Handling Infinitely Branching WSTS

Michael Blondin [1 2], Alain Finkel[1] & Pierre McKenzie [1 2]

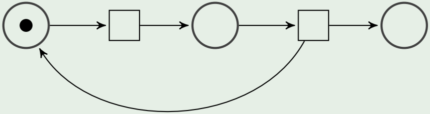
[1]LSV, ENS Cachan

[2]DIRO, Université de Montréal

PV 2015, Madrid, September 4, 2015

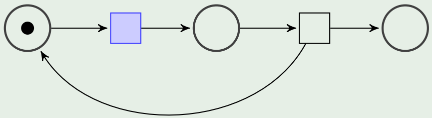
Well-structured transition systems (WSTS) encompass a large number of infinite state systems.

Example of WSTS: Petri nets (Geeraerts, Heußner, Praveen & Raskin PN'13)



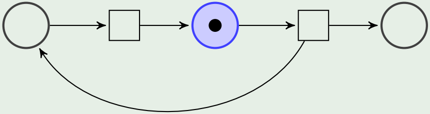
Well-structured transition systems (WSTS) encompass a large number of infinite state systems.

Example of WSTS: Petri nets (Geeraerts, Heußner, Praveen & Raskin PN'13)



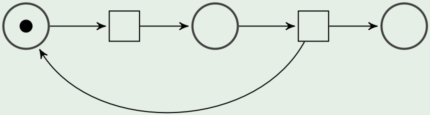
Well-structured transition systems (WSTS) encompass a large number of infinite state systems.

Example of WSTS: Petri nets (Geeraerts, Heußner, Praveen & Raskin PN'13)



Multiple decidability results are known for finitely branching WSTS.

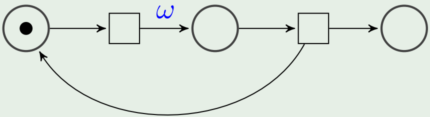
Example of WSTS: Petri nets (Geeraerts, Heußner, Praveen & Raskin PN'13)



$$\mathrm{Post}(\odot \; \bigcirc \; \bigcirc) \;=\; \bigcirc \; \odot \; \bigcirc$$

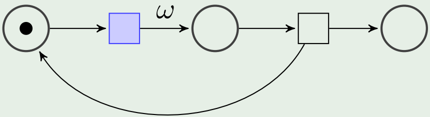
How to handle infinitely branching WSTS such as systems with infinitely many initial states, and parametric systems?

Example of WSTS: ω–Petri nets (Geeraerts, Heußner, Praveen & Raskin PN'13)



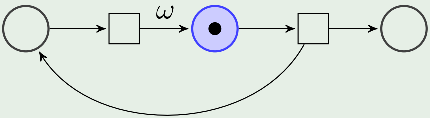
How to handle infinitely branching WSTS such as systems with infinitely many initial states, and parametric systems?

Example of WSTS: ω–Petri nets (Geeraerts, Heußner, Praveen & Raskin PN'13)



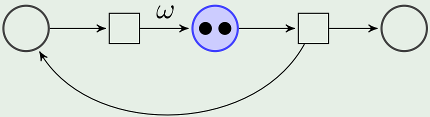
How to handle infinitely branching WSTS such as systems with infinitely many initial states, and parametric systems?

Example of WSTS: ω–Petri nets (Geeraerts, Heußner, Praveen & Raskin PN'13)



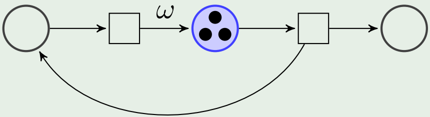
How to handle infinitely branching WSTS such as systems with infinitely many initial states, and parametric systems?

Example of WSTS: ω–Petri nets (Geeraerts, Heußner, Praveen & Raskin PN'13)



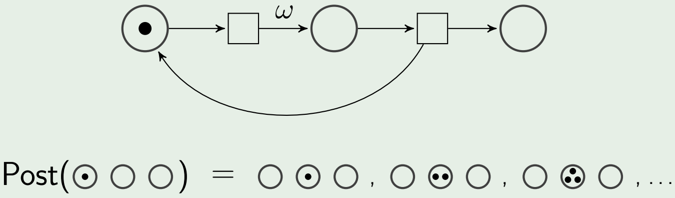
How to handle infinitely branching WSTS such as systems with infinitely many initial states, and parametric systems?

Example of WSTS: ω–Petri nets (Geeraerts, Heußner, Praveen & Raskin PN'13)



How to handle infinitely branching WSTS such as systems with infinitely many initial states, and parametric systems?

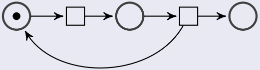
Example of WSTS: ω–Petri nets (Geeraerts, Heußner, Praveen & Raskin PN'13)



Well-structured transition system (Finkel ICALP'87, Finkel & Schnoebelen TCS'01)

$S = (X, \rightarrow, \leq)$ where

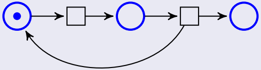
- X set,
- $\rightarrow \subseteq X \times X$,
- monotony,
- well-quasi-ordered.



Well-structured transition system (Finkel ICALP'87, Finkel & Schnoebelen TCS'01)

$S = (X, \rightarrow, \leq)$ where

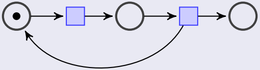
- $\mathbb{N}^3$,
- $\rightarrow \subseteq X \times X$,
- monotony,
- well-quasi-ordered.



Well-structured transition system (Finkel ICALP'87, Finkel & Schnoebelen TCS'01)

$S = (X, \rightarrow, \leq)$ where

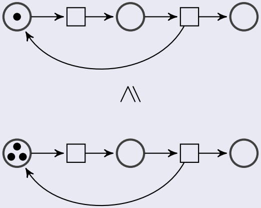
- X set,
- $\rightarrow \subseteq \mathbb{N}^3 \times \mathbb{N}^3$,
- monotony,
- well-quasi-ordered.



Well-structured transition system (Finkel ICALP'87, Finkel & Schnoebelen TCS'01)

$S = (X, \to, \leq)$ where

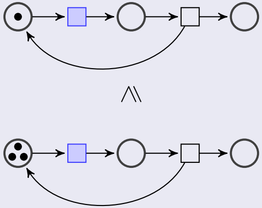
- X set,
- $\to \subseteq X \times X$,
- monotony,
- well-quasi-ordered.



Well-structured transition system (Finkel ICALP'87, Finkel & Schnoebelen TCS'01)

$S = (X, \rightarrow, \leq)$ where

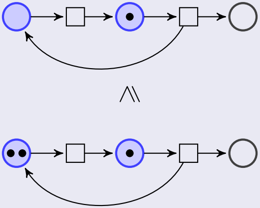
- X set,
- $\rightarrow \subseteq X \times X$,
- monotony,
- well-quasi-ordered.



Well-structured transition system (Finkel ICALP'87, Finkel & Schnoebelen TCS'01)

$S = (X, \to, \leq)$ where

- X set,
- $\to \subseteq X \times X$,
- monotony,
- well-quasi-ordered.



Well-structured transition system (Finkel ICALP'87, Finkel & Schnoebelen TCS'01)

$S = (X, \to, \leq)$ where

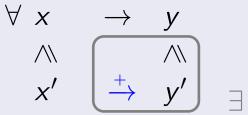
- X set,
- $\to \subseteq X \times X$,
- monotony,
- well-quasi-ordered.

$$
\begin{array}{ccccc}
\forall & x & \to & y & \\
 & \wedge\!\!\backslash & & \wedge\!\!\backslash & \\
 & x' & \xrightarrow{*} & y' & \exists
\end{array}
$$

Well-structured transition system (Finkel ICALP'87, Finkel & Schnoebelen TCS'01)

$S = (X, \rightarrow, \leq)$ where

- X set,
- $\rightarrow \subseteq X \times X$,
- transitive monotony,
- well-quasi-ordered.



Well-structured transition system (Finkel ICALP'87, Finkel & Schnoebelen TCS'01)

$S = (X, \rightarrow, \leq)$ where

- X set,
- $\rightarrow \subseteq X \times X$,
- strong monotony,
- well-quasi-ordered.

$$
\begin{array}{cccc}
\forall\ x & \rightarrow & y & \\
\leq\!\!| & & \leq\!\!| & \\
x' & \rightarrow & y' & \exists
\end{array}
$$

Well-structured transition system (Finkel ICALP'87, Finkel & Schnoebelen TCS'01)

$S = (X, \to, \leq)$ where

- X set,
- $\to \subseteq X \times X$,
- monotony,
- well-quasi-ordered:
  $\forall x_0, x_1, \ldots \ \exists i < j$ s.t. $x_i \leq x_j$.

Branching

A WSTS $(X, \to, \leq)$ is *finitely branching* if $\text{Post}(x)$ is finite for every $x \in X$.

Branching

A WSTS $(X, \rightarrow, \leq)$ is *finitely branching* if $\text{Post}(x)$ is finite for every $x \in X$.

Some finitely branching WSTS

- Petri nets, vector addition systems,

Branching

A WSTS $(X, \to, \leq)$ is *finitely branching* if $\mathrm{Post}(x)$ is finite for every $x \in X$.

Some finitely branching WSTS

- Petri nets, vector addition systems,
- Counter machines with affine updates,

Branching

A WSTS $(X, \to, \leq)$ is *finitely branching* if $\text{Post}(x)$ is finite for every $x \in X$.

Some finitely branching WSTS

- Petri nets, vector addition systems,
- Counter machines with affine updates,
- Lossy channel systems (Abdulla, Cerans, Jonsson & Tsay LICS'96),

Branching

A WSTS $(X, \rightarrow, \leq)$ is *finitely branching* if $\mathrm{Post}(x)$ is finite for every $x \in X$.

Some finitely branching WSTS

- Petri nets, vector addition systems,
- Counter machines with affine updates,
- Lossy channel systems (Abdulla, Cerans, Jonsson & Tsay LICS'96),
- Much more.

Branching

A WSTS $(X, \rightarrow, \leq)$ is *finitely branching* if $\text{Post}(x)$ is finite for every $x \in X$.

Some infinitely branching WSTS

- Inserting FIFO automata (Cécé, Finkel, Iyer IC'96),

Branching

A WSTS $(X, \rightarrow, \leq)$ is *finitely branching* if $\mathrm{Post}(x)$ is finite for every $x \in X$.

Some infinitely branching WSTS

- Inserting FIFO automata (Cécé, Finkel, Iyer IC'96),
- Inserting automata (Bouyer, Markey, Ouaknine, Schnoebelen, Worrell FAC'12),

Branching

A WSTS $(X, \to, \leq)$ is *finitely branching* if $\text{Post}(x)$ is finite for every $x \in X$.

Some infinitely branching WSTS

- Inserting FIFO automata (Cécé, Finkel, Iyer IC'96),
- Inserting automata (Bouyer, Markey, Ouaknine, Schnoebelen, Worrell FAC'12),
- ω-Petri nets (Geeraerts, Heussner, Praveen & Raskin PN'13),

Branching

A WSTS $(X, \rightarrow, \leq)$ is *finitely branching* if $\text{Post}(x)$ is finite for every $x \in X$.

Some infinitely branching WSTS

- Inserting FIFO automata (Cécé, Finkel, Iyer IC'96),
- Inserting automata (Bouyer, Markey, Ouaknine, Schnoebelen, Worrell FAC'12),
- ω-Petri nets (Geeraerts, Heussner, Praveen & Raskin PN'13),
- Parametric WSTS.

Proposition

Finite branching is undecidable for post-effective WSTS.

Proposition

Finite branching is undecidable for post-effective WSTS.

Proof

- Let $S_i = (\mathbb{N}, \rightarrow_{S_i}, \leq)$ be the WSTS such that:
  - $x \rightarrow_{S_i} x + 1$ if TM_i does not halt within $\leq x$ steps,
  - $x \rightarrow_{S_i} 0, 1, 2, \ldots$ otherwise.

Proposition

Finite branching is undecidable for post-effective WSTS.

Proof

- Let $S_i = (\mathbb{N}, \rightarrow_{S_i}, \leq)$ be the WSTS such that:
  - $x \rightarrow_{S_i} x + 1$ if TM_i does not halt within $\leq x$ steps,
  - $x \rightarrow_{S_i} 0, 1, 2, \ldots$ otherwise.
- S_i is post-effective (the cardinal of $\mathrm{Post}_{S_i}(x)$ is computable).
- S_i has strong and strict monotony since $x \rightarrow_{S_i} x + 1$ for every $x \in \mathbb{N}$.

Proposition

Finite branching is undecidable for post-effective WSTS.

Proof

- Let $S_i = (\mathbb{N}, \rightarrow_{S_i}, \leq)$ be the WSTS such that:
  - $x \rightarrow_{S_i} x + 1$ if TM_i does not halt within $\leq x$ steps,
  - $x \rightarrow_{S_i} 0, 1, 2, \ldots$ otherwise.
- S_i is post-effective (the cardinal of $\text{Post}_{S_i}(x)$ is computable).
- S_i has strong and strict monotony since $x \rightarrow_{S_i} x + 1$ for every $x \in \mathbb{N}$.
- TM_i halts iff there exist $x \in \mathbb{N}$ and an execution $0 \xrightarrow{*}_{S_i} x$ such that $\text{Post}_{S_i}(x)$ is infinite.
- The halting problem thus Turing-reduces to the infinite branching problem.

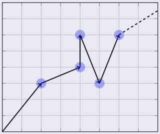
Objective

We want to study the usual reachability problems for these infinitely branching systems, e.g.,

Objective

We want to study the usual reachability problems for these infinitely branching systems, e.g.,

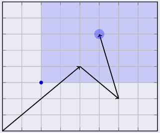
- Termination,



Objective

We want to study the usual reachability problems for these infinitely branching systems, e.g.,

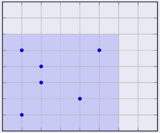
- Termination,
- Coverability,



Objective

We want to study the usual reachability problems for these infinitely branching systems, e.g.,

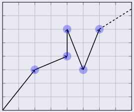
- Termination,
- Coverability,
- Boundedness.



Termination

Input: $(X, \to, \leq)$ a WSTS, $x_0 \in X$.

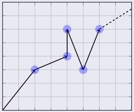
Question: $\exists x_0 \to x_1 \to x_2 \to \ldots$?



Termination

Input: $(X, \to, \leq)$ a WSTS, $x_0 \in X$.

Question: $\exists x_0 \to x_1 \to x_2 \to \ldots ?$



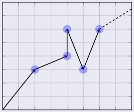
Theorem (Finkel ICALP'87)

Termination is decidable for finitely branching WSTS with transitive monotony.

Termination

Input: $(X, \to, \leq)$ a WSTS, $x_0 \in X$.

Question: $\exists x_0 \to x_1 \to x_2 \to \ldots ?$



Theorem (deduced from Dufourd, Jančar & Schnoebelen ICALP'99)

Termination is undecidable for infinitely branching WSTS.

Strong termination

Input: $(X, \to, \leq)$ a WSTS, $x_0 \in X$.

Question: $\exists k$ bounding length of executions from x_0?

Strong termination

Input: $(X, \to, \leq)$ a WSTS, $x_0 \in X$.

Question: $\exists k$ bounding length of executions from x_0?

Remark

Strong termination and termination are the same in finitely branching WSTS.

Strong termination

Input: $(X, \rightarrow, \leq)$ a WSTS, $x_0 \in X$.

Question: $\exists k$ bounding length of executions from x_0?

Theorem

Strong termination is decidable for infinitely branching WSTS under some assumptions.

Issues with finite branching techniques

Some techniques for WSTS based on finite reachability trees; impossible for infinite branching.

Some rely on upward closed sets; what about downward closed, in particular with infinite branching?

Issues with finite branching techniques

Some techniques for WSTS based on finite reachability trees; impossible for infinite branching.

Some rely on upward closed sets; what about downward closed, in particular with infinite branching?

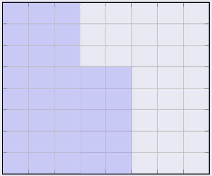
A tool

Develop from the WSTS *completion* introduced by Finkel & Goubault-Larrecq in STACS'09 and ICALP'09.

Ideals

$I \subseteq X$ is an *ideal* if

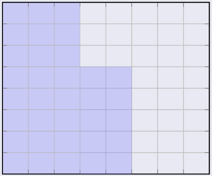
- downward closed: $I = \downarrow I$,



Ideals

$I \subseteq X$ is an *ideal* if

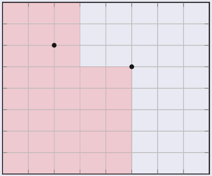
- downward closed: $I = \downarrow I$,
- directed: $a, b \in I \implies \exists c \in I$ s.t. $a \leq c$ and $b \leq c$.



Ideals

$I \subseteq X$ is an *ideal* if

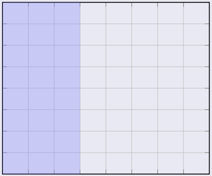
- downward closed: $I = \downarrow I$,
- directed: $a, b \in I \implies \exists c \in I$ s.t. $a \leq c$ and $b \leq c$.



Ideals

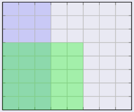
$I \subseteq X$ is an *ideal* if

- downward closed: $I = \downarrow I$,
- directed: $a, b \in I \implies \exists c \in I$ s.t. $a \leq c$ and $b \leq c$.



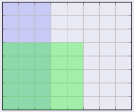
Theorem (Finkel & Goubault-Larrecq ICALP'09; Goubault-Larrecq '14)

$$D \text{ downward closed } \implies D = \bigcup_{\text{finite}} \text{Ideals}$$



Theorem (Finkel & Goubault-Larrecq ICALP'09; Goubault-Larrecq '14)

$$D \text{ downward closed} \implies D = \bigcup_{\text{finite}} \text{Ideals}$$



Corollary

Every downward closed set decomposes canonically as the union of its maximal ideals.

Completion

The *completion* of $S = (X, \rightarrow_S, \leq)$ is $\widehat{S} = (\widehat{X}, \rightarrow_{\widehat{S}}, \subseteq)$ such that

Completion

The *completion* of $S = (X, \rightarrow_S, \leq)$ is $\widehat{S} = (\widehat{X}, \rightarrow_{\widehat{S}}, \subseteq)$ such that

- $\widehat{X} = \mathrm{Ideals}(X)$,

Completion

The *completion* of $S = (X, \rightarrow_S, \leq)$ is $\widehat{S} = (\widehat{X}, \rightarrow_{\widehat{S}}, \subseteq)$ such that

- $\widehat{X} = \mathrm{Ideals}(X)$,
- $I \rightarrow_{\widehat{S}} J$ if $\downarrow \mathrm{Post}(I) = \underbrace{\ldots \cup\ J\ \cup \ldots}_{\text{canonical decomposition}}$

Theorem

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then $\widehat{S} = (\widehat{X}, \rightarrow_{\widehat{S}}, \subseteq)$ is such that:

- $\widehat{S}$ is finitely branching,

Theorem

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then $\widehat{S} = (\widehat{X}, \rightarrow_{\widehat{S}}, \subseteq)$ is such that:

- $\widehat{S}$ is finitely branching,
- $\widehat{S}$ has (strong) monotony,

Theorem

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then $\widehat{S} = (\widehat{X}, \rightarrow_{\widehat{S}}, \subseteq)$ is such that:

- $\widehat{S}$ is finitely branching,
- $\widehat{S}$ has (strong) monotony,
- $\widehat{S}$ is *not always* a WSTS (Jančar IPL'99).

Theorem

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then $\widehat{S} = (\widehat{X}, \rightarrow_{\widehat{S}}, \subseteq)$ is such that:

- $\widehat{S}$ is finitely branching,
- $\widehat{S}$ has (strong) monotony,
- $\widehat{S}$ is *not always* a WSTS (Jančar IPL'99).

Jančar IPL'99

A wqo $\leq$ is a ω^2-*wqo* iff $\leq^\#$ is a wqo, where $\leq^\#$ is the Hoare ordering defined by $A \leq^\# B$ iff $\uparrow B \subseteq \uparrow A$.

Theorem

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then $\widehat{S} = (\widehat{X}, \rightarrow_{\widehat{S}}, \subseteq)$ is such that:

- $\widehat{S}$ is finitely branching,
- $\widehat{S}$ has (strong) monotony,
- $\widehat{S}$ is *not always* a WSTS (Jančar IPL'99).

Jančar IPL'99

A wqo $\leq$ is a ω^2-*wqo* iff $\leq^\#$ is a wqo, where $\leq^\#$ is the Hoare ordering defined by $A \leq^\# B$ iff $\uparrow B \subseteq \uparrow A$.

Theorem

Let S be a WSTS, then $\widehat{S}$ is a WSTS iff S is a ω^2-WSTS.

Relating executions of S and $\widehat{S}$

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then

- if $x \xrightarrow{k}_S y$,

Relating executions of S and $\widehat{S}$

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then

- if $x \xrightarrow{k}_S y$, then for every ideal $I \supseteq \downarrow x$

Relating executions of S and $\widehat{S}$

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then

- if $x \xrightarrow{k}_S y$, then for every ideal $I \supseteq \downarrow x$ there exists an ideal $J \supseteq \downarrow y$

Relating executions of S and $\widehat{S}$

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then

- if $x \xrightarrow{k}_S y$, then for every ideal $I \supseteq \downarrow x$ there exists an ideal $J \supseteq \downarrow y$ such that $I \xrightarrow{k}_{\widehat{S}} J$,

Relating executions of S and $\widehat{S}$

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then

- if $x \xrightarrow{k}_S y$, then for every ideal $I \supseteq \downarrow x$ there exists an ideal $J \supseteq \downarrow y$ such that $I \xrightarrow{k}_{\widehat{S}} J$,
- if $I \xrightarrow{k}_{\widehat{S}} J$,

Relating executions of S and $\widehat{S}$

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then

- if $x \xrightarrow{k}_S y$, then for every ideal $I \supseteq \downarrow x$ there exists an ideal $J \supseteq \downarrow y$ such that $I \xrightarrow{k}_{\widehat{S}} J$,
- if $I \xrightarrow{k}_{\widehat{S}} J$, then for every $y \in J$

Relating executions of S and $\widehat{S}$

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then

- if $x \xrightarrow{k}_S y$, then for every ideal $I \supseteq \downarrow x$ there exists an ideal $J \supseteq \downarrow y$ such that $I \xrightarrow{k}_{\widehat{S}} J$,
- if $I \xrightarrow{k}_{\widehat{S}} J$, then for every $y \in J$ there exists $x \in I$

Relating executions of S and $\widehat{S}$

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then

- if $x \xrightarrow{k}_S y$, then for every ideal $I \supseteq \downarrow x$ there exists an ideal $J \supseteq \downarrow y$ such that $I \xrightarrow{k}_{\widehat{S}} J$,
- if $I \xrightarrow{k}_{\widehat{S}} J$, then for every $y \in J$ there exists $x \in I$ such that $x \xrightarrow{*}_S y' \geq y$.

Relating executions of S and $\widehat{S}$

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS with transitive monotony, then

- if $x \xrightarrow{k}_S y$, then for every ideal $I \supseteq \downarrow x$ there exists an ideal $J \supseteq \downarrow y$ such that $I \xrightarrow{k}_{\widehat{S}} J$,
- if $I \xrightarrow{k}_{\widehat{S}} J$, then for every $y \in J$ there exists $x \in I$ such that $x \xrightarrow{\geq k}_S y' \geq y$.

Relating executions of S and $\widehat{S}$

Let $S = (X, \to_S, \leq)$ be a WSTS with strong monotony, then

- if $x \xrightarrow{k}_S y$, then for every ideal $I \supseteq \downarrow x$ there exists an ideal $J \supseteq \downarrow y$ such that $I \xrightarrow{k}_{\widehat{S}} J$,
- if $I \xrightarrow{k}_{\widehat{S}} J$, then for every $y \in J$ there exists $x \in I$ such that $x \xrightarrow{k}_S y' \geq y$.

Relations between S and $\widehat{S}$

A generality

The completion $\widehat{S} = (\widehat{X}, \rightarrow_{\widehat{S}}, \subseteq)$ computes exactly the downward closure of the reachability set of its original system $S = (X, \rightarrow_S, \leq)$.

An equality

We have: $\mathrm{Post}^*_{\widehat{S}}(\downarrow x) = \downarrow \mathrm{Post}^*_S(x)$.

In fact, it is more exactly:

Theorem

If $Post^*_{\widehat{S}}(\downarrow x) = \{J_1, \ldots, J_n\}$ *then* $\downarrow Post^*_S(x) = J_1 \cup \ldots \cup J_n$.

Theorem

Strong termination is decidable for infinitely branching WSTS with transitive monotony and such that $\widehat{S}$ is a post-effective WSTS.

Theorem

Strong termination is decidable for infinitely branching WSTS with transitive monotony and such that $\widehat{S}$ is a post-effective WSTS.

Post-effectiveness

Possible to compute cardinality of

Theorem

Strong termination is decidable for infinitely branching WSTS with transitive monotony and such that $\widehat{S}$ is a post-effective WSTS.

Proof

- Executions bounded in S iff bounded in $\widehat{S}$.

Theorem

Strong termination is decidable for infinitely branching WSTS with transitive monotony and such that $\widehat{S}$ is a post-effective WSTS.

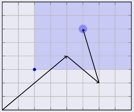
Proof

- Executions bounded in S iff bounded in $\widehat{S}$.
- $\widehat{S}$ finitely branching, can decide termination in $\widehat{S}$ by Finkel ICALP'87, Finkel & Schnoebelen TCS'01.

Coverability

Input: $(X, \rightarrow, \leq)$ a WSTS, $x_0, x \in X$.

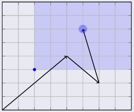
Question: $x_0 \xrightarrow{*} x' \geq x$?



Coverability

Input: $(X, \to, \leq)$ a WSTS, $x_0, x \in X$.

Question: $x_0 \in \uparrow \mathrm{Pre}^*(\uparrow x)$?



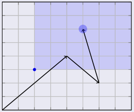
Backward method (Abdulla, Cerans, Jonsson & Tsay IC'00)

Compute $\uparrow \mathrm{Pre}^*(\uparrow x)$ iteratively assuming $\uparrow \mathrm{Pre}(\uparrow x)$ computable.

Coverability

Input: $(X, \to, \leq)$ a WSTS, $x_0, x \in X$.

Question: $x_0 \in \uparrow \mathrm{Pre}^*(\uparrow x)$?



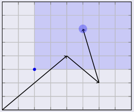
Backward method (Abdulla, Cerans, Jonsson & Tsay IC'00)

Compute $\uparrow \mathrm{Pre}^*(\uparrow x)$ iteratively assuming $\uparrow \mathrm{Pre}(\uparrow x)$ computable.

Coverability

Input: $(X, \rightarrow, \leq)$ a WSTS, $x_0, x \in X$.

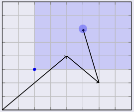
Question: $x \in \downarrow \mathrm{Post}^*(x_0)$?



Coverability

Input: $(X, \rightarrow, \leq)$ a WSTS, $x_0, x \in X$.

Question: $x \in \downarrow \mathrm{Post}^*(x_0)$?



Forward method

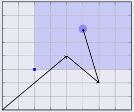
Coverability:

- Enumerate executions $\downarrow x_0 \xrightarrow{*}_{\widehat{S}} I$,
- Accept if $x \in I$.

Coverability

Input: $(X, \to, \leq)$ a WSTS, $x_0, x \in X$.

Question: $x \in \downarrow \text{Post}^*(x_0)$?



Forward method

Coverability:

- Enumerate executions $\downarrow x_0 \xrightarrow{*}_{\widehat{S}} I$,
- Accept if $x \in I$.

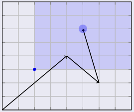
Non coverability:

- Enumerate $D \subseteq X$ downward closed, $x_0 \in D$ and $\downarrow \text{Post}_S(D) \subseteq D$
- Reject if $x \notin D$.

Coverability

Input: $(X, \to, \leq)$ a WSTS, $x_0, x \in X$.

Question: $x \in \downarrow \mathrm{Post}^*(x_0)$?



Forward method

Coverability:

- Enumerate executions $\downarrow x_0 \xrightarrow{*}_{\widehat{S}} I$,
- Accept if $x \in I$.

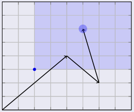
Non coverability:

- Enumerate $D = I_1 \cup \ldots \cup I_k$
- Reject if $x \notin D$.

Coverability

Input: $(X, \to, \leq)$ a WSTS, $x_0, x \in X$.

Question: $x \in \downarrow \text{Post}^*(x_0)$?



Forward method

Coverability:

- Enumerate executions $\downarrow x_0 \xrightarrow{*}_{\widehat{S}} I$,
- Accept if $x \in I$.

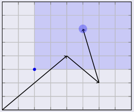
Non coverability:

- Enumerate $D \subseteq X$ downward closed
- Reject if $x \notin D$.

Coverability

Input: $(X, \to, \leq)$ a WSTS, $x_0, x \in X$.

Question: $x \in \downarrow \text{Post}^*(x_0)$?



Forward method

Coverability:

- Enumerate executions $\downarrow x_0 \xrightarrow{*}_{\widehat{S}} I$,
- Accept if $x \in I$.

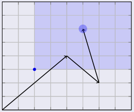
Non coverability:

- Enumerate $D \subseteq X$ downward closed, $x_0 \in D$
- Reject if $x \notin D$.

Coverability

Input: $(X, \to, \leq)$ a WSTS, $x_0, x \in X$.

Question: $x \in \downarrow \text{Post}^*(x_0)$?



Forward method

Coverability:

- Enumerate executions $\downarrow x_0 \xrightarrow{*}_{\widehat{S}} I$,
- Accept if $x \in I$.

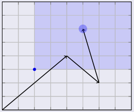
Non coverability:

- Enumerate $D \subseteq X$ downward closed, $\downarrow x_0 \subseteq I_1 \cup \ldots \cup I_k$
- Reject if $x \notin D$.

Coverability

Input: $(X, \rightarrow, \leq)$ a WSTS, $x_0, x \in X$.

Question: $x \in \downarrow \mathrm{Post}^*(x_0)$?



Forward method

Coverability:

- Enumerate executions $\downarrow x_0 \xrightarrow{*}_{\widehat{S}} I$,
- Accept if $x \in I$.

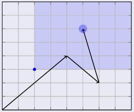
Non coverability:

- Enumerate $D \subseteq X$ downward closed, $\exists j$ s.t. $\downarrow x_0 \subseteq I_j$
- Reject if $x \notin D$.

Coverability

Input: $(X, \to, \leq)$ a WSTS, $x_0, x \in X$.

Question: $x \in \downarrow \mathrm{Post}^*(x_0)$?



Forward method

Coverability:

- Enumerate executions $\downarrow x_0 \xrightarrow{*}_{\widehat{S}} I$,
- Accept if $x \in I$.

Non coverability:

- Enumerate $D \subseteq X$ downward closed, $x_0 \in D$ and $\downarrow \mathrm{Post}_S(D) \subseteq D$
- Reject if $x \notin D$.

Prebasis computability

Prebasis computability is *sufficient*, but not *necessary*, to ensure decidability of coverability.

Prebasis computability

Prebasis computability is *sufficient*, but not *necessary*, to ensure decidability of coverability.

Coverability is decidable in $\mathcal{F}_1$

The algorithm consists to enumerate strictly increasing reachable sequences until finding an $y \geq x$.

Prebasis computability

Prebasis computability is *sufficient*, but not *necessary*, to ensure decidability of coverability.

Coverability is decidable in $\mathcal{F}_1$

The algorithm consists to enumerate strictly increasing reachable sequences until finding an $y \geq x$.

Prebasis is not computable for $\mathcal{F}_1$

Let $S_i = (\mathbb{N}, \rightarrow_{S_i}, \leq)$ be the WSTS such that:

- $x \rightarrow_{S_i} 0$ if TM_i does not halt on its encoding in $\leq x$ steps,
- $x \rightarrow_{S_i} 1$ otherwise.

Prebasis computability

Prebasis computability is *sufficient*, but not *necessary*, to ensure decidability of coverability.

Coverability is decidable in $\mathcal{F}_1$

The algorithm consists to enumerate strictly increasing reachable sequences until finding an $y \geq x$.

Prebasis is not computable for $\mathcal{F}_1$

Let $S_i = (\mathbb{N}, \rightarrow_{S_i}, \leq)$ be the WSTS such that:

- $x \rightarrow_{S_i} 0$ if TM_i does not halt on its encoding in $\leq x$ steps,
- $x \rightarrow_{S_i} 1$ otherwise.

Then $S_i \in \mathcal{F}_1$ and S_i is effective.

Three Pre sets

- $Pre_{S_i}(0) = \{x \in \mathbb{N} : TM_i$ does not halt in $\leq x$ steps $\}$,
- $Pre_{S_i}(1) = \{x \in \mathbb{N} : TM_i$ halts in $\leq x$ steps $\}$,
- $Pre_{S_i}(x) = \emptyset$ for $x \geq 2$.

Three Pre sets

- $Pre_{S_i}(0) = \{x \in \mathbb{N} : TM_i \text{ does not halt in } \leq x \text{ steps } \}$,
- $Pre_{S_i}(1) = \{x \in \mathbb{N} : TM_i \text{ halts in } \leq x \text{ steps } \}$,
- $Pre_{S_i}(x) = \emptyset$ for $x \geq 2$.

Conclusion: prebasis is not computable for $\mathcal{F}_1$

- Therefore, $\uparrow \mathrm{Pre}_{S_i}(\uparrow 1) = \uparrow \mathrm{Pre}_{S_i}(1) = \mathrm{Pre}_{S_i}(1)$.
- If an algorithm outputting a finite basis of $\uparrow \mathrm{Pre}_{S_i}(\uparrow 1)$ existed, then it would be possible to decide whether $\mathrm{Pre}_{S_i}(1) = \emptyset$.

Three Pre sets

- $Pre_{S_i}(0) = \{x \in \mathbb{N} : TM_i$ does not halt in $\leq x$ steps $\}$,
- $Pre_{S_i}(1) = \{x \in \mathbb{N} : TM_i$ halts in $\leq x$ steps $\}$,
- $Pre_{S_i}(x) = \emptyset$ for $x \geq 2$.

Conclusion: prebasis is not computable for $\mathcal{F}_1$

- Therefore, $\uparrow \mathrm{Pre}_{S_i}(\uparrow 1) = \uparrow \mathrm{Pre}_{S_i}(1) = \mathrm{Pre}_{S_i}(1)$.
- If an algorithm outputting a finite basis of $\uparrow \mathrm{Pre}_{S_i}(\uparrow 1)$ existed, then it would be possible to decide whether $\mathrm{Pre}_{S_i}(1) = \emptyset$.
- But $\mathrm{Pre}_{S_i}(1) = \emptyset$ iff TM_i does not halt.
- The halting problem thus Turing-reduces to the prebasis computation.

Boundedness for infinitely branching WSTS

Boundedness is decidable for post-effective WSTS with strict monotony and a wpo.

Boundedness for infinitely branching WSTS

Boundedness is decidable for post-effective WSTS with strict monotony and a wpo.

Proof

- We build a reachability tree T with root c_0 labelled x_0.
- If $\mathrm{Post}_S(x_0)$ is infinite, then we return "unbounded", otherwise we mark c_0 and for every $x \in \mathrm{Post}_S(x_0)$ we add a child labelled x to c_0.

Boundedness for infinitely branching WSTS

Boundedness is decidable for post-effective WSTS with strict monotony and a wpo.

Proof

- We build a reachability tree T with root c_0 labelled x_0.
- If $\text{Post}_S(x_0)$ is infinite, then we return "unbounded", otherwise we mark c_0 and for every $x \in \text{Post}_S(x_0)$ we add a child labelled x to c_0.
- If c has an ancestor c' labelled x' such that $x' < x$, we return "unbounded". Otherwise,
  - if c has an ancestor c' labelled x' such that $x' = x$, we mark c.
  - Otherwise, if $\text{Post}_S(x)$ is infinite, then we return "unbounded". Otherwise we mark c and for every $y \in \text{Post}_S(x)$ we add a child labelled y to c.

Boundedness for infinitely branching WSTS

Boundedness is decidable for post-effective WSTS with strict monotony and a wpo.

Proof

- We build a reachability tree T with root c_0 labelled x_0.
- If $\text{Post}_S(x_0)$ is infinite, then we return "unbounded", otherwise we mark c_0 and for every $x \in \text{Post}_S(x_0)$ we add a child labelled x to c_0.
- If c has an ancestor c' labelled x' such that $x' < x$, we return "unbounded". Otherwise,
  - if c has an ancestor c' labelled x' such that $x' = x$, we mark c.
  - Otherwise, if $\text{Post}_S(x)$ is infinite, then we return "unbounded". Otherwise we mark c and for every $y \in \text{Post}_S(x)$ we add a child labelled y to c.
- T is finite and correct.

Further result for infinitely branching WSTS

Strong maintainability is decidable for WSTS with strong monotony and such that $\widehat{S}$ is a post-effective WSTS.

Further work

- ∃ general class of infinitely branching WSTS with a Karp-Miller procedure?

Further work

- $\exists$ general class of infinitely branching WSTS with a Karp-Miller procedure?
- Toward the algorithmics of complete WSTS.

Further work

- $\exists$ general class of infinitely branching WSTS with a Karp-Miller procedure?
- Toward the algorithmics of complete WSTS.
- What else can we do with the WSTS completion?

Thank you!