposizione

$F \in \mathbb{F} \Leftrightarrow \exists g \in \mathbb{G}^{mr} : F(g) = F$

Esempio



$$=_{\mathcal{F}}$$

Caratterizzazione alternativa

Definizione

Scriveremo p *after* σ *MUST* X se per ogni $q \in \mathbb{P}$ tale che $p \xrightarrow{\sigma} q$ c'è un $a \in I(q)$ con $a \in X$. Porremo $p \simeq q$ se per ogni $\sigma \in Act^*$ e $X \subseteq Act :\ p\ after\ \sigma MUST\ X \Leftrightarrow q\ after\ \sigma MUST\ X$.

Caratterizzazione alternativa

Definizione

Scriveremo p *after* σ *MUST* X se per ogni $q \in \mathbb{P}$ tale che $p \xrightarrow{\sigma} q$ c'è un $a \in I(q)$ con $a \in X$. Porremo $p \simeq q$ se per ogni $\sigma \in Act^*$ e $X \subseteq Act$: p *after* σ *MUST* $X \Leftrightarrow q$ *after* σ *MUST* X.

Proposizione

Sia $p, q \in \mathbb{P}$. *Allora* $p \simeq q \Leftrightarrow p =_{\mathcal{F}} q$

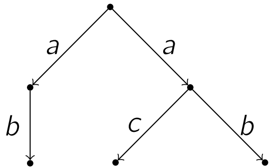
Classificazione

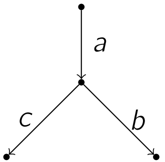
$$CT \preceq F$$

Classificazione

$$CT \preceq F$$

Esempio





$$=_{\mathcal{CT}}$$

$$\neq_{\mathcal{F}}$$

$$F \longrightarrow CT \longrightarrow T$$

Failure trace semantics $(\mathcal{FT})$

Definizione

Sia $X \subseteq Act$ il *refusal set* di un dato processo p.

- Definiamo *refusal relations* $\xrightarrow{X}$ per $X \subseteq Act$ nel seguente modo:
  $p \xrightarrow{X} q$ se $p = q$ e $I(p) \cap X = \varnothing$.
- Definiamo *failure trace relations* $\xrightarrow{\sigma}$ per $\sigma \in (Act \cup \mathcal{P}(Act))^*$ la chiusura riflessiva e transitiva dell'azione e e della refusal relations.
- $\sigma \in (Act \cup \mathcal{P}(Act))^*$ è una *failure trace* di un processo p se esiste q tale che $p \xrightarrow{\sigma} q$ e $FT(P)$ è l'insieme delle failure trace di p

Failure trace semantics ($\mathcal{FT}$)

Definizione

Sia $X \subseteq Act$ il *refusal set* di un dato processo p.

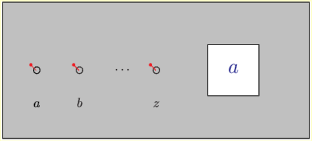
- Definiamo *refusal relations* $\xrightarrow{X}$ per $X \subseteq Act$ nel seguente modo:
  $p \xrightarrow{X} q$ se $p = q$ e $I(p) \cap X = \varnothing$.
- Definiamo *failure trace relations* $\xrightarrow{\sigma}$ per $\sigma \in (Act \cup \mathcal{P}(Act))^*$ la chiusura riflessiva e transitiva dell'azione e e della refusal relations.
- $\sigma \in (Act \cup \mathcal{P}(Act))^*$ è una *failure trace* di un processo p se esiste q tale che $p \xrightarrow{\sigma} q$ e $FT(P)$ è l'insieme delle failure trace di p

Definizione

p e q sono *failure trace* equivalent, $p =_{\mathcal{FT}} q$, se $FT(p) = FT(q)$. L'equivalenza induce la *Failure trace semantics (FT)*.

Testing Scenario

La *failure trace machine* è identica alla *failure machine* ma non rimane ferma in maniera permanente se il processo non può continuare per il fatto che tutte le azioni con cui potrebbe procedere sono state bloccate dell'osservatore. Nei momenti in cui è inattiva, finché l'osservatore non permette un'azione, lo schermo è vuoto. L'osservazione consiste dunque delle sequenza interpolata dagli insiemi rifiutati durante l'attività.



Esempio

$$\{a,b\}cdb\{b,c\}\{b,c,d\}a(Act)$$

a b c d $\varnothing$

Esempio

$$\{a, b\}cdb\{b, c\}\{b, c, d\}a(Act)$$

a	b	c	d	$\varnothing$
a	b	c	d	c

Esempio

$$\{a, b\}cdb\{b, c\}\{b, c, d\}a(Act)$$

a	b	c	d	$\varnothing$
a	b	c	d	c
a	b	c	d	d

Esempio

$$\{a, b\}cdb\{b, c\}\{b, c, d\}a(Act)$$

a	b	c	d	∅
a	b	c	d	c
a	b	c	d	d
a	b	c	d	b

Esempio

$$\{a, b\}cdb\{b, c\}\{b, c, d\}a(Act)$$

a	b	c	d	$\emptyset$
a	b	c	d	c
a	b	c	d	d
a	b	c	d	b
a	b	c	d	$\emptyset$

Esempio

$$\{a,b\}cdb\{b,c\}\{b,c,d\}a(Act)$$

a	b	c	d	$\emptyset$
a	b	c	d	c
a	b	c	d	d
a	b	c	d	b
a	b	c	d	$\emptyset$
a	b	c	d	$\emptyset$

Esempio

$$\{a, b\}cdb\{b, c\}\{b, c, d\}a(Act)$$

a	b	c	d	$\emptyset$
a	b	c	d	c
a	b	c	d	d
a	b	c	d	b
a	b	c	d	$\emptyset$
a	b	c	d	$\emptyset$
a	b	c	d	a

Esempio

$$\{a, b\}cdb\{b, c\}\{b, c, d\}a(Act)$$

a	b	c	d	∅
a	b	c	d	c
a	b	c	d	d
a	b	c	d	b
a	b	c	d	∅
a	b	c	d	∅
a	b	c	d	a
a	b	c	d	∅

Caratterizzazione modale

L'insieme $\mathcal{L}_{FT}$ di *formule della failure* su *Act* è definito ricorsivamente da:

- $\top \in \mathcal{L}_{FT}$,
- $\varphi \in \mathcal{L}_{FT} \ \wedge X \subseteq Act \Rightarrow \widetilde{X}\varphi \in \mathcal{L}_{F}$.
- $\varphi \in \mathcal{L}_{FT} \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_{F}$.

Caratterizzazione modale

L'insieme $\mathcal{L}_{FT}$ di *formule della failure* su *Act* è definito ricorsivamente da:

- $\top \in \mathcal{L}_{FT}$,
- $\varphi \in \mathcal{L}_{FT} \;\wedge X \subseteq Act \Rightarrow \widetilde{X}\varphi \in \mathcal{L}_F$.
- $\varphi \in \mathcal{L}_{FT} \;\wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_F$.

La *soddisfacibilità* $\models \subseteq \mathbb{P} \times \mathcal{L}_F$ è definita da:

•$p \models \top$ per ogni $p \in \mathbb{P}$,

•$p \models \widetilde{X}\varphi$ se $I(p) \cap X = \varnothing$ e $p \models \varphi$,

•$p \models a\varphi$ se per qualche $q \in \mathbb{P} : p \xrightarrow{a} q$ e $q \models \varphi$.

Dove $\widetilde{X}$ rappresenta le osservazioni che il processo rifiuta di quelle in X seguite da φ.

Caratterizzazione modale

L'insieme $\mathcal{L}_{FT}$ di *formule della failure* su *Act* è definito ricorsivamente da:

- $\top \in \mathcal{L}_{FT}$,
- $\varphi \in \mathcal{L}_{FT} \; \wedge X \subseteq Act \Rightarrow \widetilde{X}\varphi \in \mathcal{L}_F$.
- $\varphi \in \mathcal{L}_{FT} \; \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_F$.

La *soddisfacibilità* $\models \subseteq \mathbb{P} \times \mathcal{L}_F$ è definita da:

•$p \models \top$ per ogni $p \in \mathbb{P}$,

•$p \models \widetilde{X}\varphi$ se $I(p) \cap X = \varnothing$ e $p \models \varphi$,

•$p \models a\varphi$ se per qualche $q \in \mathbb{P} : p \xrightarrow{a} q$ e $q \models \varphi$.

Dove $\widetilde{X}$ rappresenta le osservazioni che il processo rifiuta di quelle in X seguite da φ.

Proposizione

$p =_{\mathcal{FT}} q \Leftrightarrow \forall \varphi \in \mathcal{L}_{FT} \; (p \models \varphi \Leftrightarrow q \models \varphi)$

Caratterizzazione Process Graph
Sia $g \in \mathbb{G}^{mr}$ e $\pi : s_0 \xrightarrow{a_1} s_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} s_n \ \in \mathtt{PATHS}(g)$.
Allora $FT(\pi)$ il *failure trace set* di π è il più piccolo sottoinsieme di $(Act \cup \mathcal{P}(Act))^*$ tale che

- $(Act - I(s_0))a_1(Act - I(s_1))a_2 \cdots a_n(Act - I(s_n)) \in FT(\pi)$,
- $\sigma X \rho \in FT(\pi) \Rightarrow \sigma\rho \in FT(\pi)$,
- $\sigma X \rho \in FT(\pi) \Rightarrow \sigma XX\rho \in FT(\pi)$,
- $\sigma X \rho \in FT(\pi) \wedge Y \subset X \Rightarrow \sigma Y \rho \in FT(\pi)$,

Proposizione

$FT(g) := \bigcup_{\pi \in \mathtt{PATHS}(g)} FT(\pi).$

Corollario

$g \in \mathbb{G}^{mr}$ sono failure trace equivalenti se

- *per ogni $\pi \in \mathtt{PATHS}(g)$ in g c'è $\pi' \in \mathtt{PATHS}(h)$ tale che $\pi \leq_{FT} \pi'$*
- *per ogni $\pi \in \mathtt{PATHS}(g)$ in h c'è $\pi' \in \mathtt{PATHS}(g)$ tale che $\pi \leq_{FT} \pi'$*

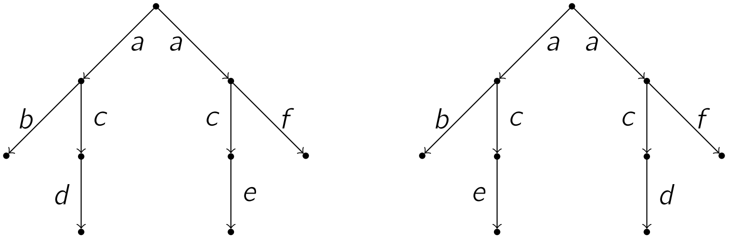
Classificazione

$$F \preceq FT$$

Classificazione

$$F \preceq FT$$

Esempio



$$=_{\mathcal{F}}$$

$$\neq_{\mathcal{FT}}$$

$$\begin{tikzcd}
\mathrm{FT} \arrow[d] \\
\mathrm{F} \arrow[d] \\
\mathrm{CT} \arrow[d] \\
\mathrm{T}
\end{tikzcd}$$

Ready trace semantics $(\mathcal{RT})$

Definizione

- Le *ready trace relations* $\xrightarrow{\sigma^\dagger}$ $\sigma \in (Act \cup \mathcal{P}(Act))^*$ è definita ricorsivamente:
  1. $p \xrightarrow{\epsilon^\dagger} p$ per ogni processo p
  2. $p \xrightarrow{a} q$ implica $p \xrightarrow{a^\dagger} q$
  3. $p \xrightarrow{X^\dagger} q$ con $X \in Act$ ogni volta che $p = q$ e $I(p) = X$
  4. $p \xrightarrow{\sigma^\dagger} q \xrightarrow{\rho^\dagger} r$ allora $p \xrightarrow{\sigma\rho^\dagger} r$
- $\sigma \in (Act \cup \mathcal{P}(Act))^*$ è la *ready trace* di p se esiste q tale che $p \xrightarrow{\sigma^\dagger} q$ e $RT(p)$ è l'insieme delle ready trace di p.

interpolata dagli insiemi rifiutati durante l'attività.

Ready trace semantics ($\mathcal{RT}$)

Definizione

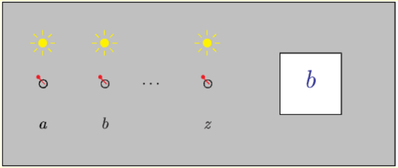
- Le *ready trace relations* $\xrightarrow{\sigma^\dagger} \sigma \in (Act \cup \mathcal{P}(Act))^*$ è definita ricorsivamente:
  1. $p \xrightarrow{\epsilon^\dagger} p$ per ogni processo p
  2. $p \xrightarrow{a} q$ implica $p \xrightarrow{a^\dagger} q$
  3. $p \xrightarrow{X^\dagger} q$ con $X \in Act$ ogni volta che $p = q$ e $I(p) = X$
  4. $p \xrightarrow{\sigma^\dagger} q \xrightarrow{\rho^\dagger} r$ allora $p \xrightarrow{\sigma\rho^\dagger} r$
- $\sigma \in (Act \cup \mathcal{P}(Act))^*$ è la *ready trace* di p se esiste q tale che $p \xrightarrow{\sigma^\dagger} q$ e $RT(p)$ è l'insieme delle ready trace di p.

interpolata dagli insiemi rifiutati durante l'attività.

Definizione

p e q sono *ready trace* equivalenti, $p =_{\mathcal{RT}} q$, se $RT(p) = RT(q)$. L'equivalenza induce la *Ready trace semantics (RT)*.

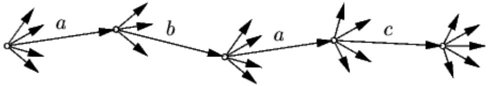
Testing Scenario



La *ready trace machine* è una variante della *failure machine* che ha per ogni elemento di *Act* ha una lampadina. Ogni volta che il processo è inattivo le lampadine delle azioni che il processo è pronto a intraprendere sono accese (tali azioni sono bloccate). L'osservazione consiste dunque delle sequenza azioni (ready trace) e insiemi di parti di *Act* (menus), che rappresentano rispettivante le informazioni date dal display e quelle date dalle lampadine. L'insieme di tali coppie sono i comportamenti osservabili ossia il *ready trace set*.

Esempio

Per ogni nodo si indica la transizione e il menu.



Caratterizzazione modale

L'insieme $\mathcal{L}_{RT}$ di *formule della ready trace* su *Act* è definito ricorsivamente da:

- $\top \in \mathcal{L}_{RT}$,
- $\varphi \in \mathcal{L}_{RT} \ \wedge X \subseteq Act \Rightarrow X\varphi \in \mathcal{L}_{RT}$,
- $\varphi \in \mathcal{L}_{RT} \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_{RT}$.

Caratterizzazione modale

L'insieme $\mathcal{L}_{RT}$ di *formule della ready trace* su *Act* è definito ricorsivamente da:

- $\top \in \mathcal{L}_{RT}$,
- $\varphi \in \mathcal{L}_{RT} \ \wedge X \subseteq Act \Rightarrow X\varphi \in \mathcal{L}_{RT}$,
- $\varphi \in \mathcal{L}_{RT} \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_{RT}$.

La *soddisfacibilità* $\models \subseteq \mathbb{P} \times \mathcal{L}_{RT}$ è definita da:

•$p \models \top$ per ogni $p \in \mathbb{P}$,

•$p \models X\varphi$ se $I(p) = X$ e $p \models a\varphi$,

•$p \models a\varphi$ se per qualche $q \in \mathbb{P} : p \xrightarrow{a} q$ e $q \models \varphi$.

$X\varphi$ rappresenta le osservazioni del menu seguita da φ,

Caratterizzazione modale

L'insieme $\mathcal{L}_{RT}$ di *formule della ready trace* su *Act* è definito ricorsivamente da:

- $\top \in \mathcal{L}_{RT}$,
- $\varphi \in \mathcal{L}_{RT} \ \wedge X \subseteq Act \Rightarrow X\varphi \in \mathcal{L}_{RT}$,
- $\varphi \in \mathcal{L}_{RT} \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_{RT}$.

La *soddisfacibilità* $\models \subseteq \mathbb{P} \times \mathcal{L}_{RT}$ è definita da:

•$p \models \top$ per ogni $p \in \mathbb{P}$,

•$p \models X\varphi$ se $I(p) = X$ e $p \models a\varphi$,

•$p \models a\varphi$ se per qualche $q \in \mathbb{P} : p \xrightarrow{a} q$ e $q \models \varphi$.

$X\varphi$ rappresenta le osservazioni del menu seguita da φ,

Proposizione

$p =_{\mathcal{RT}} q \Leftrightarrow \forall \varphi \in \mathcal{L}_{RT} \ (p \models \varphi \Leftrightarrow q \models \varphi)$

Modello Esplicito

Il *dominio dei fallimenti* $\mathbb{RT}$ è l'insieme dei sottoinsiemi $RT \subseteq Act \times \mathcal{P}(Act)$ che soddisfa

RT1 $\exists X\ (X \in RT)$

RT2 $\sigma X \in RT \wedge a \in X \Leftrightarrow \exists Y\ (\sigma X a Y \in RT)$

Modello Esplicito

Il *dominio dei fallimenti* $\mathbb{RT}$ è l'insieme dei sottoinsiemi $RT \subseteq Act \times \mathcal{P}(Act)$ che soddisfa

RT1 $\exists X \; (X \in RT)$

RT2 $\sigma X \in RT \wedge a \in X \Leftrightarrow \exists Y \; (\sigma X a Y \in RT)$

Proposizione

$R \in \mathbb{RT} \Leftrightarrow \exists g \in \mathbb{G}^{mr} : RT(g) = RT$

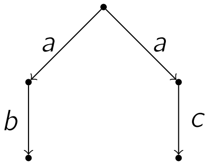
Classificazione

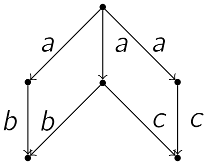
$$FT \preceq RT$$

Classificazione

$$FT \preceq RT$$

Esempio





$$=_{\mathcal{F}}$$
$$=_{\mathcal{FT}}$$
$$\neq_{\mathcal{RT}}$$

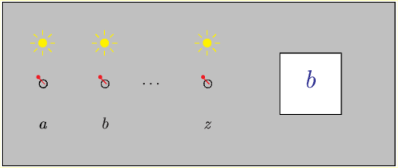
$$\begin{array}{c} \mathsf{RT} \\ \downarrow \\ \mathsf{FT} \\ \downarrow \\ \mathsf{F} \\ \downarrow \\ \mathsf{CT} \\ \downarrow \\ \mathsf{T} \end{array}$$

Readiness semantics $(\mathcal{R})$

Definizione

$\langle \sigma, X \rangle \subseteq Act^* \times \mathcal{P}(Act)$ è detta *ready pair* di un processo p se esiste q tale che $p \xrightarrow{\sigma} q$ e $I(q) = X$.
$R(p)$ è l'insieme delle ready pairs.
p e q sono *ready* equivalenti, $p =_{\mathcal{R}} q$, se $R(p) = R(q)$.
L'equivalenza induce la *Readiness semantics (R)*.

Testing Scenario



La *readiness machine* ha la stessa forma della *ready trace machine*, ma come per la *failure* non può ripredere da uno stato di inattività. La funzione delle lampadine è utilizzata solo una volta durante l'esecuzione. L'osservazione consiste delle sequenza azioni (trace) e di un sottoinsieme di *Act* (menu), che rappresenta una possibile estensione del processo se l'osservatore potesse sbloccarlo.

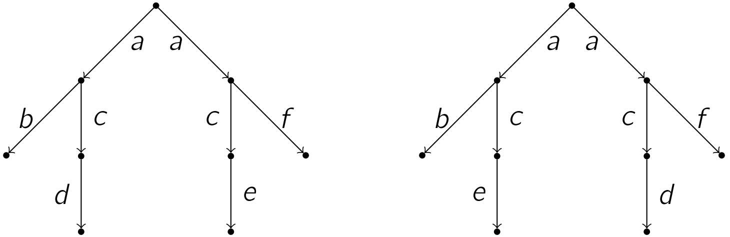
Classificazione

$F \preceq R \preceq RT$ ma R e FT sono indipendenti

Classificazione

$F \preceq R \preceq RT$ ma R e FT sono indipendenti

Esempio



$$=_{\mathcal{R}}$$

$$\neq_{\mathcal{RT}}$$

$$
\begin{tikzcd}
 & RT \arrow[dl] \arrow[d] \\
R \arrow[dr] & FT \arrow[d] \\
 & F \arrow[d] \\
 & CT \arrow[d] \\
 & T
\end{tikzcd}
$$

Simulation semantics ($\mathcal{S}$)

Definizione

Una *simulazione* è una relazione binaria η sui processi tale che $\forall a \in Act$ se $p \eta q$ e $p \xrightarrow{a} p'$ allora $\exists q' : q \xrightarrow{a} q'$ e $p' \eta q'$.
Un processo p *può essere simulato da* q, $p \xrightarrow{\subseteq} q$, se esiste η tale che $p \eta q$.
p e q sono *simili* $p \rightleftarrows q$, se $p \xrightarrow{\subseteq} q$ e $q \xrightarrow{\subseteq} p$.

Simulation semantics ($\mathcal{S}$)

Definizione

Una *simulazione* è una relazione binaria η sui processi tale che $\forall a \in Act$ se $p\eta q$ e $p \xrightarrow{a} p'$ allora $\exists q' : q \xrightarrow{a} q'$ e $p'\eta q'$.
Un processo p *può essere simulato da* q, $p \xrightarrow{\subseteq} q$, se esiste η tale che $p\eta q$.
p e q sono *simili* $p \rightleftarrows q$, se $p \xrightarrow{\subseteq} q$ e $q \xrightarrow{\subseteq} p$.

Proposizione

$\rightleftarrows$ è una relazione d'equivalenza sul dominio dei processi.

Testing Scenario
La situazione è la stessa della trece semantics, ma in più l'osservatore ha, in ogni momento di un'esecuzione del processo in esame, la possibilità di fare un numero arbitrario di copie del processo nello stato corrente e osservarli indipendentemente. L'osservazione consiste dunque di un albero che può essere codificato tramite un espressione.

Caratterizzazione modale

L'insieme $\mathcal{L}_S$ di *formule di simulazione* su Act è definito ricorsivamente da:

- I è un insieme e $\varphi_i \in \mathcal{L}_S$ $\forall i \in I$ allora $\bigwedge_{i \in I} \varphi_i \in \mathcal{L}_S$,
- $\varphi \in \mathcal{L}_S \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_S$.

Caratterizzazione modale

L'insieme $\mathcal{L}_S$ di *formule di simulazione* su *Act* è definito ricorsivamente da:

- I è un insieme e $\varphi_i \in \mathcal{L}_S \; \forall i \in I$ allora $\bigwedge_{i \in I} \varphi_i \in \mathcal{L}_S$,
- $\varphi \in \mathcal{L}_S \;\wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_S$.

La *soddisfacibilità* $\models \subseteq \mathbb{P} \times \mathcal{L}_S$ è definita da:

- $p \models \bigwedge_{i \in I} \varphi_i$ se per ogni $i \in I$ $p \models \varphi_i$,
- $p \models a\varphi$ se per qualche $q \in \mathbb{P} : p \xrightarrow{a} q$ e $q \models \varphi$.

$\top := \bigwedge_{i \in \varnothing} \varphi_i$

Definizione

$S(p) := \{\varphi \in \mathcal{L}_S | \; p \models \varphi\}$ è l'insieme delle classi, e dunque $p =_{\mathcal{S}} q$ se $S(p) = S(q)$.

Caratterizzazione modale

L'insieme $\mathcal{L}_S$ di *formule di simulazione* su *Act* è definito ricorsivamente da:

- I è un insieme e $\varphi_i \in \mathcal{L}_S\ \forall i \in I$ allora $\bigwedge_{i\in I} \varphi_i \in \mathcal{L}_S$,
- $\varphi \in \mathcal{L}_S \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_S$.

La *soddisfacibilità* $\models \subseteq \mathbb{P} \times \mathcal{L}_S$ è definita da:

- $p \models \bigwedge_{i\in I} \varphi_i$ se per ogni $i \in I$ $p \models \varphi_i$,
- $p \models a\varphi$ se per qualche $q \in \mathbb{P} : p \xrightarrow{a} q$ e $q \models \varphi$.

$\top := \bigwedge_{i\in\varnothing} \varphi_i$

Definizione

$S(p) := \{\varphi \in \mathcal{L}_S |\ p \models \varphi\}$ è l'insieme delle classi, e dunque $p =_{\mathcal{S}} q$ se $S(p) = S(q)$.

Proposizione

$p \rightleftarrows q \Leftrightarrow p =_{\mathcal{S}} q$

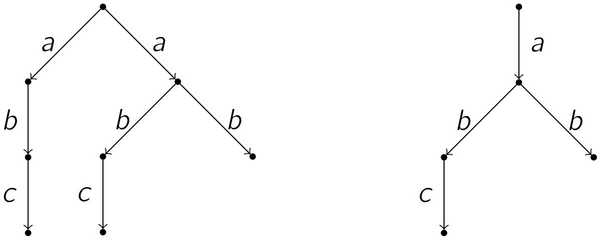
Classificazione

$T \preceq S$ ma è indipendente da CT, R, F, FT, RT

Classificazione

$T \preceq S$ ma è indipendente da CT, R, F, FT, RT

Esempio



$=_S$

$$
\begin{tikzcd}
& RT \arrow[dl] \arrow[d] & \\
R \arrow[dr] & FT \arrow[d] & \\
& F \arrow[d] & S \arrow[ddl] \\
& CT \arrow[d] & \\
& T &
\end{tikzcd}
$$

Bisimulation semantics $(\mathcal{B})$

Definizione

Una *Bisimulazione* è una relazione binaria η sui processi tale che $\forall a \in Act$

- $p\eta q$ e $p \xrightarrow{a} p'$ allora $\exists q' : q \xrightarrow{a} q'$ e $p'\eta q'$.
- $p\eta q$ e $q \xrightarrow{a} q'$ allora $\exists p' : p \xrightarrow{a} p'$ e $p'\eta q'$.

p e q sono *bisimili* $p \leftrightsquigarrow q$, se esiste η tale che $p\eta q$.

Bisimulation semantics $(\mathcal{B})$

Definizione

Una *Bisimulazione* è una relazione binaria η sui processi tale che $\forall a \in Act$

- $p \eta q$ e $p \xrightarrow{a} p'$ allora $\exists q' : q \xrightarrow{a} q'$ e $p' \eta q'$.
- $p \eta q$ e $q \xrightarrow{a} q'$ allora $\exists p' : p \xrightarrow{a} p'$ e $p' \eta q'$.

p e q sono *bisimili* $p \leftrightsquigarrow q$, se esiste η tale che $p \eta q$.

Proposizione

$\leftrightsquigarrow$ è una bisimulazione e una relazione d'equivalenza su $\mathbb{P}$.

Caratterizzazione modale

La classe $\mathcal{L}_B$ di *formule infinitarie di Hennessy-Milner* su *Act* è definita ricorsivamente da:

- I è un insieme e $\varphi_i \in \mathcal{L}_B$ $\forall i \in I$ allora $\bigwedge_{i \in I} \varphi_i \in \mathcal{L}_B$,
- $\varphi \in \mathcal{L}_B \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_B$.
- $\varphi \in \mathcal{L}_B \Rightarrow \neg\varphi \in \mathcal{L}_B$.

Caratterizzazione modale

La classe $\mathcal{L}_B$ di *formule infinitarie di Hennessy-Milner* su *Act* è definita ricorsivamente da:

- I è un insieme e $\varphi_i \in \mathcal{L}_B$ $\forall i \in I$ allora $\bigwedge_{i \in I} \varphi_i \in \mathcal{L}_B$,
- $\varphi \in \mathcal{L}_B \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_B$.
- $\varphi \in \mathcal{L}_B \Rightarrow \neg\varphi \in \mathcal{L}_B$.

La *soddisfacibilità* $\models \subseteq \mathbb{P} \times \mathcal{L}_B$ è definita da:

- $p \models \bigwedge_{i \in I} \varphi_i$ se per ogni $i \in I$ $p \models \varphi_i$,
- $p \models a\varphi$ se per qualche $q \in \mathbb{P} : p \xrightarrow{a} q$ e $q \models \varphi$.
- $p \models \neg\varphi$ se $p \not\models \varphi$.

Definizione

$B(p) := \{\varphi \in \mathcal{L}_B \mid p \models \varphi\}$ è l'insieme delle classi, e dunque $p \sqsubseteq_{\mathcal{B}} q$ se $B(p) \subseteq B(q)$ e $p =_{\mathcal{B}} q$ se $B(p) = B(q)$.

Caratterizzazione modale

La classe $\mathcal{L}_B$ di *formule infinitarie di Hennessy-Milner* su *Act* è definita ricorsivamente da:

- I è un insieme e $\varphi_i \in \mathcal{L}_B$ $\forall i \in I$ allora $\bigwedge_{i \in I} \varphi_i \in \mathcal{L}_B$,
- $\varphi \in \mathcal{L}_B \ \wedge a \in Act \Rightarrow a\varphi \in \mathcal{L}_B$.
- $\varphi \in \mathcal{L}_B \Rightarrow \neg\varphi \in \mathcal{L}_B$.

La *soddisfacibilità* $\models \subseteq \mathbb{P} \times \mathcal{L}_B$ è definita da:

- $p \models \bigwedge_{i \in I} \varphi_i$ se per ogni $i \in I$ $p \models \varphi_i$,
- $p \models a\varphi$ se per qualche $q \in \mathbb{P} : p \xrightarrow{a} q$ e $q \models \varphi$.
- $p \models \neg\varphi$ se $p \not\models \varphi$.

Definizione

$B(p) := \{\varphi \in \mathcal{L}_B |\ p \models \varphi\}$ è l'insieme delle classi, e dunque $p \sqsubseteq_{\mathcal{B}} q$ se $B(p) \subseteq B(q)$ e $p =_{\mathcal{B}} q$ se $B(p) = B(q)$.

Proposizione

$p \leftrightsquigarrow q \Leftrightarrow p \sqsubseteq_{\mathcal{B}} q \Leftrightarrow p =_{\mathcal{B}} q$

Testing Scenario

È una variante della failure trace machine con replicatore ed in più l'osservatore ha la capacità di *global testing*, ossia può enumerare tutti i possibili *ambienti operativi* per ogni stato del test, in modo da garantire che tutti i rami non deterministici siano percorsi da varie copie del processo in esame, e così tutte le possibili mosse di un processo investigate.

Le assunzioni per le implentazioni del *global testing* sono

- le condizioni ambientale determinano la scelta della transizione in ogni momento
- ci sono un numero finito di condizioni possibili
- non sono controllabili.

Caratterizzazione Process Graph
Siano $g, h \in \mathbb{G}$. Una *bisimulazione* tra g e h è una relazione binaria $\eta \subseteq \mathtt{NODES}(g) \times \mathtt{NODES}(h)$ tale che:
• $\mathtt{ROOT}(G)\eta\mathtt{ROOT}(h)$
• Se $s\eta t$ e $(s, a, s') \in \mathtt{EDGES}(g)$, allora c'è $(t, a, t') \in \mathtt{EDGES}(h)$ tale che $s'\eta t'$
• Se $s\eta t$ e $(t, a, t') \in \mathtt{EDGES}(h)$, allora c'è $(s, a, s') \in \mathtt{EDGES}(g)$ tale che $s'\eta t'$

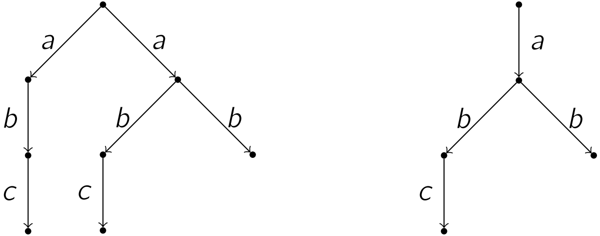
Classificazione

La bisimulazione è più fine di tutte le semantiche precedenti.

Classificazione

La bisimulazione è più fine di tutte le semantiche precedenti.

Esempio



$$=_{\mathcal{S}}$$
$$\neq_{\mathcal{B}}$$

Teorema

$g, h \in \mathbb{G}$ $g \cong h \Rightarrow g$ *e* h *sono bisimili*

Teorema

$g, h \in \mathbb{G}$ $g \cong h \Rightarrow$ g e h sono bisimili

Esempio

Consideriamo $Act = \{a, b, c\}$ e $g, h \in \mathbb{G}$ dati da

Sono bisimili ma non isomorfi.

$$
\begin{tikzcd}
& B \arrow[d] \arrow[dddr] & \\
& RT \arrow[dl] \arrow[d] & \\
R \arrow[dr] & FT \arrow[d] & \\
& F \arrow[d] & S \arrow[ddl] \\
& CT \arrow[d] & \\
& T &
\end{tikzcd}
$$

Possiamo estendere le semantiche date con vari accorgimenti in modo da ottenere altre semantiche per la classe di processi in esame:

$$
\begin{array}{rl}
\mathcal{L}_T & \varphi ::= \top \mid a\varphi'(\varphi' \in \mathcal{L}_T) \\
\mathcal{L}_{CT} & \varphi ::= \top \mid a\varphi'(\varphi' \in \mathcal{L}_{CT}) \mid 0 \\
\mathcal{L}_F & \varphi ::= \top \mid a\varphi'(\varphi' \in \mathcal{L}_F) \mid \widetilde{X}(X \subseteq Act) \\
\mathcal{L}_R & \varphi ::= \top \mid a\varphi'(\varphi' \in \mathcal{L}_R) \mid X(X \subseteq Act) \\
\mathcal{L}_{FT} & \varphi ::= \top \mid a\varphi'(\varphi' \in \mathcal{L}_{FT}) \mid \widetilde{X}\varphi'(X \subseteq Act, \varphi' \in \mathcal{L}_{FT}) \\
\mathcal{L}_{RT} & \varphi ::= \top \mid a\varphi'(\varphi' \in \mathcal{L}_{RT}) \mid X\varphi'(X \subseteq Act, \varphi' \in \mathcal{L}_{RT}) \\
\mathcal{L}_{PF} & \varphi ::= a\varphi'(\varphi' \in \mathcal{L}_{PF}) \mid \bigwedge_{i \in I} \varphi_i \wedge \bigwedge_{j \in J} \neg\varphi'_j(\varphi_i, \varphi'_j \in \mathcal{L}_T) \\
\mathcal{L}_S & \varphi ::= a\varphi'(\varphi' \in \mathcal{L}_S) \mid \bigwedge_{i \in I} \varphi_i(\varphi_i \in \mathcal{L}_S) \\
\mathcal{L}_{CS} & \varphi ::= a\varphi'(\varphi' \in \mathcal{L}_{CS}) \mid \bigwedge_{i \in I} \varphi_i(\varphi_i \in \mathcal{L}_{CS}) \mid 0 \\
\mathcal{L}_{RS} & \varphi ::= a\varphi'(\varphi' \in \mathcal{L}_{RS}) \mid X(X \subseteq Act) \\
\mathcal{L}_{PW} & \bigwedge_{a \in X} a\varphi_a(\varphi_a \in \mathcal{L}_{PW}, X \subseteq Act) \mid X(X \subseteq Act) \\
\mathcal{L}_{2S} & \varphi ::= a\varphi'(\varphi' \in \mathcal{L}_{2S}) \mid \bigwedge_{i \in I} \varphi_i(\varphi_i \in \mathcal{L}_{2S}) \mid \neg\varphi'(\varphi' \in \mathcal{L}_S) \\
\mathcal{L}_B & \varphi ::= \varphi'(\varphi' \in \mathcal{L}_B) \mid \bigwedge_{i \in I} \varphi_i(\varphi_i \in \mathcal{L}_B) \mid \neg\varphi'(\varphi' \in \mathcal{L}_B)
\end{array}
$$

Proposizione

Ognuno dei linguaggi $\mathcal{L}_{\mathcal{O}}$ definiti sopra è una sottolinguaggio di $\mathcal{L}_B$

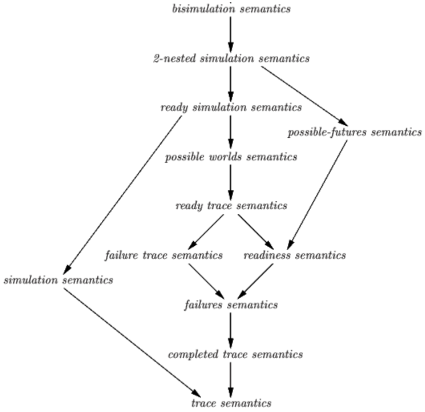
Proposizione

Ognuno dei linguaggi $\mathcal{L}_{\mathcal{O}}$ definiti sopra è una sottolinguaggio di $\mathcal{L}_B$

Teorema di classificazione per le semantiche LTS

$T \preceq CT \preceq F \preceq R \preceq RT$,
$T \preceq F \preceq FT \preceq RT \preceq PW \preceq RS \preceq 2S \preceq B$,
$R \preceq PF \preceq 2S$,
$T \preceq S \preceq CS \preceq RS$ e $CT \preceq CS$

Spettro


bisimulation semantics
2-nested simulation semantics
ready simulation semantics
possible-futures semantics
possible worlds semantics
ready trace semantics
failure trace semantics
readiness semantics
simulation semantics
failures semantics
completed trace semantics
trace semantics

Quali sono i criteri per scegliere la semantica da utilizzare?

Quali sono i criteri per scegliere la semantica da utilizzare?
Alcuni:

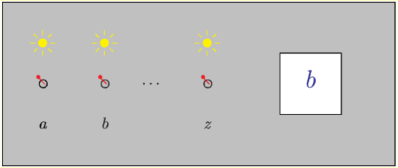
- La possibilità di interagire con i processi.
- Il grado di finezza richiesto per l'equivalenza
- La complessità della verifica

In generale la questione della scelta della semantica per i comportamenti osservabili è un problema molto difficile.

Reactive versus generative testing scenarios

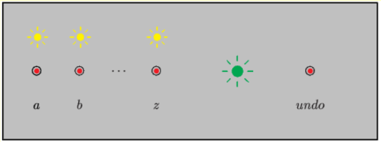
Generative testing scenario

Nei processi investigati la scelta delle azioni de eseguire è fatta in maniera autonoma. L'osservatore può al più limitare il comportamento della *generative machine* ponendo restrizioni sul percorso delle azioni.



Reactive testing scenario

I processi agiscono stimolati dall'ambiente esterno. La *reactive machine* può essere ottenuta dalle generative sostituendo gli interruttori con pulsanti e il display con una luce verde.
Le varie semantiche si ottengono scegliendo il modo in cui l'osservatore può premere i pulsanti (uno per volta o più di uno), il tempo di validità del'input oppure aggiungendo pulsanti "speciali" come *undo*.



Se allarghiamo il dominio ai **processi infiniti**, per ognuna delle semantiche trattate si presentano molteplici possibilità di estensione e ciò rende molto complessa la trattazione.

Se allarghiamo il dominio ai **processi infiniti**, per ognuna delle semantiche trattate si presentano molteplici possibilità di estensione e ciò rende molto complessa la trattazione.

Spettro