Refinements and Abstractions of Signal-Event (Timed) Languages

Paul Gastin

LSV
ENS de Cachan & CNRS
Paul.Gastin@lsv.ens-cachan.fr

Joint work with Béatrice Bérard and Antoine Petit

MFPS, May 26th, 2006

Outline

Refinements and Abstractions

Abstract level		Concrete level
ConnectToServer	$\xrightarrow{\text{refinement}}$ $\xleftarrow{\text{abstraction}}$	Details used to establish the connection

Formalisation of refinement

Let $\sigma : A \to \mathcal{P}(B^*)$ be a substitution.

Abstract level		Concrete level
Action $a \in A$	$\xrightarrow{\text{refinement}}$	$\sigma(a) \subseteq B^*$
Behavior $w = abaac \in A^*$	$\xrightarrow{\text{refinement}}$	$\sigma(w) = \sigma(a)\sigma(b)\sigma(a)\sigma(a)\sigma(c) \subseteq B^*$
Language $K \subseteq A^*$	$\xrightarrow{\text{refinement}}$	$\sigma(K) = \bigcup_{w \in K} \sigma(w) \subseteq B^*$

Refinements and Abstractions

Abstract level		Concrete level
ConnectToServer	$\xrightarrow{\text{refinement}}$ $\xleftarrow{\text{abstraction}}$	Details used to establish the connection

Formalisation of refinement

Let $\sigma : A \to \mathcal{P}(B^*)$ be a substitution.

Abstract level		Concrete level
Action $a \in A$	$\xrightarrow{\text{refinement}}$	$\sigma(a) \subseteq B^*$
Behavior $w = abaac \in A^*$	$\xrightarrow{\text{refinement}}$	$\sigma(w) = \sigma(a)\sigma(b)\sigma(a)\sigma(a)\sigma(c) \subseteq B^*$
Language $K \subseteq A^*$	$\xrightarrow{\text{refinement}}$	$\sigma(K) = \bigcup_{w \in K} \sigma(w) \subseteq B^*$

Refinements and Abstractions

Abstract level	Concrete level
ConnectToServer $\underset{\xleftarrow{\text{abstraction}}}{\xrightarrow{\text{refinement}}}$	Details used to establish the connection

Formalisation of abstraction

Let $\sigma : A \to \mathcal{P}(B^*)$ be a substitution.

Abstract level		Concrete level
$\sigma^{-1}(L) = \{w \in A^* \mid \sigma(w) \cap L \neq \emptyset\}$	$\xleftarrow{\text{abstraction}}$	$L \subseteq B^*$

Adding time to the picture

Timed refinement

Abstract level	$\xrightarrow{\text{refinement}}$ $\xleftarrow{\text{abstraction}}$	Concrete level
$\mathsf{ConnectToServer}^2$		$\mathsf{Req} \cdot \mathsf{Wait}^2 \cdot \mathsf{Ack}$
$\mathsf{ConnectToServer}^{4.5}$		$\mathsf{Req} \cdot \mathsf{Wait}^1 \cdot \mathsf{Nack} \cdot \mathsf{Wait}^{0.5} \cdot \mathsf{Retry} \cdot \mathsf{Wait}^3 \cdot \mathsf{Ack}$

An abstract action a with duration d should be replaced by a concrete execution (word) w with the same duration $\|w\| = d$.

Adding time to the picture

Timed refinement

Abstract level $\xrightarrow{\text{refinement}}$ Concrete level

Abstract level $\xleftarrow{\text{abstraction}}$ Concrete level

ConnectToServer^2	$\text{Req} \cdot \text{Wait}^2 \cdot \text{Ack}$
$\text{ConnectToServer}^{4.5}$	$\text{Req} \cdot \text{Wait}^1 \cdot \text{Nack} \cdot \text{Wait}^{0.5} \cdot \text{Retry} \cdot \text{Wait}^3 \cdot \text{Ack}$

An abstract action a with duration d should be replaced by a concrete execution (word) w with the same duration $\|w\| = d$.

Outline

Signal-Event (Timed) Words

Asarin - Caspi - Maler 2002

- Σ_e finite set of (instantaneous) events
- Σ_s finite set of signals
- $\mathbb{T}$ time domain, $\overline{\mathbb{T}} = \mathbb{T} \cup \{\infty\}$
- $\Sigma = \Sigma_e \cup (\Sigma_s \times \mathbb{T})$
- Notation: a^d for $(a, d) \in \Sigma_s \times \overline{\mathbb{T}}$
- Σ^∞ set of signal-event (timed) words
  Example: $a^3 f f g b^{1.5} a^2 f$
- Signal stuttering: $a^2 a^3 \approx a^5$, $\quad a^\infty = a^2 a^2 a^2 \cdots$

Signal-Event (Timed) Words

Asarin - Caspi - Maler 2002

- Σ_e finite set of (instantaneous) events
- Σ_s finite set of signals
- $\mathbb{T}$ time domain, $\overline{\mathbb{T}} = \mathbb{T} \cup \{\infty\}$
- $\Sigma = \Sigma_e \cup (\Sigma_s \times \mathbb{T})$
- Notation: a^d for $(a, d) \in \Sigma_s \times \overline{\mathbb{T}}$
- Σ^∞ set of signal-event (timed) words
  Example: $a^3 f f g b^{1.5} a^2 f$
- Signal stuttering: $a^2 a^3 \approx a^5$, $\quad a^\infty = a^2 a^2 a^2 \cdots$

Signal-Event (Timed) Words

Asarin - Caspi - Maler 2002

- Σ_e finite set of (instantaneous) events
- Σ_s finite set of signals
- $\mathbb{T}$ time domain, $\overline{\mathbb{T}} = \mathbb{T} \cup \{\infty\}$
- $\Sigma = \Sigma_e \cup (\Sigma_s \times \mathbb{T})$
- Notation: a^d for $(a, d) \in \Sigma_s \times \overline{\mathbb{T}}$
- Σ^∞ set of signal-event (timed) words
  Example: $a^3 f f g b^{1.5} a^2 f$
- Signal stuttering: $a^2 a^3 \approx a^5$, $\qquad a^\infty = a^2 a^2 a^2 \cdots$

Signal-Event (Timed) Words

Unobservable signal τ

- Useful to hide signals:

  Signal-event word $\xrightarrow{\text{hiding signals}}$ Classical timed words

  $a^3 f b^1 g f a^2 f$ $\quad\quad$ $\tau^3 f \tau^1 g f \tau^2 f = (f,3)(g,4)(f,4)(f,6)$

- $\tau^0 \approx \varepsilon$: an hidden signal with zero duration is not observable.
  $a^0 \not\approx \varepsilon$: a signal, even of zero duration, is observable.
  $\tau^2 \not\approx \varepsilon$: we still observe a time delay but the actual signal has been hidden.
  Example : $a^2\tau^0 a^1 f \tau^0 g \tau^1 f b^2 b^2 b^2 \cdots \approx a^3 f g \tau^1 f b^\infty$
- Signal-event words $SE(\Sigma) = \Sigma^\infty / \approx$

Signal-Event (Timed) Words

Unobservable signal τ

- Useful to hide signals:

  Signal-event word $\xrightarrow{\text{hiding signals}}$ Classical timed words

  $a^3 f b^1 g f a^2 f$ $\qquad\qquad$ $\tau^3 f \tau^1 g f \tau^2 f = (f,3)(g,4)(f,4)(f,6)$

- $\tau^0 \approx \varepsilon$: an hidden signal with zero duration is not observable.
  $a^0 \not\approx \varepsilon$: a signal, even of zero duration, is observable.
  $\tau^2 \not\approx \varepsilon$: we still observe a time delay but the actual signal has been hidden.
  Example : $a^2\tau^0 a^1 f \tau^0 g \tau^1 f b^2 b^2 b^2 \cdots \approx a^3 f g \tau^1 f b^\infty$
- Signal-event words $SE(\Sigma) = \Sigma^\infty / \approx$

Signal-Event (Timed) Words

Unobservable signal τ

- Useful to hide signals:

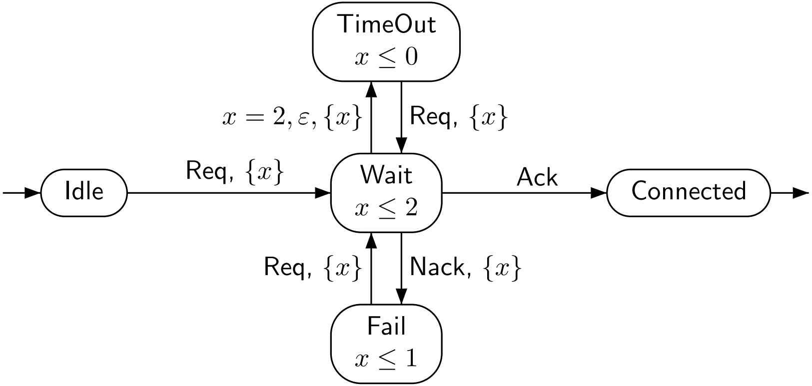
  Signal-event word $\xrightarrow{\text{hiding signals}}$ Classical timed words

  $a^3 f b^1 g f a^2 f$ $\qquad$ $\tau^3 f \tau^1 g f \tau^2 f = (f, 3)(g, 4)(f, 4)(f, 6)$

- $\tau^0 \approx \varepsilon$: an hidden signal with zero duration is not observable.
  $a^0 \not\approx \varepsilon$: a signal, even of zero duration, is observable.
  $\tau^2 \not\approx \varepsilon$: we still observe a time delay but the actual signal has been hidden.
  Example : $a^2 \tau^0 a^1 f \tau^0 g \tau^1 f b^2 b^2 b^2 \cdots \approx a^3 f g \tau^1 f b^\infty$
- Signal-event words $SE(\Sigma) = \Sigma^\infty / \approx$

Signal-Event (Timed) automata

- States emit signals
- Transitions emit (instantaneous) events



- Run : Idle^3 · Req · Wait^2 · TimeOut^0 · Req · Wait^1 · Ack · Connected^8
- SEL : languages accepted by SE-automata without ε-transitions.
- SEL_ε : languages accepted by SE-automata with ε-transitions.

Outline

Signal-Event (Timed) Substitutions

Definition

- Abstract alphabet : Σ_e and Σ_s
- Concrete alphabet : Σ'_e and Σ'_s
- Substitution σ from $SE(\Sigma)$ to $SE(\Sigma')$ defined by:

$$a \in \Sigma_e : \; L_a \subseteq (\Sigma'_e \cup \Sigma'_s \times \{0\})^*$$
$$\sigma(a) = L_a$$
$$a \in \Sigma_s \setminus \{\tau\} : \; L_a \subseteq SE(\Sigma') \text{ not containing Zeno words.}$$
$$\sigma(a^d) = \{w \in L_a \mid \|w\| = d\}$$
$$a = \tau : \; L_\tau = \{\tau\} \times \overline{\mathbb{T}}$$
$$\sigma(\tau^d) = \{\tau^d\}$$

Remark

If we allow Zeno words in L_a then we may get transfinite words as refinements.
Example: if $b^1 f b^{1/2} f b^{1/4} f \cdots \in L_a$ and $L_g = \{g\}$ then $\sigma(a^2 g)$ is transfinite.

Signal-Event (Timed) Substitutions

Definition

- Abstract alphabet : Σ_e and Σ_s
- Concrete alphabet : Σ'_e and Σ'_s
- Substitution σ from $SE(\Sigma)$ to $SE(\Sigma')$ defined by:

$$a \in \Sigma_e : \; L_a \subseteq (\Sigma'_e \cup \Sigma'_s \times \{0\})^*$$

$$\sigma(a) = L_a$$

$a \in \Sigma_s \setminus \{\tau\} : \; L_a \subseteq SE(\Sigma')$ not containing Zeno words.

$$\sigma(a^d) = \{w \in L_a \mid \|w\| = d\}$$

$$a = \tau : \; L_\tau = \{\tau\} \times \overline{\mathbb{T}}$$

$$\sigma(\tau^d) = \{\tau^d\}$$

Remark

If we allow Zeno words in L_a then we may get transfinite words as refinements.
Example: if $b^1 f b^{1/2} f b^{1/4} f \cdots \in L_a$ and $L_g = \{g\}$ then $\sigma(a^2 g)$ is transfinite.

Signal-Event (Timed) Substitutions

Remark

In general, SE-substitutions are not morphisms

Example: if $L_a = \{b^2\}$ then $\sigma(a^1) = \emptyset$ and $\sigma(a^2) \neq \sigma(a^1)\sigma(a^1)$

Substitutions are applied to SE-words in normal form:

$\sigma(a^2\tau^0 a^1 f\tau^0 g\tau^1 f b^2 b^2 b^2 \cdots) = \sigma(a^3)\sigma(f)\sigma(g)\tau^1\sigma(f)\sigma(b^\infty)$

Proposition

Let σ be a timed substitution, given by a family $(L_a)_{a\in\Sigma_e\cup\Sigma_s}$.
Then, σ is a morphism if and only if for each signal $a \in \Sigma_s$ we have

1. L_a is closed under concatenation:
   for all $u, v \in L_a$ with $\|u\| < \infty$, we have $uv \in L_a$,
2. L_a is closed under decomposition:
   for each $v \in L_a$ with $\|v\| = d$, for all $d_1 \in \mathbb{T}$, $d_2 \in \overline{\mathbb{T}}$ such that $d = d_1 + d_2$, there exist $v_i \in L_a$ with $\|v_i\| = d_i$ such that $v = v_1 v_2$.

Signal-Event (Timed) Substitutions

Remark

In general, SE-substitutions are not morphisms

Example: if $L_a = \{b^2\}$ then $\sigma(a^1) = \emptyset$ and $\sigma(a^2) \neq \sigma(a^1)\sigma(a^1)$

Substitutions are applied to SE-words in normal form:

$\sigma(a^2\tau^0 a^1 f \tau^0 g \tau^1 f b^2 b^2 b^2 \cdots) = \sigma(a^3)\sigma(f)\sigma(g)\tau^1\sigma(f)\sigma(b^\infty)$

Proposition

Let σ be a timed substitution, given by a family $(L_a)_{a \in \Sigma_e \cup \Sigma_s}$.
Then, σ is a morphism if and only if for each signal $a \in \Sigma_s$ we have

1. L_a is closed under concatenation:
   for all $u, v \in L_a$ with $\|u\| < \infty$, we have $uv \in L_a$,
2. L_a is closed under decomposition:
   for each $v \in L_a$ with $\|v\| = d$, for all $d_1 \in \mathbb{T}$, $d_2 \in \overline{\mathbb{T}}$ such that $d = d_1 + d_2$, there exist $v_i \in L_a$ with $\|v_i\| = d_i$ such that $v = v_1 v_2$.

Recognizable substitutions

Definition

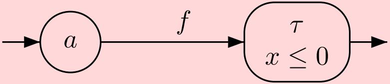
Let σ be a substitution defined by $(L_a)_{a \in \Sigma_e \cup \Sigma_s}$. Then,

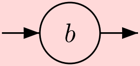
- σ is a $SEL - substitution$ if each L_a is in SEL
- σ is a $SEL_\varepsilon - substitution$ if each L_a is in SEL_ε

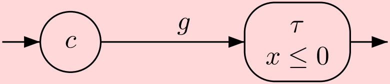
Outline

Closure under substitutions

SEL is not closed under $SEL - substitution$

- $L = \{a^0 f\}$ is recognized by 

- $L_a = \{b\} \times \overline{\mathbb{T}}$ is recognized by 

- $L_f = \{c^0 g\}$ is recognized by 

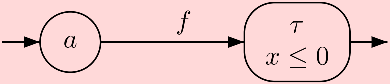
- $\sigma(L) = \{b^0 c^0 g\}$ cannot be accepted without ε-transitions.

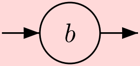
Theorem

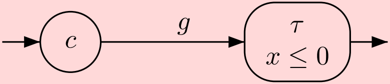
1. The class SEL_ε is closed under SEL_ε-substitutions.
2. The class SEL is closed under SEL-substitution satisfying $L_f \subseteq \Sigma'_e((\Sigma'_s \times \{0\})\Sigma'_e)^*$ for each $f \in \Sigma_e$, i.e., each word in L_f must start and end with an instantaneous event.

Closure under substitutions

SEL is not closed under $SEL - substitution$

- $L = \{a^0 f\}$ is recognized by 

- $L_a = \{b\} \times \overline{\mathbb{T}}$ is recognized by 

- $L_f = \{c^0 g\}$ is recognized by 

- $\sigma(L) = \{b^0 c^0 g\}$ cannot be accepted without ε-transitions.

Theorem

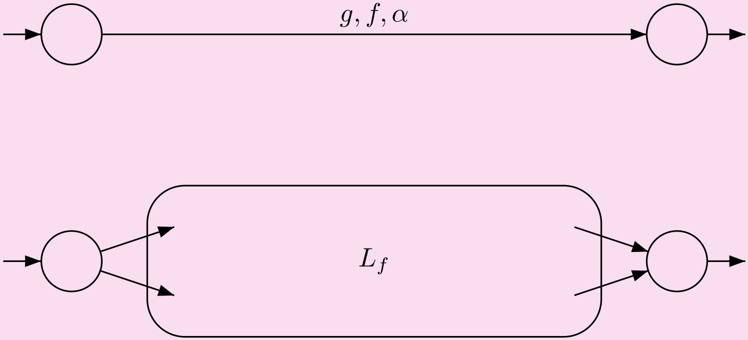
1. The class SEL_ε is closed under SEL_ε-substitutions.
2. The class SEL is closed under SEL-substitution satisfying $L_f \subseteq \Sigma'_e((\Sigma'_s \times \{0\})\Sigma'_e)^*$ for each $f \in \Sigma_e$,
   i.e., each word in L_f must start and end with an instantaneous event.

Closure under substitutions

Theorem

1. The class SEL_ε is closed under SEL_ε-substitutions.
2. The class SEL is closed under SEL-substitution satisfying $L_f \subseteq \Sigma'_e((\Sigma'_s \times \{0\})\Sigma'_e)^*$ for each $f \in \Sigma_e$.

Handling events is easy for SEL_ε-substitutions.

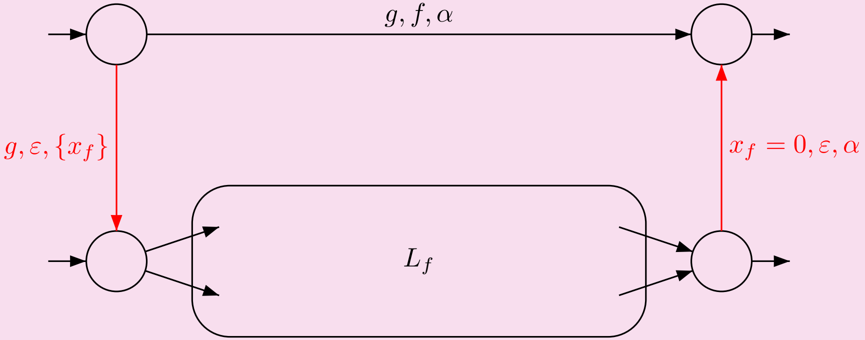


Closure under substitutions

Theorem

1. The class SEL_ε is closed under SEL_ε-substitutions.
2. The class SEL is closed under SEL-substitution satisfying $L_f \subseteq \Sigma'_e((\Sigma'_s \times \{0\})\Sigma'_e)^*$ for each $f \in \Sigma_e$.

Handling events is easy for SEL_ε-substitutions.

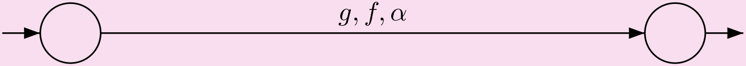


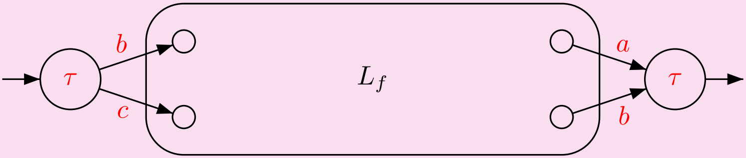
Closure under substitutions

Theorem

1. The class SEL_ε is closed under SEL_ε-substitutions.
2. The class SEL is closed under SEL-substitution satisfying $L_f \subseteq \Sigma'_e((\Sigma'_s \times \{0\})\Sigma'_e)^*$ for each $f \in \Sigma_e$.

Handling events is easy for SEL-substitutions.



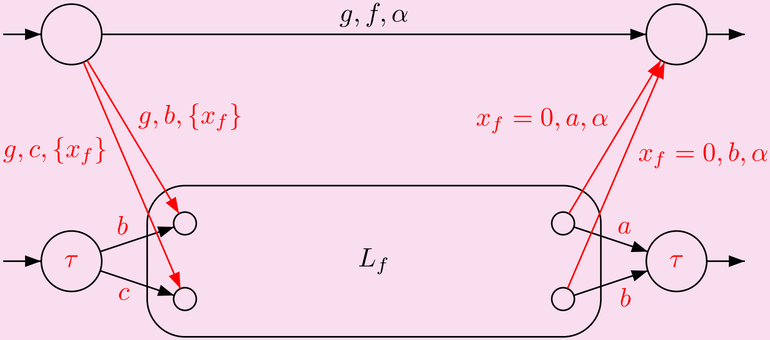


Closure under substitutions

Theorem

1. The class SEL_ε is closed under SEL_ε-substitutions.
2. The class SEL is closed under SEL-substitution satisfying $L_f \subseteq \Sigma_e'((\Sigma_s' \times \{0\})\Sigma_e')^*$ for each $f \in \Sigma_e$.

Handling events is easy for SEL-substitutions.

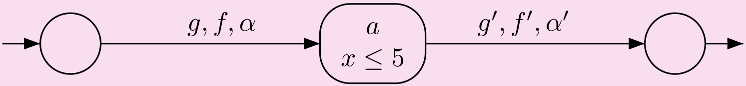


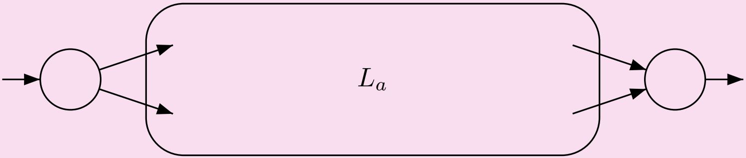
Closure under substitutions

Theorem

1. The class SEL_ε is closed under SEL_ε-substitutions.
2. The class SEL is closed under SEL-substitution satisfying $L_f \subseteq \Sigma'_e((\Sigma'_s \times \{0\})\Sigma'_e)^*$ for each $f \in \Sigma_e$.

Handling signals is easy for SEL-substitutions.



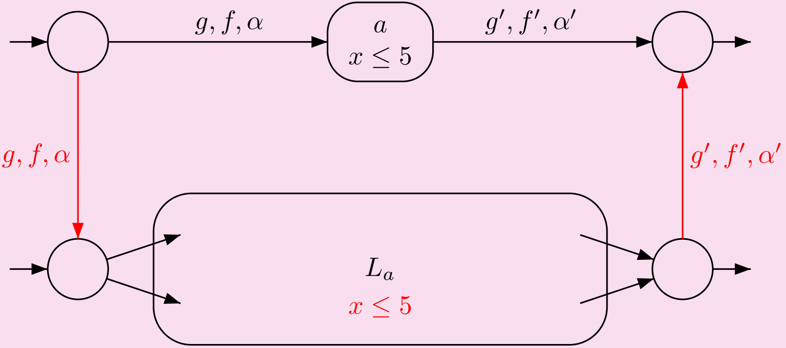


Closure under substitutions

Theorem

1. The class SEL_ε is closed under SEL_ε-substitutions.
2. The class SEL is closed under SEL-substitution satisfying $L_f \subseteq \Sigma'_e((\Sigma'_s \times \{0\})\Sigma'_e)^*$ for each $f \in \Sigma_e$.

Handling signals is easy for SEL-substitutions.

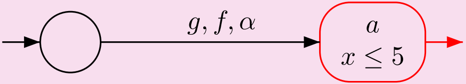


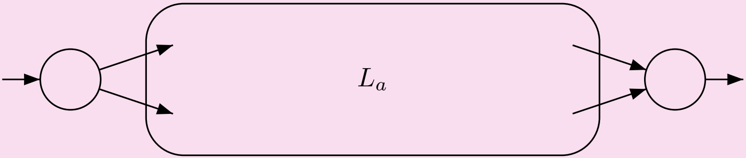
Closure under substitutions

Theorem

1. The class SEL_ε is closed under SEL_ε-substitutions.
2. The class SEL is closed under SEL-substitution satisfying $L_f \subseteq \Sigma'_e((\Sigma'_s \times \{0\})\Sigma'_e)^*$ for each $f \in \Sigma_e$.

Handling signals is easy for SEL-substitutions.



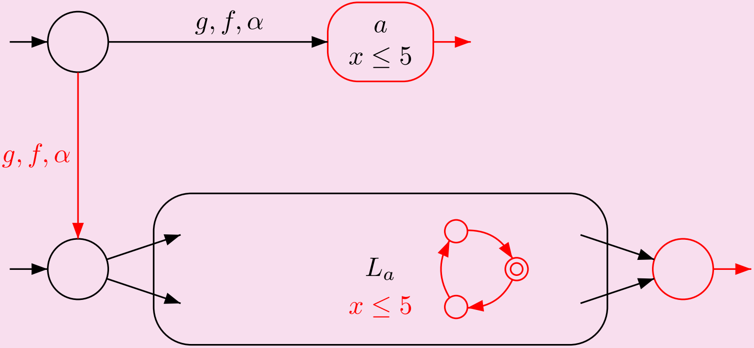


Closure under substitutions

Theorem

1. The class SEL_ε is closed under SEL_ε-substitutions.
2. The class SEL is closed under SEL-substitution satisfying $L_f \subseteq \Sigma'_e((\Sigma'_s \times \{0\})\Sigma'_e)^*$ for each $f \in \Sigma_e$.

Handling signals is easy for SEL-substitutions.



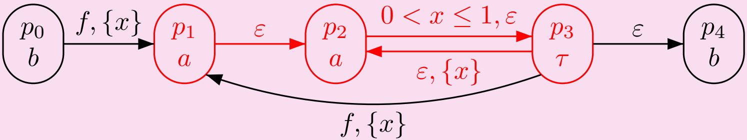
Closure under substitutions

Theorem

1. The class SEL_ε is closed under SEL_ε-substitutions.
2. The class SEL is closed under SEL-substitution satisfying $L_f \subseteq \Sigma'_e((\Sigma'_s \times \{0\})\Sigma'_e)^*$ for each $f \in \Sigma_e$.

Handling signals for SEL_ε-substitutions is harder.

Remember that substitutions are applied to SE-words in normal form.



A possible run gives : $f a^{0.3} a^{0.6} \tau^0 a^{0.5} \tau^1 a^{0.6} \tau^0 a^{0.5} \tau^0 b^3 \quad \approx \quad f a^{1.4} \tau^1 a^{1.1} b^3$

We have to synchronize a-blocks of $\mathcal{A}$ with the automaton $\mathcal{A}_a$ taking into account τ-states that may be crossed instantaneously.

Closure under substitutions

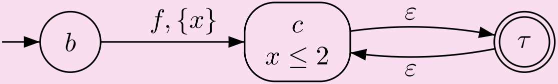
Theorem

1. The class SEL_ε is closed under SEL_ε-substitutions.
2. The class SEL is closed under SEL-substitution satisfying $L_f \subseteq \Sigma'_e((\Sigma'_s \times \{0\})\Sigma'_e)^*$ for each $f \in \Sigma_e$.

Handling signals for SEL_ε-substitutions is harder.

Zeno runs in $\mathcal{A}_a$ must be removed.

- L_a accepted by



- $b^1 f c^2$ is accepted by an infinite Zeno run: $b^1 f c^1 \tau^0 c^{1/2} \tau^0 c^{1/4} \dots$
- We get $g b^1 f c^2 g \in \sigma(g a^3 g)$.

We have to replace the Zeno run of $\mathcal{A}_a$ by a finite run.

Normal form for SE-automata

Theorem

Let $\mathcal{A}$ be a SE-automaton. We can effectively construct an equivalent SE-automaton $\mathcal{A}'$ such that:

1. no infinite run of $\mathcal{A}'$ accepts a finite word with finite duration, and
2. no finite run of $\mathcal{A}'$ accepts a word with infinite duration.

Main problem

We have to replace infinite accepting ε-loops

by finite accepting runs.

Normal form for SE-automata

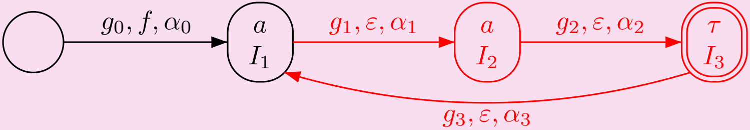
Theorem

Let $\mathcal{A}$ be a SE-automaton. We can effectively construct an equivalent SE-automaton $\mathcal{A}'$ such that:

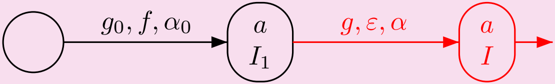
1. no infinite run of $\mathcal{A}'$ accepts a finite word with finite duration, and
2. no finite run of $\mathcal{A}'$ accepts a word with infinite duration.

Main problem

We have to replace infinite accepting ε-loops



by finite accepting runs.



Outline

Closure under inverse substitutions

Theorem

1. The class SEL_ε is closed under inverse SEL_ε-substitution.
2. The class SEL is closed under inverse SEL-substitution acting only on events: $L_a = \{a\} \times \overline{\mathbb{T}}$ for all $a \in \Sigma_s$.

The class SEL is not closed under arbitrary inverse SEL-substitution

- Let $\Sigma_s = \Sigma_s' = \{a, b\}$ and $\Sigma_e = \Sigma_e' = \{f\}$.
- Let σ be the SEL-substitution defined by $L_a = \{a^1 f\}$, $L_b = \{b^0\}$ and $L_f = \{f\}$.
- $L = \{a^1 f b^0\}$ is a SEL.
- $\sigma^{-1}(L) = \{a^1 b^0\}$ is not a SEL.

Closure under inverse substitutions

Theorem

1. The class SEL_ε is closed under inverse SEL_ε-substitution.
2. The class SEL is closed under inverse SEL-substitution acting only on events: $L_a = \{a\} \times \overline{\mathbb{T}}$ for all $a \in \Sigma_s$.

The class SEL is not closed under arbitrary inverse SEL-substitution

- Let $\Sigma_s = \Sigma'_s = \{a, b\}$ and $\Sigma_e = \Sigma'_e = \{f\}$.
- Let σ be the SEL-substitution defined by $L_a = \{a^1 f\}$, $L_b = \{b^0\}$ and $L_f = \{f\}$.
- $L = \{a^1 f b^0\}$ is a SEL.
- $\sigma^{-1}(L) = \{a^1 b^0\}$ is not a SEL.

Closure under inverse substitutions

Theorem

1. The class SEL_ε is closed under inverse SEL_ε-substitution.
2. The class SEL is closed under inverse SEL-substitution acting only on events: $L_a = \{a\} \times \overline{\mathbb{T}}$ for all $a \in \Sigma_s$.

Proof: word case

- Let $\sigma : A \to \mathcal{P}(B^*)$ be a rational substitution
- Let $\Pi_A : (A \uplus B)^* \to A^*$ and $\Pi_B : (A \uplus B)^* \to B^*$ be the projections
- Let $M = \left(\bigcup_{a \in A} a\sigma(a) \right)^* \subseteq (A \uplus B)^*$ is rational.
- Then, $\sigma^{-1}(L) = \Pi_A(\Pi_B^{-1}(L) \cap M)$

Closure under inverse substitutions

Theorem

1. The class SEL_ε is closed under inverse SEL_ε-substitution.
2. The class SEL is closed under inverse SEL-substitution acting only on events: $L_a = \{a\} \times \overline{\mathbb{T}}$ for all $a \in \Sigma_s$.

Proof: Signal-event words

- Let $\hat{\Sigma}_e = \Sigma_e \uplus \Sigma'_e$ and $\hat{\Sigma}_s = \Sigma_s \times \Sigma'_s$.
- Let $\Pi_1 : SE(\hat{\Sigma}) \to SE(\Sigma)$ and $\Pi_2 : SE(\hat{\Sigma}) \to SE(\Sigma')$ be the natural projections
- $\sigma^{-1}(L) = \Pi_1(\Pi_2^{-1}(L) \cap M)$ for a suitable language M, in the class SEL_ε.

  if $f \in \Sigma_e$ then $f \cdot \sigma(f)$ is replaced by

$$f \cdot \{w \in SE(\hat{\Sigma}) \mid w = (\tau, b_0)^0 f_1 (\tau, b_1)^0 f_2 \cdots (\tau, b_n)^0 \text{ with } b_0^0 f_1 b_1^0 f_2 \cdots b_n^0 \in \sigma(f)\}$$

  if $a \in \Sigma_s$ then $a \cdot \sigma(a)$ is replaced by

$$\{w \in SE(\hat{\Sigma}) \mid w = (a, b_0)^{d_0} f_1 (a, b_1)^{d_1} f_2 \cdots \text{ with } b_0^{d_0} f_1 b_1^{d_1} f_2 \cdots \in \sigma(a^{d_0+d_1+\cdots})\}$$

- If L is in the class SEL_ε, then so is $\Pi_2^{-1}(L)$.
- The class SEL_ε is closed under intersection.

Closure under intersection

Remark

- Easy for the class SEL or for classical timed languages.
- More difficult with signals and ε-transitions due to stuttering and unobservability of τ^0.
- Asarin, Caspi and Maler 97 did not deal with these difficulties and considered finite runs only.
  Asarin, Caspi and Maler 02 deals with the intersection of classical timed automata only.
- Dima 00 gives a construction to remove stuttering for automata with a single clock.
- Durand-Lose 04 gives a construction for intersection taking stuttering into account but restricted to finite runs and without zero-duration signals.
  His approach do not extend to infinite runs since it would introduce Zeno runs leading to transfinite problems

Outline

Conclusion

- Signal-event words are the natural objects for studying refinements, abstractions and other problems.
- Extending classical results to SE-automata is not always easy due to ε-transitions, signal stuttering, unobservability of τ^0, Zeno runs, ...
- We have proved closure properties (intersection, refinement, abstraction) for the general case of SE-automata.