Decidability of Weak Simulation on One-Counter Nets

Piotr Hofman\textsuperscript{1}  Richard Mayr\textsuperscript{2}  Patrick Totzke\textsuperscript{2}

University of Warsaw\textsuperscript{1}  University of Edinburgh\textsuperscript{2}

June 22, 2013
One-Counter Nets

\((Q, Act, \delta)\) \hspace{1cm} \delta \subseteq (Q \times Act \times \{-1, 0, +1\} \times Q)
One-Counter Nets

\((Q, \text{Act}, \delta)\) \quad \delta \subseteq (Q \times \text{Act} \times \{-1, 0, +1\} \times Q)

Induced LTS over \(Q \times \mathbb{N}\)
One-Counter Nets

\((Q, \text{Act}, \delta) \quad \delta \subseteq (Q \times \text{Act} \times \{-1, 0, +1\} \times Q)\)

Induced LTS over \(Q \times \mathbb{N}\)
One-Counter Nets

\((Q, Act, \delta)\) \quad \delta \subseteq (Q \times Act \times \{-1, 0, +1\} \times Q)
Simulation Games

...are played in rounds between Spoiler and Duplicator. If a player cannot move the other wins. Infinite plays are won by Duplicator.
...are played in rounds between Spoiler and Duplicator. If a player cannot move the other wins. Infinite plays are won by Duplicator.

In each round

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
<td>vs.</td>
<td>$\beta$</td>
</tr>
</tbody>
</table>

1. Spoiler moves from $\alpha$
2. Duplicator responds from $\beta$
3. game continues from $\alpha'$ vs. $\beta'$
Simulation Games

... are played in rounds between Spoiler and Duplicator. If a player cannot move the other wins. Infinite plays are won by Duplicator.

In each round

\[
\alpha \quad \text{vs.} \quad \beta
\]

1. Spoiler moves from \( \alpha \)
2. Duplicator responds from \( \beta \)
3. game continues from \( \alpha' \) vs. \( \beta' \)
Simulation Games

...are played in rounds between Spoiler and Duplicator. If a player cannot move the other wins. Infinite plays are won by Duplicator.

In each round

1. Spoiler moves from $\alpha$
2. Duplicator responds from $\beta$
3. game continues from $\alpha'$ vs. $\beta'$
Simulation Games

...are played in rounds between Spoiler and Duplicator. If a player cannot move the other wins. Infinite plays are won by Duplicator.

In each round

| α    vs.  β       |
|---|---|
| a  | a |
| α' vs. β'         |

1. Spoiler moves from α
2. Duplicator responds from β
3. game continues from α' vs. β'
Simulation Games

... are played in rounds between Spoiler and Duplicator. If a player cannot move the other wins. Infinite plays are won by Duplicator.

In each round

\[
\begin{array}{ccc}
\alpha & \text{vs.} & \beta \\
 a & \text{vs.} & \beta' \\
\end{array}
\]

1. Spoiler moves from \( \alpha \)
2. Duplicator responds from \( \beta \)
3. game continues from \( \alpha' \) vs. \( \beta' \)

Def: Simulation (\( \preceq \))

\( \alpha \preceq \beta \) iff Duplicator has a strategy to win from \( \alpha \) vs. \( \beta \).
...are played in rounds between Spoiler and Duplicator. If a player cannot move the other wins.

In round from $\alpha, \beta, i$

1. Spoiler moves from $\alpha$; picks ordinal $j < i$

2. Duplicator responds from $\beta$

3. game continues from $\alpha', \beta', j$

Def: Simulation Approximant $(\preceq_i)$

$\alpha \preceq_i \beta$ iff Duplicator has a strategy to win from $\alpha$ vs. $\beta$. 
### Weak Notions

#### Weak Steps ($a \neq \tau \in \text{Act}$)

<table>
<thead>
<tr>
<th>Action</th>
<th>Weak Notion</th>
<th>Approximants</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau \to \ast$</td>
<td>$\tau \to \ast$</td>
<td>$\tau \to \ast$</td>
</tr>
<tr>
<td>$a \to \ast$</td>
<td>$\tau \to \ast, a \to \ast, \tau \to \ast$</td>
<td>$\tau \to \ast$</td>
</tr>
</tbody>
</table>
Weak Notions

Weak Steps \((a \neq \tau \in \text{Act})\)

\[\begin{align*}
\tau \rightarrow := & \quad \tau \rightarrow^* \\
a \rightarrow := & \quad \tau \rightarrow^* \rightarrow a \rightarrow \tau \rightarrow^*
\end{align*}\]

Def: Weak Simulation \(\leq\) and Approximants \(\leq_i\)

by 2-player games as before where Duplicator makes weak steps...
Example

Countdown game

```
a, 0  \quad a, -1  \quad \tau, +1  \quad a, -1
S \quad D \quad C \quad B
```

Strong Simulation: \( S_0 \preceq_0 D_0 \)

Weak Simulation: \( S_0 \preceq_\omega D_0 \)
**Example**

**Countdown game**

- $S$: $a, 0$
- $D$: $a, -1$
- $C$: $\tau, +1$
- $B$: $a, -1$

**Strong Simulation:**

- $S0 \preceq_0 D0$
Example

**Countdown game**

<table>
<thead>
<tr>
<th>State</th>
<th>Action</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S$</td>
<td>$a, 0$</td>
<td>$D$</td>
</tr>
<tr>
<td></td>
<td>$\tau, 0$</td>
<td>$C$</td>
</tr>
<tr>
<td>$D$</td>
<td>$a, -1$</td>
<td>$\tau, +1$</td>
</tr>
<tr>
<td>$C$</td>
<td>$a, 0$</td>
<td>$B$</td>
</tr>
</tbody>
</table>

**Strong Simulation:**

- $S_0 \preceq_0 D_0$
- $S_0 \not\preceq_1 D_0$
Example

Countdown game

\[
\begin{align*}
S & \quad a, 0 & D & \quad a, -1 & C & \quad \tau, +1 & B & \quad a, -1 \\
S & \quad a, 0 & D & \quad \tau, 0 & C & \quad a, 0 & B & \quad \tau, 0
\end{align*}
\]

Strong Simulation:
- \( S_0 \preceq_0 D_0 \)
- \( S_0 \npreceq_1 D_0 \)

Weak Simulation:
- \( S_0 \preceq_\omega D_0 \)
Example

Countdown game

<table>
<thead>
<tr>
<th>State</th>
<th>Transition</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>$a, 0$</td>
<td>D</td>
</tr>
<tr>
<td>D</td>
<td>$a, -1$</td>
<td>C</td>
</tr>
<tr>
<td>C</td>
<td>$\tau, +1$</td>
<td>B</td>
</tr>
<tr>
<td>B</td>
<td>$a, -1$</td>
<td></td>
</tr>
</tbody>
</table>

Strong Simulation:
- $S_0 \preceq_0 D_0$
- $S_0 \not\preceq_1 D_0$

Weak Simulation:
- $S_0 \preceq_\omega D_0$
- $S_0 \not\preceq_\omega+1 D_0$
Example

Countdown game

- $a, 0$ (S)
- $a, -1$ (D)
- $\tau, +1$ (C)
- $a, -1$ (B)

Strong Simulation:
- $S_0 \preceq_0 D_0$
- $S_0 \not\preceq_1 D_0$

Weak Simulation:
- $S_0 \preceq_\omega D_0$
- $S_0 \not\preceq_\omega+1 D_0$
- $S_0 \not\preceq D_0$
Our Main Contribution

We show decidability of the

**OCN Weak Simulation Problem**

**Input:** A net $N = (Q, Act, \delta)$ and configurations $pm, qn$.

**Question:** $pm \leq qn$?
We show decidability of the

**OCN Weak Simulation Problem**

**Input:** A net $N = (Q, Act, \delta)$ and configurations $pm, qn$.

**Question:** $pm \leq qn$?

**Theorem**

*For a given net, the relation $\leq$ is effectively semilinear.*
Why should you care?

In practice, modelling might use both $\infty$-states and branching:
- network protocols/queues keeping track of their workload
- random guesses

Theoretically, surprising:
- rare positive result for behavioral preorder that is not finitely approximable $\preceq \neq \preceq_\omega$.
- goes against the usual ‘finer is easier’ trend
Some Context – Strong Case

- PDA
- OCA
- OCN
- NFA
- Petri Nets

\[ \subseteq \text{undecidable} \]
\[ \preceq \text{decidable} \]
\[ \preceq \text{PSPACE-hard} \]
\[ \sim \text{PSPACE-}c \]
\[ \subseteq \text{undec.} \]
\[ \preceq \text{undec.} \]
\[ \sim \text{undec.} \]
\[ \subseteq \text{PSPACE-comp.} \]
\[ \preceq \text{P-comp.} \]
Some Context – Strong Case

\[ \subseteq \text{undecidable} \]
\[ \preceq \text{undecidable} \]
\[ \sim \text{decidable} \ [\text{Sén98}] \]

\[ \subseteq \text{undecidable} \]
\[ \preceq \text{undecidable} \ [\text{JMS99}] \]
\[ \sim \text{PSPACE}-\text{c} \ [\text{Srb09, BGJ10}] \]

\[ \subseteq \text{undecidable} \ [\text{HMT13}] \]
\[ \preceq \text{decidable} \ [\text{AC98}, \text{PSPACE}-\text{hard} \ [\text{Srb09}] \]
\[ \sim \text{PSPACE}-\text{c} \ [\text{Srb09, BGJ10}] \]

\[ \subseteq \text{PSPACE-comp.} \]
\[ \preceq \text{P-comp.} \]
\[ \sim \text{P-comp.} \ [\text{PJ95}] \]
Some Context – Weak Case

- Undecidable
- Decidable [HMT13]
- PSPACE-hard [Srb09]
- Decidable [HMT13]
- P-comp.
- PSPACE-comp.
- Undec.
- Undec.
- Undec.
- Undec.
- Undec.
- Undec.

References:
- [HMT13]
- [PJ95]
- [Hüt94]
- [Srb09]
- [May03]
- [JMS99]
Monotonicity in Nets

If \( pm \xrightarrow{a} qn \) Then \( p(m + 1) \xrightarrow{a} q(n + 1) \).
Monotonicity in Nets

If $pm \xrightarrow{a} qn$ then $p(m + 1) \xrightarrow{a} q(n + 1)$.

If $m' \leq m$ then $pm' \leq pm$. 
Monotonicity in Nets

If $pm \xrightarrow{a} qn$ Then $p(m + 1) \xrightarrow{a} q(n + 1)$.

If $m' \leq m$ Then $pm' \leq pm$.

If $m' \leq m$, $pm \leq qn$ and $n \leq n'$ Then $pm' \leq qn'$. 
(m, n) is black iff pm \preceq qn
Monotonicity illustrated

\[(m, n) \text{ is black iff } pm \preceq qn\]
Monotonicity illustrated

$(m, n)$ is black iff $pm \preceq qn$
Belt Theorem [JKM00, AC98]

“Every frontier lies in a belt with rational slope”.
Theorem [JKM00, AC98]

For any given OCN, $\preceq$ is an *effectively semilinear* set.
Proof of the main result

Symbolic infinite branching

Reduce \((\text{OCN} \leq \text{OCN}) \sim \to (\text{OCN} \leq \omega\text{-Net})\)
Proof of the main result

Symbolic infinite branching
Reduce \((OCN \leq OCN) \leadsto (OCN \preceq \omega\text{-Net})\)

Approximants for the new game
\[
\exists \text{ finite sequence } \preceq^0 \supseteq \preceq^1 \supseteq \preceq^2 \supseteq \cdots \supseteq \preceq^k = \preceq
\]
### Proof of the main result

<table>
<thead>
<tr>
<th>Symbolic infinite branching</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduce ((\text{OCN} \leq \text{OCN}) \sim (\text{OCN} \preceq \omega\text{-Net}))</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Approximants for the new game</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\exists) finite sequence (\preceq^0 \supseteq \preceq^1 \supseteq \preceq^2 \supseteq \cdots \supseteq \preceq^k = \preceq)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Compute approximants for finite (k)</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recursively compute (\preceq^k) by reduction to ((\text{OCN} \preceq \text{OCN}))</td>
<td></td>
</tr>
</tbody>
</table>
Proof of the main result

Symbolic infinite branching
Reduce \((OCN \leq OCN) \leadsto (OCN \leq \omega\text{-Net})\)

Approximants for the new game
\(\exists\) finite sequence \(\preceq^0 \supseteq \preceq^1 \supseteq \preceq^2 \supseteq \cdots \supseteq \preceq^k = \preceq\)

Compute approximants for finite \(k\)
Recursively compute \(\preceq^k\) by reduction to \((OCN \preceq OCN)\)
Symbolic Infinite Branching

\( \omega \)-Net \( \mathcal{N} = (Q, \text{Act}, \delta) \) with transitions

\[
\delta \subseteq Q \times \text{Act} \times \{-1, 0, 1, \omega\} \times Q
\]

...induces LTS over \( Q \times \mathbb{N} \) like OCN. A transition

introduces strong steps \( pm \overset{a, \omega}{\rightarrow} qn \) for any \( n \geq m \).
### Context

**Reduction to Strong Simulation (OCN vs. $\omega$-Net)**

### Lemma

For a OCN $N$ one can construct a OCN $M \supseteq N$ and an $\omega$-net $M' \supseteq N$ where for all configurations $pm, qn$ holds that

\[
pm \leq qn \text{ w.r.t. } N \iff pm \preceq qn \text{ w.r.t. } M, M'.
\]
Reduction to Strong Simulation (OCN vs. $\omega$-Net)

$\omega$-Countdown net

- Transition: $a, -1$
- Transition: $\tau, 0$
- Transition: $\tau, +1$
- Transition: $a, 0$
- Transition: $a, -1$
Reduction to Strong Simulation (OCN vs. $\omega$-Net)

$\omega$-Countdown net

\[ a, -1 \xrightarrow{\tau, 0} \tau, +1 \xrightarrow{a, 0} a, -1 \]

\[ a, -1 \xrightarrow{a, \omega} a, -1 \]
Reduction to Strong Simulation (OCN vs. \(\omega\)-Net)

\(\omega\)-Countdown net

\[
\begin{align*}
D & \xrightarrow{a, -1} \tau, 0 \\
C & \xrightarrow{\tau, +1} a, 0 \\
B & \xrightarrow{a, -1}
\end{align*}
\]
Reduction to Strong Simulation (OCN vs. $\omega$-Net)

$\omega$-Countdown net

\[
\begin{align*}
D & \xrightarrow{a, -1} (x, 0) & \xrightarrow{x, 0} & \xrightarrow{x, -1} D & \xrightarrow{a, -1} x, 0 & \xrightarrow{x, 0} & \xrightarrow{x, +1} a, \omega & \xrightarrow{a, \omega} x, 0 & \xrightarrow{x, 0} \\
\xrightarrow{\tau, 0} & C & \xrightarrow{a, 0} & B & \xrightarrow{a, \omega} & \xrightarrow{x, 0} & \xrightarrow{x, 0} & \xrightarrow{x, 0} & \xrightarrow{x, 0} \end{align*}
\]
Proof of the main result

Symbolic infinite branching

Reduce \((\text{OCN} \leq \text{OCN}) \leadsto (\text{OCN} \leq \omega\text{-Net})\)

Approximants for the new game

\(\exists\) finite sequence \(\leq^0 \supseteq \leq^1 \supseteq \leq^2 \supseteq \cdots \supseteq \leq^k = \leq\)

Compute approximants for finite \(k\)

Recursively compute \(\leq^k\) by reduction to \((\text{OCN} \leq \text{OCN})\)
# Proof of the main result

<table>
<thead>
<tr>
<th>Symbolic infinite branching</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduce ((\mathrm{OCN} \leq \mathrm{OCN}) \rightsquigarrow (\mathrm{OCN} \leq \omega\text{-Net}))</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Approximants for the new game</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\exists) finite sequence (\preceq^0 \supseteq \preceq^1 \supseteq \preceq^2 \supseteq \cdots \supseteq \preceq^k = \preceq)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Compute approximants for finite (k)</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recursively compute (\preceq^k) by reduction to ((\mathrm{OCN} \preceq \mathrm{OCN}))</td>
<td></td>
</tr>
</tbody>
</table>
Approximants for strong simulation (OCN vs. $\omega$-Net)

\[ \leq \beta \alpha \]
Approximants for strong simulation (OCN vs. $\omega$-Net)

$\leq^\beta \alpha$

... holds if Duplicator can guarantee to either

- survive $\alpha$ (ordinal) rounds or
- make an $\omega$-move at least $\beta$ times.
Approximants for strong simulation (OCN vs. $\omega$-Net)

\[
\preceq^\beta \alpha
\]

...holds if Duplicator can guarantee to either

- survive $\alpha$ (ordinal) rounds or
- make an $\omega$-move at least $\beta$ times.

\[
\preceq_\alpha = \bigcap_\beta \preceq^\beta \quad \preceq^\beta = \bigcap_\alpha \preceq_\alpha
\]
Approximants illustrated
Example

\((\omega \cdot 2)\)-Countdown game

\[
\begin{array}{cccc}
S & a,0 & \rightarrow & a,-1 \\
& & & a,-1 \\
D & & & a,-1 \\
& \rightarrow & \rightarrow & \rightarrow \\
& a,\omega & a,\omega & a,\omega \\
C & & & B \\
& \rightarrow & \rightarrow & \rightarrow \\
& & & a,-1 \\
B \\
\end{array}
\]
Example

(\(\omega \cdot 2\))-Countdown game

<table>
<thead>
<tr>
<th>Node</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>a, 0</td>
</tr>
<tr>
<td>D</td>
<td>a, -1</td>
</tr>
<tr>
<td>C</td>
<td>a, -1</td>
</tr>
<tr>
<td>B</td>
<td>a, -1</td>
</tr>
</tbody>
</table>

\[ S_0 \preceq^2 D_0 \]
Example

(ω · 2)-Countdown game

<table>
<thead>
<tr>
<th>State</th>
<th>Transition</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>a, 0</td>
<td>D</td>
</tr>
<tr>
<td>D</td>
<td>a, -1</td>
<td>C</td>
</tr>
<tr>
<td>C</td>
<td>a, ω</td>
<td>B</td>
</tr>
<tr>
<td>B</td>
<td>a, -1</td>
<td></td>
</tr>
</tbody>
</table>

- $S_0 \preceq^2 D_0$
- $S_0 \preceq_{\omega \cdot 2} D_0$
### Example

\((\omega \cdot 2)\)-Countdown game

<table>
<thead>
<tr>
<th>State</th>
<th>Action</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>a, 0</td>
<td>D</td>
</tr>
<tr>
<td>D</td>
<td>a, -1</td>
<td>C</td>
</tr>
<tr>
<td>C</td>
<td>a, ω</td>
<td>B</td>
</tr>
</tbody>
</table>

- \(S_0 \preceq^2 D_0\)
- \(S_0 \preceq_{\omega \cdot 2} D_0\)
- \(S_0 \preceq^3_{\omega \cdot 2+1} D_0\)
Example

(ω · 2)-Countdown game

\[
\begin{align*}
S & \quad (a, 0) \quad S \\
D & \quad (a, -1) \quad \quad (a, \omega) \quad D \\
C & \quad (a, -1) \quad \quad (a, \omega) \quad C \\
B & \quad (a, -1) \quad \quad \text{End} \\
\end{align*}
\]

- \( S_0 \preceq^2 D_0 \)
- \( S_0 \preceq^\omega 2 D_0 \)
- \( S_0 \preceq^3 D_0 \)
- \( S_0 \preceq^\omega 2+1 D_0 \)
Example

(\(\omega \cdot 2\))-Countdown game

\[
\begin{align*}
S &: a, 0 \\
D &: a, -1 \quad a, \omega \\
C &: a, -1 \quad a, \omega \\
B &: a, -1
\end{align*}
\]

- \(S0 \preceq^2 D0\)
- \(S0 \preceq^\omega 2 D0\)
- \(S0 \preceq^\omega 2+1 D0\)
- \(S0 \not\preceq^3 D0\)
- \(\preceq = \preceq^3\)
Example

(ω · 2)-Countdown game

\[
\begin{align*}
S & \overset{a, 0}{\rightarrow} a, -1 & D & \overset{a, \omega}{\rightarrow} a, -1 & C & \overset{a, \omega}{\rightarrow} B
\end{align*}
\]

- \( S_0 \preceq^2 D_0 \)
- \( S_0 \preceq^\omega D_0 \)
- \( S_0 \preceq^\omega 2 + 1 D_0 \)

Lemma

For any OCN \( N \) and \( \omega \)-Net \( M \), there is \( k \in \mathbb{N} \) such that

\( \preceq = \preceq^k \)
Proof of the main result

Symbolic infinite branching
Reduce \((OCN \leq OCN) \leadsto (OCN \preceq \omega\text{-Net})\)

Approximants for the new game
\[
\exists \text{ finite sequence } \preceq^0 \supseteq \preceq^1 \supseteq \preceq^2 \supseteq \cdots \supseteq \preceq^k = \preceq
\]

Compute approximants for finite \(k\)
Recursively compute \(\preceq^k\) by reduction to \((OCN \preceq OCN)\)
**Proof of the main result**

**Symbolic infinite branching**
Reduce \((OCN \equiv OCN) \rightsquigarrow (OCN \leq \omega\text{-Net})\)

**Approximants for the new game**
\[\exists \text{ finite sequence } \preceq^0 \supseteq \preceq^1 \supseteq \preceq^2 \supseteq \cdots \supseteq \preceq^k = \preceq\]

**Compute approximants for finite } k**
Recursively compute } \preceq^k \text{ by reduction to } (OCN \preceq OCN)
### Observation

If a response via $\downarrow^\omega$ leads to (game) position $pm \not\trianglelefteq^k qn$ then $pm \not\trianglelefteq^k qn'$ for all $n' \in \mathbb{N}$. 
Observation

If a response via $\rightarrow_\omega$ leads to (game) position $pm \not\preceq^k qn$ then $pm \not\preceq^k qn'$ for all $n' \in \mathbb{N}$.

For any pair $p, q$ of states there is a minimal sufficient value $m$ with

$$pm \not\preceq^k qn \text{ for all } n$$
Computing $\leq^{k+1}$

- Compute minimal sufficient values $\in \mathbb{N} \cup \{\infty\}$ for all $(p, q)$
- Build gadget nets that test if Spoiler’s counter is sufficient.
Compute minimal sufficient values $\in \mathbb{N} \cup \{\infty\}$ for all $(p, q)$.

- Build gadget nets that test if Spoiler’s counter is sufficient.
- Use *Defenders Forcing* to substitute $\omega$-transitions by the ability to move into testing gadgets.
Compute minimal sufficient values $\in \mathbb{N} \cup \{\infty\}$ for all $(p, q)$.

- Build gadget nets that test if Spoiler’s counter is sufficient.
- Use *Defenders Forcing* to substitute $\omega$-transitions by the ability to move into testing gadgets.

$\Rightarrow$ Strong simulation game OCN vs. OCN.
### Proof of the main result

<table>
<thead>
<tr>
<th>Symbolic infinite branching</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduce $(OCN \leq OCN) \leadsto (OCN \leq \omega\text{-Net})$</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Approximants for the new game</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\exists$ finite sequence $\leq^0 \supseteq \leq^1 \supseteq \leq^2 \supseteq \cdots \supseteq \leq^k = \leq$</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Compute approximants for finite $k$</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recursively compute $\leq^k$ by reduction to $(OCN \leq OCN)$</td>
<td></td>
</tr>
</tbody>
</table>
Conclusion

- Weak Simulation is decidable for One-Counter Nets
- Our proof crucially depends on monotonicity! We
  - symbolically capture $\infty$ branching,
  - derive finite sequence of approximants and
  - use semilinearity of $OCN \preceq OCN$ to compute approximants and check convergence.
- We also consider (weak) trace inclusion for OCN and (weak) Simulation between OCN and NFA.


