Preorder-Constrained Simulation

(Early Idea)

Koko Muroya[1], Takahiro Sanada[1] & Natsuki Urabe[2]

[1] RIMS, Kyoto University, Japan
[2] National Institute of Informatics, Japan

Outline

- Overview
- Preorder-Constrained Simulation without up-to
- Preorder-Constrained Simulation with up-to
- Conclusion and Future Work

Simulation

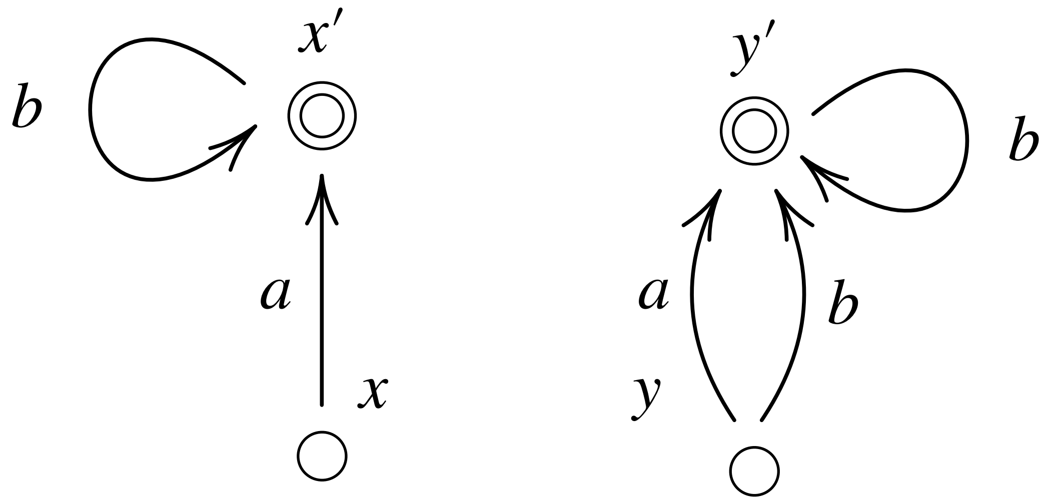
for behavioral **equivalence:**
bisimulation

- Step-wise formalization for behavioral inclusion
- Useful for proving trace inclusion

Simulation

for behavioral **equivalence:**
bisimulation

- Step-wise formalization for behavioral inclusion
- Useful for proving trace inclusion
- Example:



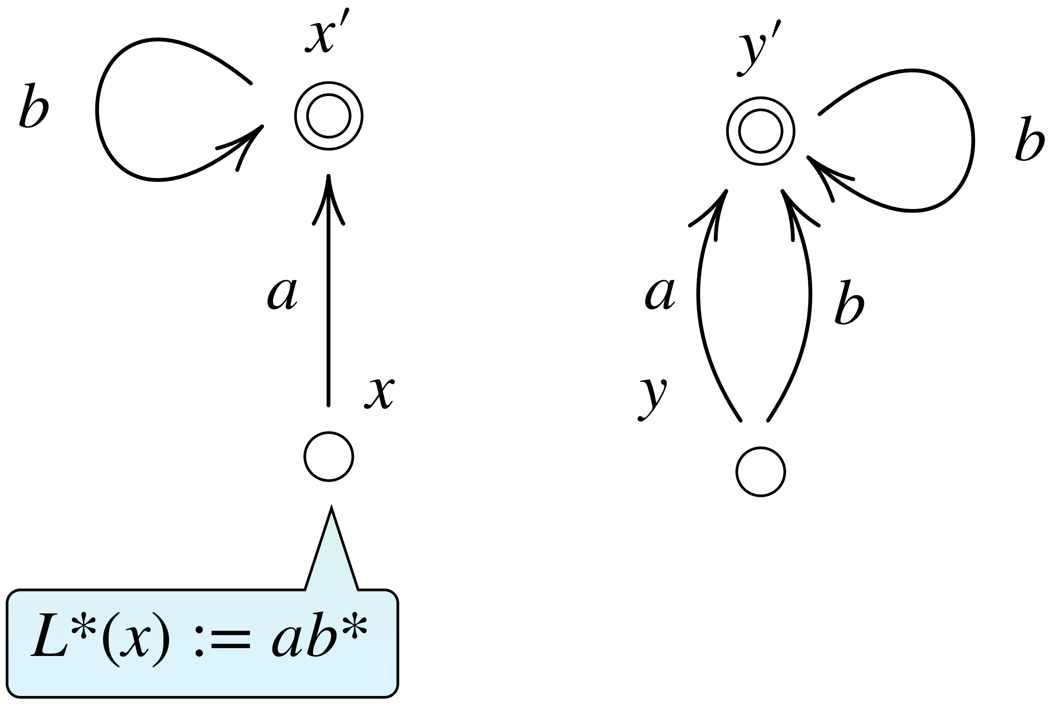
Simulation

for behavioral **equivalence:**
bisimulation

- Step-wise formalization for behavioral inclusion

- Useful for proving trace inclusion

- Example:



$L^*(x) := ab^*$

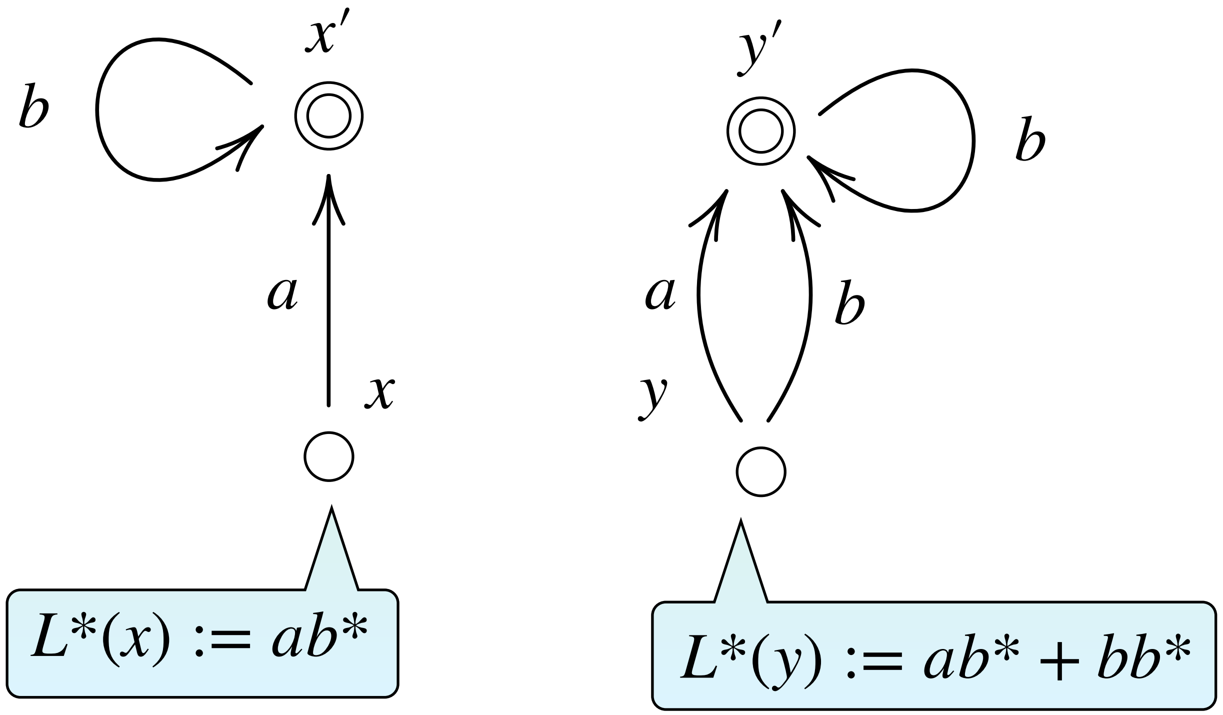
Simulation

for behavioral **equivalence:**
bisimulation

- Step-wise formalization for behavioral inclusion

- Useful for proving trace inclusion

- Example:



$L^*(x) := ab^*$

$L^*(y) := ab^* + bb^*$

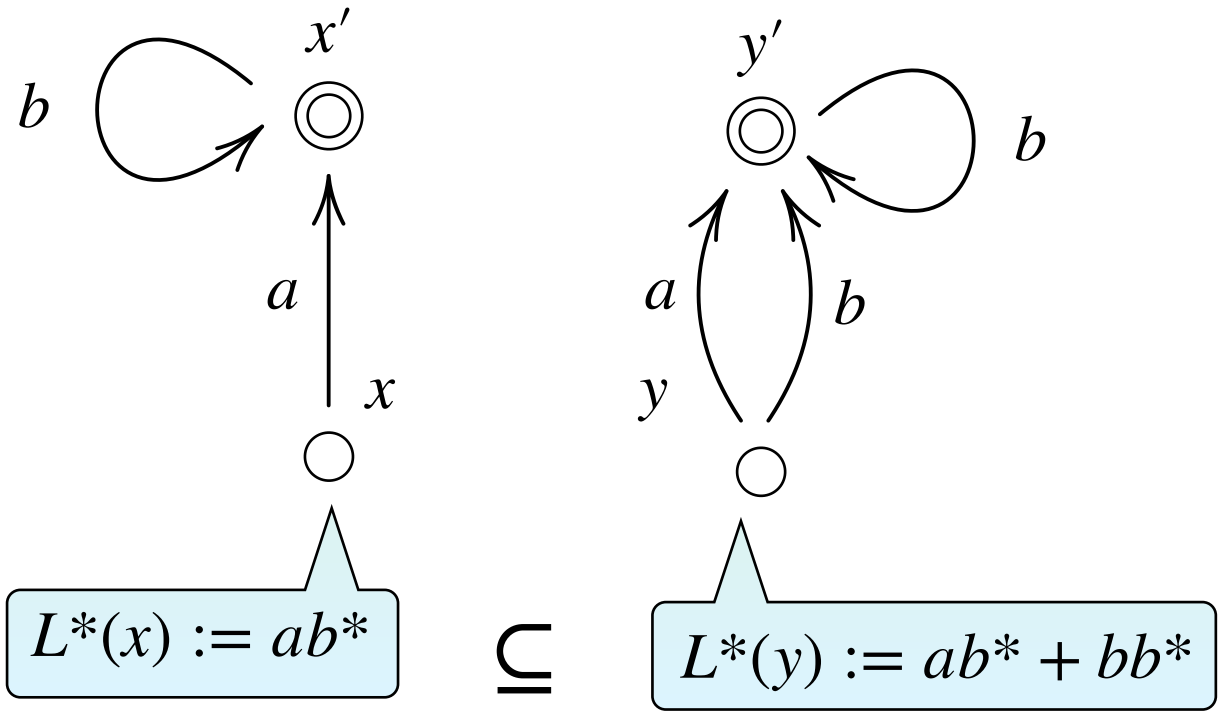
Simulation

for behavioral **equivalence:**
bisimulation

- Step-wise formalization for behavioral inclusion

- Useful for proving trace inclusion

- Example:



$L^*(x) := ab^*$ $\subseteq$ $L^*(y) := ab^* + bb^*$

trace inclusion (language inclusion)

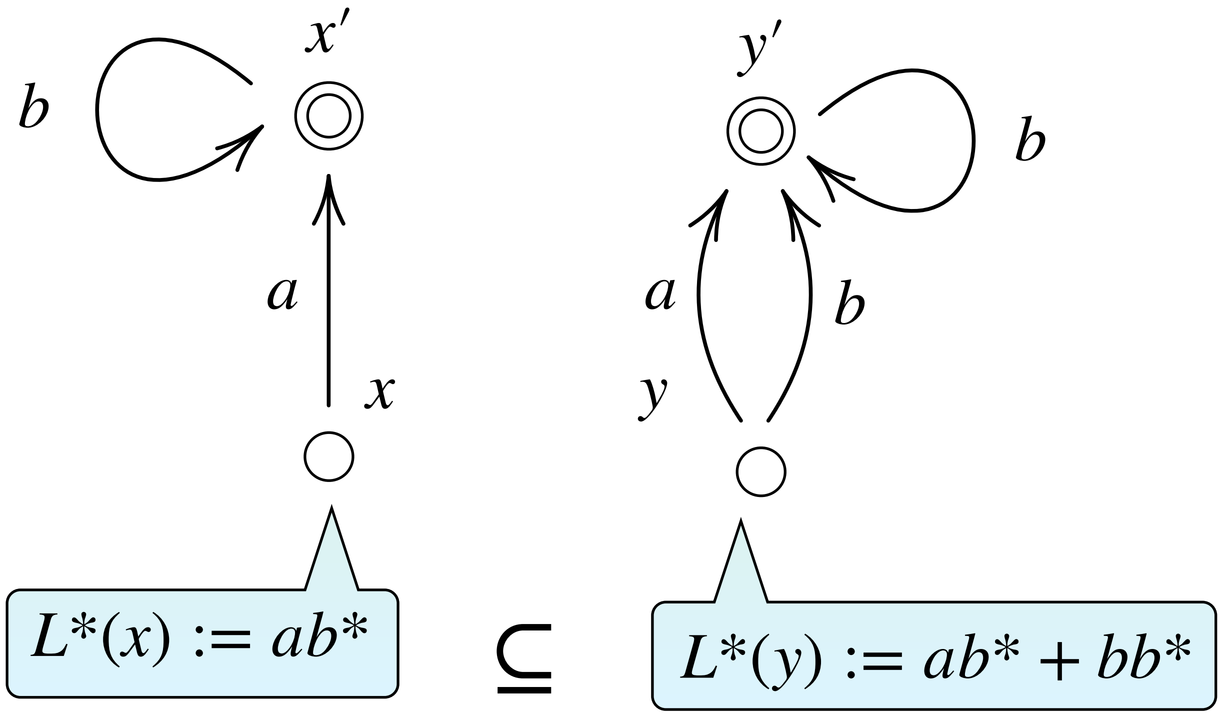
Simulation

for behavioral **equivalence:**
bisimulation

- Step-wise formalization for behavioral inclusion

- Useful for proving trace inclusion

- Example:



$L^*(x) := ab^*$ $\subseteq$ $L^*(y) := ab^* + bb^*$

PSPACE-complete

trace inclusion (language inclusion)

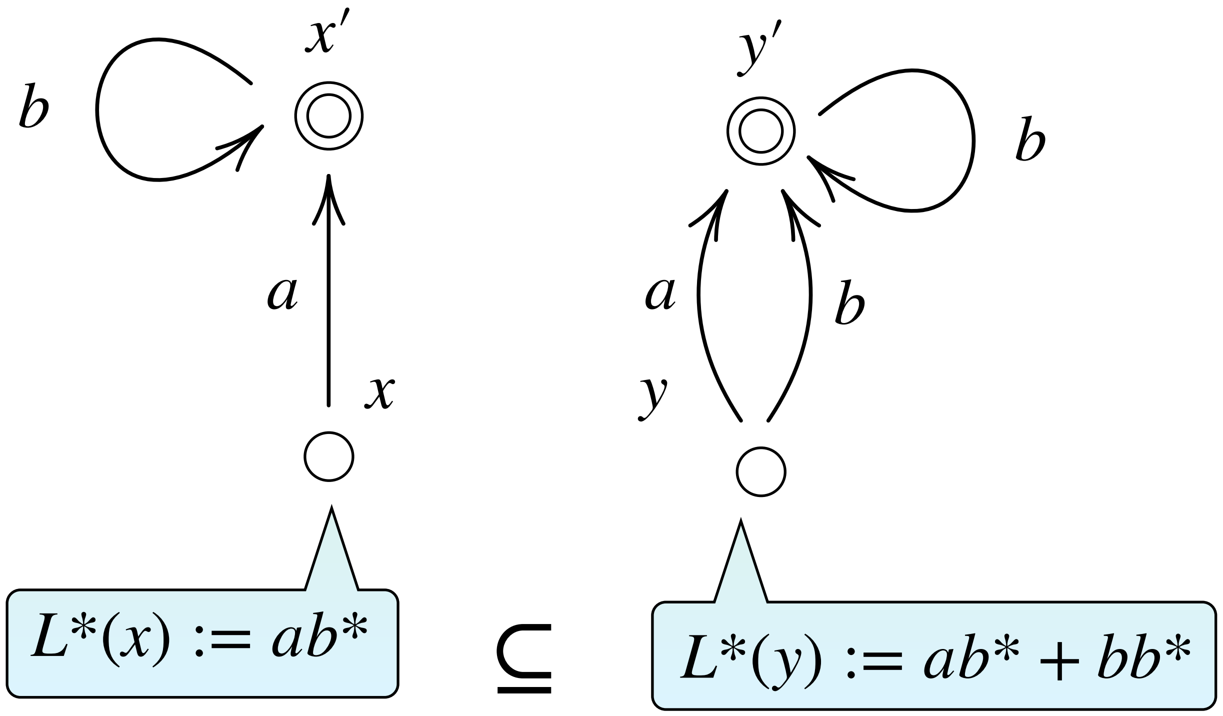
Forward Simulation [Lynch & Vaandrager, ’95]

Definition:

A ***forward simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$ and
- $\forall (a, x') \in c(x)\,.\ \exists y'$ s.t. $(a, y') \in d(y)\,.\ x' R y'$



PSPACE-complete

trace inclusion

Forward Simulation [Lynch & Vaandrager, '95]

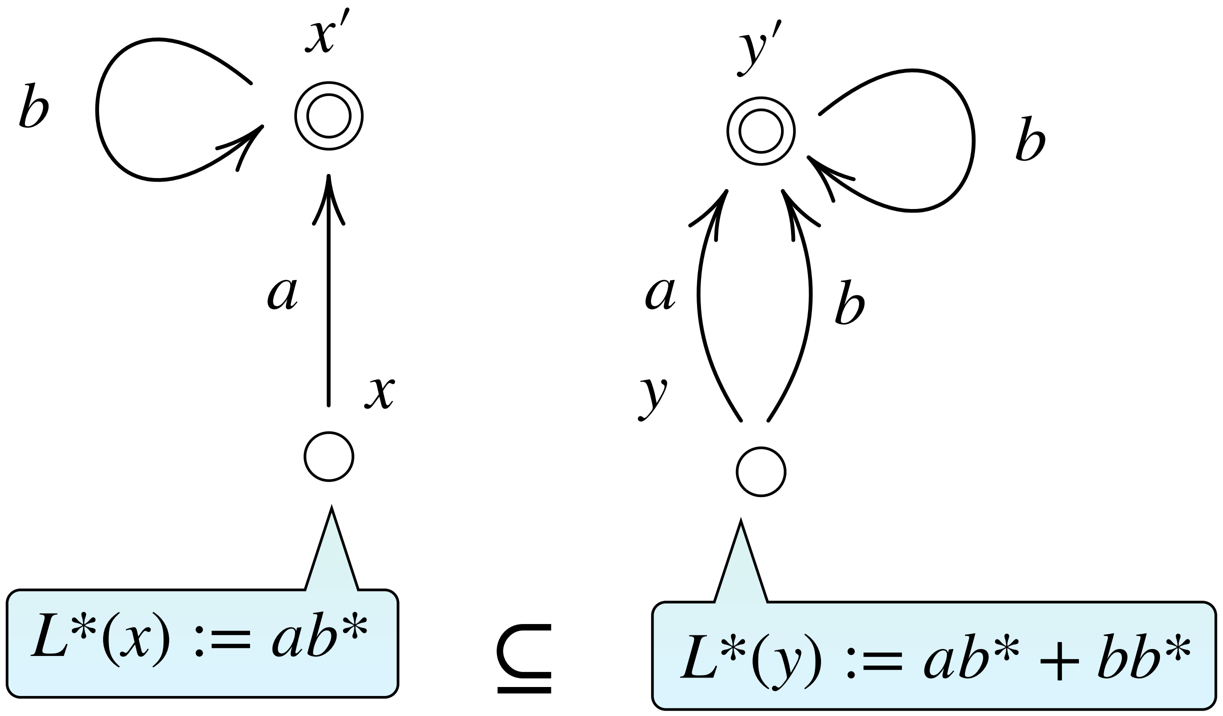
Definition:

A ***forward simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$ and
- $\forall (a, x') \in c(x)\,.\, \exists y'$ s.t. $(a, y') \in d(y)\,.\, x'Ry'$

$$x \overset{R}{\dashrightarrow} y$$



$L^*(x) := ab^*$ $\subseteq$ $L^*(y) := ab^* + bb^*$

trace inclusion

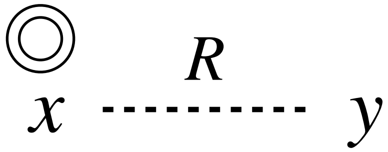
PSPACE-complete

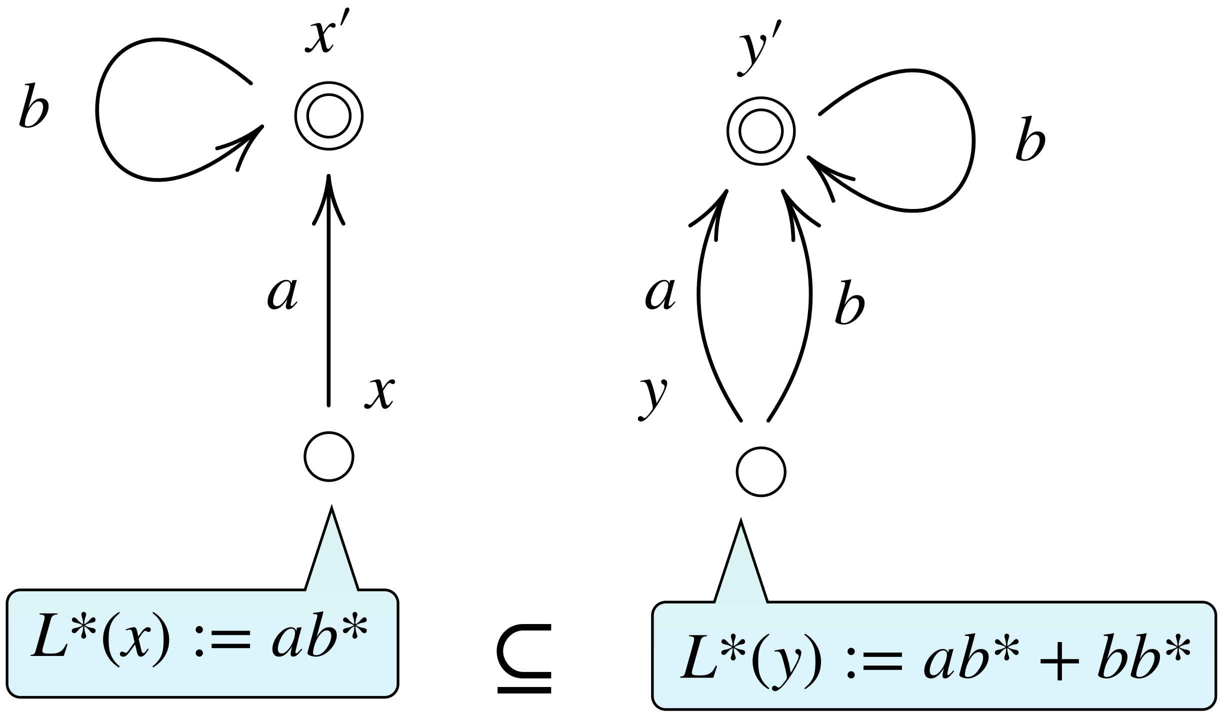
Forward Simulation [Lynch & Vaandrager, '95]

Definition:

A ***forward simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$
- $x \in F_1 \implies y \in F_2$ and
- $\forall (a, x') \in c(x)\,.\ \exists y'$ s.t. $(a, y') \in d(y)\,.\ x' R y'$

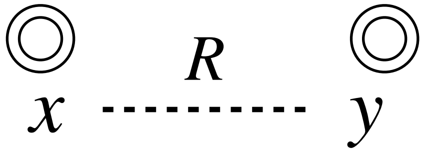




trace inclusion

PSPACE-complete

Forward Simulation [Lynch & Vaandrager, ’95]

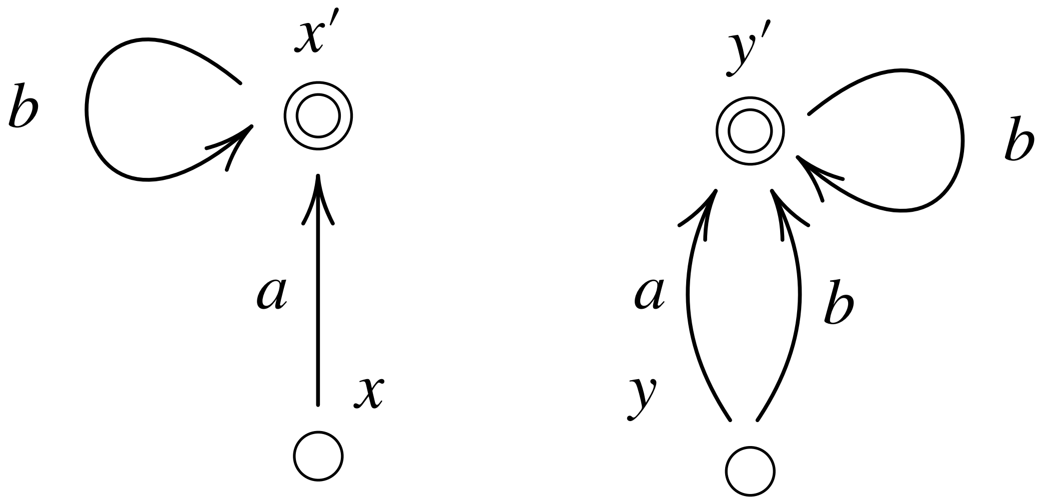


Definition:

A ***forward simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$ and
- $\forall (a, x') \in c(x)\,.\ \exists y' \text{ s.t. } (a, y') \in d(y)\,.\ x' R y'$



$L^*(x) := ab^*$ $\subseteq$ $L^*(y) := ab^* + bb^*$

trace inclusion

PSPACE-complete

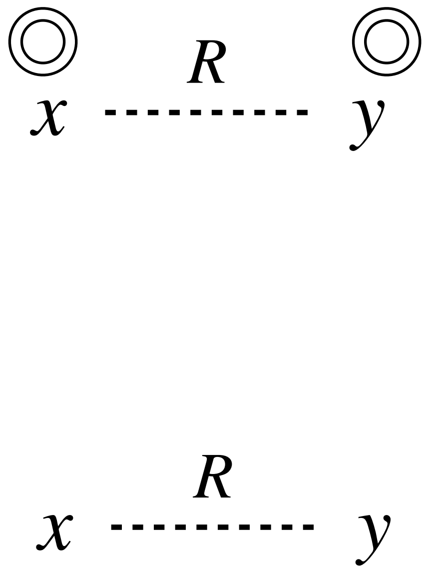
Forward Simulation [Lynch & Vaandrager, '95]

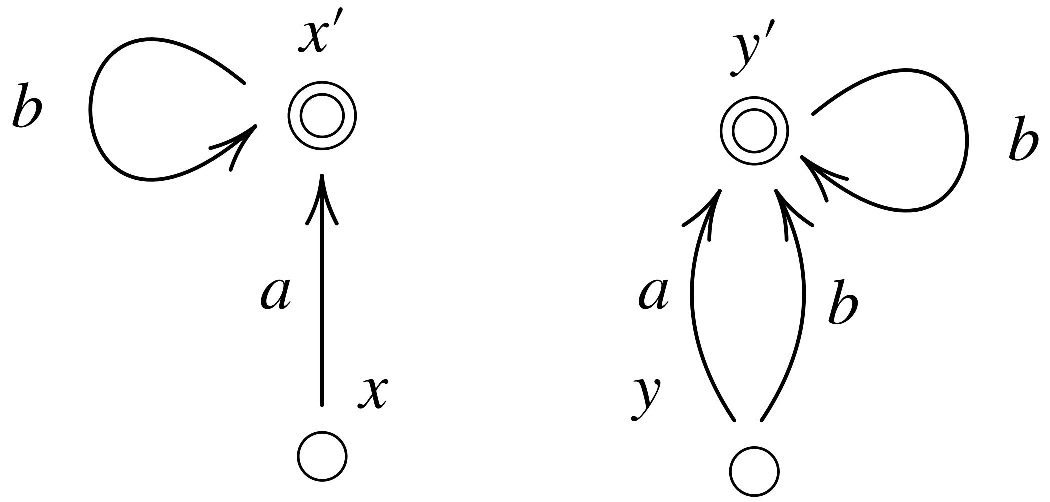
Definition:

A ***forward simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$ and
- $\forall (a, x') \in c(x)\,.\ \exists y'$ s.t. $(a, y') \in d(y)\,.\ x' R y'$





$L^*(x) := ab^*$ $\subseteq$ $L^*(y) := ab^* + bb^*$

PSPACE-complete

trace inclusion

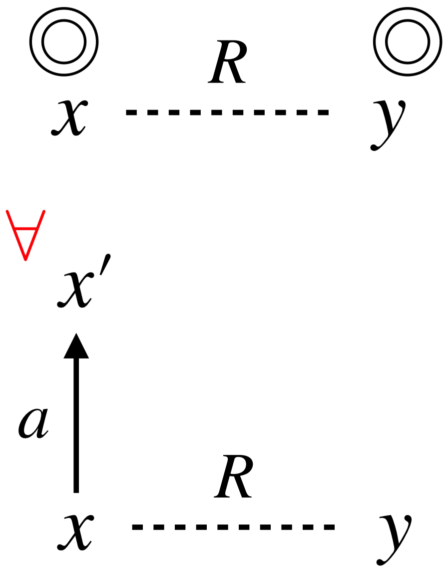
Forward Simulation [Lynch & Vaandrager, ’95]

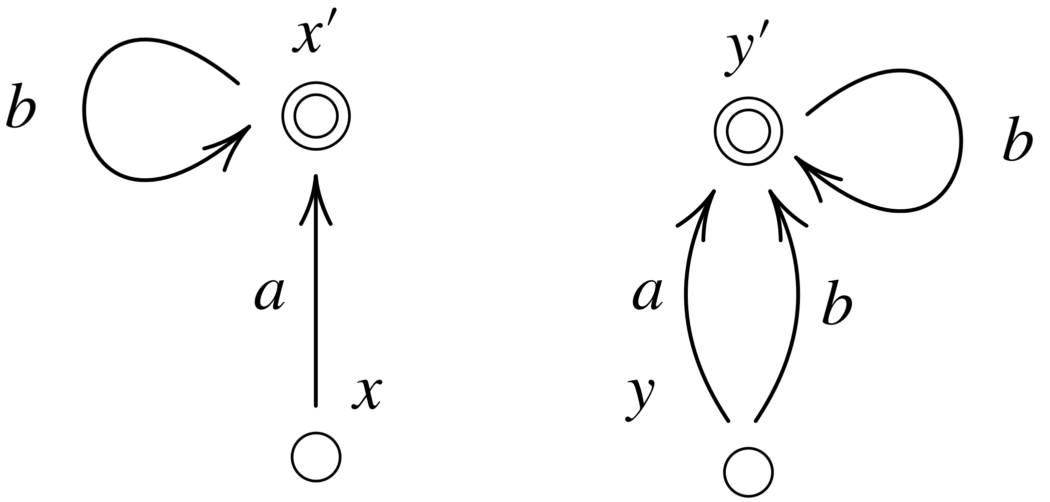
Definition:

A ***forward simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$ and
- $\forall (a, x') \in c(x)\,.\ \exists y'$ s.t. $(a, y') \in d(y)\,.\ x' R y'$





$L^*(x) := ab^*$ $\subseteq$ $L^*(y) := ab^* + bb^*$

trace inclusion

PSPACE-complete

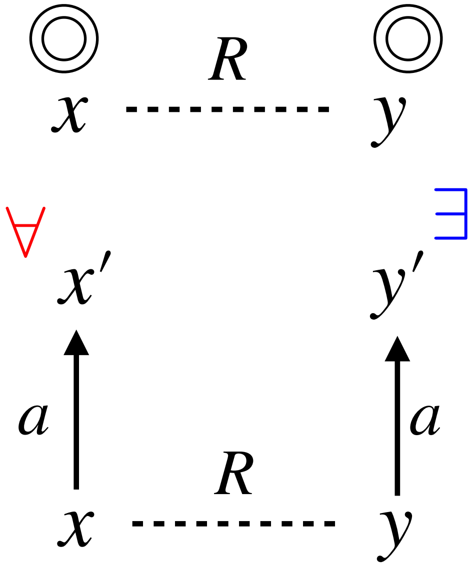
Forward Simulation [Lynch & Vaandrager, '95]

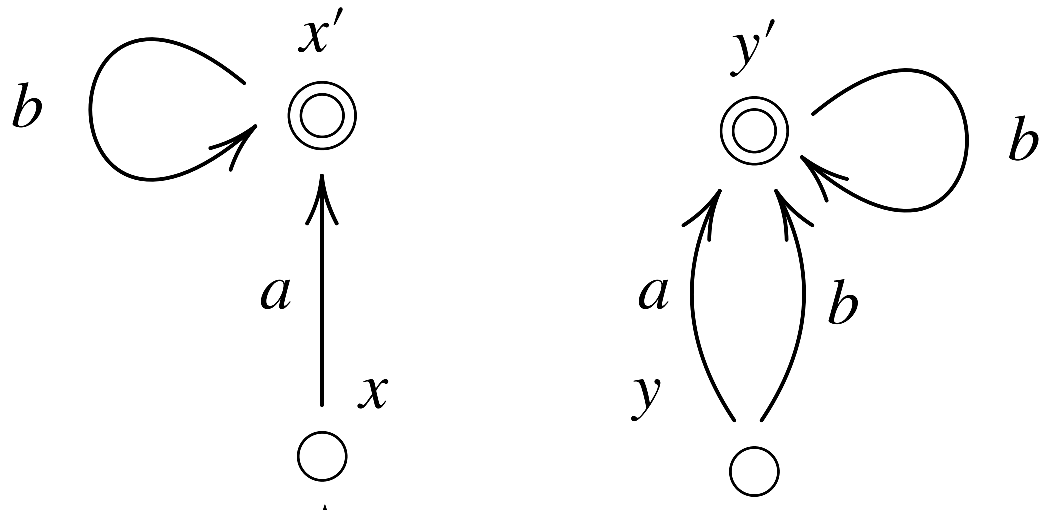
Definition:

A ***forward simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$ and
- $\forall (a, x') \in c(x)\,.\ \exists y'$ s.t. $(a, y') \in d(y)\,.\ x' R y'$





$L^*(x) := ab^*$ $\subseteq$ $L^*(y) := ab^* + bb^*$

PSPACE-complete

trace inclusion

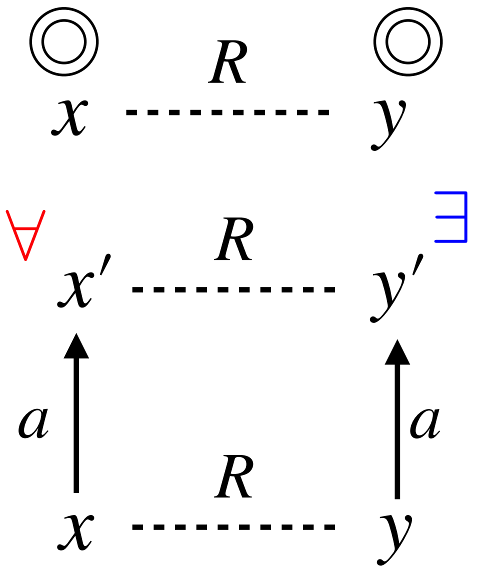
Forward Simulation [Lynch & Vaandrager, ’95]

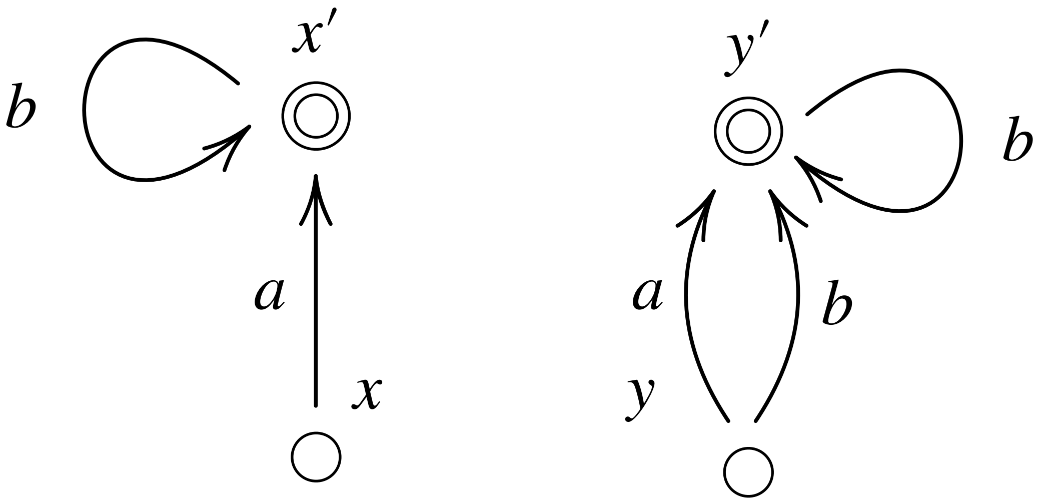
Definition:

A ***forward simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$ and
- $\forall (a, x') \in c(x)\,.\ \exists y'$ s.t. $(a, y') \in d(y)\,.\ x' R y'$





$L^*(x) := ab^*$ $\subseteq$ $L^*(y) := ab^* + bb^*$

trace inclusion

PSPACE-complete

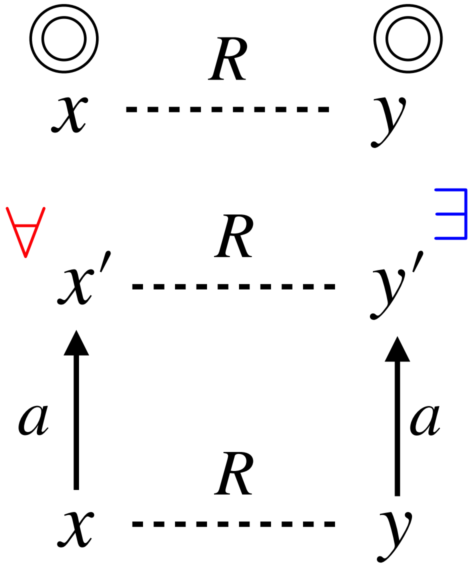
Forward Simulation [Lynch & Vaandrager, '95]

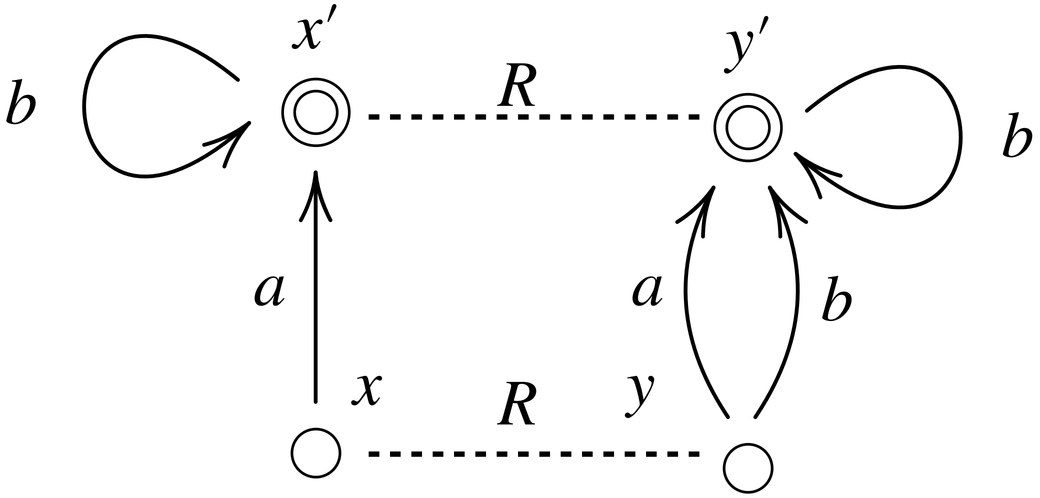
Definition:

A ***forward simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$ and
- $\forall (a, x') \in c(x)\,.\ \exists y'$ s.t. $(a, y') \in d(y)\,.\ x' R y'$





$L^*(x) := ab^*$ $\subseteq$ $L^*(y) := ab^* + bb^*$

PSPACE-complete

trace inclusion

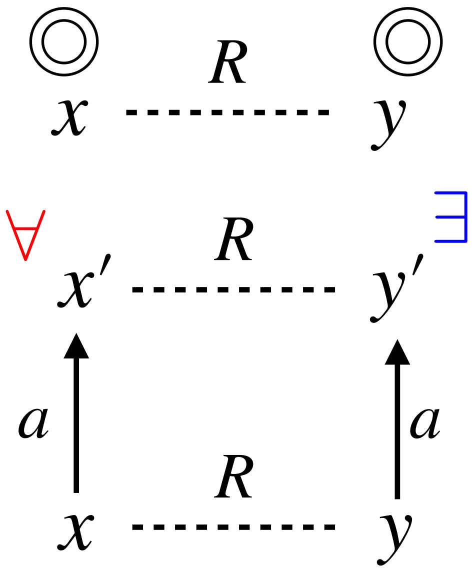
Forward Simulation [Lynch & Vaandrager, '95]

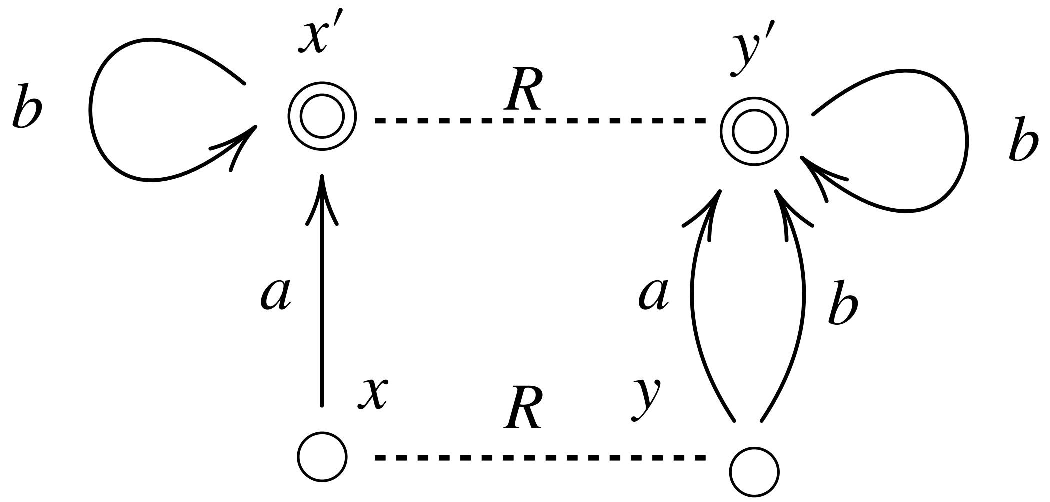
Definition:

A ***forward simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x,y) \in R\,.$

- $x \in F_1 \implies y \in F_2$ and
- $\forall (a,x') \in c(x)\,.\ \exists y'$ s.t. $(a,y') \in d(y)\,.\ x'Ry'$





$L^*(x) := ab^*$ $\subseteq$ $L^*(y) := ab^* + bb^*$

trace inclusion

Theorem (**soundness**):

If R is a forward simulation,

$$xRy \implies L^*(x) \subseteq L^*(y)$$

PSPACE-complete

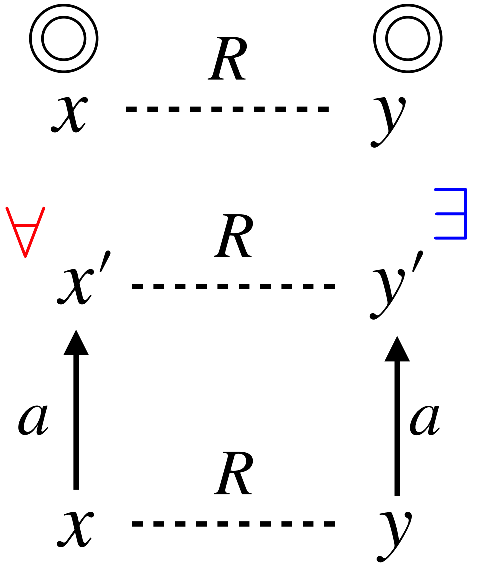
Forward Simulation [Lynch & Vaandrager, '95]

Definition:

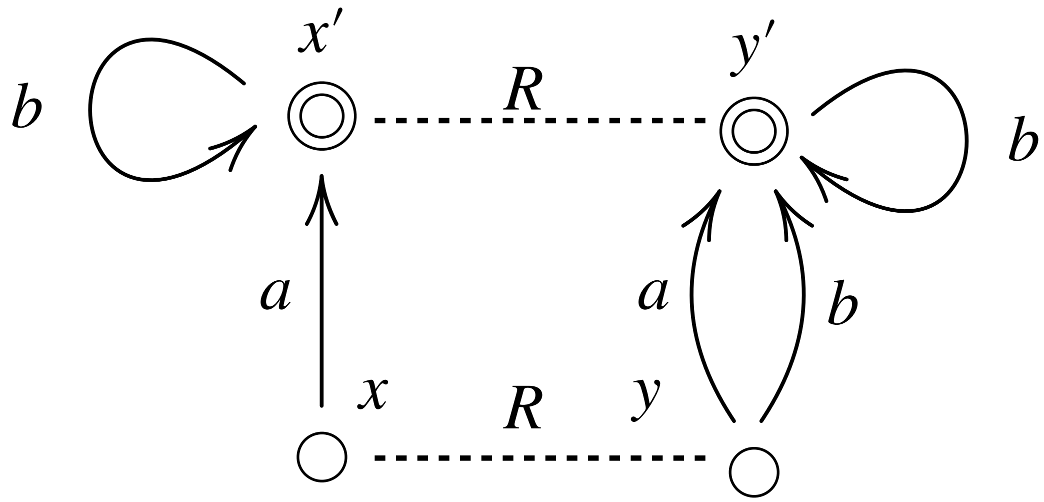
A ***forward simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$ and
- $\forall (a, x') \in c(x)\,.\, \exists y'$ s.t. $(a, y') \in d(y)\,.\, x' R y'$



in P



$L^*(x) := ab^*$ $\subseteq$ $L^*(y) := ab^* + bb^*$

trace inclusion

Theorem (**soundness**):

If R is a forward simulation,

$$xRy \implies L^*(x) \subseteq L^*(y)$$

PSPACE-complete

Various Simulation Notions

- Simulation up-to
- Weak Simulation
- Improvement
- Preorder-Constrained Simulation

Simulation up-to & Weak Simulation

Simulation up-to

- Enhancement with prior knowledge on trace inclusion

$$\begin{tikzcd}
x' & & y' \\
x \arrow[u, "a"] \arrow[rr, dash, dashed, "R"] & & y \arrow[u, "a"']
\end{tikzcd}$$

Simulation up-to & Weak Simulation

Simulation up-to

- Enhancement with prior knowledge on trace inclusion

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "L^*(x') \subseteq L^*(x'')"] & x'' & y'' \arrow[r, dash, dashed, "L^*(y') \subseteq L^*(y'')"] & y' \\
x \arrow[u, "a"] \arrow[rrr, dash, dashed, "R"] & & & y \arrow[u, "a"']
\end{tikzcd}$$

Simulation up-to & Weak Simulation

Simulation up-to

- Enhancement with prior knowledge on trace inclusion

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "L^*(x') \subseteq L^*(x'')"] & x'' \arrow[r, dash, dashed, "R"] & y'' \arrow[r, dash, dashed, "L^*(y') \subseteq L^*(y'')"] & y' \\
x \arrow[u, "a"] \arrow[rrr, dash, dashed, "R"] & & & y \arrow[u, "a"']
\end{tikzcd}$$

Simulation up-to & Weak Simulation

Simulation up-to

- Enhancement with prior knowledge on trace inclusion

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "L^*(x') \subseteq L^*(x'')"] & x'' \arrow[r, dash, dashed, "R"] & y'' \arrow[r, dash, dashed, "L^*(y') \subseteq L^*(y'')"] & y' \\
x \arrow[u, "a"] \arrow[rrr, dash, dashed, "R"] & & & y \arrow[u, "a"']
\end{tikzcd}
\qquad \Longrightarrow \qquad L^*(x) \subseteq L^*(y)$$

Simulation up-to & Weak Simulation

Simulation up-to

- Enhancement with prior knowledge on trace inclusion

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "L^*(x') \subseteq L^*(x'')"] & x'' \arrow[r, dash, dashed, "R"] & y'' \arrow[r, dash, dashed, "L^*(y') \subseteq L^*(y'')"] & y' \\
x \arrow[u, "a"] \arrow[rrr, dash, dashed, "R"] & & & y \arrow[u, "a"']
\end{tikzcd}$$

$$\Longrightarrow \qquad L^*(x) \subseteq L^*(y)$$

Weak simulation

- Simulation for systems with **silent moves,** e.g. $c : X \to \mathscr{P}((\{\tau\} + \Sigma) \times X)$

silent letter

Simulation up-to & Weak Simulation

Simulation up-to

- Enhancement with prior knowledge on trace inclusion

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "L^*(x') \subseteq L^*(x'')"] & x'' \arrow[r, dash, dashed, "R"] & y'' \arrow[r, dash, dashed, "L^*(y') \subseteq L^*(y'')"] & y' \\
x \arrow[u, "a"] \arrow[rrr, dash, dashed, "R"] & & & y \arrow[u, "a"']
\end{tikzcd}
\qquad \Longrightarrow \qquad L^*(x) \subseteq L^*(y)$$

Weak simulation

- Simulation for systems with **silent moves,** e.g. $c : X \to \mathscr{P}((\{\tau\} + \Sigma) \times X)$

silent letter

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, "\tau"] \arrow[r, dash, dashed, "R"] & y \arrow[u, dotted, "\tau^*"']
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
 & {} \arrow[u, dotted, "\tau^*"'] \\
 & {} \arrow[u, "a"'] \\
x \arrow[uuu, "a"] \arrow[r, dash, dashed, "R"] & y \arrow[u, dotted, "\tau^*"']
\end{tikzcd}$$

Simulation up-to & Weak Simulation

Simulation up-to

- Enhancement with prior knowledge on trace inclusion

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "L^*(x') \subseteq L^*(x'')"] & x'' \arrow[r, dash, dashed, "R"] & y'' \arrow[r, dash, dashed, "L^*(y') \subseteq L^*(y'')"] & y' \\
x \arrow[u, "a"] \arrow[rrr, dash, dashed, "R"] & & & y \arrow[u, "a"']
\end{tikzcd}$$

$$\Longrightarrow \qquad L^*(x) \subseteq L^*(y)$$

Weak simulation

- Simulation for systems with **silent moves,** e.g. $c : X \to \mathscr{P}((\{\tau\} + \Sigma) \times X)$

silent letter

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, "\tau"] \arrow[r, dash, dashed, "R"] & y \arrow[u, dotted, "\tau^*"']
\end{tikzcd}
\qquad
\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
 & \cdot \arrow[u, dotted, "\tau^*"'] \\
 & \cdot \arrow[u, "a"'] \\
x \arrow[uuu, "a"] \arrow[r, dash, dashed, "R"] & y \arrow[u, dotted, "\tau^*"']
\end{tikzcd}$$

$$\Longrightarrow \qquad \text{remove}_\tau(L^*(x)) \subseteq \text{remove}_\tau(L^*(y))$$

weak trace inclusion

Simulation up-to & Weak Simulation

Simulation up-to

- Enhancement with prior knowledge on trace inclusion

$L^*(x') \subseteq L^*(x'')$ $\quad$ $L^*(y') \subseteq L^*(y'')$

$$\begin{tikzcd}
x' \arrow[r, dash, dashed] & x'' \arrow[r, dash, dashed, "R"] & y'' \arrow[r, dash, dashed] & y' \\
x \arrow[u, "a"] \arrow[rrr, dash, dashed, "R"] & & & y \arrow[u, "a"']
\end{tikzcd}$$

$$\Longrightarrow \qquad L^*(x) \subseteq L^*(y)$$

Weak simulation

- Simulation for systems with **silent moves,** e.g. $c : X \to \mathscr{P}((\{\tau\} + \Sigma) \times X)$

silent letter

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, "\tau"] \arrow[r, dash, dashed, "R"] & y \arrow[u, dotted, "\tau^*"']
\end{tikzcd}
\qquad
\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
 & \bullet \arrow[u, dotted, "\tau^*"'] \\
 & \bullet \arrow[u, "a"'] \\
x \arrow[uuu, "a"] \arrow[r, dash, dashed, "R"] & y \arrow[u, dotted, "\tau^*"']
\end{tikzcd}$$

$$\Longrightarrow \qquad \mathrm{remove}_\tau(L^*(x)) \subseteq \mathrm{remove}_\tau(L^*(y))$$

weak trace inclusion

Naive combination of weak and up-to is NOT sound

Simulation up-to & Weak Simulation

Simulation up-to

- Enhancement with prior knowledge on trace inclusion

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "L^*(x') \subseteq L^*(x'')"] & x'' \arrow[r, dash, dashed, "R"] & y'' \arrow[r, dash, dashed, "L^*(y') \subseteq L^*(y'')"] & y' \\
x \arrow[u, "a"] \arrow[rrr, dash, dashed, "R"] & & & y \arrow[u, "a"']
\end{tikzcd}
\quad \Longrightarrow \quad L^*(x) \subseteq L^*(y)$$

Weak simulation

- Simulation for systems with **silent moves,** e.g. $c : X \to \mathscr{P}((\{\tau\} + \Sigma) \times X)$

silent letter

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, "\tau"] \arrow[r, dash, dashed, "R"] & y \arrow[u, dotted, "\tau^*"']
\end{tikzcd}
\qquad
\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
 & \cdot \arrow[u, dotted, "\tau^*"'] \\
 & \cdot \arrow[u, "a"'] \\
x \arrow[uuu, "a"] \arrow[r, dash, dashed, "R"] & y \arrow[u, dotted, "\tau^*"']
\end{tikzcd}
\quad \Longrightarrow \quad
\begin{array}{l} \mathrm{remove}_\tau(L^*(x)) \\ \quad \subseteq \mathrm{remove}_\tau(L^*(y)) \end{array}$$

weak trace inclusion

Naive combination of weak and up-to is NOT sound

→ need special care, e.g. [Pous, '05]

Simulations in Program Semantics Literature

$$e \to^{n} w \not\to \implies \exists n' : n \geq n' .\ e' \to^{n'} w' \not\to$$

Improvement [Accattoli, Dal Lago & Vanoni, 2020]

- (Bi)simulation for comparing **reduction lengths** of λ-terms
- Automata-theoretically: for deterministic & unlabeled systems $(c : X \to X, F_1 \subseteq X)$
- Simulate multiple steps with multiple steps

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, dotted, "\exists\, m > 0 \text{ steps}"] \arrow[r, dash, dashed, "R"'] \arrow[ur, phantom, "m \geq n"] & y \arrow[u, dotted, "n \text{ steps}\ \exists"']
\end{tikzcd}$$

Simulations in Program Semantics Literature

Improvement [Accattoli, Dal Lago & Vanoni, 2020]

$e \to^n w \not\to \implies \exists n' : n \geq n' .\, e' \to^{n'} w' \not\to$

- (Bi)simulation for comparing **reduction lengths** of λ-terms
- Automata-theoretically: for deterministic & unlabeled systems $(c : X \to X, F_1 \subseteq X)$
- Simulate multiple steps with multiple steps

$$\begin{tikzcd}
x' \arrow[rr, dash, dashed, "R"] & & y' \\
& m \geq n & \\
x \arrow[uu, dotted, "\exists\, m > 0 \text{ steps}"] \arrow[rr, dash, dashed, "R"] & & y \arrow[uu, dotted, "n \text{ steps}\ \exists"']
\end{tikzcd}$$

$e \to^n w \not\to \implies \exists n' : n Q n' .\, e' \to^{n'} w' \not\to$

Preorder-constrained simulation [Muroya, PhD thesis, 2020]

≅ (One-directional) improvement + up-to + generalization

(+ some restriction)

- Parameterized by a preorder $Q \subseteq \mathbb{N} \times \mathbb{N}$ closed under addition

(i.e. $iQi' \wedge jQj' \implies (i+j)Q(i'+j')$)

$$\begin{tikzcd}
x' \arrow[r, dash, dashed] & x'' \arrow[rr, dash, dashed, "R"] & & y'' \arrow[r, dash, dashed] & y' \\
& & mQn & & \\
x \arrow[uu, dotted, "m > 0"] \arrow[rrrr, dash, dashed, "R"] & & & & y \arrow[uu, dotted, "n"']
\end{tikzcd}$$

Simulations in Program Semantics Literature

$$e \to^{n} w \not\to \implies \exists n' : n \geq n' .\ e' \to^{n'} w' \not\to$$

Improvement [Accattoli, Dal Lago & Vanoni, 2020]

- (Bi)simulation for comparing **reduction lengths** of λ-terms
- Automata-theoretically: for deterministic & unlabeled systems $(c : X \to X, F_1 \subseteq X)$

  $\cong c : X \to \{\tau\} \times X$

  variant of weak simulation
- Simulate multiple steps with multiple steps

$$\begin{tikzcd} x' \arrow[r, dash, dashed, "R"] & y' \\ x \arrow[u, dotted, "\exists\; m > 0 \text{ steps}"] \arrow[r, dash, dashed, "R"] \arrow[ur, phantom, "m \geq n"] & y \arrow[u, dotted, "n \text{ steps}\; \exists"'] \end{tikzcd}$$

$$e \to^{n} w \not\to \implies \exists n' : n Q n' .\ e' \to^{n'} w' \not\to$$

Preorder-constrained simulation [Muroya, PhD thesis, 2020]

≅ (One-directional) improvement + up-to + generalization

(+ some restriction)

- Parameterized by a preorder $Q \subseteq \mathbb{N} \times \mathbb{N}$ closed under addition

(i.e. $iQi' \wedge jQj' \implies (i+j)Q(i'+j')$)

$$\begin{tikzcd} x' \arrow[r, dash, dashed] & x'' \arrow[r, dash, dashed, "R"] & y'' \arrow[r, dash, dashed] & y' \\ x \arrow[u, dotted, "m > 0"] \arrow[rrr, dash, dashed, "R"] \arrow[urrr, phantom, "mQn"] & & & y \arrow[u, dotted, "n"'] \end{tikzcd}$$

Simulations in Program Semantics Literature

Improvement [Accattoli, Dal Lago & Vanoni, 2020]

$$e \to^{n} w \not\to \implies \exists n' : n \geq n' .\, e' \to^{n'} w' \not\to$$

- (Bi)simulation for comparing **reduction lengths** of λ-terms
- Automata-theoretically: for deterministic & unlabeled systems $(c : X \to X, F_1 \subseteq X)$

$\cong c : X \to \{\tau\} \times X$

- Simulate multiple steps with multiple steps

variant of weak simulation

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, dotted, "\exists\; m > 0 \text{ steps}"] \arrow[r, dash, dashed, "R"] \arrow[ur, phantom, "m \geq n"] & y \arrow[u, dotted, "n \text{ steps}\; \exists"']
\end{tikzcd}$$

$$e \to^{n} w \not\to \implies \exists n' : nQn' .\, e' \to^{n'} w' \not\to$$

Preorder-constrained simulation [Muroya, PhD thesis, 2020]

≅ (One-directional) improvement + up-to + generalization

special care required

(+ some restriction)

- Parameterized by a preorder $Q \subseteq \mathbb{N} \times \mathbb{N}$ closed under addition

(i.e. $iQi' \wedge jQj' \implies (i+j)Q(i'+j')$)

$$\begin{tikzcd}
x' \arrow[r, dash, dashed] & x'' \arrow[r, dash, dashed, "R"] & y'' \arrow[r, dash, dashed] & y' \\
x \arrow[u, dotted, "m > 0"] \arrow[rrr, dash, dashed, "R"] \arrow[urrr, phantom, "mQn"] & & & y \arrow[u, dotted, "n"']
\end{tikzcd}$$

Main Contribution

Preorder-constrained simulation
for
nondeterministic automata

- Parameterized by a preorder $Q \subseteq \mathbb{N} \times \mathbb{N}$
  closed under addition (i.e. $iQi' \wedge jQj' \implies (i+j)Q(i'+j')$)

- Parameterized by a preorder $\mathbb{Q} \subseteq \Sigma^* \times \Sigma^*$
  closed under concatenation (i.e. $w\mathbb{Q}w' \wedge v\mathbb{Q}v' \implies wv\mathbb{Q}w'v'$)

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, dotted, "w"] \arrow[r, dash, dashed, "R"'] \arrow[ur, phantom, "w\mathbb{Q}w'"] & y \arrow[u, dotted, "w'"']
\end{tikzcd}$$

Main Contribution

Preorder-constrained simulation
for
nondeterministic automata

$$\begin{tikzcd} x' \arrow[r, dash, dashed, "R"] & y' \\ x \arrow[u, dotted, "m > 0"] \arrow[r, dash, dashed, "R"'] \arrow[ur, phantom, "mQn"] & y \arrow[u, dotted, "n"'] \end{tikzcd}$$

- Parameterized by a preorder $Q \subseteq \mathbb{N} \times \mathbb{N}$ closed under addition (i.e. $iQi' \wedge jQj' \implies (i+j)Q(i'+j')$)

$$\begin{tikzcd} x' \arrow[r, dash, dashed, "R"] & y' \\ x \arrow[u, dotted, "w"] \arrow[r, dash, dashed, "R"'] \arrow[ur, phantom, "w\mathcal{Q}w'"] & y \arrow[u, dotted, "w'"'] \end{tikzcd}$$

- Parameterized by a preorder $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$ closed under concatenation (i.e. $w\mathcal{Q}w' \wedge v\mathcal{Q}v' \implies wv\mathcal{Q}w'v'$)

Main Contribution

Preorder-constrained simulation
for
nondeterministic automata

- Parameterized by a preorder $Q \subseteq \mathbb{N} \times \mathbb{N}$ closed under addition (i.e. $iQi' \wedge jQj' \implies (i+j)Q(i'+j')$)

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, dotted, "m>0"] \arrow[r, dash, dashed, "R"'] \arrow[ur, phantom, "mQn"] & y \arrow[u, dotted, "n"']
\end{tikzcd}$$

- Parameterized by a preorder $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$ closed under concatenation (i.e. $w\mathcal{Q}w' \wedge v\mathcal{Q}v' \implies wv\mathcal{Q}w'v'$)

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, dotted, "w"] \arrow[r, dash, dashed, "R"'] \arrow[ur, phantom, "w\mathcal{Q}w'"] & y \arrow[u, dotted, "w'"']
\end{tikzcd}$$

Theorem (**soundness**):

$$xRy \implies \forall w \in L^*(x)\,.\,\exists w' \in L^*(y)\,.\,w\mathcal{Q}w'$$

$\mathcal{Q}$**-trace inclusion** $L^*(x) \preceq_{\mathcal{Q}} L^*(y)$

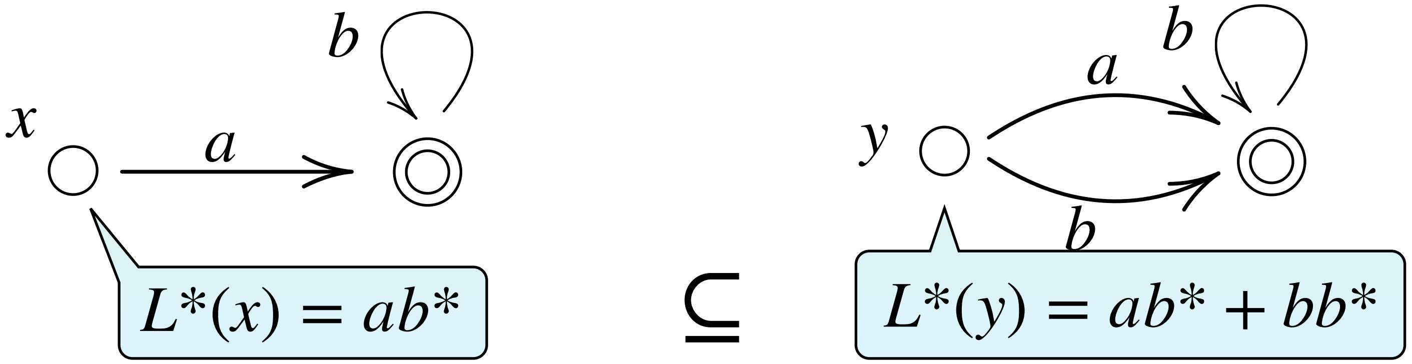
Examples for $\mathcal{Q}$-trace Inclusion

$$\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$$

$$\forall w \in L^*(x)\,.\,\exists w' \in L^*(y)\,.\,w\mathcal{Q}w'$$

- When $w\mathcal{Q}w' \overset{\text{def}}{\iff} w = w'$

$$\mathcal{Q}\text{-trace inclusion} \iff L^*(x) \subseteq L^*(y)$$ **(finite trace inclusion)**



$L^*(x) = ab^*$ $\subseteq$ $L^*(y) = ab^* + bb^*$

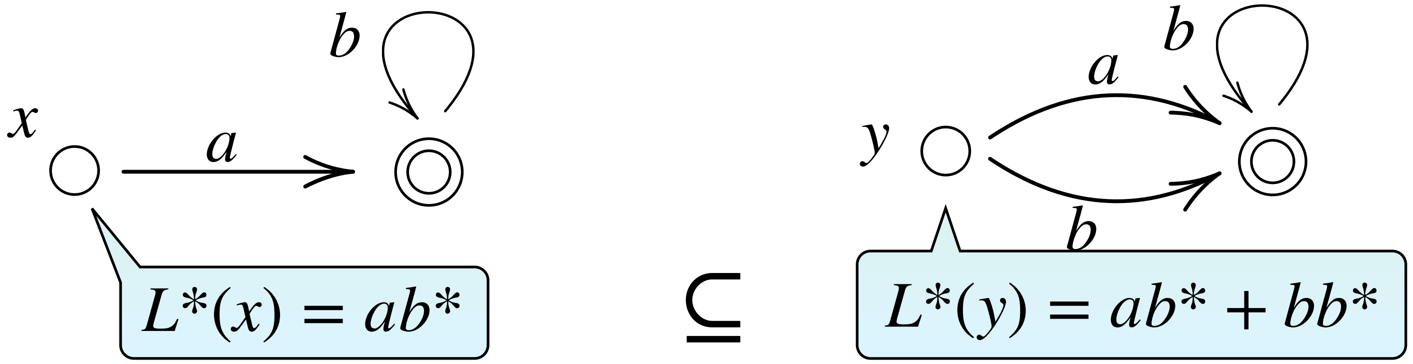
Examples for $\mathcal{Q}$-trace Inclusion

$\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$

$\forall w \in L^*(x) \,.\, \exists w' \in L^*(y) \,.\, w\mathcal{Q}w'$

- When $w\mathcal{Q}w' \overset{\text{def}}{\iff} w = w'$

$\mathcal{Q}$-trace inclusion $\iff L^*(x) \subseteq L^*(y)$ **(finite trace inclusion)**

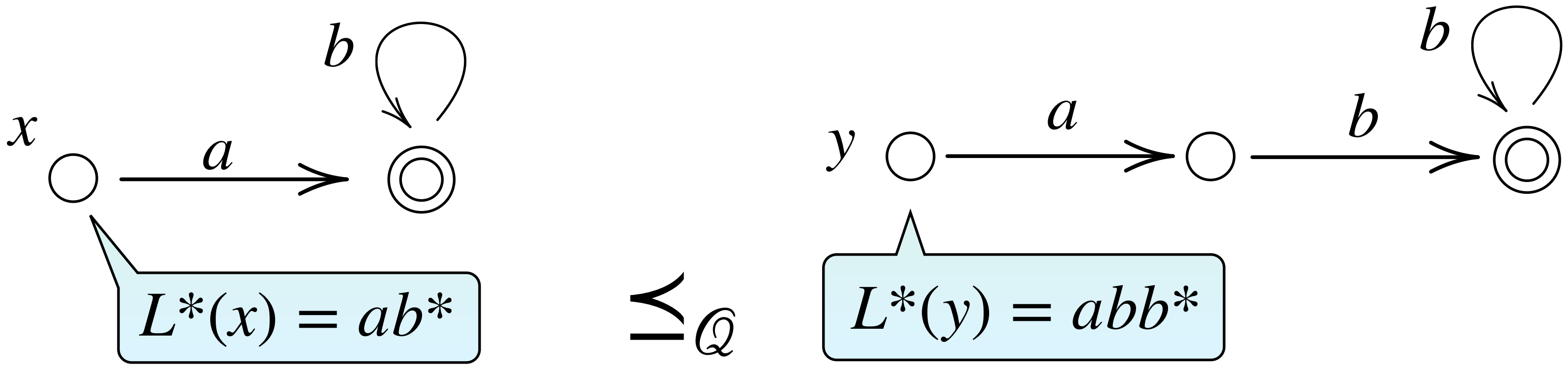


$L^*(x) = ab^*$ $\subseteq$ $L^*(y) = ab^* + bb^*$

- When $w\mathcal{Q}w' \overset{\text{def}}{\iff} w$ is a substring of w'

$\mathcal{Q}$-trace inclusion $\iff \forall w \in L^*(x) \,.\, \exists w' \in L^*(y) \,.\, w$ is a substring of w'

(trace inclusion wrt. substring)



$L^*(x) = ab^*$ $\preceq_{\mathcal{Q}}$ $L^*(y) = abb^*$

Examples for $\mathcal{Q}$-trace Inclusion

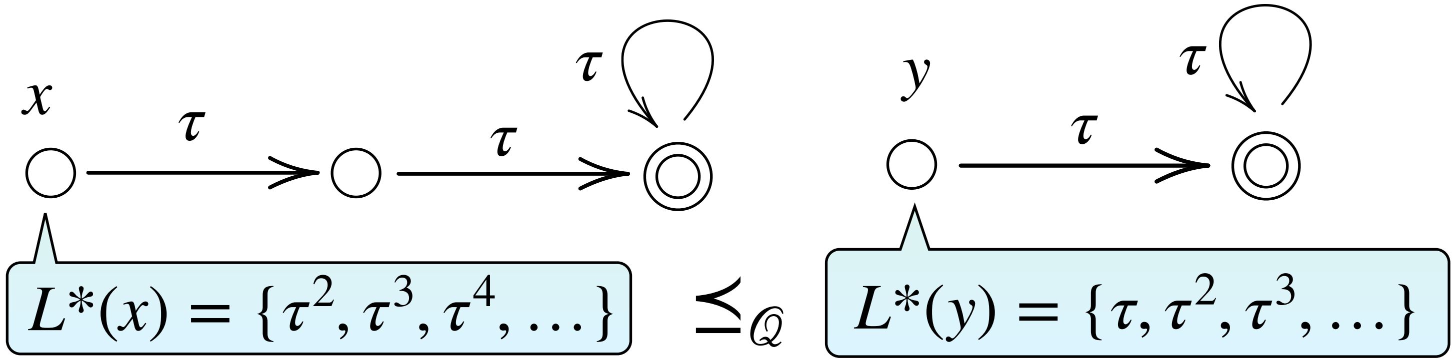
$$\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$$

$$\forall w \in L^*(x)\,.\, \exists w' \in L^*(y)\,.\, w\mathcal{Q}w'$$

- When $\Sigma = \{\tau\}$ and $w\mathcal{Q}w' \overset{\text{def}}{\iff} |w| \geq |w'|$

$$\mathcal{Q}\text{-trace inclusion} \iff \min \mathsf{length}(x \to \cdots \to \checkmark) \geq \min \mathsf{length}(y \to \cdots \to \checkmark)$$

(compare minimum distance to accepting state)



$$L^*(x) = \{\tau^2, \tau^3, \tau^4, \dots\} \preceq_{\mathcal{Q}} L^*(y) = \{\tau, \tau^2, \tau^3, \dots\}$$

Examples for $\mathcal{Q}$-trace Inclusion

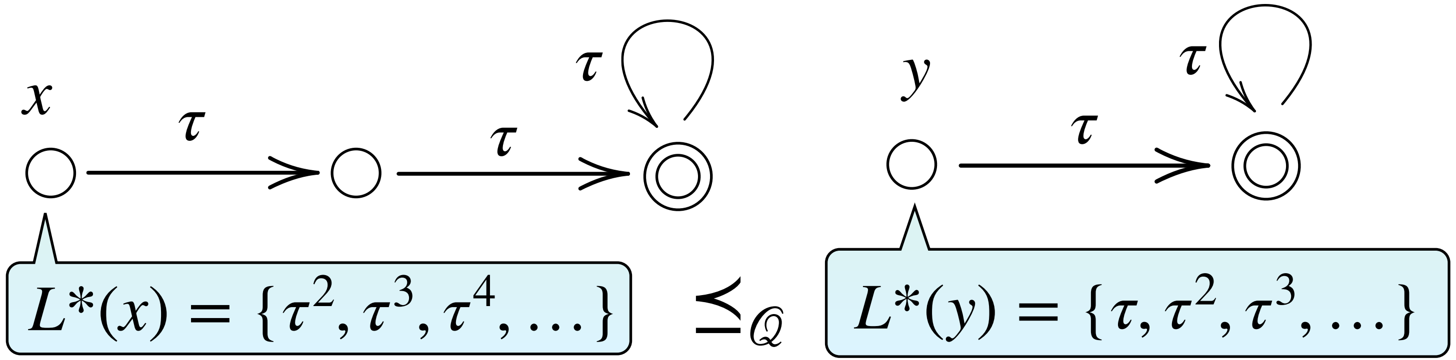
$$\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$$

$$\forall w \in L^*(x) \,.\, \exists w' \in L^*(y) \,.\, w\mathcal{Q}w'$$

- When $\Sigma = \{\tau\}$ and $w\mathcal{Q}w' \overset{\text{def}}{\iff} |w| \geq |w'|$

$$\mathcal{Q}\text{-trace inclusion} \iff \min \mathsf{length}(x \to \cdots \to \checkmark) \geq \min \mathsf{length}(y \to \cdots \to \checkmark)$$

(compare minimum distance to accepting state)

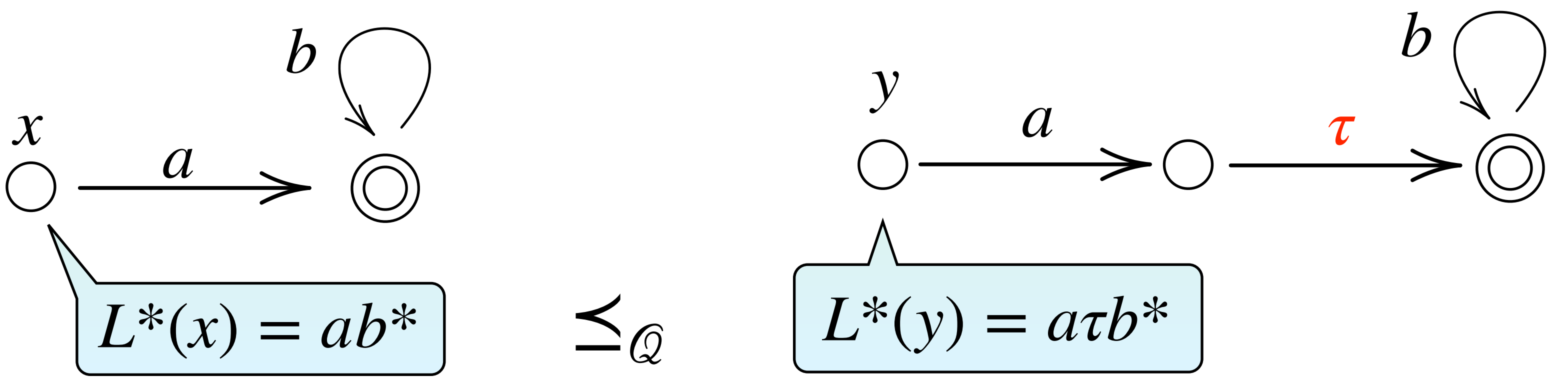


$L^*(x) = \{\tau^2, \tau^3, \tau^4, \ldots\}$ $\preceq_{\mathcal{Q}}$ $L^*(y) = \{\tau, \tau^2, \tau^3, \ldots\}$

- When $\Sigma = \{\tau\} + \Sigma'$ and $w\mathcal{Q}w' \overset{\text{def}}{\iff} \mathsf{remove}_\tau(w) = \mathsf{remove}_\tau(w')$

$$\mathcal{Q}\text{-trace inclusion} \iff \mathsf{remove}_\tau(L^*(x)) \subseteq \mathsf{remove}_\tau(L^*(y))$$

(weak trace inclusion)



$L^*(x) = ab^*$ $\preceq_{\mathcal{Q}}$ $L^*(y) = a\tau b^*$

Examples for $\mathcal{Q}$-trace Inclusion

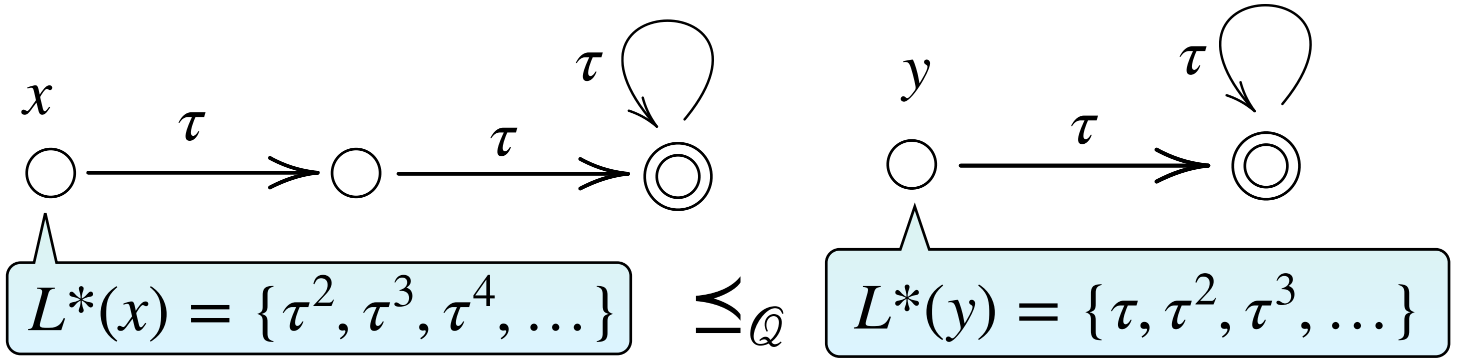
$\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$

$\forall w \in L^*(x) \,.\, \exists w' \in L^*(y) \,.\, w\mathcal{Q}w'$

- When $\Sigma = \{\tau\}$ and $w\mathcal{Q}w' \overset{\text{def}}{\iff} |w| \geq |w'|$

$$\mathcal{Q}\text{-trace inclusion} \iff \min \mathsf{length}(x \to \cdots \to \checkmark) \geq \min \mathsf{length}(y \to \cdots \to \checkmark)$$

(compare minimum distance to accepting state)

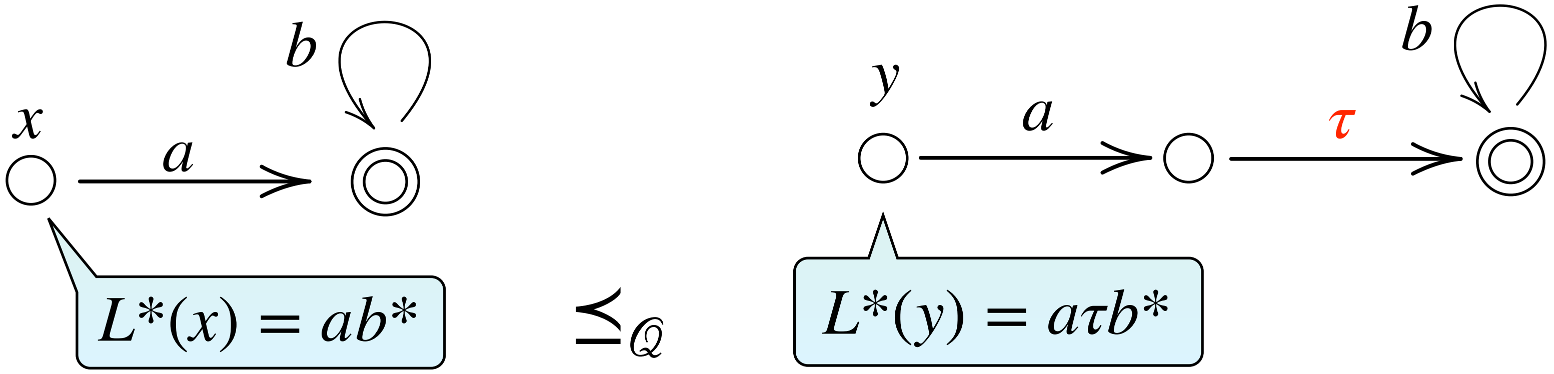


$L^*(x) = \{\tau^2, \tau^3, \tau^4, \dots\} \preceq_{\mathcal{Q}} L^*(y) = \{\tau, \tau^2, \tau^3, \dots\}$

- When $\Sigma = \{\tau\} + \Sigma'$ and $w\mathcal{Q}w' \overset{\text{def}}{\iff} \mathsf{remove}_\tau(w) = \mathsf{remove}_\tau(w')$

$$\mathcal{Q}\text{-trace inclusion} \iff \mathsf{remove}_\tau(L^*(x)) \subseteq \mathsf{remove}_\tau(L^*(y))$$

(weak trace inclusion)



$L^*(x) = ab^* \preceq_{\mathcal{Q}} L^*(y) = a\tau b^*$

- And more (e.g. weighted automata)

Outline

- Overview
- Preorder-Constrained Simulation without up-to
- Preorder-Constrained Simulation with up-to
- Conclusion and Future Work

Towards Generalization

preorder-constrained simulation
for **deterministic & unlabeled systems**
[Accattoli, Dal Lago & Vanoni, 2020] [Muroya, Phd thesis]

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, dotted, "\exists\ m > 0"] \arrow[r, dash, dashed, "R"'] \arrow[ur, phantom, "mQn"] & y \arrow[u, dotted, "\exists\ n"']
\end{tikzcd}$$

Towards Generalization

preorder-constrained simulation
for **deterministic & unlabeled systems**
[Accattoli, Dal Lago & Vanoni, 2020] [Muroya, Phd thesis]

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, dotted, "\exists \; m > 0"] \arrow[r, dash, dashed, "R"'] \arrow[ur, phantom, "mQn"] & y \arrow[u, dotted, "n \; \exists"']
\end{tikzcd}$$

+

forward simulation
for **nondeterministic automata**
[Lynch & Vaandrager, ‘95]

$$\begin{tikzcd}
\forall \; x' \arrow[r, dash, dashed, "R"] & y' \; \exists \\
x \arrow[u, "a"] \arrow[r, dash, dashed, "R"'] & y \arrow[u, "a"']
\end{tikzcd}$$

preorder-constrained simulation
for **nondeterministic automata**

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, dotted, "w \neq \varepsilon"] \arrow[r, dash, dashed, "R"'] \arrow[ur, phantom, "w \mathbb{Q} w'"] & y \arrow[u, dotted, "w'"']
\end{tikzcd}$$

Towards Generalization

preorder-constrained simulation
for **deterministic & unlabeled systems**
[Accattoli, Dal Lago & Vanoni, 2020] [Muroya, Phd thesis]

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, dotted, "\exists \; m > 0"] \arrow[r, dash, dashed, "R"'] \arrow[ur, phantom, "mQn"] & y \arrow[u, dotted, "n \; \exists"']
\end{tikzcd}$$

+

forward simulation
for **nondeterministic automata**
[Lynch & Vaandrager, '95]

$$\begin{tikzcd}
\forall \; x' \arrow[r, dash, dashed, "R"] & y' \; \exists \\
x \arrow[u, "a"] \arrow[r, dash, dashed, "R"'] & y \arrow[u, "a"']
\end{tikzcd}$$

preorder-constrained simulation
for **nondeterministic automata**

$$\begin{tikzcd}
x' \arrow[r, dash, dashed, "R"] & y' \\
x \arrow[u, dotted, "w \neq \varepsilon"] \arrow[r, dash, dashed, "R"'] \arrow[ur, phantom, "w \mathbb{Q} w'"] & y \arrow[u, dotted, "w'"']
\end{tikzcd}$$

w and x' are quantified by ∀
$|w|$ is quantified by ∃

Main Result:

Preorder-Constrained Simulation for Nondeterministic Automata

Definition:

Let $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$ be a preorder. A ***$\mathcal{Q}$-constrained simulation*** from $(c : X \to \mathcal{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathcal{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies \exists w' \in \Sigma^* \,.\, \varepsilon \mathcal{Q} w', y \xrightarrow{w'}{}^{*} y' \in F_2$
- $\forall a_1 \ldots a_n \in \Sigma^* \,.\, \forall x_1 \ldots x_n \in X_1^* \,.\, x \xrightarrow{a_1} x_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} x_n \in F_1$

  $\implies \exists k \in \{1, \ldots, n\} \,.\, \exists w' \in \Sigma^* \,.\, a_1 \ldots a_k \mathcal{Q} w', y \xrightarrow{w'}{}^{*} y'$ and $x_k R y'$

Main Result:
Preorder-Constrained Simulation for Nondeterministic Automata

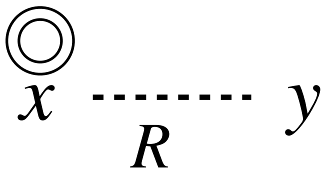
Definition:

Let $\mathscr{Q} \subseteq \Sigma^* \times \Sigma^*$ be a preorder. A ***$\mathscr{Q}$-constrained simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies \exists w' \in \Sigma^*\,.\, \varepsilon \mathscr{Q} w', y \xrightarrow{w'}{}^{*} y' \in F_2$
- $\forall a_1 \ldots a_n \in \Sigma^*\,.\, \forall x_1 \ldots x_n \in X_1^*\,.\, x \xrightarrow{a_1} x_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} x_n \in F_1$

$\implies \exists k \in \{1, \ldots, n\}\,.\, \exists w' \in \Sigma^*\,.\, a_1 \ldots a_k \mathscr{Q} w', y \xrightarrow{w'}{}^{*} y'$ and $x_k R y'$



Main Result:

Preorder-Constrained Simulation for Nondeterministic Automata

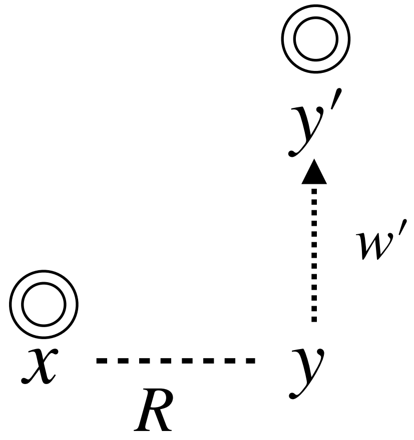
Definition:

Let $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$ be a preorder. A ***$\mathcal{Q}$-constrained simulation*** from $(c : X \to \mathcal{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathcal{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies \exists w' \in \Sigma^*\,.\ \varepsilon \mathcal{Q} w', y \xrightarrow{{w'}^*} y' \in F_2$
- $\forall a_1 \ldots a_n \in \Sigma^*\,.\ \forall x_1 \ldots x_n \in X_1^*\,.\ x \xrightarrow{a_1} x_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} x_n \in F_1$

  $\implies \exists k \in \{1, \ldots, n\}\,.\ \exists w' \in \Sigma^*\,.\ a_1 \ldots a_k \mathcal{Q} w', y \xrightarrow{{w'}^*} y'$ and $x_k R y'$



Main Result:
Preorder-Constrained Simulation for Nondeterministic Automata

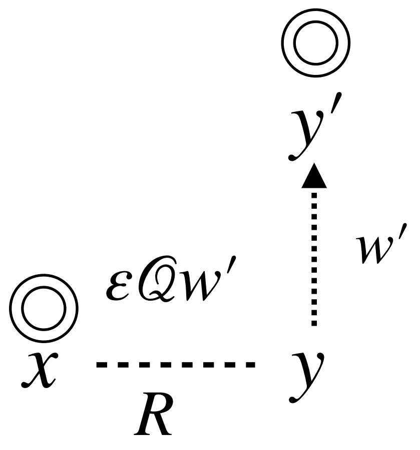
Definition:

Let $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$ be a preorder. A ***$\mathcal{Q}$-constrained simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies \exists w' \in \Sigma^*\,.\ \varepsilon \mathcal{Q} w', y \xrightarrow{w'^*} y' \in F_2$
- $\forall a_1 \ldots a_n \in \Sigma^*\,.\ \forall x_1 \ldots x_n \in X_1^*\,.\ x \xrightarrow{a_1} x_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} x_n \in F_1$

  $\implies \exists k \in \{1, \ldots, n\}\,.\ \exists w' \in \Sigma^*\,.\ a_1 \ldots a_k \mathcal{Q} w', y \xrightarrow{w'^*} y'$ and $x_k R y'$



Main Result:
Preorder-Constrained Simulation for Nondeterministic Automata

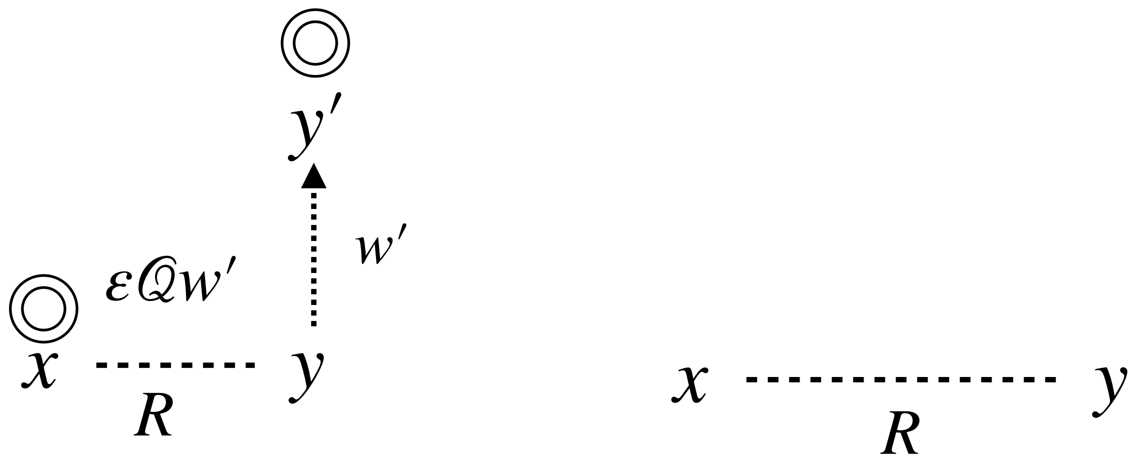
Definition:

Let $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$ be a preorder. A ***$\mathcal{Q}$-constrained simulation*** from $(c : X \to \mathcal{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathcal{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$\forall (x,y) \in R\,.$

- $x \in F_1 \implies \exists w' \in \Sigma^*\,.\ \varepsilon \mathcal{Q} w', y \xrightarrow{w'}{}^{*} y' \in F_2$
- $\forall a_1 \ldots a_n \in \Sigma^*\,.\ \forall x_1 \ldots x_n \in X_1^*\,.\ x \xrightarrow{a_1} x_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} x_n \in F_1$

  $\implies \exists k \in \{1, \ldots, n\}\,.\ \exists w' \in \Sigma^*\,.\ a_1 \ldots a_k \mathcal{Q} w', y \xrightarrow{w'}{}^{*} y'$ and $x_k R y'$



Main Result:

Preorder-Constrained Simulation for Nondeterministic Automata

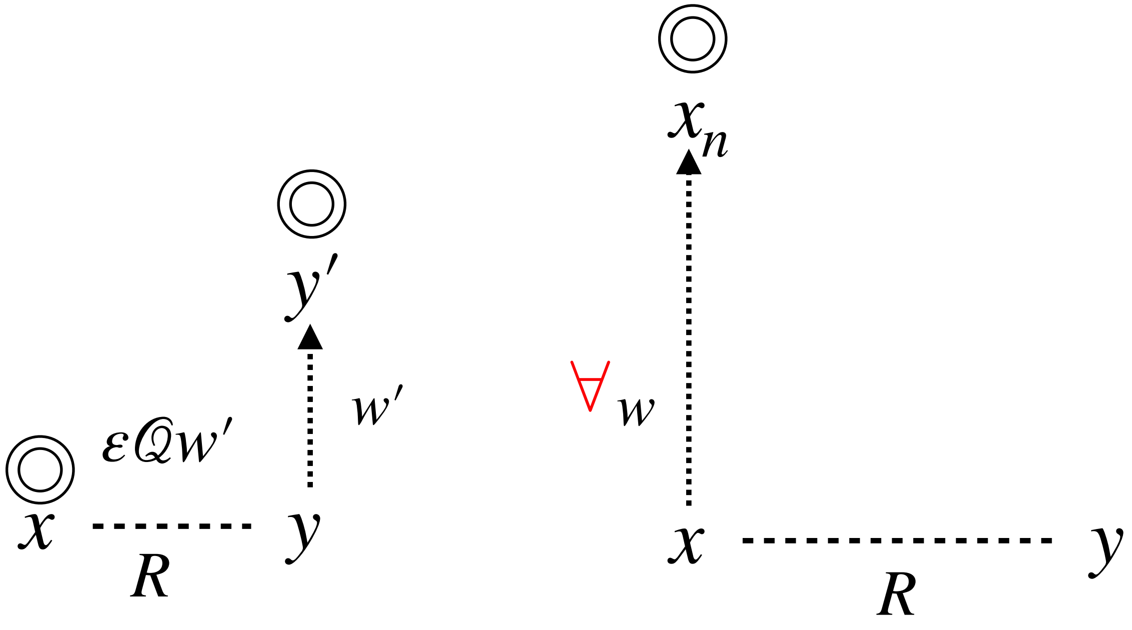
Definition:

Let $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$ be a preorder. A ***$\mathcal{Q}$-constrained simulation*** from $(c : X \to \mathcal{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathcal{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies \exists w' \in \Sigma^*\,.\ \varepsilon \mathcal{Q} w', y \xrightarrow{w'}{}^{*}\ y' \in F_2$
- $\forall a_1 \ldots a_n \in \Sigma^*\,.\ \forall x_1 \ldots x_n \in X_1^*\,.\ x \xrightarrow{a_1} x_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} x_n \in F_1$

  $\implies \exists k \in \{1, \ldots, n\}\,.\ \exists w' \in \Sigma^*\,.\ a_1 \ldots a_k \mathcal{Q} w', y \xrightarrow{w'}{}^{*}\ y'$ and $x_k R y'$



Main Result:

Preorder-Constrained Simulation for Nondeterministic Automata

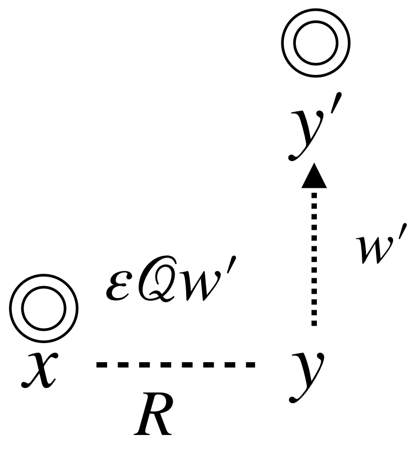
Definition:

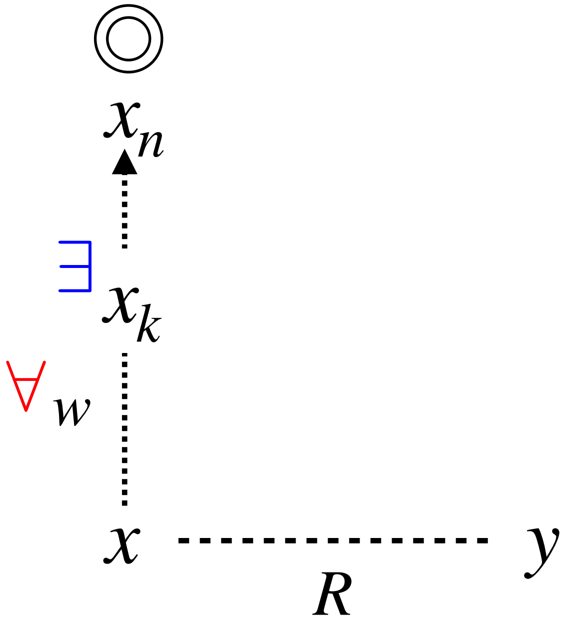
Let $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$ be a preorder. A ***$\mathcal{Q}$-constrained simulation*** from $(c : X \to \mathcal{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathcal{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies \exists w' \in \Sigma^*\,.\ \varepsilon \mathcal{Q} w', y \overset{w'^*}{\to} y' \in F_2$
- $\forall a_1 \ldots a_n \in \Sigma^*\,.\ \forall x_1 \ldots x_n \in X_1^*\,.\ x \overset{a_1}{\to} x_1 \overset{a_2}{\to} \cdots \overset{a_n}{\to} x_n \in F_1$

$$\implies \exists k \in \{1, \ldots, n\}\,.\ \exists w' \in \Sigma^*\,.\ a_1 \ldots a_k \mathcal{Q} w', y \overset{w'^*}{\to} y' \text{ and } x_k R y'$$





Main Result:
Preorder-Constrained Simulation for Nondeterministic Automata

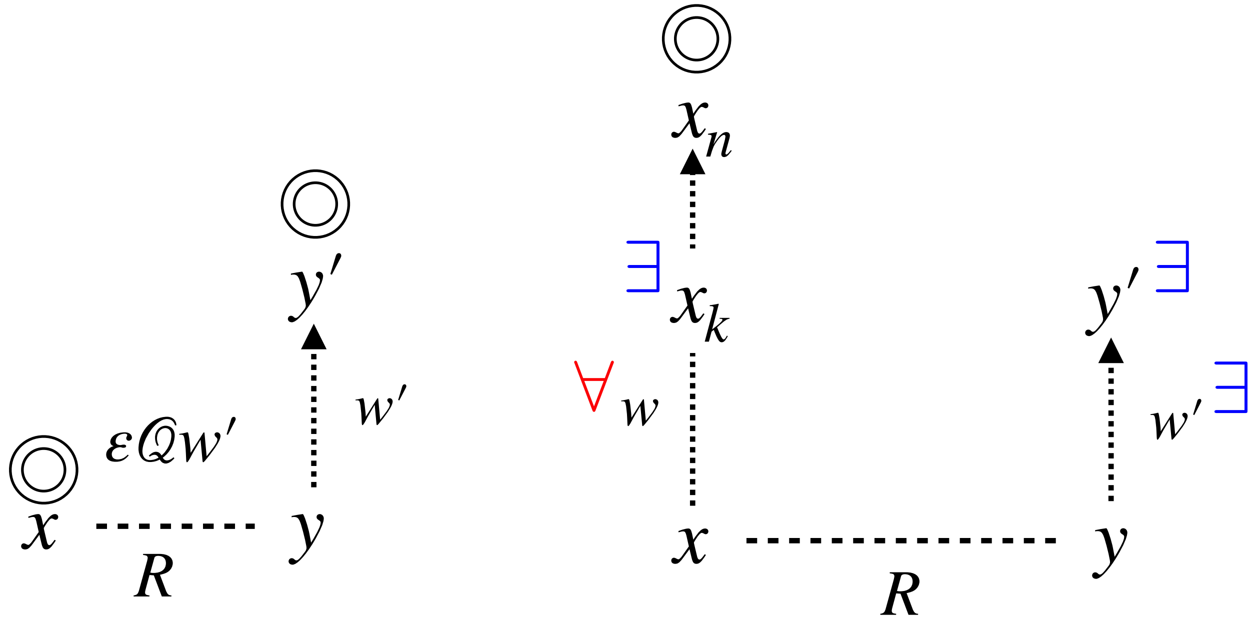
Definition:

Let $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$ be a preorder. A ***$\mathcal{Q}$-constrained simulation*** from $(c : X \to \mathcal{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathcal{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies \exists w' \in \Sigma^*\,.\; \varepsilon \mathcal{Q} w', y \overset{w'^*}{\to} y' \in F_2$
- $\forall a_1 \ldots a_n \in \Sigma^*\,.\; \forall x_1 \ldots x_n \in X_1^*\,.\; x \overset{a_1}{\to} x_1 \overset{a_2}{\to} \cdots \overset{a_n}{\to} x_n \in F_1$

  $\implies \exists k \in \{1, \ldots, n\}\,.\; \exists w' \in \Sigma^*\,.\; a_1 \ldots a_k \mathcal{Q} w', y \overset{w'^*}{\to} y'$ and $x_k R y'$



Main Result:
Preorder-Constrained Simulation for Nondeterministic Automata

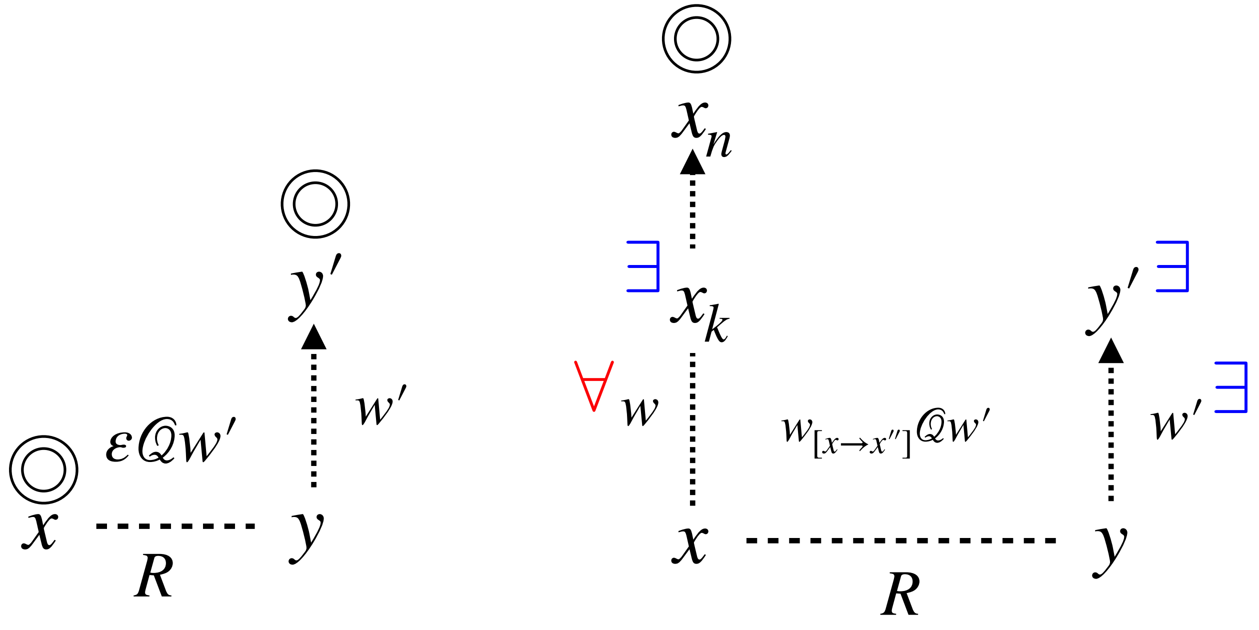
Definition:

Let $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$ be a preorder. A ***$\mathcal{Q}$-constrained simulation*** from $(c : X \to \mathcal{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathcal{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies \exists w' \in \Sigma^* \,.\, \varepsilon \mathcal{Q} w', y \overset{w'}{\to}{}^{*} y' \in F_2$
- $\forall a_1 \ldots a_n \in \Sigma^* \,.\, \forall x_1 \ldots x_n \in X_1^* \,.\, x \xrightarrow{a_1} x_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} x_n \in F_1$

  $\implies \exists k \in \{1, \ldots, n\} \,.\, \exists w' \in \Sigma^* \,.\, a_1 \ldots a_k \mathcal{Q} w', y \overset{w'}{\to}{}^{*} y'$ and $x_k R y'$



Main Result:

Preorder-Constrained Simulation for Nondeterministic Automata

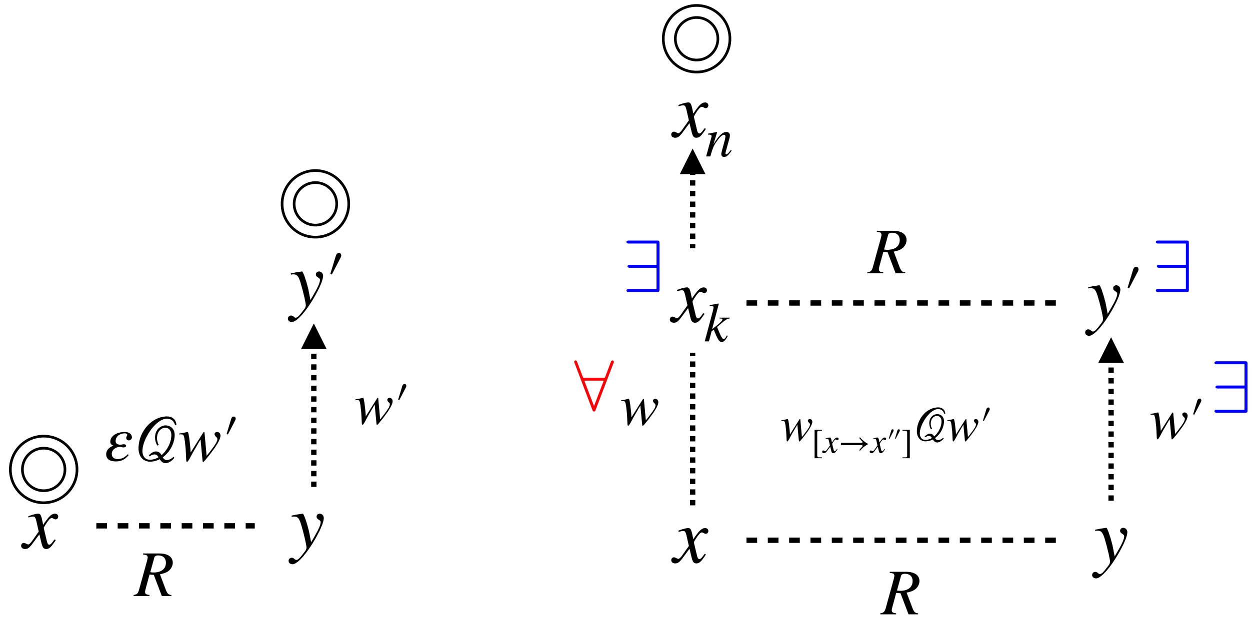
Definition:

Let $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$ be a preorder. A ***$\mathcal{Q}$-constrained simulation*** from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies \exists w' \in \Sigma^*\,.\,\varepsilon \mathcal{Q} w', y \xrightarrow{w'}{}^{*} y' \in F_2$
- $\forall a_1 \ldots a_n \in \Sigma^*\,.\,\forall x_1 \ldots x_n \in X_1^*\,.\,x \xrightarrow{a_1} x_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} x_n \in F_1$

$\implies \exists k \in \{1, \ldots, n\}\,.\,\exists w' \in \Sigma^*\,.\,a_1 \ldots a_k \mathcal{Q} w', y \xrightarrow{w'}{}^{*} y'$ and $x_k R y'$



Main Result:

Preorder-Constrained Simulation for Nondeterministic Automata

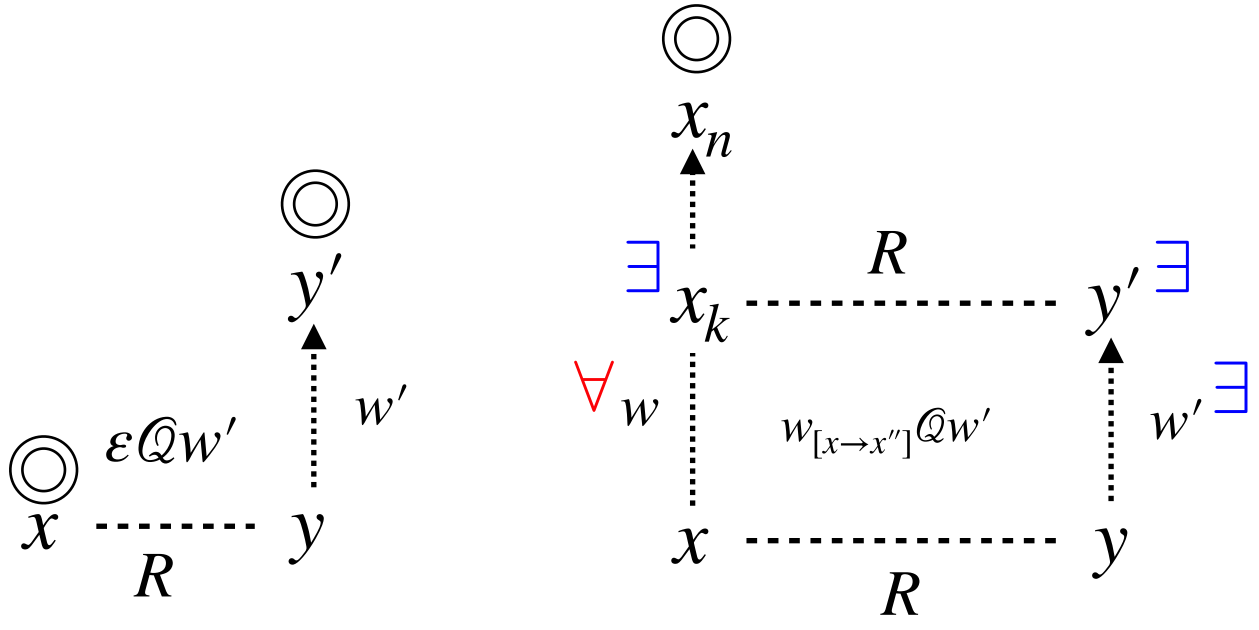
Definition:

Let $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$ be a preorder. A ***$\mathcal{Q}$-constrained simulation*** from $(c : X \to \mathcal{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathcal{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$$\forall (x, y) \in R\,. \qquad \text{- } x \in F_1 \implies \exists w' \in \Sigma^*\,.\, \varepsilon \mathcal{Q} w', y \xrightarrow{w'}{}^{*}\, y' \in F_2$$

$$\text{- } \forall a_1 \ldots a_n \in \Sigma^*\,.\, \forall x_1 \ldots x_n \in X_1^*\,.\, x \xrightarrow{a_1} x_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} x_n \in F_1$$

$$\implies \exists k \in \{1, \ldots, n\}\,.\, \exists w' \in \Sigma^*\,.\, a_1 \ldots a_k \mathcal{Q} w', y \xrightarrow{w'}{}^{*}\, y' \text{ and } x_k R y'$$



Theorem (**soundness**):

When $\mathcal{Q}$ is closed under concatenation,

$$xRy \implies$$

$$\forall w \in L^*(x)\,.\, \exists w' \in L^*(y)\,.\, w \mathcal{Q} w'$$

Characterization as Safety Game

$S_\forall := \{\checkmark\} + \Sigma^* \times X \times Y$

$S_\exists := \{\text{last\_turn}\} \times \Sigma^* \times X \times Y + \Sigma^+ \times X \times Y$

$\rightarrow_\forall \subseteq S_\forall \times S_\exists$ is given by:

$$\left\{\left((w,x,y),(wa,x',y)\right) \mid x \xrightarrow{a} x'\right\} \cup \left\{\left((w,x,y),(\text{last\_turn},w,x,y)\right) \mid x \in F_1\right\}$$

$\rightarrow_\exists \subseteq S_\exists \times S_\forall$ is given by:

$$\left\{\left((w,x',y),(w,x',y)\right)\right\} \cup \left\{\left((w,x',y),(\varepsilon,x',y')\right) \mid y \xrightarrow{w'^*} y', w\mathcal{Q}w'\right\}$$

$$\cup \left\{\left((\text{last\_turn},w,x,y),\checkmark\right) \mid y \xrightarrow{w'^*} y' \in F_2, w\mathcal{Q}w'\right\}$$

Conjecture:

Simulator is winning from (ε,x,y) $\iff$

$(x,y) \in R$ *for some* $\mathcal{Q}$*-constrained simulation*

Characterization as Safety Game

$S_\forall := \{\checkmark\} + \Sigma^* \times X \times Y$ **(state space for Challenger)**

$S_\exists := \{\text{last\_turn}\} \times \Sigma^* \times X \times Y + \Sigma^+ \times X \times Y$ **(state space for Simulator)**

$\rightarrow_\forall \subseteq S_\forall \times S_\exists$ is given by:

$$\left\{ \left((w,x,y), (wa,x',y) \right) \mid x \xrightarrow{a} x' \right\} \cup \left\{ \left((w,x,y), (\text{last\_turn}, w, x, y) \right) \mid x \in F_1 \right\}$$

$\rightarrow_\exists \subseteq S_\exists \times S_\forall$ is given by:

$$\left\{ \left((w,x',y), (w,x',y) \right) \right\} \cup \left\{ \left((w,x',y), (\varepsilon,x',y') \right) \mid y \overset{w'^*}{\rightarrow} y', w \mathcal{Q} w' \right\}$$

$$\cup \left\{ \left((\text{last\_turn}, w,x,y), \checkmark \right) \mid y \overset{w'^*}{\rightarrow} y' \in F_2, w \mathcal{Q} w' \right\}$$

Conjecture:

Simulator is winning from $(\varepsilon, x, y) \iff$

$(x,y) \in R$ *for some* $\mathcal{Q}$*-constrained simulation*

Characterization as Safety Game

$S_\forall := \{\checkmark\} + \Sigma^* \times X \times Y$ **(state space for Challenger)**

queue

$S_\exists := \{\text{last\_turn}\} \times \Sigma^* \times X \times Y + \Sigma^+ \times X \times Y$ **(state space for Simulator)**

$\rightarrow_\forall \subseteq S_\forall \times S_\exists$ is given by:

$$\left\{ \big((w,x,y),(wa,x',y)\big) \mid x \xrightarrow{a} x' \right\} \cup \left\{ \big((w,x,y),(\text{last\_turn},w,x,y)\big) \mid x \in F_1 \right\}$$

$\rightarrow_\exists \subseteq S_\exists \times S_\forall$ is given by:

$$\left\{ \big((w,x',y),(w,x',y)\big) \right\} \cup \left\{ \big((w,x',y),(\varepsilon,x',y')\big) \mid y \xrightarrow{w'}{}^{*} y', w\mathcal{Q}w' \right\}$$

$$\cup \left\{ \big((\text{last\_turn},w,x,y),\checkmark\big) \mid y \xrightarrow{w'}{}^{*} y' \in F_2, w\mathcal{Q}w' \right\}$$

Conjecture:

Simulator is winning from $(\varepsilon, x, y) \iff$

$(x, y) \in R$ *for some* $\mathcal{Q}$*-constrained simulation*

Characterization as Safety Game

$S_\forall := \{\checkmark\} + \Sigma^* \times X \times Y$ **(state space for Challenger)**

queue

$S_\exists := \{\mathsf{last\_turn}\} \times \Sigma^* \times X \times Y + \Sigma^+ \times X \times Y$ **(state space for Simulator)**

$\to_\forall \subseteq S_\forall \times S_\exists$ is given by:

$$\left\{\big((w,x,y),(wa,x',y)\big) \mid x \xrightarrow{a} x'\right\} \cup \left\{\big((w,x,y),(\mathsf{last\_turn},w,x,y)\big) \mid x \in F_1\right\}$$

choose successor state and enqueue

$\to_\exists \subseteq S_\exists \times S_\forall$ is given by:

$$\left\{\big((w,x',y),(w,x',y)\big)\right\} \cup \left\{\big((w,x',y),(\varepsilon,x',y')\big) \mid y \overset{w'}{\to}{}^{*} y', w\mathcal{Q}w'\right\}$$

$$\cup \left\{\big((\mathsf{last\_turn},w,x,y),\checkmark\big) \mid y \overset{w'}{\to}{}^{*} y' \in F_2, w\mathcal{Q}w'\right\}$$

Conjecture:

Simulator is winning from (ε, x, y) $\iff$

$(x, y) \in R$ *for some* $\mathcal{Q}$*-constrained simulation*

Characterization as Safety Game

$S_\forall := \{\checkmark\} + \Sigma^* \times X \times Y$ **(state space for Challenger)**

queue

$S_\exists := \{\mathsf{last\_turn}\} \times \Sigma^* \times X \times Y + \Sigma^+ \times X \times Y$ **(state space for Simulator)**

$\rightarrow_\forall \subseteq S_\forall \times S_\exists$ is given by:

$$\left\{\big((w,x,y),(wa,x',y)\big) \mid x \xrightarrow{a} x'\right\} \cup \left\{\big((w,x,y),(\mathsf{last\_turn},w,x,y)\big) \mid x \in F_1\right\}$$

choose successor state and enqueue

$\rightarrow_\exists \subseteq S_\exists \times S_\forall$ is given by:

$$\left\{\big((w,x',y),(w,x',y)\big)\right\} \cup \left\{\big((w,x',y),(\varepsilon,x',y')\big) \mid y \overset{w'^*}{\rightarrow} y', wQw'\right\}$$

pass the turn

dequeue all, and move

$$\cup \left\{\big((\mathsf{last\_turn},w,x,y),\checkmark\big) \mid y \overset{w'^*}{\rightarrow} y' \in F_2, wQw'\right\}$$

Conjecture:

Simulator is winning from $(\varepsilon,x,y) \iff$

$(x,y) \in R$ *for some* Q*-constrained simulation*

Characterization as Safety Game

$S_\forall := \{✓\} + \Sigma^* \times X \times Y$ **(state space for Challenger)**

queue

$S_\exists := \{\mathsf{last\_turn}\} \times \Sigma^* \times X \times Y + \Sigma^+ \times X \times Y$ **(state space for Simulator)**

$\to_\forall \subseteq S_\forall \times S_\exists$ is given by:

$$\Big\{\big((w,x,y),(wa,x',y)\big) \mid x \xrightarrow{a} x'\Big\} \cup \Big\{\big((w,x,y),(\mathsf{last\_turn},w,x,y)\big) \mid x \in F_1\Big\}$$

choose successor state and enqueue **declare "last turn" (possible when x is accepting)**

$\to_\exists \subseteq S_\exists \times S_\forall$ is given by:

$$\Big\{\big((w,x',y),(w,x',y)\big)\Big\} \cup \Big\{\big((w,x',y),(\varepsilon,x',y')\big) \mid y \xrightarrow{w'}{}^{*} y', w\mathcal{Q}w'\Big\}$$

pass the turn **dequeue all, and move**

$$\cup \Big\{\big((\mathsf{last\_turn},w,x,y),✓\big) \mid y \xrightarrow{w'}{}^{*} y' \in F_2, w\mathcal{Q}w'\Big\}$$

Conjecture:

Simulator is winning from $(\varepsilon,x,y) \iff$

$(x,y) \in R$ *for some* $\mathcal{Q}$*-constrained simulation*

Characterization as Safety Game

$S_\forall := \{\checkmark\} + \boxed{\Sigma^*} \times X \times Y$ **(state space for Challenger)**

queue

$S_\exists := \{\text{last\_turn}\} \times \Sigma^* \times X \times Y + \Sigma^+ \times X \times Y$ **(state space for Simulator)**

$\rightarrow_\forall \subseteq S_\forall \times S_\exists$ is given by:

$$\left\{\left((w,x,y),(wa,x',y)\right) \mid x \xrightarrow{a} x'\right\} \cup \left\{\left((w,x,y),(\text{last\_turn},w,x,y)\right) \mid x \in F_1\right\}$$

choose successor state and enqueue **declare “last turn” (possible when x is accepting)**

$\rightarrow_\exists \subseteq S_\exists \times S_\forall$ is given by:

$$\left\{\left((w,x',y),(w,x',y)\right)\right\} \cup \left\{\left((w,x',y),(\varepsilon,x',y')\right) \mid y \xrightarrow{w'}{}^{*}\, y', w\mathcal{Q}w'\right\}$$

pass the turn **dequeue all, and move**

$$\cup \left\{\left((\text{last\_turn},w,x,y),\checkmark\right) \mid y \xrightarrow{w'}{}^{*}\, y' \in F_2, w\mathcal{Q}w'\right\}$$

reach accepting state, and win the game

Conjecture:

Simulator is winning from (ε, x, y) $\iff$

$(x, y) \in R$ *for some* $\mathcal{Q}$*-constrained simulation*

Characterization as Safety Game

$S_\forall := \{\checkmark\} + \Sigma^* \times X \times Y$ **(state space for Challenger)**

queue

$S_\exists := \{\mathsf{last\_turn}\} \times \Sigma^* \times X \times Y + \Sigma^+ \times X \times Y$ **(state space for Simulator)**

$\to_\forall \subseteq S_\forall \times S_\exists$ is given by:

$$\left\{\left((w,x,y),(wa,x',y)\right) \mid x \xrightarrow{a} x'\right\} \cup \left\{\left((w,x,y),(\mathsf{last\_turn},w,x,y)\right) \mid x \in F_1\right\}$$

choose successor state and enqueue **declare "last turn" (possible when x is accepting)**

$\to_\exists \subseteq S_\exists \times S_\forall$ is given by:

$$\left\{\left((w,x',y),(w,x',y)\right)\right\} \cup \left\{\left((w,x',y),(\varepsilon,x',y')\right) \mid y \xrightarrow{w'}{}^{*} y', w @ w'\right\}$$

pass the turn **dequeue all, and move**

$$\cup \left\{\left((\mathsf{last\_turn},w,x,y),\checkmark\right) \mid y \xrightarrow{w'}{}^{*} y' \in F_2, w @ w'\right\}$$

reach accepting state, and win the game

- **Simulator loses iff it gets stuck (i.e. infinite play is winning)**

Conjecture:

Simulator is winning from (ε,x,y) $\iff$

$(x,y) \in R$ *for some* $@$*-constrained simulation*

Outline

- Overview
- Preorder-Constrained Simulation without up-to
- Preorder-Constrained Simulation with up-to
- Conclusion and Future Work

Preorder-Constrained Simulation with up-to

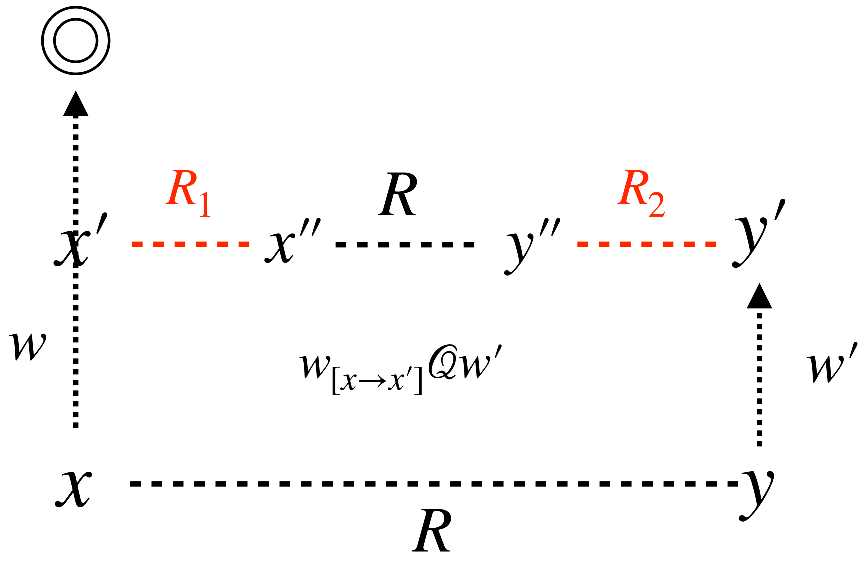
Definition:

Let $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$. A $\mathcal{Q}$-*constrained simulation* **up-to** $(R_1 \subseteq X \times X, R_2 \subseteq Y \times Y)$ from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies \exists w' \in \Sigma^* \,.\, \varepsilon \mathcal{Q} w', y \overset{w'^*}{\to} y' \in F_2$
- $\forall a_1 \ldots a_n \in \Sigma^* \,.\, \forall x_1 \ldots x_n \in X_1^* \,.\, x \xrightarrow{a_1} x_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} x_n \in F_1$

  $\implies \exists k \in \{1, \ldots, n\} \,.\, \exists w' \in \Sigma^* \,.\, a_1 \ldots a_k \mathcal{Q} w', y \overset{w'^*}{\to} y'$ and $x_k R_1 R R_2 y'$



Theorem (**soundness**):

When $\mathcal{Q}$ is closed under concatenation,

xR_1x' and yR_2y' imply $\mathcal{Q}$-trace inclusion, and

xR_1x' implies $(|\cdot| \geq |\cdot|)$-trace inclusion,

$$xRy \implies$$

$$\forall w \in L^*(x) \,.\, \exists w' \in L^*(y) \,.\, w \mathcal{Q} w'$$

Preorder-Constrained Simulation with up-to

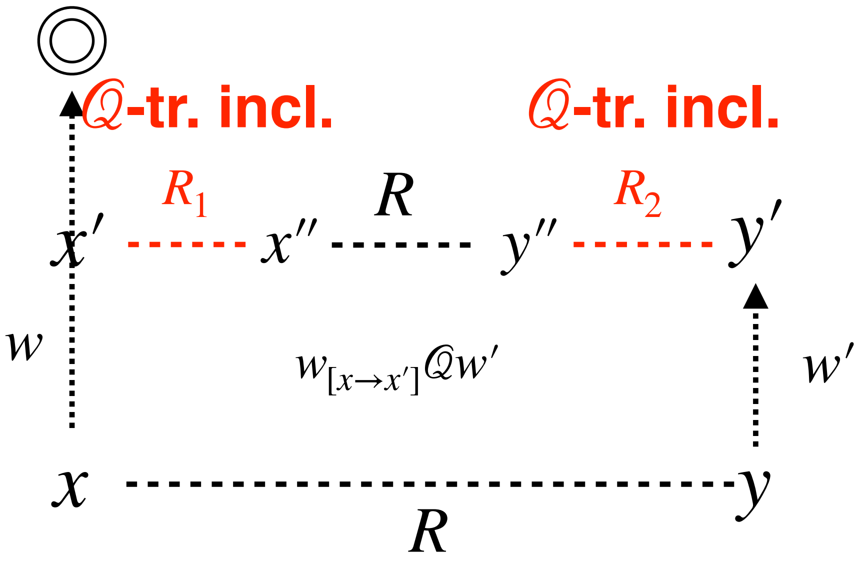
Definition:

Let $\mathscr{Q} \subseteq \Sigma^* \times \Sigma^*$. A ***$\mathscr{Q}$-constrained simulation*** **up-to** $(R_1 \subseteq X \times X, R_2 \subseteq Y \times Y)$ from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$$\forall (x,y) \in R \,. \quad \text{-}\ x \in F_1 \Longrightarrow \exists w' \in \Sigma^* \,.\, \varepsilon \mathscr{Q} w', y \overset{{w'}^{*}}{\rightarrow} y' \in F_2$$

$$\text{-}\ \forall a_1 \ldots a_n \in \Sigma^* \,.\, \forall x_1 \ldots x_n \in X_1^* \,.\, x \xrightarrow{a_1} x_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} x_n \in F_1$$

$$\Longrightarrow \exists k \in \{1,\ldots,n\} \,.\, \exists w' \in \Sigma^* \,.\, a_1 \ldots a_k \mathscr{Q} w', y \overset{{w'}^{*}}{\rightarrow} y' \text{ and } x_k R_1 R R_2 y'$$



Theorem (**soundness**):

When $\mathscr{Q}$ is closed under concatenation,

xR_1x' and yR_2y' imply $\mathscr{Q}$-trace inclusion, and

xR_1x' implies $(|\cdot| \geq |\cdot|)$-trace inclusion,

$$xRy \Longrightarrow$$

$$\forall w \in L^*(x) \,.\, \exists w' \in L^*(y) \,.\, w \mathscr{Q} w'$$

Preorder-Constrained Simulation with up-to

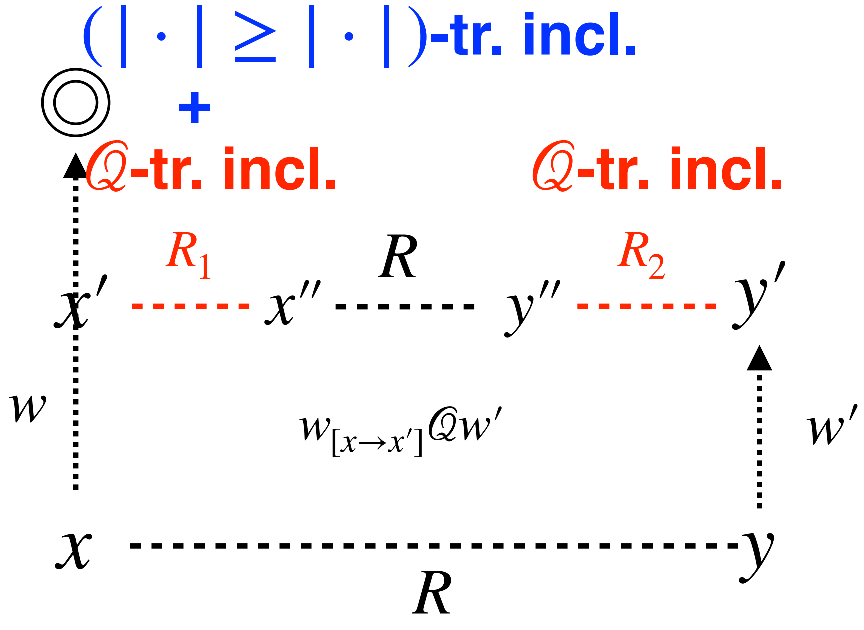
Definition:

Let $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$. A ***$\mathcal{Q}$-constrained simulation*** **up-to** $(R_1 \subseteq X \times X, R_2 \subseteq Y \times Y)$ from $(c : X \to \mathscr{P}(\Sigma \times X), F_1 \subseteq X)$ to $(d : Y \to \mathscr{P}(\Sigma \times Y), F_2 \subseteq Y)$ is $R \subseteq X \times Y$ s.t.

$$\forall (x,y) \in R\,. \qquad -\ x \in F_1 \implies \exists w' \in \Sigma^*\,.\ \varepsilon \mathcal{Q} w', y \overset{w'^*}{\to} y' \in F_2$$

$$-\ \forall a_1 \ldots a_n \in \Sigma^*\,.\ \forall x_1 \ldots x_n \in X_1^*\,.\ x \xrightarrow{a_1} x_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} x_n \in F_1$$

$$\implies \exists k \in \{1, \ldots, n\}\,.\ \exists w' \in \Sigma^*\,.\ a_1 \ldots a_k \mathcal{Q} w', y \overset{w'^*}{\to} y' \text{ and } x_k R_1 R R_2 y'$$



$(|\cdot| \geq |\cdot|)$**-tr. incl.**

◎ **+**

$\mathcal{Q}$**-tr. incl.** $\qquad$ $\mathcal{Q}$**-tr. incl.**

Theorem (**soundness**):

When $\mathcal{Q}$ is closed under concatenation,

xR_1x' and yR_2y' imply $\mathcal{Q}$-trace inclusion, and

xR_1x' implies $(|\cdot| \geq |\cdot|)$-trace inclusion,

$$xRy \implies \forall w \in L^*(x)\,.\ \exists w' \in L^*(y)\,.\ w \mathcal{Q} w'$$

for safely combining weak & up-to

Outline

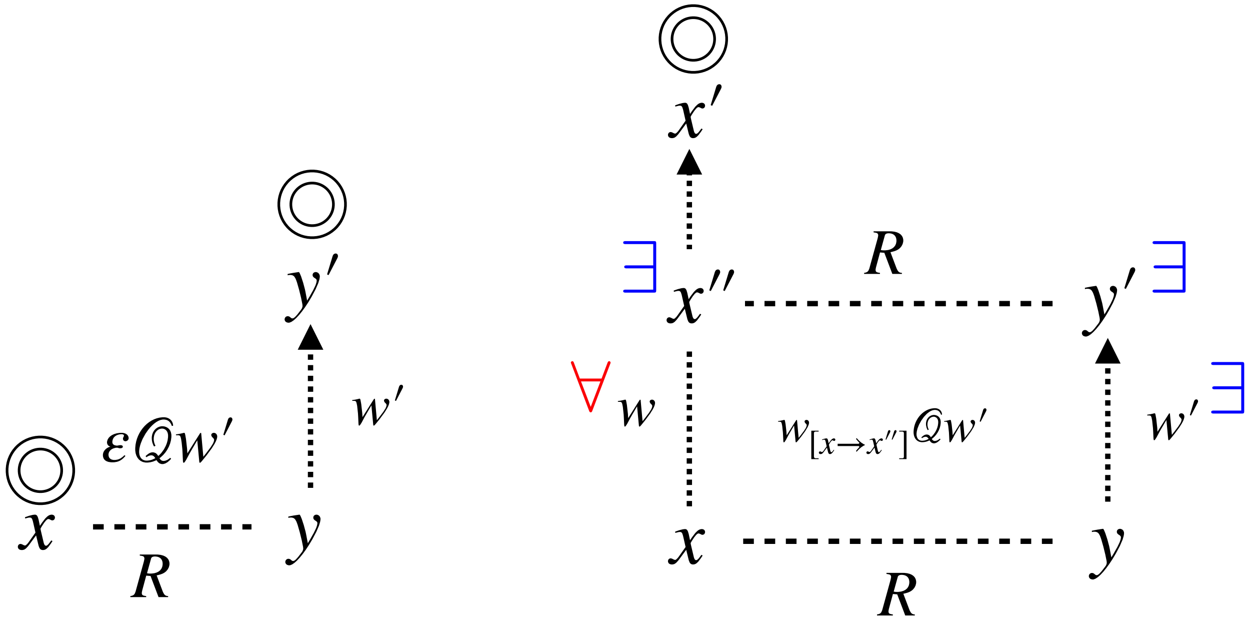
- Overview
- Preorder-Constrained Simulation without up-to
- Preorder-Constrained Simulation with up-to
- Conclusion and Future Work

Conclusion

- New simulation notion for witnessing **$\mathscr{Q}$-trace inclusion**

$$\forall w \in L^*(x)\,.\,\exists w' \in L^*(y)\,.\,w\mathscr{Q}w'$$

- Enhancement with up-to



Future Directions

- Extension to **infinitary** trace inclusion

Future Directions

- Extension to **infinitary** trace inclusion
- Relaxing condition in up-to

Theorem (**soundness**):

When $\mathcal{Q}$ is closed under concatenation,

xR_1x' *and* yR_2y' *imply* $\mathcal{Q}$*-trace inclusion, and*

xR_1x' *implies* $(|\cdot| \geq |\cdot|)$*-trace inclusion,*

$$xRy \implies \forall w \in L^*(x)\,.\,\exists w' \in L^*(y)\,.\,w\mathcal{Q}w'$$

Conjecture:
finitely many violation is ok

Future Directions

- Extension to **infinitary** trace inclusion
- Relaxing condition in up-to

Theorem (**soundness**):

When Q is closed under concatenation,

xR_1x' and yR_2y' imply Q-trace inclusion, and

xR_1x' implies $(|\cdot| \geq |\cdot|)$-trace inclusion,

$$xRy \implies \forall w \in L^*(x)\,.\,\exists w' \in L^*(y)\,.\,wQw'$$

Conjecture:
finitely many violation is ok

- Coalgebraic characterization
  - Coalgebraic simulation: [Hughes & Jacobs, ’04] [Hasuo, ’06]
  - Coalgebraic simulation with queues: [U. & Hasuo, ’14]
  - Coalgebraic bisimulation up-to: [Rot, Bonsangue & Rutten, ‘13]

Addenda

Addendum I: Buffered Simulation

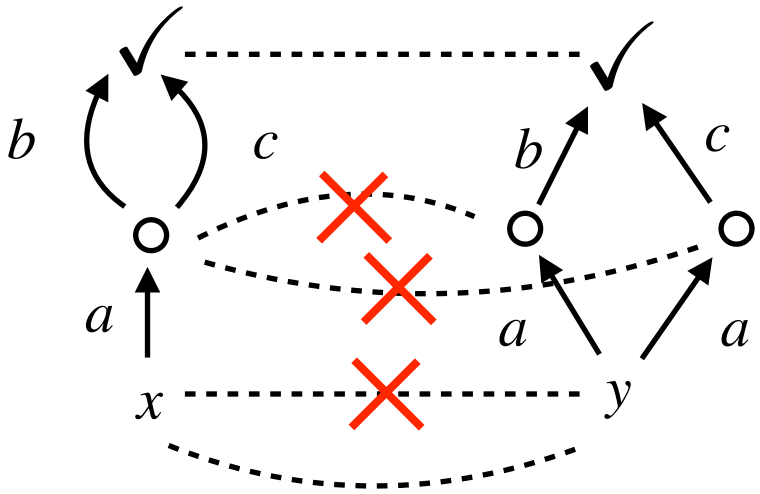
- Simulation with queueing (**buffering**) is used for fixing **incompleteness**
  - e.g. [Hutagalung, Lange & Lozes, AFL 2014] for Büchi automata

Theorem (**soundness**):

If R is a forward simulation,

$$xRy \underset{\cancel{\Longleftarrow}}{\Longrightarrow} L^*(x) \subseteq L^*(y)$$

incomplete



- $L^*(x) = L^*(y) = \{ab, ac\}$, but no forward simulation can prove it
- However, it does exist if buffering is allowed

Addendum I: Buffered Simulation

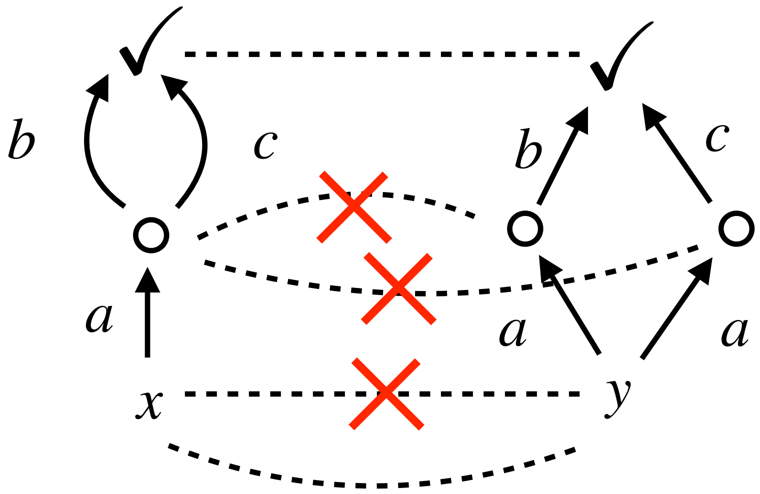
- Simulation with queueing (**buffering**) is used for fixing **incompleteness**
    - e.g. [Hutagalung, Lange & Lozes, AFL 2014] for Büchi automata

Theorem (**soundness**):

If R is a forward simulation,

$$xRy \;\underset{\cancel{\Longleftarrow}}{\Longrightarrow}\; L^*(x) \subseteq L^*(y)$$

incomplete



- $L^*(x) = L^*(y) = \{ab, ac\}$, but no forward simulation can prove it
- However, it does exist if buffering is allowed

- Preorder-constrained simulation also has this property

Addendum II: Categorical Buffered Simulation

- Kleisli Simulation [Hasuo, '06]
  - System as a coalgebra in Kleisli category

    $c : X \to FX$ in $\mathscr{K}\ell(T)$ whose homsets are order-enriched
  - Simulation as an oplax homomorphism

$$\begin{tikzcd} FX & FY \arrow[l, "\overline{F}f"'] \\ X \arrow[u, "c"] \arrow[ur, phantom, "\sqsubseteq"] & Y \arrow[l, "f"] \arrow[u, "d"'] \end{tikzcd}$$

in $\mathscr{K}\ell(T)$

$\overline{F} : \mathscr{K}\ell(T) \to \mathscr{K}\ell(T)$: lifting of F

Addendum II: Categorical Buffered Simulation

- Kleisli Simulation [Hasuo, '06]
  - System as a coalgebra in Kleisli category

    $c : X \to FX$ in $\mathscr{K}\ell(T)$ whose homsets are order-enriched

  - Simulation as an oplax homomorphism

$$\begin{tikzcd}
FX & FY \arrow[l, "\overline{F}f"'] \\
X \arrow[u, "c"] \arrow[ur, phantom, "\sqsubseteq"] & Y \arrow[l, "f"] \arrow[u, "d"']
\end{tikzcd}$$

in $\mathscr{K}\ell(T)$

$\overline{F} : \mathscr{K}\ell(T) \to \mathscr{K}\ell(T)$: lifting of F

- Forward partial execution [U. & Hasuo, '14]

$$\begin{tikzcd}
FX & & F^{n+1}X \\
X \arrow[u, "c"] & \longmapsto & F^n X \arrow[u, "\overline{F}^n c"]
\end{tikzcd}$$

in $\mathscr{K}\ell(T)$

Addendum II: Categorical Buffered Simulation

- Kleisli Simulation [Hasuo, ’06]
    - System as a coalgebra in Kleisli category

      $c : X \to FX$ in $\mathscr{K}\ell(T)$ whose homsets are order-enriched

    - Simulation as an oplax homomorphism

$$\begin{tikzcd}
FX & FY \arrow[l, "\overline{F}f"'] \\
X \arrow[u, "c"] \arrow[ur, phantom, "\sqsubseteq"] & Y \arrow[l, "f"] \arrow[u, "d"']
\end{tikzcd}$$

in $\mathscr{K}\ell(T)$

$\overline{F} : \mathscr{K}\ell(T) \to \mathscr{K}\ell(T)$: lifting of F

- Forward partial execution [U. & Hasuo, ’14]

$$\begin{tikzcd}
FX & & F^{n+1}X \\
X \arrow[u, "c"] & \mapsto & F^nX \arrow[u, "\overline{F}^n c"']
\end{tikzcd}$$

in $\mathscr{K}\ell(T)$

$$\begin{tikzcd}
F^{n+1}X & \cdots \arrow[l, "\overline{F}^n c"'] & & FX \arrow[ll, "\overline{F}c"'] & FY \arrow[l, "\overline{F}f"'] \\
F^nX \arrow[u, "\overline{F}^n c"] & \cdots \arrow[l, "\overline{F}^{n-1}c"] & = & X \arrow[ll, "c"] \arrow[u, "c"] & Y \arrow[l, "f"] \arrow[u, "d"'] \arrow[ul, phantom, "\sqsubseteq"]
\end{tikzcd}$$

in $\mathscr{K}\ell(T)$

Addendum II: Categorical Buffered Simulation

- Kleisli Simulation [Hasuo, '06]
  - System as a coalgebra in Kleisli category

    $c : X \to FX$ in $\mathscr{K}\ell(T)$ whose homsets are order-enriched

  - Simulation as an oplax homomorphism

$$\begin{tikzcd} FX & FY \arrow[l, "\overline{F}f"'] \\ X \arrow[u, "c"] \arrow[ur, phantom, "\sqsubseteq"] & Y \arrow[l, "f"] \arrow[u, "d"'] \end{tikzcd}$$

in $\mathscr{K}\ell(T)$

$\overline{F} : \mathscr{K}\ell(T) \to \mathscr{K}\ell(T)$: lifting of F

- Forward partial execution [U. & Hasuo, '14]

$$\begin{tikzcd} FX \\ X \arrow[u, "c"] \end{tikzcd} \mapsto \begin{tikzcd} F^{n+1}X \\ F^nX \arrow[u, "\overline{F}^n c"] \end{tikzcd}$$

in $\mathscr{K}\ell(T)$

➡

$$\begin{tikzcd} F^{n+1}X & \cdots \arrow[l, "\overline{F}^n c"'] & FX \arrow[l, "\overline{F}c"'] & FY \arrow[l, "\overline{F}f"'] \\ F^nX \arrow[u, "\overline{F}^n c"] & \cdots \arrow[l, "\overline{F}^{n-1}c"] \arrow[u, phantom, "="] & X \arrow[l, "c"] \arrow[u, "c"] \arrow[ur, phantom, "\sqsubseteq"] & Y \arrow[l, "f"] \arrow[u, "d"'] \end{tikzcd}$$

in $\mathscr{K}\ell(T)$

  - Essentially the same as buffering one step

e.g. when $F = 1 + \Sigma \times (\,\cdot\,)$, $F^nX = \bigcup_{i \leq n} \Sigma^i \times X$

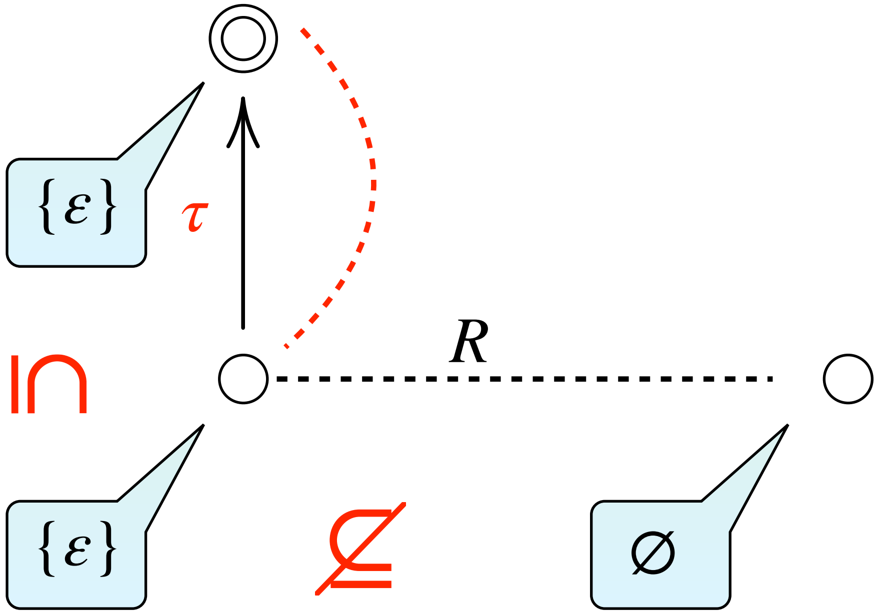
Addendum III: Unsound Weak Simulation up-to

[Sangiorgi & Milner, ’92]

$L^*(x') \subseteq L^*(x'')$ $L^*(y') \subseteq L^*(y'')$

$$\begin{tikzcd} x' \arrow[r, dash, dashed] & x'' \arrow[r, dash, dashed, "R"] & y'' \arrow[r, dash, dashed] & y' \\ x \arrow[u, "a"] \arrow[rrr, dash, dashed, "R"] & & & y \arrow[u, "a"'] \end{tikzcd}$$

$$\begin{tikzcd} x' \arrow[r, dash, dashed, "R"] & y' \\ x \arrow[u, "\tau"] \arrow[r, dash, dashed, "R"] & y \arrow[u, dotted, "\tau^*"'] \end{tikzcd}$$



Addendum IV: Towards Computation

- Conjecture: preorder-constrained simulation is not only sound but also **complete**

Conjecture (**completeness**):

When $\mathcal{Q}$ is closed under concatenation,

$$xRy \Longleftarrow \quad \forall w \in L^*(x)\,.\, \exists w' \in L^*(y)\,.\, w\mathcal{Q}w'$$

Hard to compute

Addendum IV: Towards Computation

- Conjecture: preorder-constrained simulation is not only sound but also **complete**

Conjecture (**completeness**):

When $\mathcal{Q}$ is closed under concatenation,

$$xRy \Longleftarrow \quad \forall w \in L^*(x)\,.\, \exists w' \in L^*(y)\,.\, w\mathcal{Q}w'$$

Hard to compute

- We may have to finitely restrict the size of queue

Addendum IV: Towards Computation

$S_\forall := \{\checkmark\} + \bigcup_{i \le M} \Sigma^i \times X \times Y$

queue of size M

$S_\exists :=$ {last_turn} $\times \bigcup_{i \le M} \Sigma^i \times X \times Y + \bigcup_{1 < i \le M} \Sigma^i \times X \times Y$

$\to_\forall \subseteq S_\forall \times S_\exists$ is given by:

$$\Big\{\big((w,x,y),(wa,x',y)\big) \mid x \xrightarrow{a} x'\Big\} \cup \Big\{\big((w,x,y),(\mathsf{last\_turn},w,x,y)\big) \mid x \in F_1\Big\}$$

$\to_\exists \subseteq S_\exists \times S_\forall$ is given by:

$$\Big\{\big((w,x',y),(w,x',y)\big) \mid |w| < M\Big\} \cup \Big\{\big((w,x',y),(\varepsilon,x',y')\big) \mid y \xrightarrow{w'^*} y', w@w'\Big\}$$

pass the turn when queue is not full

$$\cup \Big\{\big((\mathsf{last\_turn},w,x,y),\checkmark\big) \mid y \xrightarrow{w'^*} y' \in F_2, w@w'\Big\}$$

Addendum IV: Towards Computation

$$S_\forall := \{\checkmark\} + \bigcup_{i \le M} \Sigma^i \times X \times Y$$

queue of size M

$$S_\exists := \{\text{last\_turn}\} \times \bigcup_{i \le M} \Sigma^i \times X \times Y + \bigcup_{1 < i \le M} \Sigma^i \times X \times Y$$

$\to_\forall \subseteq S_\forall \times S_\exists$ is given by:

$$\Big\{\big((w,x,y),(wa,x',y)\big) \mid x \xrightarrow{a} x'\Big\} \cup \Big\{\big((w,x,y),(\text{last\_turn},w,x,y)\big) \mid x \in F_1\Big\}$$

$\to_\exists \subseteq S_\exists \times S_\forall$ is given by:

$$\Big\{\big((w,x',y),(w,x',y)\big) \mid |w| < M\Big\} \cup \Big\{\big((w,x',y),(\varepsilon,x',y')\big) \mid y \xrightarrow{w'^*} y', w \mathcal{Q} w'\Big\}$$

pass the turn when queue is not full

$$\cup \Big\{\big((\text{last\_turn},w,x,y),\checkmark\big) \mid y \xrightarrow{w'^*} y' \in F_2, w \mathcal{Q} w'\Big\}$$

- When M is fixed, solvable in polynomial time

Addendum IV: Towards Computation

$S_\forall := \{\checkmark\} + \bigcup_{i \le M} \Sigma^i \times X \times Y$

queue of size M

$S_\exists := \{\text{last\_turn}\} \times \bigcup_{i \le M} \Sigma^i \times X \times Y + \bigcup_{1 < i \le M} \Sigma^i \times X \times Y$

$\to_\forall \subseteq S_\forall \times S_\exists$ is given by:

$$\Big\{\big((w, x, y), (wa, x', y)\big) \mid x \overset{a}{\to} x'\Big\} \cup \Big\{\big((w, x, y), (\text{last\_turn}, w, x, y)\big) \mid x \in F_1\Big\}$$

$\to_\exists \subseteq S_\exists \times S_\forall$ is given by:

$$\Big\{\big((w, x', y), (w, x', y)\big) \mid |w| < M\Big\} \cup \Big\{\big((w, x', y), (\varepsilon, x', y')\big) \mid y \overset{w'^*}{\to} y', w @ w'\Big\}$$

pass the turn when queue is not full

$$\cup \Big\{\big((\text{last\_turn}, w, x, y), \checkmark\big) \mid y \overset{w'^*}{\to} y' \in F_2, w @ w'\Big\}$$

- When M is fixed, solvable in polynomial time
- Bigger M → more simulations & higher time complexity

Addendum V: Examples for $\mathcal{Q}$-trace Inclusion

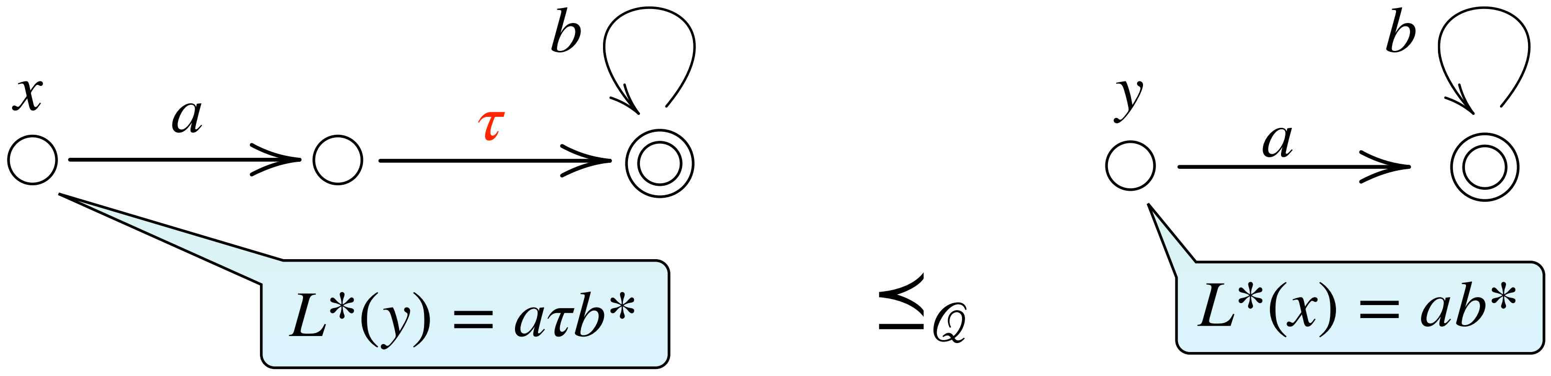
$$\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$$

$$\forall w \in L^*(x) \,.\, \exists w' \in L^*(y) \,.\, w\mathcal{Q}w'$$

- When $\Sigma = \{\tau\} + \Sigma'$ and $w\mathcal{Q}w' \overset{\text{def}}{\iff} \text{remove}_\tau(w) = \text{remove}_\tau(w')$ and $|w| \geq |w'|$

$$\mathcal{Q}\text{-trace inclusion} \iff \forall w \in L^*(x) \,.\, \exists w' \in L^*(y) \,.\, \text{remove}_\tau(w) = \text{remove}_\tau(w') \text{ and } |w| \geq |w'|$$

(weak trace inclusion & compare distance to accepting state)



Addendum V: Examples for $\mathcal{Q}$-trace Inclusion

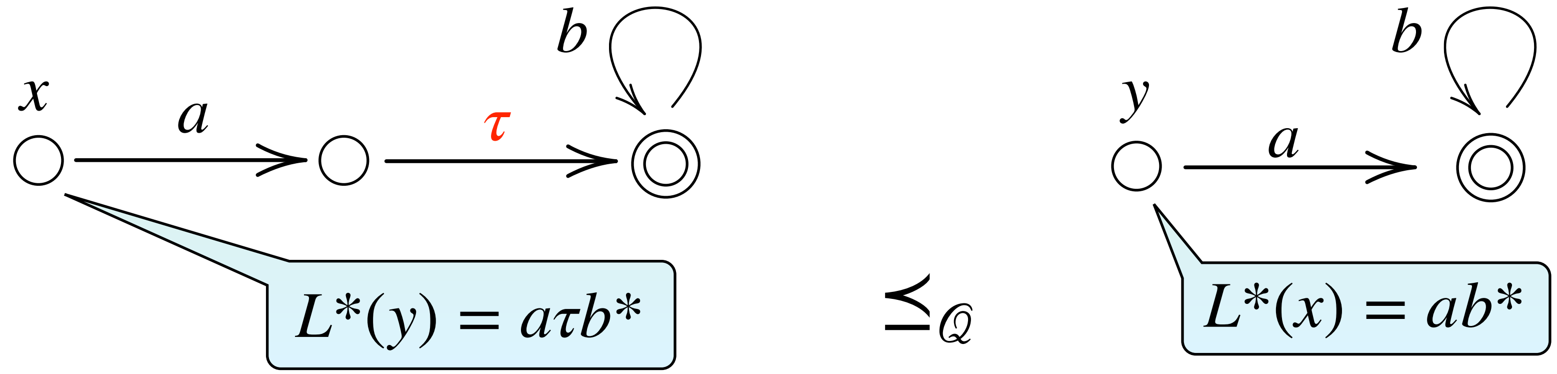
$$\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$$

$$\forall w \in L^*(x) \,.\, \exists w' \in L^*(y) \,.\, w \mathcal{Q} w'$$

- When $\Sigma = \{\tau\} + \Sigma'$ and $w \mathcal{Q} w' \overset{\text{def}}{\iff} \text{remove}_\tau(w) = \text{remove}_\tau(w')$ and $|w| \geq |w'|$

$$\mathcal{Q}\text{-trace inclusion} \iff \forall w \in L^*(x) \,.\, \exists w' \in L^*(y) \,.\, \text{remove}_\tau(w) = \text{remove}_\tau(w') \text{ and } |w| \geq |w'|$$

(weak trace inclusion & compare distance to accepting state)

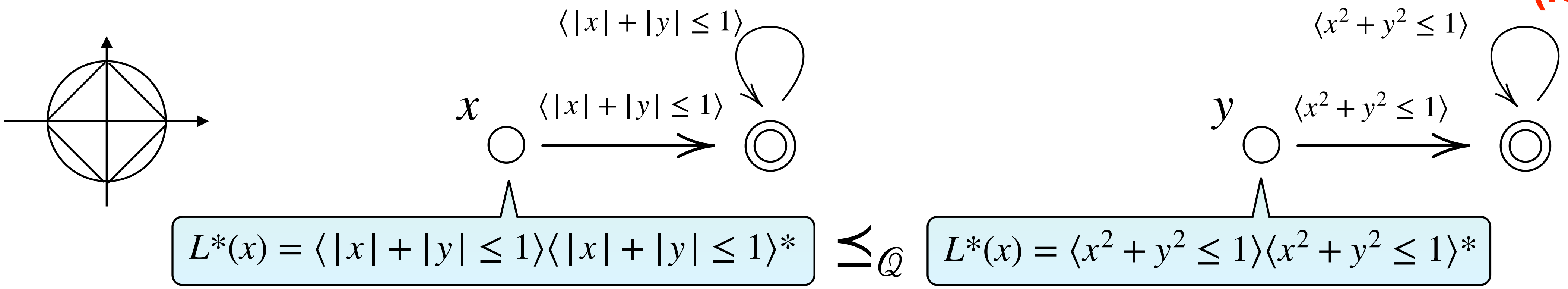


$L^*(y) = a\tau b^*$ $\preceq_{\mathcal{Q}}$ $L^*(x) = ab^*$

- When $\Sigma = \mathscr{P}\mathbb{R}^m$ and $a_1 \ldots a_k \mathcal{Q} a'_1 \ldots a'_{k'} \overset{\text{def}}{\iff} k = k'$ and $\forall i \,.\, a_i \subseteq a'_i$

$$\mathcal{Q}\text{-trace inclusion} \iff \forall a_1 \ldots a_k \in L^*(x) \,.\, \exists a'_1 \ldots a'_k \in L^*(y) \,.\, \forall i \,.\, a_i \subseteq a'_i$$

(letter-wise inclusion)



$L^*(x) = \langle |x| + |y| \leq 1 \rangle \langle |x| + |y| \leq 1 \rangle^*$ $\preceq_{\mathcal{Q}}$ $L^*(x) = \langle x^2 + y^2 \leq 1 \rangle \langle x^2 + y^2 \leq 1 \rangle^*$

Addendum V: Examples for $\mathcal{Q}$-trace Inclusion

$$\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$$

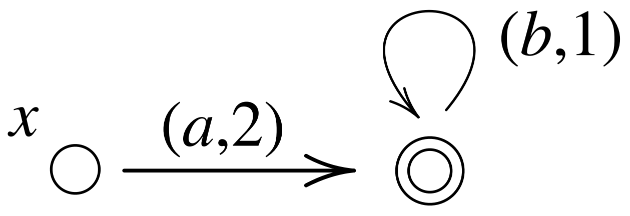
$$\forall w \in L^*(x) \,.\, \exists w' \in L^*(y) \,.\, w \mathcal{Q} w'$$

- When $\Sigma = \Sigma' \times \mathbb{N}$ and

  $(a_1, n_1)\ldots(a_k, n_k)\mathcal{Q}(a'_1, n'_1)\ldots(a'_{k'}, n'_{k'}) \overset{\mathrm{def}}{\iff} k = k', a_1\ldots a_k = a'_1\ldots a'_{k'}$ and $\sum_i n_i \geq \sum_j n'_j$

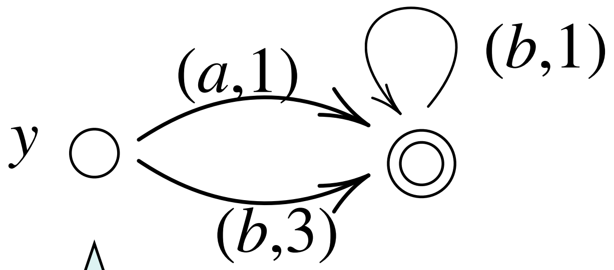
$$\mathcal{Q}\text{-trace inclusion} \iff \forall a_1\ldots a_k \in \Sigma'^* \,.\, \min_{x\xrightarrow{a_1,n_1}\cdots\xrightarrow{a_k,n_k}\checkmark} \sum_i n_i \geq \min_{y\xrightarrow{a'_1,n_1}\cdots\xrightarrow{a'_k,n_k}\checkmark} \sum_i n'_i$$

(quantitative language inclusion)



$L^*(x) = (a,2)(b,1)^*$

$\preceq_{\mathcal{Q}}$



$L^*(y) = (a,1)(b,1)^* + (b,3)(b,1)^*$

$a \mapsto 2$
$ab \mapsto 3$
$abb \mapsto 4$
$\vdots$

$a \mapsto 1$
$ab \mapsto 2$
$abb \mapsto 3$
$\vdots$

$b \mapsto 3$
$bb \mapsto 4$
$bbb \mapsto 5$
$\vdots$

Addendum VI:
Preorder-Constrained Simulation for Deterministic Unlabeled Systems

[M., PhD thesis]

Definition (when no deadend):

Let $Q \subseteq \mathbb{N} \times \mathbb{N}$ be a preorder. A ***Q-constrained simulation*** from $(c : X \to X, F_1 \subseteq X)$ to $(d : Y \to Y, F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$
- $\exists m > 0, n \in \mathbb{N}\,.\ x \underbrace{\to \cdots \to}_{m} x'', y \underbrace{\to \cdots \to}_{n} y', x'' R y'$ and mQn

Addendum VI:
Preorder-Constrained Simulation for Deterministic Unlabeled Systems

[M., PhD thesis]

Definition (when no deadend):

Let $Q \subseteq \mathbb{N} \times \mathbb{N}$ be a preorder. A ***Q-constrained simulation*** from $(c : X \to X, F_1 \subseteq X)$ to $(d : Y \to Y, F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$$\forall (x, y) \in R\,. \quad \begin{array}{l} \text{- } x \in F_1 \implies y \in F_2 \\ \text{- } \exists m > 0, n \in \mathbb{N}\,.\, x \underbrace{\to \cdots \to}_{m} x'', y \underbrace{\to \cdots \to}_{n} y', x'' R y' \ \text{ and } \ mQn \end{array}$$

$$\begin{tikzcd} x \arrow[r, dash, dashed, "R"] & y \end{tikzcd}$$

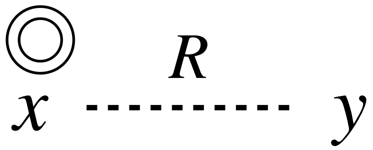
Addendum VI: Preorder-Constrained Simulation for Deterministic Unlabeled Systems

[M., PhD thesis]

Definition (when no deadend):

Let $Q \subseteq \mathbb{N} \times \mathbb{N}$ be a preorder. A ***Q*-constrained simulation** from $(c : X \to X, F_1 \subseteq X)$ to $(d : Y \to Y, F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$$\forall (x, y) \in R\,. \quad \begin{array}{l} \text{- } x \in F_1 \implies y \in F_2 \\ \text{- } \exists m > 0, n \in \mathbb{N}\,.\ x \underbrace{\to \cdots \to}_{m} x'', y \underbrace{\to \cdots \to}_{n} y', x'' R y' \ \text{ and } \ mQn \end{array}$$



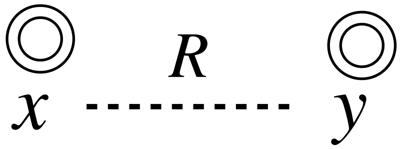
Addendum VI:
Preorder-Constrained Simulation for Deterministic Unlabeled Systems

[M., PhD thesis]

Definition (when no deadend):

Let $Q \subseteq \mathbb{N} \times \mathbb{N}$ be a preorder. A ***Q-constrained simulation*** from $(c : X \to X, F_1 \subseteq X)$ to $(d : Y \to Y, F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$$\forall (x, y) \in R\,. \quad \begin{array}{l} \text{- } x \in F_1 \implies y \in F_2 \\ \text{- } \exists m > 0, n \in \mathbb{N}\,.\, x \underbrace{\to \cdots \to}_{m} x'', y \underbrace{\to \cdots \to}_{n} y', x''Ry' \text{ and } mQn \end{array}$$



Addendum VI:
Preorder-Constrained Simulation for Deterministic Unlabeled Systems

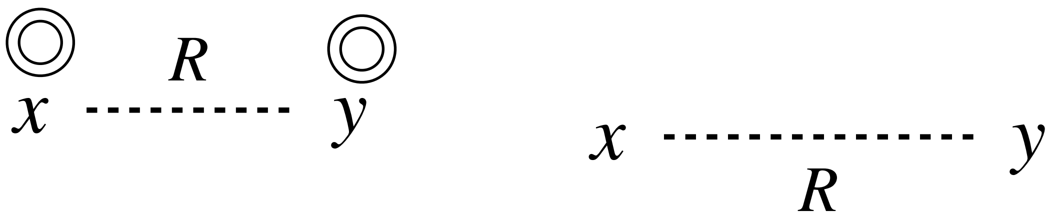
[M., PhD thesis]

Definition (when no deadend):

Let $Q \subseteq \mathbb{N} \times \mathbb{N}$ be a preorder. A ***Q-constrained simulation*** from $(c : X \to X, F_1 \subseteq X)$ to $(d : Y \to Y, F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$
- $\exists m > 0, n \in \mathbb{N}\,.\; x \underbrace{\to \cdots \to}_{m} x'', y \underbrace{\to \cdots \to}_{n} y', x'' R y'$ and mQn



Addendum VI: Preorder-Constrained Simulation for Deterministic Unlabeled Systems

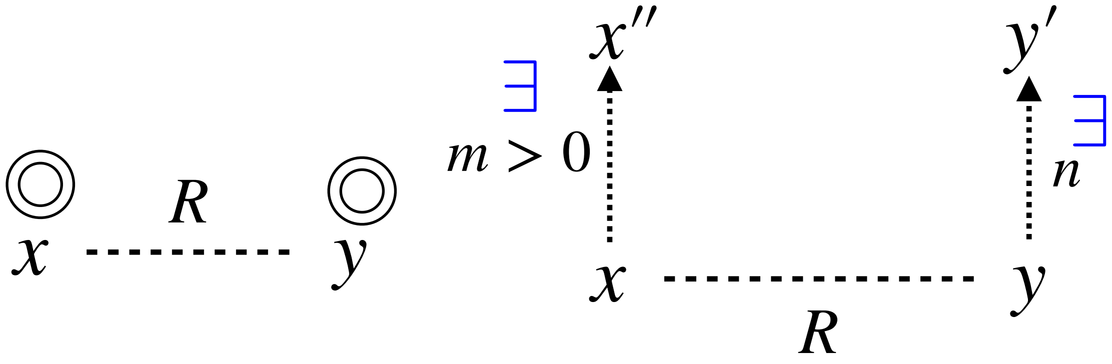
[M., PhD thesis]

Definition (when no deadend):

Let $Q \subseteq \mathbb{N} \times \mathbb{N}$ be a preorder. A ***Q-constrained simulation*** from $(c : X \to X, F_1 \subseteq X)$ to $(d : Y \to Y, F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$
- $\exists m > 0, n \in \mathbb{N}\,.\; x \underbrace{\to \cdots \to}_{m} x'', y \underbrace{\to \cdots \to}_{n} y', x'' R y'$ and mQn



Addendum VI:
Preorder-Constrained Simulation for Deterministic Unlabeled Systems

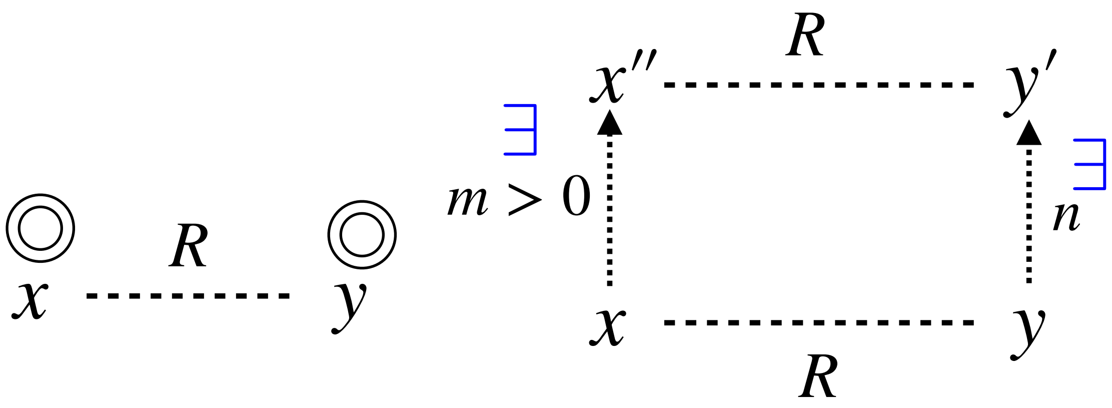
[M., PhD thesis]

Definition (when no deadend):

Let $Q \subseteq \mathbb{N} \times \mathbb{N}$ be a preorder. A ***Q-constrained simulation*** from $(c : X \to X, F_1 \subseteq X)$ to $(d : Y \to Y, F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$
- $\exists m > 0, n \in \mathbb{N}\,.\; x \underbrace{\to \cdots \to}_{m} x'', y \underbrace{\to \cdots \to}_{n} y', x'' R y'$ and mQn



Addendum VI: Preorder-Constrained Simulation for Deterministic Unlabeled Systems

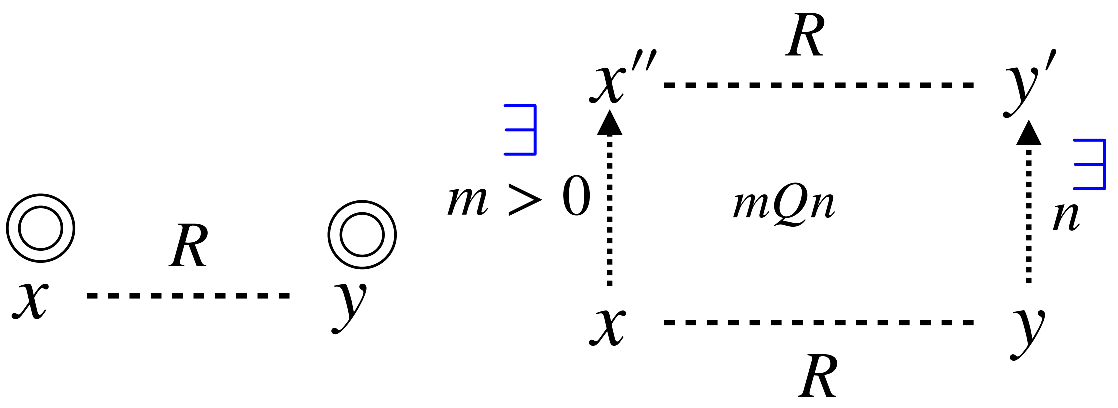
[M., PhD thesis]

Definition (when no deadend):

Let $Q \subseteq \mathbb{N} \times \mathbb{N}$ be a preorder. A ***Q-constrained simulation*** from $(c : X \to X, F_1 \subseteq X)$ to $(d : Y \to Y, F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$\forall (x, y) \in R\,.$

- $x \in F_1 \implies y \in F_2$
- $\exists m > 0, n \in \mathbb{N}\,.\; x \underbrace{\to \cdots \to}_{m} x'', y \underbrace{\to \cdots \to}_{n} y', x'' R y'$ and mQn



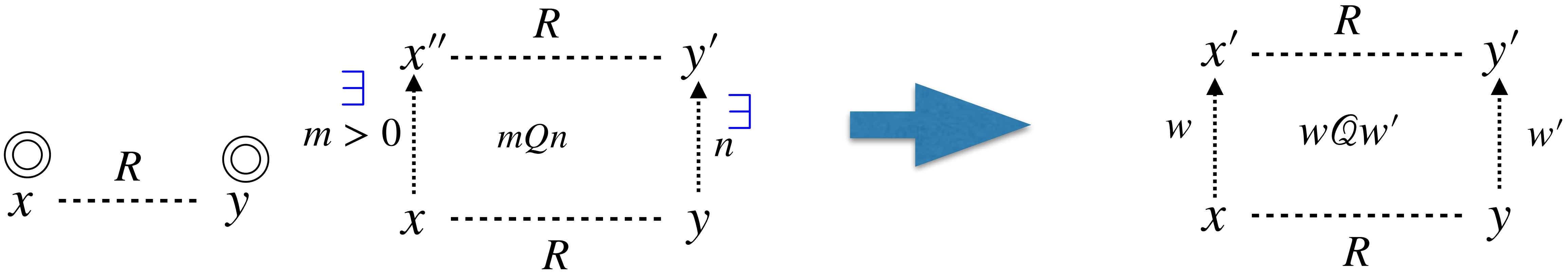
Addendum VI: Preorder-Constrained Simulation for Deterministic Unlabeled Systems

[M., PhD thesis]

Definition (when no deadend):

Let $Q \subseteq \mathbb{N} \times \mathbb{N}$ be a preorder. A ***Q-constrained simulation*** from $(c : X \to X, F_1 \subseteq X)$ to $(d : Y \to Y, F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$$\forall (x, y) \in R\,. \quad \begin{array}{l} \text{- } x \in F_1 \implies y \in F_2 \\ \text{- } \exists m > 0, n \in \mathbb{N}\,.\; x \underbrace{\to \cdots \to}_{m} x'', y \underbrace{\to \cdots \to}_{n} y', x'' R y' \text{ and } mQn \end{array}$$



- Goal: generalization to nondeterministic automata

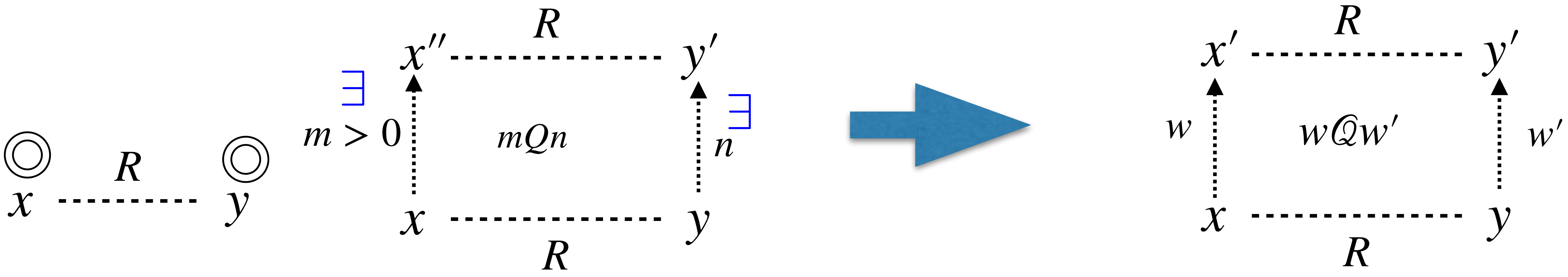
Addendum VI:
Preorder-Constrained Simulation for Deterministic Unlabeled Systems

[M., PhD thesis]

Definition (when no deadend):

Let $Q \subseteq \mathbb{N} \times \mathbb{N}$ be a preorder. A ***Q-constrained simulation*** from $(c : X \to X, F_1 \subseteq X)$ to $(d : Y \to Y, F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

$$\forall (x, y) \in R\,.\quad \begin{array}{l} \text{- } x \in F_1 \implies y \in F_2 \\ \text{- } \exists m > 0, n \in \mathbb{N}\,.\; x \underbrace{\to \cdots \to}_{m} x'', y \underbrace{\to \cdots \to}_{n} y', x'' R y' \text{ and } mQn \end{array}$$



- Goal: generalization to nondeterministic automata

- Difficulty: both m and n are chosen by ∃

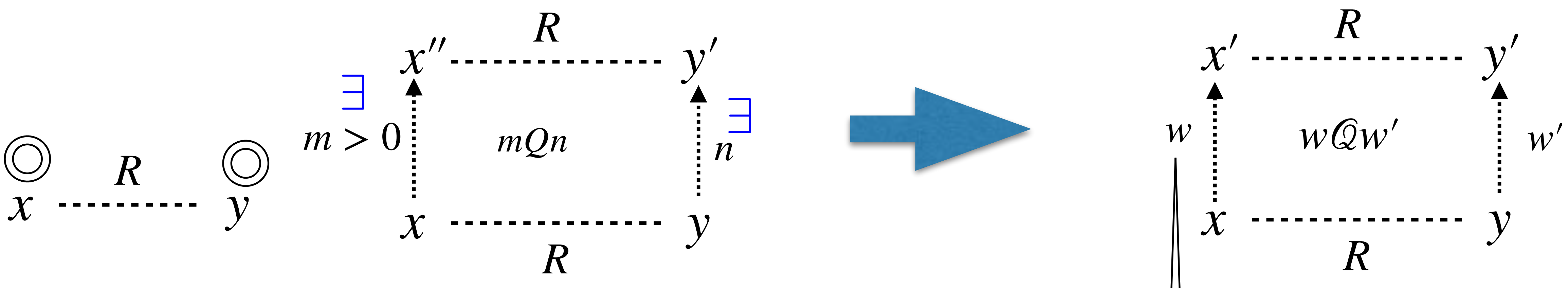
Addendum VI: Preorder-Constrained Simulation for Deterministic Unlabeled Systems

[M., PhD thesis]

Definition (when no deadend):

Let $Q \subseteq \mathbb{N} \times \mathbb{N}$ be a preorder. A ***Q-constrained simulation*** from $(c : X \to X, F_1 \subseteq X)$ to $(d : Y \to Y, F_2 \subseteq Y)$ is a relation $R \subseteq X \times Y$ such that

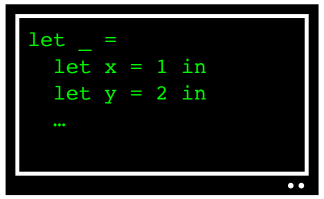
$$\forall (x, y) \in R\,. \quad \begin{array}{l} \text{- } x \in F_1 \implies y \in F_2 \\ \text{- } \exists m > 0, n \in \mathbb{N}\,.\; x \underbrace{\to \cdots \to}_{m} x'', y \underbrace{\to \cdots \to}_{n} y', x''Ry' \;\text{ and }\; mQn \end{array}$$



- Goal: generalization to nondeterministic automata
- Difficulty: both m and n are chosen by ∃

We wish:
nondeterminism is resolved by ∀,
length is determined by ∃

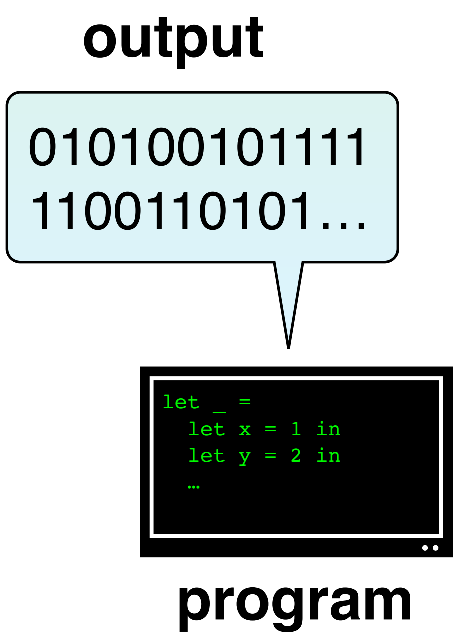
Addendum VII: Verification via Behavioral Inclusion



```
let _ =
  let x = 1 in
  let y = 2 in
  …
```

program

Addendum VII: Verification via Behavioral Inclusion



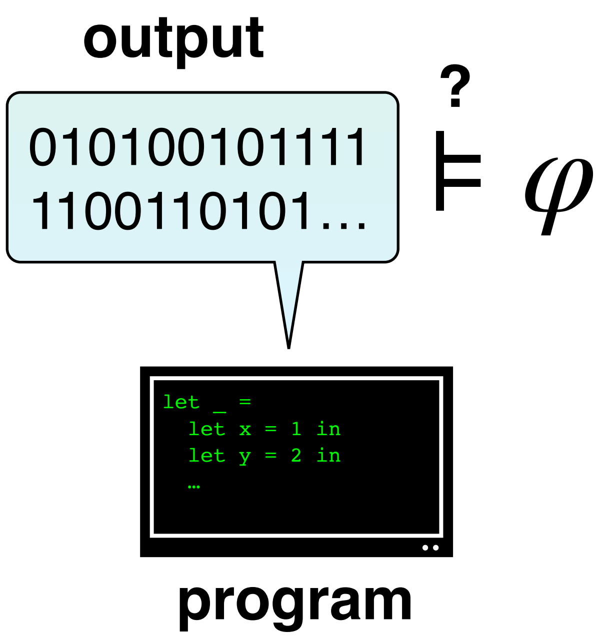
output

010100101111
1100110101…

```
let _ =
  let x = 1 in
  let y = 2 in
  …
```

program

Addendum VII: Verification via Behavioral Inclusion



output

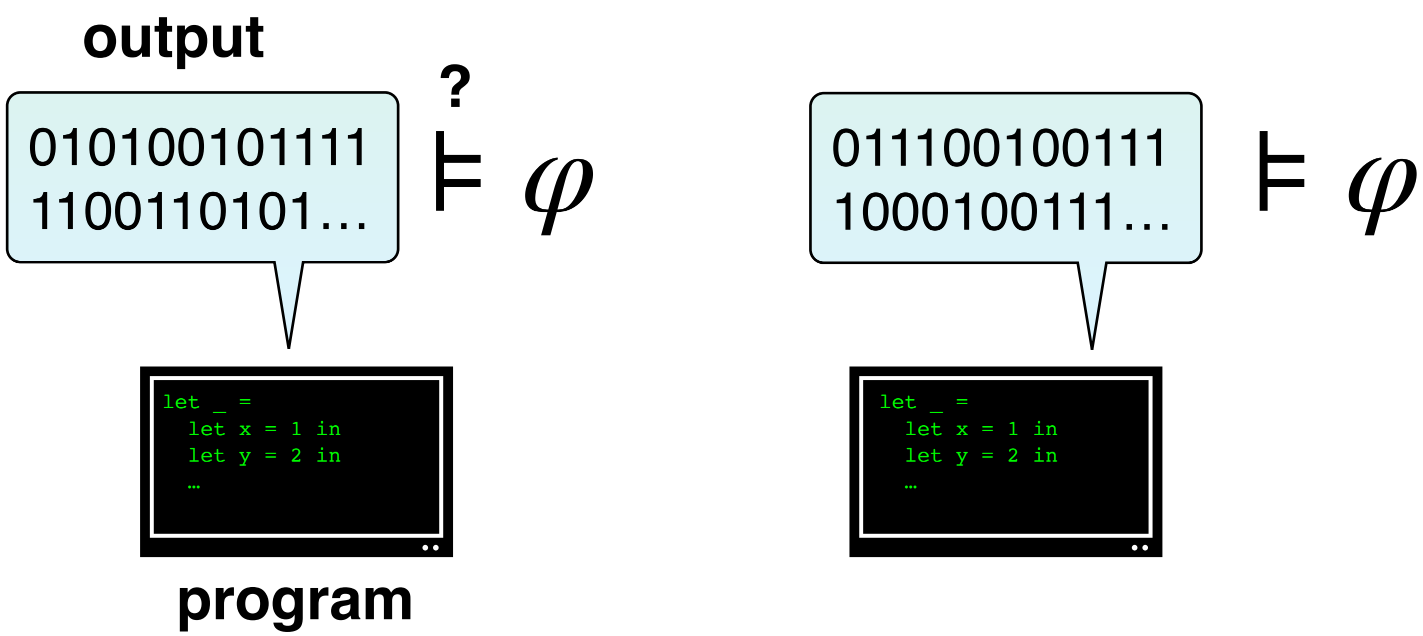
010100101111
1100110101…

$\overset{?}{\models} \varphi$

```
let _ =
  let x = 1 in
  let y = 2 in
  …
```

program

Addendum VII: Verification via Behavioral Inclusion



output

010100101111
1100110101…

$\overset{?}{\models} \varphi$

```
let _ =
  let x = 1 in
  let y = 2 in
  …
```

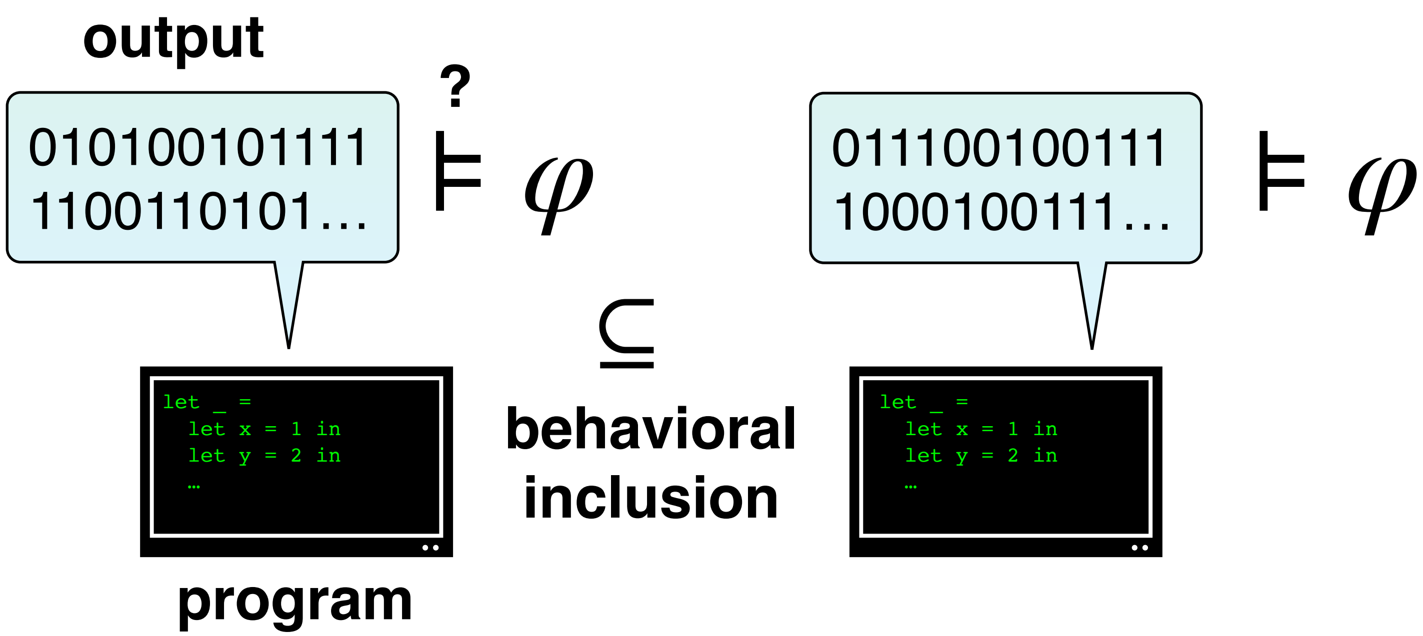
program

011100100111
1000100111…

$\models \varphi$

```
let _ =
  let x = 1 in
  let y = 2 in
  …
```

Addendum VII: Verification via Behavioral Inclusion



output

010100101111
1100110101…

$\overset{?}{\models} \varphi$

```
let _ =
  let x = 1 in
  let y = 2 in
  …
```

program

$\subseteq$

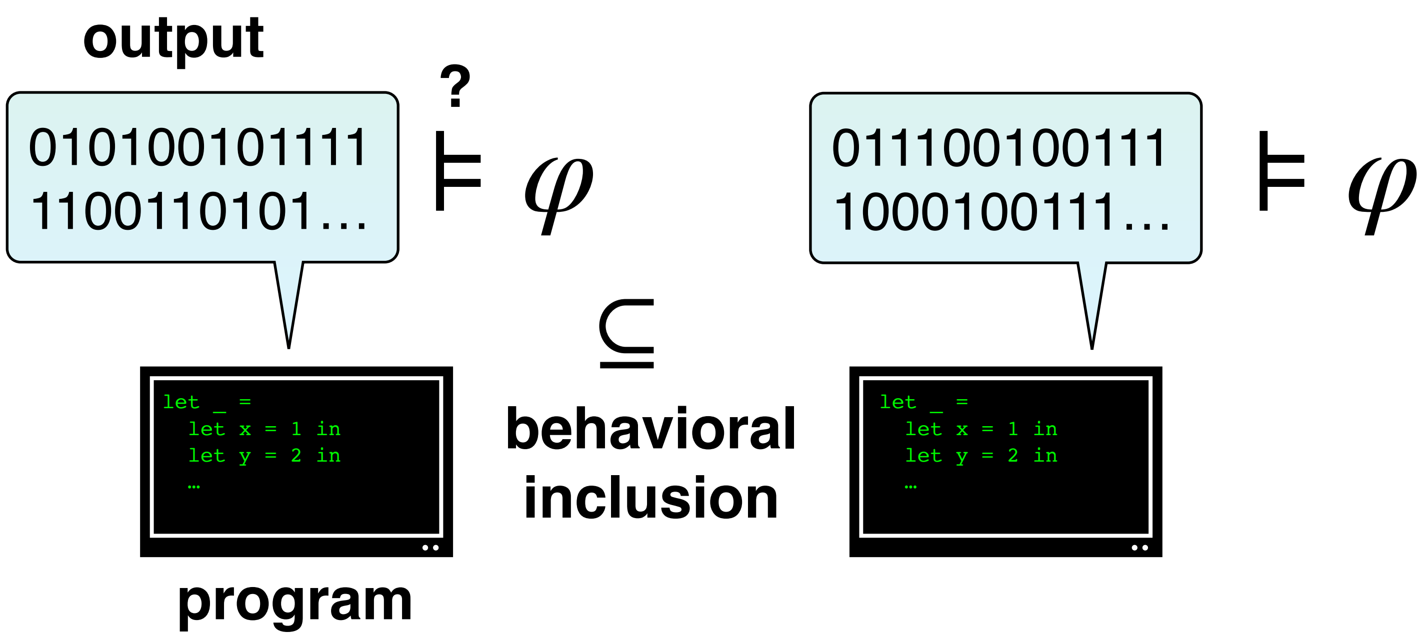
behavioral inclusion

011100100111
1000100111…

$\models \varphi$

```
let _ =
  let x = 1 in
  let y = 2 in
  …
```

Addendum VII: Verification via Behavioral Inclusion



output

010100101111
1100110101… $\overset{?}{\models} \varphi$

011100100111
1000100111… $\models \varphi$

```
let _ =
  let x = 1 in
  let y = 2 in
  …
```

program

$\subseteq$

behavioral inclusion

```
let _ =
  let x = 1 in
  let y = 2 in
  …
```

- Behavioral inclusion between nondeterministic automata
  - Two formalizations: **trace inclusion** and **simulation**