Handling Infinite Branching WSTS

Michael Blondin\textsuperscript{1, 2}, Alain Finkel\textsuperscript{1} & Pierre McKenzie \textsuperscript{1, 2}

\textsuperscript{1}LSV, ENS Cachan

\textsuperscript{2}DIRO, Université de Montréal

January 6, 2014
Well-structured transition systems (WSTS) are known to encompass a large number of infinite state systems.
Well-structured transition systems (WSTS) are known to encompass a large number of infinite state systems.

Moreover, multiple decidability results are known on WSTS.
Well-structured transition systems (WSTS) are known to encompass a large number of infinite state systems. Moreover, multiple decidability results are known on WSTS. However, most results and techniques known suppose finite branching.
Well-structured transition systems (WSTS) are known to encompass a large number of infinite state systems.

Moreover, multiple decidability results are known on WSTS.

However, most results and techniques known suppose finite branching.

Developing from a theory elaborated by Finkel and Goubault-Larrecq, we introduce a way to work with infinitely branching WSTS.
Ordered transition systems

\[ S = (X, \rightarrow_S, \leq) \]

where

- \( X \) set,
- \( \rightarrow_S \subseteq X \times X \),
- \( \leq \) quasi-ordering \( X \).
Ordered transition systems

\[ S = (X, \rightarrow_S, \leq) \text{ where} \]

- \( X \) set: \textit{recursively enumerable},
- \( \rightarrow_S \subseteq X \times X \): \textit{decidable},
- \( \leq \) quasi-ordering \( X \): \textit{decidable}. 
Well-ordered transition system (WSTS)

A WSTS is an ordered transition system \((X, \rightarrow, \leq)\) with

- well-quasi-ordering: \(\forall x_0, x_1, \ldots \exists i < j \text{ s.t. } x_i \leq x_j,\)
- monotony:

\[
\forall x \quad \xrightarrow{\neg \neg} y \\
\forall x' \quad \xrightarrow{\neg \neg} y' \quad \exists
\]

\[x \rightarrow y^* y' \]

(Some) types of monotony

Standard monotony:

\[ \forall x \rightarrow y \]

\[ x' \rightarrow y' \]
(Some) types of monotony

**Strong** monotony:

\[
\forall x \quad \rightarrow y \quad \land \quad x' \quad \rightarrow y' \quad \exists
\]
(Some) types of monotony

**Transitive** monotony:

\[
\forall x \rightarrow y \quad \land \\
\land \\
\land \\
\exists \quad x' \rightarrow y'
\]
(Some) types of monotony

**Strict monotony:**

\[
\forall x \xrightarrow{} y \\
\wedge \\
\forall x' \xrightarrow{*} y' \\
\exists
\]
Branching

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

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

Some infinitely branching WSTS

- Inserting FIFO automata (Cécé, Finkel, Iyer 1996)
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 1996)
- Inserting automata (Bouyer, Markey, Ouaknine, Schnoebelen, Worrell 2012)
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 1996)
- Inserting automata (Bouyer, Markey, Ouaknine, Schnoebelen, Worrell 2012)
- \(\omega\)-Petri nets (Geeraerts, Heussner, Praveen & Raskin 2013),
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 1996)
- Inserting automata (Bouyer, Markey, Ouaknine, Schnoebelen, Worrell 2012)
- \(\omega\)-Petri nets (Geeraerts, Heussner, Praveen & Raskin 2013)
- Parameterized WSTS,
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 1996)
- Inserting automata (Bouyer, Markey, Ouaknine, Schnoebelen, Worrell 2012)
- \(\omega\)-Petri nets (Geeraerts, Heussner, Praveen & Raskin 2013),
- Parameterized WSTS,
- etc.
### Definitions

#### Effectiveness

A WSTS \((X, \rightarrow, \leq)\) is post-effective if it is possible to compute \(|\text{Post}(x)|\) for every \(x \in X\).
Effectiveness

A WSTS \((X, \rightarrow, \leq)\) is post-effective if it is possible to compute \(|\text{Post}(x)|\) for every \(x \in X\).

Remark

If \(\text{Post}(x)\) is finite, then it is computable by minimal hypotheses. Therefore, our definition generalizes post-effectiveness for finitely branching WSTS.
### 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 & Schnoebelen 2001)

Decidable for finitely branching post-effective WSTS with transitive monotony.
Termination

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

**Question:** \(\exists x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow \ldots\)?

**Theorem (Blondin, Finkel & McKenzie in progress)**

Undecidable for infinitely branching post-effective WSTS with transitive monotony.
Boundedness

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

**Question:** \(\text{Post}^*(x_0)\) finite?

**Theorem (Finkel & Schnoebelen 2001)**

Decidable for finitely branching post-effective WSTS with strict monotony.
Boundedness

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

**Question:** \(\text{Post}^*(x_0)\) finite?

**Theorem (Blondin, Finkel & McKenzie in progress)**

Decidable for infinitely branching post-effective WSTS with strict monotony.
Coverability

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

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

Theorem (Abdulla, Cerans, Jonsson & Tsay 2000; Finkel & Schnoebelen 2001)

Decidable for some classes of infinitely branching WSTS.
Coverability

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

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

**Theorem (Blondin, Finkel & McKenzie in progress)**

Decidable for *some classes* of infinitely branching WSTS.
Control-state maintainability

**Input:** \((X, \to, \leq)\) a WSTS, \(x_0 \in X\) and \(\{t_1, \ldots, t_n\} \subseteq X\).

**Question:** \(\exists\) maximal execution \(x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow \ldots\) such that \(\forall i \ x_i \in \uparrow\{t_1, \ldots, t_n\}\)?

**Theorem (Finkel & Schnoebelen 2001)**

Decidable for finitely branching post-effective WSTS with stuttering monotony.
Control-state maintainability

**Input:** \((X, \rightarrow, \leq)\) a WSTS, \(x_0 \in X\) and \(\{t_1, \ldots, t_n\} \subseteq X\).

**Question:** \(\exists\) maximal execution \(x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow \ldots\) such that \(\forall i \ x_i \in \uparrow \{t_1, \ldots, t_n\}\)?

**Theorem (Blondin, Finkel & McKenzie in progress)**

**Undecidable for infinitely branching post-effective WSTS with stuttering monotony.**
Downward closure

\[ \downarrow D = \{ x \in X : \exists d \in D \ x \leq d \} . \]

Ideals

\( I \subseteq X \) is an ideal if it is

- downward closed: \( I = \downarrow I \),
- directed: \( a, b \in I \implies \exists c \in I \ \text{s.t.} \ a \leq c \ \text{and} \ b \leq c \).
Theorem (Finkel & Goubault-Larrecq 2009)

Every downward closed set in $X$ is a finite union of ideals of $X$. 
Theorem (Finkel & Goubault-Larrecq 2009)

Every downward closed set in $X$ is a finite union of ideals of $X$.

Corollary (FGL 2009; Blondin, Finkel & McKenzie in progress)

Every downward closed subset decomposes canonically as the union of its maximal ideals.
Completion (FGL 2009; Blondin, Finkel & McKenzie in progress)

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

- $\hat{X} = \text{Ideals}(X)$,
- $I \rightarrow_{\hat{S}} J$ if $J$ appears in the canonical decomposition of $\downarrow\text{Post}(I)$. 
Theorem (FGL 2009; Blondin, Finkel & McKenzie in progress)

Let $S = (X, \rightarrow_S, \leq)$ be a WSTS, then
- $\hat{S}$ is finitely branching.
Theorem (FGL 2009; Blondin, Finkel & McKenzie in progress)

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

- $\hat{S}$ is finitely branching.
- $\hat{S}$ has strong monotony.
Theorem (FGL 2009; Blondin, Finkel & McKenzie in progress)

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

- \( \hat{S} \) is finitely branching.
- \( \hat{S} \) has strong monotony.
- \( \hat{S} \) is a WSTS iff \( S \) is a \( \omega^2 \)-WSTS iff \( A \leq \# B \iff \uparrow A \subseteq \uparrow B \) is a wqo (by Jančar 1999).
Ideals in $\mathbb{N}^d$

$I \subseteq \mathbb{N}^d$ is an ideal iff $I = \downarrow x_1 \times \cdots \times \downarrow x_d$ with $x_i \in \mathbb{N}$ or $x_i = \mathbb{N}$. 

Representation

$\downarrow 5 \times \mathbb{N} \times \downarrow 10$ can be represented by $(5, \omega, 10)$,

$\downarrow 5 \times \mathbb{N} \times \downarrow 10 \subseteq \mathbb{N} \times \mathbb{N} \times \downarrow 20$ can be tested by $(5, \omega, 10) \leq (\omega, \omega, 20)$. 

36/95
Ideals in \( \mathbb{N}^d \)

\( I \subseteq \mathbb{N}^d \) is an ideal iff \( I = \downarrow x_1 \times \cdots \times \downarrow x_d \) with \( x_i \in \mathbb{N} \) or \( x_i = \mathbb{N} \).

Representation

- \( \downarrow 5 \times \mathbb{N} \times \downarrow 10 \) can be represented by \((5, \omega, 10)\),
Ideals and completion
Examples

Ideals in $\mathbb{N}^d$

$I \subseteq \mathbb{N}^d$ is an ideal iff $I = \downarrow x_1 \times \cdots \times \downarrow x_d$ with $x_i \in \mathbb{N}$ or $x_i = \mathbb{N}$.

Representation

- $\downarrow 5 \times \mathbb{N} \times \downarrow 10$ can be represented by $(5, \omega, 10)$,
- $\downarrow 5 \times \mathbb{N} \times \downarrow 10 \subseteq \mathbb{N} \times \mathbb{N} \times \downarrow 20$ can be tested by $(5, \omega, 10) \leq (\omega, \omega, 20)$. 

VAS completions are post-effective

- Transitions can be carried in $\mathbb{N}_\omega^d$. 

The maximal elements obtained are the ideals of $\hat{\text{Post}}(I)$. 

Example: $\text{VAS}_A = \{(2, -3, -5), (4, 5, -1), (-6, -2, 5)\}$ and ideal $I = ↓[5 \times \mathbb{N} \times 10]$. 

$\text{Post}(I) = \frac{39}{95}$.
VAS completions are post-effective

- Transitions can be carried in $\mathbb{N}^d_\omega$,
- The maximal elements obtained are the ideals of $\text{Post}^\wedge_S(I)$. 
VAS completions are post-effective

- Transitions can be carried in $\mathbb{N}_\omega^d$,
- The maximal elements obtained are the ideals of $\text{Post}_{\widehat{S}}(I)$.

Example

VAS $A = \{(2, -3, -5), (4, 5, -1), (-6, -2, 5)\}$ and ideal $I = \downarrow 5 \times \mathbb{N} \times \downarrow 10$: 

\[
\begin{align*}
(5, \omega, 10) + (2, -3, -5) &= (7, \omega, 5) \\
(4, 5, -1) + (2, -3, -5) &= (6, 2, -1) \\
(-6, -2, 5) + (2, -3, -5) &= (-4, -5, 0)
\end{align*}
\]
VAS completions are post-effective

- Transitions can be carried in $\mathbb{N}_{\omega}^d$,
- The maximal elements obtained are the ideals of $\text{Post}_{\tilde{S}}(I)$.

Example

VAS $A = \{(2, -3, -5), (4, 5, -1), (-6, -2, 5)\}$ and ideal $I = \downarrow 5 \times \mathbb{N} \times \downarrow 10$:

$$(5, \omega, 10) + (2, -3, -5) = (7, \omega, 5)$$

$\downarrow \text{Post}(I) = \downarrow 7 \times \mathbb{N} \times \downarrow 5$
**VAS completions are post-effective**

- Transitions can be carried in $\mathbb{N}_\omega^d$.
- The maximal elements obtained are the ideals of $\text{Post}^\sim(S(I))$.

**Example**

VAS $A = \{(2, -3, -5), (4, 5, -1), (-6, -2, 5)\}$ and ideal $I = \downarrow 5 \times \mathbb{N} \times \downarrow 10$:

$$(5, \omega, 10) + (4, 5, -1) = (9, \omega, 9)$$

$$\downarrow \text{Post}(I) = \downarrow 7 \times \mathbb{N} \times \downarrow 5 \cup \downarrow 9 \times \mathbb{N} \times \downarrow 9$$
VAS completions are post-effective

- Transitions can be carried in $\mathbb{N}_\omega^d$,
- The maximal elements obtained are the ideals of $\text{Post}_\hat{S}(I)$.

Example

VAS $A = \{(2, -3, -5), (4, 5, -1), (-6, -2, 5)\}$ and ideal $I = \downarrow 5 \times \mathbb{N} \times \downarrow 10$:

$$(5, \omega, 10) + (-6, -2, 5) = \emptyset$$

$\downarrow \text{Post}(I) = \downarrow 7 \times \mathbb{N} \times \downarrow 5 \cup \downarrow 9 \times \mathbb{N} \times \downarrow 9$$
VAS completions are post-effective

- Transitions can be carried in $\mathbb{N}_\omega^d$,
- The maximal elements obtained are the ideals of $\text{Post}^\wedge_S(I)$.

Example

VAS $A = \{(2, -3, -5), (4, 5, -1), (-6, -2, 5)\}$ and ideal $I = \downarrow 5 \times \mathbb{N} \times \downarrow 10$:

$$\downarrow \text{Post}(I) = \downarrow 7 \times \mathbb{N} \times \downarrow 5 \cup \downarrow 9 \times \mathbb{N} \times \downarrow 9$$
VAS completions are post-effective

- Transitions can be carried in $\mathbb{N}^d_\omega$,
- The maximal elements obtained are the ideals of $\text{Post}_{\hat{S}}(I)$.

Example

VAS $A = \{(2, -3, -5), (4, 5, -1), (-6, -2, 5)\}$ and ideal $I = \downarrow 5 \times \mathbb{N} \times \downarrow 10$:

$$\text{Post}_{\hat{S}}(I) = \{\downarrow 9 \times \mathbb{N} \times \downarrow 9\}$$
### Coverability

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

**Question:** \(x_0 \xrightarrow{*} x' \geq x\)?
Coverability

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

**Question:** \(x_0 \in \uparrow \text{Pre}^*(\uparrow x)\)?
Coverability

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

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

**Backward method (Abdulla, Cerans, Jonsson & Tsay 2000)**

Compute sequence converging to \(\uparrow \text{Pre}^* (\uparrow x)\):

\[
\begin{align*}
Y_0 &= \uparrow x \\
Y_1 &= Y_0 \cup \uparrow \text{Pre}(Y_0) \\
& \vdots \\
Y_n &= Y_{n-1} \cup \uparrow \text{Pre}(Y_{n-1})
\end{align*}
\]

and verify if \(x_0 \in Y_n\).
Coverability

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

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

**Backward method (Abdulla, Cerans, Jonsson & Tsay 2000)**

Compute sequence converging to \(\uparrow \text{Pre}^* (\uparrow x)\):

\[
egin{align*}
Y_0 & = \uparrow x \\
Y_1 & = Y_0 \cup \uparrow \text{Pre}(Y_0) \\
\vdots & \vdots \\
Y_n & = Y_{n-1} \cup \uparrow \text{Pre}(Y_{n-1})
\end{align*}
\]

and verify if \(x_0 \in Y_n\). Computing Pre not always efficient!
Coverability

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

**Question:** \(x_0 \xrightarrow{*} x' \geq x\)?
### Coverability

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

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

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

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

**Theorem (Blondin, Finkel & McKenzie in progress)**

Coverability is decidable for WSTS with post-effective completion.
Proof: two semi-algorithms to decide coverability

Coverability:
- Enumerate execution \( \downarrow x_0 \xrightarrow{\ast} S \downarrow I \),
- Accept if \( x \in I \).
Proof: two semi-algorithms to decide coverability

**Coverability:**
- Enumerate execution $x_0 \xrightarrow{*} S I,$
- Accept if $x \in I.$

**Non coverability:**
- Enumerate $D \subseteq X$ downward closed
Proof: two semi-algorithms to decide coverability

<table>
<thead>
<tr>
<th>Coverability:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enumerate execution $\downarrow x_0 \xrightarrow[*]{S} I$,</td>
</tr>
<tr>
<td>Accept if $x \in I$.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Non coverability:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enumerate $D = I_1 \cup \ldots \cup I_k$</td>
</tr>
</tbody>
</table>
Proof: two semi-algorithms to decide coverability

**Coverability:**
- Enumerate execution $\downarrow x_0 \xrightarrow{S} I,$
- Accept if $x \in I.$

**Non coverability:**
- Enumerate $D \subseteq X$ downward closed
Proof: two semi-algorithms to decide coverability

Coverability:

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

Non coverability:

- Enumerate $D \subseteq X$ downward closed, $x_0 \in D$
Proof: two semi-algorithms to decide coverability

**Coverability:**
- Enumerate execution $\downarrow x_0 \xrightarrow{*} S I$,
- Accept if $x \in I$.

**Non coverability:**
- Enumerate $D \subseteq X$ downward closed, $\downarrow x_0 \subseteq D$
Proof: two semi-algorithms to decide coverability

**Coverability:**

- Enumerate execution $\downarrow x_0 \xrightarrow{\ast} S I$,
- Accept if $x \in I$.

**Non coverability:**

- Enumerate $D \subseteq X$ downward closed, $\downarrow x_0 \subseteq I_1 \cup \ldots \cup I_k$
Proof: two semi-algorithms to decide coverability

Coverability:

- Enumerate execution $\downarrow x_0 \xrightarrow{\ast} S \subseteq I$,
- Accept if $x \in I$.

Non coverability:

- Enumerate $D \subseteq X$ downward closed, $\exists i \text{ t.q. } \downarrow x_0 \subseteq l_i$
Proof: two semi-algorithms to decide coverability

Coverability:
- Enumerate execution $\downarrow x_0 \xrightarrow{*} \hat{S} I$,
- Accept if $x \in I$.

Non coverability:
- Enumerate $D \subseteq X$ downward closed, $x_0 \in D$
**Proof: two semi-algorithms to decide coverability**

### Coverability:
- Enumerate execution \( \downarrow x_0 \xrightarrow{\ast} S \downarrow I \),
- Accept if \( x \in I \).

### Non coverability:
- Enumerate \( D \subseteq X \) downward closed, \( x_0 \in D \) and \( \downarrow \text{Post}_S(D) \subseteq D \).
Proof: two semi-algorithms to decide coverability

Coverability:

- Enumerate execution $\downarrow x_0 \xrightarrow{\ast} \hat{S} l$,
- Accept if $x \in I$.

Non-coverability:

- Enumerate $D \subseteq X$ downward closed, $x_0 \in D$ and $\downarrow \text{Post}_S(I_1 \cup \ldots \cup I_k) \subseteq I_1 \cup \ldots \cup I_k$
Proof: two semi-algorithms to decide coverability

**Coverability:**
- Enumerate execution $\downarrow x_0 \xrightarrow{\ast} I$,
- Accept if $x \in I$.

**Non coverability:**
- Enumerate $D \subseteq X$ downward closed, $x_0 \in D$ and $\downarrow \text{Post}_S(I_1) \cup \ldots \cup \downarrow \text{Post}_S(I_k) \subseteq I_1 \cup \ldots \cup I_k$
Proof: two semi-algorithms to decide coverability

Coverability:
- Enumerate execution $\downarrow x_0 \xrightarrow{*} \tilde{S} I$, 
- Accept if $x \in I$.

Non-coverability:
- Enumerate $D \subseteq X$ downward closed, $x_0 \in D$ and
  \[ (J_{1,1} \cup \ldots \cup J_{1,n_1}) \cup \ldots \cup (J_{k,1} \cup \ldots \cup J_{k,n_k}) \subseteq I_1 \cup \ldots \cup I_k \]
  \[ \text{Post}_{\tilde{S}}(I_1) = \{J_{1,1}, \ldots, J_{1,n_1}\} \]
  \[ \text{Post}_{\tilde{S}}(I_k) = \{J_{k,1}, \ldots, J_{k,n_k}\} \]
### Proof: two semi-algorithms to decide coverability

**Coverability:**
- Enumerate execution \(\downarrow x_0 \xrightarrow{*} \hat{S} \in I\),
- Accept if \(x \in I\).

**Non coverability:**
- Enumerate \(D \subseteq X\) downward closed, \(x_0 \in D\) and \(\exists i, j, i'\) t.q. \(J_{i,j} \subseteq I_{i'}\).
Proof: two semi-algorithms to decide coverability

Coverability:

- Enumerate execution \(\downarrow x_0 \xrightarrow{\ast} \hat{S} I\),
- Accept if \(x \in I\).

Non coverability:

- Enumerate \(D \subseteq X\) downward closed, \(x_0 \in D\) and \(\downarrow \text{Post}_S(D) \subseteq D\).
Proof: two semi-algorithms to decide coverability

**Coverability:**
- Enumerate execution \( \downarrow x_0 \xrightarrow{\ast} S I \),
- 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 \( x \notin D \).
Proof: two semi-algorithms to decide coverability

**Coverability:**
- Enumerate execution $\downarrow x_0 \xrightarrow{S} I$,
- 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 $\downarrow x \not\subseteq I_1 \cup \ldots \cup I_k$. 
Proof: two semi-algorithms to decide coverability

**Coverability:**

- Enumerate execution $\downarrow x_0 \xrightarrow{*}_{\Sigma} I$,  
- Accept if $x \in I$.

**Non coverability:**

- Enumerate $D \subseteq X$ downward closed, $x_0 \in D$ and $\downarrow Post_{\Sigma}(D) \subseteq D$,  
- Reject if $\forall i \downarrow x \not\subseteq l_i$. 
Proof: two semi-algorithms to decide coverability

Coverability:
- Enumerate execution \( \downarrow x_0 \xrightarrow{\ast} I \),
- 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 \( x \not\in D \). **Witness:** \( D = \downarrow \text{Post}_S^\ast(x_0) \)
### Termination

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

**Question:** \(\exists x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow \ldots\)?
Termination

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

*Question:* \(\exists x_0 \to x_1 \to x_2 \to \ldots?\)

**Theorem (Blondin, Finkel & McKenzie in progress)**

Termination is undecidable, even for post-effective \(\omega^2\)-WSTS with strong and strict monotony, and with post-effective completion.
Termination

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

**Question:** \(\exists x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow \ldots\) ?

Proof

Structural termination is undecidable for Transfer Petri nets (Dufourd, Jančar & Schnoebelen 1999). Structural termination reduces to termination by adding a new element that branches on every other elements.
Execution boundedness

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

Question: \(\exists k\) bounding length of executions?
**Execution boundedness**

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

**Question:** \(\exists k\) bounding length of executions?

**Remark**

Termination and execution boundedness are the same in finitely branching WSTS.
Relating executions of $S$ and $\hat{S}$

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 $I \xrightarrow{k} \hat{S} J$,

- if $I \xrightarrow{k} \hat{S} J$, then for every $y \in J$ there exists $x \in I$ such that $x \xrightarrow{*} S y' \geq y$. 
Relating executions of $S$ and $\hat{S}$

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 $I \xrightarrow{k_{\hat{S}}} J$,

- if $I \xrightarrow{k_{\hat{S}}} J$, 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 $\hat{S}$

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

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

- if $I \xrightarrow{k} \hat{S} J$, then for every $y \in J$ there exists $x \in I$ such that $x \xrightarrow{k} y' \geq y$. 
Theorem (Blondin, Finkel & McKenzie in progress)

Execution boundedness is decidable for $\omega^2$-WSTS with transitive monotony, and with post-effective completion.
Theorem (Blondin, Finkel & McKenzie in progress)

Execution boundedness is decidable for $\omega^2$-WSTS with transitive monotony, and with post-effective completion.

Proof

Executions are bounded in $S$ iff bounded in $\hat{S}$. Since $\hat{S}$ is finitely branching, it suffices to solve termination in $\hat{S}$. 
Control-state maintainability

**Input:** $(X, \rightarrow, \leq)$ a WSTS, $x_0 \in X$ and $\{t_1, \ldots, t_n\} \subseteq X$.

**Question:** $\exists$ maximal execution $x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow \ldots$ such that $\forall i \ x_i \in \uparrow \{t_1, \ldots, t_n\}$?
Control-state maintainability

Input: \((X, \rightarrow, \leq)\) a WSTS, \(x_0 \in X\) and \(\{t_1, \ldots, t_n\} \subseteq X\).

Question: \(\exists\) maximal execution \(x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow \ldots\) such that \(\forall i \ x_i \in \uparrow\{t_1, \ldots, t_n\}\)?

Theorem (Blondin, Finkel & McKenzie in progress)

Control-state maintainability is undecidable, even for post-effective \(\omega^2\)-WSTS with strong and strict monotony, and with post-effective completion.
Control-state maintainability boundedness

**Input:** \((X, \to, \leq)\) a WSTS, \(x_0 \in X\) and \(\{t_1, \ldots, t_n\} \subseteq X\).

**Question:** \(\exists k\) bounding lengths of executions \(x_0 \to x_1 \to x_2 \to \ldots\) such that \(\forall i\ x_i \in \uparrow \{t_1, \ldots, t_n\}\)?
Control-state maintainability boundedness

**Input:** \((X, \rightarrow, \leq)\) a WSTS, \(x_0 \in X\) and \(\{t_1, \ldots, t_n\} \subseteq X\).

**Question:** \(\exists k\) bounding lengths of executions \(x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow \ldots\) such that \(\forall i \ x_i \in \uparrow \{t_1, \ldots, t_n\}\)?

**Remark**

Control-state maintainability and control-state maintainability boundedness are (almost) the same in finitely branching WSTS.
Theorem (Blondin, Finkel & McKenzie in progress)

Control-state maintainability boundedness is decidable for $\omega^2$-WSTS with transitive monotony, and with post-effective completion.
Theorem (Blondin, Finkel & McKenzie in progress)

Control-state maintainability boundedness is decidable for \(\omega^2\)-WSTS with transitive monotony, and with post-effective completion.

Proof

“Good” executions are bounded in \(S\) iff “good” executions are bounded in \(\hat{S}\). Since \(\hat{S}\) is finitely branching, it suffices to solve control-state maintainability in \(\hat{S}\).
**Boundedness**

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

**Question:** \(\text{Post}^*(x_0)\) finite?
Boundedness

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

Question: \(\text{Post}^*(x_0)\) finite?

Theorem (Blondin, Finkel & McKenzie in progress)

Boundedness is decidable for post-effective WSTS with strict monotony.
**Boundedness**

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

*Question:* \(\text{Post}^*(x_0)\) finite?

**Proof**

Build a finite reachability tree as in (Finkel & Schnoebelen 2001) returning “unbounded” if some infinite \(\text{Post}(x)\) is encountered.
Open questions

- What hypotheses make termination and control-state maintainability decidable?
Open questions

- What hypotheses make termination and control-state maintainability decidable?
- Other problems can be solved for infinitely branching WSTS?
Open questions

- What hypotheses make termination and control-state maintainability decidable?
- Other problems can be solved for infinitely branching WSTS?
- What other applications has the completion?
Thank you! Merci!