A Theory of Nondeterministic and Probabilistic Processes

(joint work with Yuxin Deng, Rob van Glabbeek, Carroll Morgan)

Computing Laboratory, Oxford February 2011
Goal:

An introduction to:

- *Characterising Testing Preorders for Finite Probabilistic Processes*, Lmcs 2008

All authored by: Yuxin Deng, Rob van Glabbeek, Matthew Hennessy, Carroll Morgan, Chenyi Zhang

Previous work:

Goal:

An introduction to:

- *Characterising Testing Preorders for Finite Probabilistic Processes*, Lmcs 2008

All authored by: Yuxin Deng, Rob van Glabbeek, Matthew Hennessy, Carroll Morgan, Chenyi Zhang

Previous work:

Outline

**Background**  why bother ?

Probabilistic Labelled Transition Systems

**Testing Theory**

Simulations

**Results**  some
Outline

**Background** why bother?

Probabilistic Labelled Transition Systems

Testing Theory

Simulations

Results some
Background

Goal: Specification and proof methodologies for probabilistic concurrent systems

Nondeterminism + Probability – why necessary?

- “Nondeterminism” intrinsic to specification development à la CSP
  - underspecified components expressed using “nondeterminism”

\[
\text{COMP} \sqcap \text{OPTION} \leq \text{COMP} \\
\text{underspecified} \quad \text{more specified}
\]

- Analysis of concurrent systems requires “nondeterminism”

□ - internal choice of CSP
Background

Goal: Specification and proof methodologies for probabilistic concurrent systems

Nondeterminism + Probability – why necessary?

▶ “Nondeterminism” intrinsic to specification development à la CSP
  ▶ underspecified components expressed using “nondeterminism”

\[
\text{COMP} \sqcap \text{OPTION} \leq \text{COMP}
\]

underspecified \hspace{2cm} more specified

▶ Analysis of concurrent systems requires “nondeterminism”

\(-\) internal choice of CSP
Background

Goal: Specification and proof methodologies for probabilistic concurrent systems

Nondeterminism + Probability – why necessary?

- “Nondeterminism” intrinsic to specification development à la CSP
  - underspecified components expressed using “nondeterminism”

\[
\text{COMP } \square \text{ OPTION } \leq \text{ COMP}
\]

\[\text{underspecified} \quad \text{more specified}\]

- Analysis of concurrent systems requires “nondeterminism”

\[\square - \text{internal choice of CSP}\]
Background

Goal: Specification and proof methodologies for probabilistic concurrent systems

Nondeterminism + Probability – why necessary?

- “Nondeterminism” intrinsic to specification development à la CSP
  - underspecified components expressed using “nondeterminism”

\[
\text{COMP } \triangleleft \text{ OPTION } \leq \text{ COMP}
\]

  underspecified     more specified

- Analysis of concurrent systems requires “nondeterminism”

\(\triangleleft\) - internal choice of CSP
Analysis of concurrent systems

Sys1:

Sys1 ⇐ (new s)(A | Sw)
A ⇐ up.U + s?down.D
Sw ⇐ s!stop

Sys2:

Sys2 ⇐ (new s)(B | Sw)
B ⇐ s?(up.U + down.D) + s?down.D
Sw ⇐ s!stop
Analysis of concurrent systems

In CSP theory:

\[ Sys1 \approx Sys2 \]

semantically equivalent

Both equivalent to the nondeterministic

\[ \tau.(up.U + down.D) + \tau.down.D \]

concurrency = nondeterminism + interleaving

probabilistic concurrency = probability + nondeterminism + interleaving
Analysis of concurrent systems

In CSP theory:

\[
\text{Sys}_1 \approx \text{Sys}_2
\]

semantically equivalent

Both equivalent to the nondeterministic

\[
\tau.(up.U + down.D) + \tau.down.D
\]

concurrency = nondeterminism + interleaving

probabilistic concurrency = probability + nondeterminism + interleaving
Analysis of concurrent systems

In CSP theory:

\[ \text{Sys1} \approx \text{Sys2} \]

semantically equivalent

Both equivalent to the nondeterministic

\[ \tau. (up.U + down.D) + \tau. down.D \]

concurrency = nondeterminism + interleaving

probabilistic concurrency = probability + nondeterminism + interleaving
Analysis of concurrent systems

In CSP theory:

\[ \text{Sys}_1 \approx \text{Sys}_2 \]

semantically equivalent

Both equivalent to the nondeterministic

\[ \tau.(up.U + down.D) + \tau.down.D \]

concurrency = nondeterminism + interleaving

probabilistic concurrency = probability + nondeterminism + interleaving
Outline

Background  why bother ?

Probabilistic Labelled Transition Systems

Testing Theory

Simulations

Results  some
Nondeterministic processes

Intensional semantics:

A process is a state in an LTS

Labelled Transition Systems:

\[ \langle S, \text{Act}_{\tau}, \rightarrow \rangle \]

- \( S \) - states
- \( \rightarrow \subseteq S \times \text{Act}_{\tau} \times S \)

\( s_1 \xrightarrow{\mu} s_2 \): process \( s_1 \) can perform action \( \mu \) and continue as \( s_2 \)

\( s_1 \xrightarrow{\tau} s_2 \) special internal action
Nondeterministic processes

Intensional semantics:

A process is a state in an LTS

Labelled Transition Systems:

\[ \langle S, \text{Act}_\tau, \rightarrow \rangle \]

- \( S \) - states
- \( \rightarrow \subseteq S \times \text{Act}_\tau \times S \)

\( s_1 \xrightarrow{\mu} s_2 \): process \( s_1 \) can perform action \( \mu \) and continue as \( s_2 \)

\( s_1 \xrightarrow{\tau} s_2 \) special internal action
Probabilistic processes \textit{a la} Roberto Segala

Intensional semantics:

\begin{center}
A process is a \textit{distribution} in an pLTS
\end{center}

Probabilistic Labelled Transition Systems:

\[ \langle S, \text{Act}_\tau, \rightarrow \rangle \]

\begin{itemize}
\item $S$ - states
\item $\rightarrow \subseteq S \times \text{Act}_\tau \times D(S)$
\end{itemize}

$D(S)$: Mappings $\Delta : S \rightarrow [0,1]$ with $\sum_{s \in S} \Delta(s) = 1$

$s_1 \xrightarrow{\mu} \Delta$: process $s_1$

\begin{itemize}
\item can perform action $\mu$
\item with probability $\Delta(s_2)$ it continues as process $s_2$
\end{itemize}
Probabilistic processes a la Roberto Segala

Intensional semantics:

A process is a distribution in an pLTS

Probabilistic Labelled Transition Systems:

\( \langle S, \text{Act}_\tau, \rightarrow \rangle \)

- \( S \) - states
- \( \rightarrow \subseteq S \times \text{Act}_\tau \times \mathcal{D}(S) \)

\( \mathcal{D}(S) \): Mappings \( \Delta : S \rightarrow [0, 1] \) with \( \sum_{s \in S} \Delta(s) = 1 \)

\( s_1 \xrightarrow{\mu} \Delta \): process \( s_1 \)

- can perform action \( \mu \)
- with probability \( \Delta(s_2) \) it continues as process \( s_2 \)
Probabilistic processes a la Roberto Segala

Intensional semantics:

A process is a distribution in an pLTS

Probabilistic Labelled Transition Systems:

\[ \langle S, \text{Act}_\tau, \rightarrow \rangle \]

- \( S \) - states
- \( \rightarrow \subseteq S \times \text{Act}_\tau \times \mathcal{D}(S) \)

\( \mathcal{D}(S) \): Mappings \( \Delta : S \rightarrow [0, 1] \) with \( \sum_{s \in S} \Delta(s) = 1 \)

\( s_1 \xrightarrow{\mu} \Delta \): process \( s_1 \)

- can perform action \( \mu \)
- with probability \( \Delta(s_2) \) it continues as process \( s_2 \)
Example: Tossing coins

Fair coin: $F = \mu X.F(X)$

Biased coin: $B = \mu X.B(X)$
Example: tossing coins

Picks a coin once: \( BF = (\mu X. F(X)) \sqcap (\mu X. B(X)) \)
Example: tossing coins

Picks a coin after every toss: $\mu_{BF} = \mu X. F(X) \sqcap B(X)$
Example process

What is the probability of action $a$ happening?

What is the probability of passing the test $\overline{a.\omega}$?  

Pessimistic view $\frac{1}{2}$

Optimistic view 1
What is the probability of action $a$ happening?

What is the probability of passing the test $\overline{a}.\omega$? 

Pessimistic view $\frac{1}{2}$

Optimistic view $1$
Example process

- What is the probability of action $a$ happening?
- What is the probability of passing the test $\overline{a}\omega$? Please do an $a$
- Pessimistic view $\frac{1}{2}$
- Optimistic view 1
Example process

What is the probability of action \( a \) happening?

What is the probability of passing the test \( \overline{a}.\omega \)?

Pessimistic view \( \frac{1}{2} \)

Optimistic view 1
Example process

- What is the probability of action $a$ happening?
- What is the probability of passing the test $\bar{a}.\omega$? Please do an $a$
- Pessimistic view $\frac{1}{2}$
- Optimistic view $1$
Outline

Background why bother?

Probabilistic Labelled Transition Systems

Testing Theory

Simulations

Results some
Testing scenario

- a set of processes $\mathcal{Proc}$
- a set of tests $\mathcal{T}$
- a set of ordered outcomes $\mathcal{O}$
- $\text{Apply} : \mathcal{T} \times \mathcal{Proc} \rightarrow \mathcal{P}^+(\mathcal{O})$ – the non-empty set of possible results of applying a test to a process

Comparing sets of outcomes:

- $\mathcal{O}_1 \sqsubseteq_{Ho} \mathcal{O}_2$ if for every $o_1 \in \mathcal{O}_1$ there exists some $o_2 \in \mathcal{O}_2$ such that $o_1 \leq o_2$
- $\mathcal{O}_1 \sqsubseteq_{Sm} \mathcal{O}_2$ if for every $o_2 \in \mathcal{O}_2$ there exists some $o_1 \in \mathcal{O}_1$ such that $o_1 \leq o_2$

$o_1 \leq o_2$ : means $o_2$ is as least as good as $o_1$
Testing scenario

- a set of processes $\mathcal{P}_{roc}$
- a set of tests $\mathcal{T}$
- a set of ordered outcomes $\mathcal{O}$

$\textit{Apply} : \mathcal{T} \times \mathcal{P}_{roc} \rightarrow \mathcal{P}^+(\mathcal{O})$ – the non-empty set of possible results of applying a test to a process

Comparing sets of outcomes:

- $\mathcal{O}_1 \sqsubseteq_{Ho} \mathcal{O}_2$ if for every $o_1 \in \mathcal{O}_1$ there exists some $o_2 \in \mathcal{O}_2$ such that $o_1 \leq o_2$
- $\mathcal{O}_1 \sqsubseteq_{Sm} \mathcal{O}_2$ if for every $o_2 \in \mathcal{O}_2$ there exists some $o_1 \in \mathcal{O}_1$ such that $o_1 \leq o_2$

$o_1 \leq o_2$ : means $o_2$ is as least as good as $o_1$
Testing preorders

- $P \sqsubseteq_{\text{may}} Q$ if $\text{Apply}(T, P) \sqsubseteq_{\text{Ho}} \text{Apply}(T, Q)$ for every test $T$
- $P \sqsubseteq_{\text{must}} Q$ if $\text{Apply}(T, P) \sqsubseteq_{\text{Sm}} \text{Apply}(T, Q)$ for every test $T$

Standard testing:
Use as outcomes $\mathcal{O} = \{\top, \bot\}$ with $\bot \leq \top$

Comparisons:
Possible outcome sets: $\{\bot\} \quad \{\bot, \top\} \quad \{\top\}$
- May: $\{\bot\} <_{\text{Ho}} \{\bot, \top\} =_{\text{Ho}} \{\top\}$
- Must: $\{\bot\} =_{\text{Sm}} \{\bot, \top\} <_{\text{Sm}} \{\top\}$
Testing preorders

- \( P \sqsubseteq_{\text{may}} Q \) if \( \text{Apply}(T, P) \sqsubseteq_{\text{Ho}} \text{Apply}(T, Q) \) for every test \( T \)
- \( P \sqsubseteq_{\text{must}} Q \) if \( \text{Apply}(T, P) \sqsubseteq_{\text{Sm}} \text{Apply}(T, Q) \) for every test \( T \)

Standard testing:
Use as outcomes \( \mathcal{O} = \{\top, \bot\} \) with \( \bot \leq \top \)

Comparisons:
Possible outcome sets: \{\bot\} \quad \{\bot, \top\} \quad \{\top\}
- May: \{\bot\} <_{\text{Ho}} \{\bot, \top\} =_{\text{Ho}} \{\top\}
- Must: \{\bot\} =_{\text{Sm}} \{\bot, \top\} <_{\text{Sm}} \{\top\}
Testing preorders

- \( P \sqsubseteq_{\text{may}} Q \) if \( \text{Apply}(T, P) \sqsubseteq_{\text{Ho}} \text{Apply}(T, Q) \) for every test \( T \)
- \( P \sqsubseteq_{\text{must}} Q \) if \( \text{Apply}(T, P) \sqsubseteq_{\text{Sm}} \text{Apply}(T, Q) \) for every test \( T \)

Standard testing:
Use as outcomes \( \mathcal{O} = \{ \top, \bot \} \) with \( \bot \leq \top \)

Comparisons:
Possible outcome sets: \( \{ \bot \} \quad \{ \bot, \top \} \quad \{ \top \} \)
- May: \( \{ \bot \} <_{\text{Ho}} \{ \bot, \top \} =_{\text{Ho}} \{ \top \} \)
- Must: \( \{ \bot \} =_{\text{Sm}} \{ \bot, \top \} <_{\text{Sm}} \{ \top \} \)
Testing preorders

- $P \sqsubseteq_{\text{may}} Q$ if $\text{Apply}(T, P) \sqsubseteq_{\text{Ho}} \text{Apply}(T, Q)$ for every test $T$
- $P \sqsubseteq_{\text{must}} Q$ if $\text{Apply}(T, P) \sqsubseteq_{\text{Sm}} \text{Apply}(T, Q)$ for every test $T$

Probabilistic testing:

Use as outcomes $O$ the unit interval $[0, 1]$

Intuition: with $0 \leq p \leq q \leq 1$, passing a test with probability $q$ is better than passing with probability $p$

Comparisons:

- May: $O_1 \sqsubseteq_{\text{Ho}} O_2$ is every possibility $p \in O_1$ can be improved on by some $q \in O_2$
- Must: $O_1 \sqsubseteq_{\text{Sm}} O_2$ if every possibility $q \in O_2$ is an improvement on some $p \in O_1$
Testing preorders

- $P \sqsubseteq_{\text{may}} Q$ if $\text{Apply}(T, P) \sqsubseteq_{\text{Ho}} \text{Apply}(T, Q)$ for every test $T$
- $P \sqsubseteq_{\text{must}} Q$ if $\text{Apply}(T, P) \sqsubseteq_{\text{Sm}} \text{Apply}(T, Q)$ for every test $T$

Probabilistic testing:

Use as outcomes $O$ the unit interval $[0, 1]$

Intuition: with $0 \leq p \leq q \leq 1$, passing a test with probability $q$ better than passing with probability $p$

Comparisons:

- May: $O_1 \sqsubseteq_{\text{Ho}} O_2$ is every possibility $p \in O_1$ can be improved on by some $q \in O_2$
- Must: $O_1 \sqsubseteq_{\text{Sm}} O_2$ if every possibility $q \in O_2$ is an improvement on some $p \in O_1$
Applying tests to processes $\text{Apply}(T, P)$

Run the combined process $(T \mid P)$

Nondeterministic case:

- Nondeterministic $(T \mid P)$ resolved to a set of deterministic executions
- Each execution succeeds or fails
- Each execution contributes $\top$ or $\bot$ to $\text{Apply}(T, P)$

Probabilistic case:

- Probabilistic $(T \mid P)$ resolved to a set of deterministic but probabilistic executions
- Each execution contributes a probability of success to $\text{Apply}(T, P)$
- Formalisation: resolutions, policies, strategies, derivations, …
Applying tests to processes $Apply(T, P)$

Run the combined process $(T | P)$

**Nondeterministic case:**

- Nondeterministic $(T | P)$ resolved to a set of deterministic executions
- Each execution succeeds or fails
- Each execution contributes $\top$ or $\bot$ to $Apply(T, P)$

**Probabilistic case:**

- Probabilistic $(T | P)$ resolved to a set of deterministic but probabilistic executions
- Each execution contributes a probability of success to $Apply(T, P)$
- Formalisation: resolutions, policies, strategies, derivations, ...
Applying tests to processes \( \text{Apply}(T, P) \)

Run the combined process \( (T \mid P) \)

**Nondeterministic case:**

- Nondeterministic \( (T \mid P) \) resolved to a set of deterministic executions
- Each execution succeeds or fails
- Each execution contributes \( \top \) or \( \bot \) to \( \text{Apply}(T, P) \)

**Probabilistic case:**

- Probabilistic \( (T \mid P) \) resolved to a set of deterministic but probabilistic executions
- Each execution contributes a probability of success to \( \text{Apply}(T, P) \)
- Formalisation: resolutions, policies, strategies, derivations, …
Applying tests to processes \( \text{Apply}(T, P) \)

Run the combined process \( (T \mid P) \)

Nondeterministic case:

- Nondeterministic \( (T \mid P) \) resolved to a set of deterministic executions
- Each execution succeeds or fails
- Each execution contributes \( \top \) or \( \bot \) to \( \text{Apply}(T, P) \)

Probabilistic case:

- Probabilistic \( (T \mid P) \) resolved to a set of deterministic but probabilistic executions
- Each execution contributes a probability of success to \( \text{Apply}(T, P) \)
- Formalisation: resolutions, policies, strategies, derivations, ...
Example

\[ r_1 = a.(\tau.b + \tau.c) \quad r_2 = a.b + a.c \]

\[ T = \overline{a}.(\overline{\omega} \frac{1}{2} \oplus \overline{c}.\omega) \]

\[ \text{Apply}(T, r_1) = \begin{cases} \inf : & 0 \\ \sup : & 1 \end{cases} \]

\[ \text{Apply}(T, r_2) = \begin{cases} \inf : & \frac{1}{2} \\ \sup : & \frac{1}{2} \end{cases} \]

So choice points do matter:

\[ r_1 \not\equiv \text{pmay} \quad r_2 \]

\[ r_1 \not\equiv \text{pmust} \quad r_2 \]
Example

\[ r_1 = a.(\tau.b + \tau.c) \quad r_2 = a.b + a.c \]

\[ T = \overline{a}.(\overline{b}.\omega_{\frac{1}{2}} \oplus \overline{c}.\omega) \]

Apply\((T, r_1) = \begin{cases} 
\text{inf} : & 0 \\
\text{sup} : & 1 
\end{cases} \]

Apply\((T, r_2) = \begin{cases} 
\text{inf} : & \frac{1}{2} \\
\text{sup} : & \frac{1}{2} 
\end{cases} \)

So choice points do matter:

\[ r_1 \not\equiv_{\text{p}\text{may}} r_2 \]

\[ r_1 \not\equiv_{\text{p}\text{must}} r_2 \]
Example

\[ \text{Apply}(\bar{a}.\omega, P) = \begin{cases} \inf : & \frac{1}{2} \\ \sup : & 1 \end{cases} \]

\[ P \sim_{p\text{may}} \bar{a}.0 \]

\[ P \sqsubseteq_{p\text{must}} a.0 \]

\[ a.0 \not\sqsubseteq_{p\text{must}} P \]
Example

Apply(\overline{a}.\omega, P) = \begin{cases} 
\inf : & \frac{1}{2} \\
\sup : & 1 
\end{cases}

\[
P \simeq \text{pMay} a.0
\]

\[
P \sqsubseteq \text{pMust} a.0
\]

\[
a.0 \not\sqsubseteq \text{pMust} P
\]
Example

Apply(\overline{a}.\omega, P) = \begin{cases} 
\inf & : \frac{1}{2} \\
\sup & : 1 
\end{cases}

\[ P \simeq_{p\text{may}} a.0 \quad P \sqsubseteq_{p\text{must}} a.0 \quad a.0 \nsubseteq_{p\text{must}} P \]
Example

Apply($\bar{a}.\omega, P$) = \[
\begin{cases}
  \text{inf} : & \frac{1}{2} \\
  \text{sup} : & 1
\end{cases}
\]

$P \simeq_{\text{pmay}} a.0$

$P \sqsubseteq_{\text{pmust}} a.0$

$a.0 \not\sqsubseteq_{\text{pmust}} P$
Outline

Background  why bother?

Probabilistic Labelled Transition Systems

Testing Theory

Simulations

Results  some
Simulations in LTSs

Largest relation over $S \times S$ satisfying:

\[
\begin{array}{c}
    s \\ \downarrow \mu \\
    s'
\end{array}
\triangleleft_s
\begin{array}{c}
    t \\ \downarrow \mu \\
    t'
\end{array}
\]

implies

\[
\begin{array}{c}
    s \\ \downarrow \mu \\
    s'
\end{array}
\triangleleft_s
\begin{array}{c}
    t \\ \downarrow \mu \\
    t'
\end{array}
\]

Weak moves:

- $s \xrightarrow{a} s'$ means $s \xrightarrow{\tau} s_1 \xrightarrow{\tau} \ldots s_n \xrightarrow{a} s'_1 \xrightarrow{\tau} \ldots \xrightarrow{\tau} s'$
- $s \xrightarrow{\tau} s'$ means $s \xrightarrow{\tau} s_1 \xrightarrow{\tau} \ldots \xrightarrow{\tau} \ldots \xrightarrow{\tau} s'$

Logic characterisation: in finite branching LTS

\[
\phi ::= \tt \mid \phi \lor \phi' \mid \langle \mu \rangle \phi
\]
Simulations in LTSs

Largest relation over $S \times S$ satisfying:

$$s \triangleleft_s t \quad \text{implies} \quad s' \triangleleft_s t'$$

Weak moves:

- $s \xrightarrow{a} s'$ means $s \xrightarrow{\tau} s_1 \xrightarrow{\tau} \ldots s_n \xrightarrow{a} s'_1 \xrightarrow{\tau} \ldots \xrightarrow{\tau} s'$
- $s \xrightarrow{\tau} s'$ means $s \xrightarrow{\tau} s_1 \xrightarrow{\tau} \ldots \xrightarrow{\tau} \ldots \xrightarrow{\tau} s'$

Logic characterisation: in finite branching LTS

$$\phi ::= \texttt{tt} \mid \phi \lor \phi' \mid \langle \mu \rangle \phi$$
Simulations in a pLTS

Largest relation $\triangleleft_s \subseteq S \times S$ satisfying:

\[
\begin{array}{ccc}
  s & \triangleleft_s & t \\
  \mu & \text{implies} & \mu \\
  \Delta & & \Delta
\end{array}
\]

Lifting relations:

\[
\text{lift}(\triangleleft_s) \subseteq \mathcal{D}(S) \times \mathcal{D}(S)
\]
Simulations in a pLTS

Largest relation $\triangleleft_s \subseteq S \times S$ satisfying:

$$s \triangleleft_s t \quad \implies \quad \Delta \quad \implies \quad \text{lift}(\triangleleft_s) \quad \Theta$$

Lifting relations:

$\text{lift}(\triangleleft)$ lifts $R \subseteq S \times S$ to $\text{lift}(R) \subseteq \mathcal{D}(S) \times \mathcal{D}(S)$
Lifting relations

From $\mathcal{R} \subseteq S \times \mathcal{D}(S)$, to $\text{lift}(\mathcal{R}) \subseteq \mathcal{D}(S) \times \mathcal{D}(S)$

$\Delta \text{ lift}(\mathcal{R}) \Theta$ whenever

$\Delta = \sum_{i \in I} p_i \cdot s_i$, $I$ a finite index set

For each $i \in I$ there is a distribution $\Theta_i$ s.t. $s_i \mathcal{R} \Theta_i$

$\Theta = \sum_{i \in I} p_i \cdot \Theta_i$

$\sum_{i \in I} p_i = 1$

Many different formulations

Note: in decomposition $\sum_{i \in I} p_i \cdot s_i$ states $s_i$ are not necessarily unique
Lifting relations

From $\mathcal{R} \subseteq S \times \mathcal{D}(S)$, to $\text{lift}(\mathcal{R}) \subseteq \mathcal{D}(S) \times \mathcal{D}(S)$

\[ \Delta \text{ lift}(\mathcal{R}) \Theta \text{ whenever} \]

- $\Delta = \sum_{i \in I} p_i \cdot s_i$, $I$ a finite index set
- For each $i \in I$ there is a distribution $\Theta_i$ s.t. $s_i \mathcal{R} \Theta_i$
- $\Theta = \sum_{i \in I} p_i \cdot \Theta_i$
- $\sum_{i \in I} p_i = 1$

Many different formulations

Note: in decomposition $\sum_{i \in I} p_i \cdot s_i$ states $s_i$ are not necessarily unique
Lifting relations

From $\mathcal{R} \subseteq S \times \mathcal{D}(S)$, to $\text{lift}(\mathcal{R}) \subseteq \mathcal{D}(S) \times \mathcal{D}(S)$

\[ \Delta \text{ lift}(\mathcal{R}) \Theta \]

whenever

\[ \Delta = \sum_{i \in I} p_i \cdot s_i \ , \quad I \text{ a finite index set} \]

\[ \text{For each } i \in I \text{ there is a distribution } \Theta_i \text{ s.t. } s_i \mathcal{R} \Theta_i \]

\[ \Theta = \sum_{i \in I} p_i \cdot \Theta_i \]

\[ \sum_{i \in I} p_i = 1 \]

Many different formulations

Note: in decomposition $\sum_{i \in I} p_i \cdot s_i$ states $s_i$ are not necessarily unique
Lifting relations

From $\mathcal{R} \subseteq S \times \mathcal{D}(S)$, to $\text{lift}(\mathcal{R}) \subseteq \mathcal{D}(S) \times \mathcal{D}(S)$

$$\Delta \; \text{lift}(\mathcal{R}) \Theta$$

whenever

- $\Delta = \sum_{i \in I} p_i \cdot s_i$, $\forall i$ a finite index set
- For each $i \in I$ there is a distribution $\Theta_i$ s.t. $s_i \; \mathcal{R} \; \Theta_i$
- $\Theta = \sum_{i \in I} p_i \cdot \Theta_i$
- $\sum_{i \in I} p_i = 1$

Many different formulations

Note: in decomposition $\sum_{i \in I} p_i \cdot s_i$ states $s_i$ are not necessarily unique
Lifting actions: from \( s \xrightarrow{\mu} \Theta \) to \( \Delta \xrightarrow{\mu} \Theta \)

- \( \Delta \) represents a cloud of possible process states
- each possible state must be able to perform \( \mu \)
- all possible residuals combine to \( \Theta \)

**Examples:**

- \( (a \cdot b + a \cdot c) \frac{1}{2} \oplus a \cdot d \xrightarrow{a} b \frac{1}{2} \oplus d \)
- \( (a \cdot b + a \cdot c) \frac{1}{2} \oplus a \cdot d \xrightarrow{a} (b \frac{1}{2} \oplus c) \frac{1}{2} \oplus d \)
- \( (a \cdot b + a \cdot c) \frac{1}{2} \oplus a \cdot d \xrightarrow{a} (b \oplus c) \frac{1}{2} \oplus d \)
- \( (\tau \cdot a + \tau \cdot b) \frac{1}{2} \oplus (\tau \cdot a + \tau \cdot c) \xrightarrow{\tau} a \frac{1}{2} \oplus (b \frac{1}{2} \oplus c) \)
Lifting actions: from $s \xrightarrow{\mu} \Theta$ to $\Delta \xrightarrow{\mu} \Theta$

- $\Delta$ represents a cloud of possible process states
- each possible state must be able to perform $\mu$
- all possible residuals combine to $\Theta$

Examples:

- $(a.b + a.c) \frac{1}{2} \oplus a.d \xrightarrow{a} b \frac{1}{2} \oplus d$
- $(a.b + a.c) \frac{1}{2} \oplus a.d \xrightarrow{a} (b \frac{1}{2} \oplus c) \frac{1}{2} \oplus d$
- $(a.b + a.c) \frac{1}{2} \oplus a.d \xrightarrow{a} (b_p \oplus c) \frac{1}{2} \oplus d$
- $(\tau.a + \tau.b) \frac{1}{2} \oplus (\tau.a + \tau.c) \xrightarrow{\tau} a \frac{1}{2} \oplus (b \frac{1}{2} \oplus c)$
Lifting actions: from $s \xrightarrow{\mu} \Theta$ to $\Delta \xrightarrow{\mu} \Theta$

- $\Delta$ represents a cloud of possible process states
- each possible state must be able to perform $\mu$
- all possible residuals combine to $\Theta$

Examples:

- $(a.b + a.c) \frac{1}{2} \bigoplus a.d \xrightarrow{a} b \frac{1}{2} \bigoplus d$
- $(a.b + a.c) \frac{1}{2} \bigoplus a.d \xrightarrow{a} (b \frac{1}{2} \bigoplus c) \frac{1}{2} \bigoplus d$
- $(a.b + a.c) \frac{1}{2} \bigoplus a.d \xrightarrow{a} (b_p \bigoplus c) \frac{1}{2} \bigoplus d$
- $(\tau.a + \tau.b) \frac{1}{2} \bigoplus (\tau.a + \tau.c) \xrightarrow{\tau} a \frac{1}{2} \bigoplus (b \frac{1}{2} \bigoplus c)$
Lifting actions: from $s \xrightarrow{\mu} \Theta$ to $\Delta \xrightarrow{\mu} \Theta$

$\Delta$ represents a cloud of possible process states

- each possible state must be able to perform $\mu$
- all possible residuals combine to $\Theta$

Examples:

1. $(a.b + a.c) \frac{1}{2} \oplus a.d \xrightarrow{a} b \frac{1}{2} \oplus d$
2. $(a.b + a.c) \frac{1}{2} \oplus a.d \xrightarrow{a} (b \frac{1}{2} \oplus c) \frac{1}{2} \oplus d$
3. $(a.b + a.c) \frac{1}{2} \oplus a.d \xrightarrow{a} (b_p \oplus c) \frac{1}{2} \oplus d$
4. $(\tau.a + \tau.b) \frac{1}{2} \oplus (\tau.a + \tau.c) \xrightarrow{\tau} a \frac{1}{2} \oplus (b \frac{1}{2} \oplus c)$
Lifting actions: from $s \xrightarrow{\mu} \Theta$ to $\Delta \xrightarrow{\mu} \Theta$

- $\Delta$ represents a cloud of possible process states
- each possible state must be able to perform $\mu$
- all possible residuals combine to $\Theta$

Examples:

- $(a.b + a.c)_{\frac{1}{2}} \oplus a.d \xrightarrow{a} b_{\frac{1}{2}} \oplus d$
- $(a.b + a.c)_{\frac{1}{2}} \oplus a.d \xrightarrow{a} (b_{\frac{1}{2}} \oplus c)_{\frac{1}{2}} \oplus d$
- $(a.b + a.c)_{\frac{1}{2}} \oplus a.d \xrightarrow{a} (b_p \oplus c)_{\frac{1}{2}} \oplus d$
- $(\tau.a + \tau.b)_{\frac{1}{2}} \oplus (\tau.a + \tau.c) \xrightarrow{\tau} a_{\frac{1}{2}} \oplus (b_{\frac{1}{2}} \oplus c)$
Lifting actions: from \( s \xrightarrow{\mu} \Theta \) to \( \Delta \xrightarrow{\mu} \Theta \)

- \( \Delta \) represents a cloud of possible process states
- each possible state must be able to perform \( \mu \)
- all possible residuals combine to \( \Theta \)

Examples:

- \((a.b + a.c)\frac{1}{2} \oplus a.d \xrightarrow{a} b\frac{1}{2} \oplus d\)
- \((a.b + a.c)\frac{1}{2} \oplus a.d \xrightarrow{a} (b\frac{1}{2} \oplus c)\frac{1}{2} \oplus d\)
- \((a.b + a.c)\frac{1}{2} \oplus a.d \xrightarrow{a} (b_p \oplus c)\frac{1}{2} \oplus d\)
- \((\tau.a + \tau.b)\frac{1}{2} \oplus (\tau.a + \tau.c) \xrightarrow{\tau} a\frac{1}{2} \oplus (b\frac{1}{2} \oplus c)\)
Simulations in a pLTS

Largest relation $\triangleleft_s \subseteq S \times \mathcal{D}(S)$ satisfying:

$\Delta \xrightarrow{\mu} \trianglerighteq_s \Theta$

implies

$\Delta \xrightarrow{\mu} \text{lift}(\triangleleft_s) \Theta$

Logic characterisation: in finitary pLTS

$\phi ::= \text{tt} \mid \phi \lor \phi' \mid \phi \land \phi' \mid \langle \mu \rangle (\phi' \oplus \phi')$
Simulations in a pLTS

Largest relation $\trianglerighteq S \subseteq S \times D(S)$ satisfying:

$$
\begin{align*}
  s \trianglerighteq S \Theta & \quad \text{implies} \quad s \trianglerighteq S \Theta \\
  \mu & \quad \text{implies} \quad \mu \\
  \Delta & \quad \text{implies} \quad \mu
\end{align*}
$$

Logic characterisation: in finitary pLTS

$$
\phi ::= \operatorname{tt} \mid \phi \lor \phi' \mid \phi \land \phi' \mid \langle \mu \rangle (\phi_p \oplus \phi')
$$
Example simulation

\[ d \cdot (a \frac{1}{2} \oplus b) \trianglelefteq_S d \cdot ((a \frac{1}{2} \oplus b) \frac{1}{2} \oplus (a + b)) \text{ because} \]

\[ a \frac{1}{2} \oplus b \quad \text{lift}(\trianglelefteq_S) \quad (a \frac{1}{2} \oplus b) \frac{1}{2} \oplus (a + b) \]

\[ \frac{1}{2} \cdot a + \frac{1}{2} \cdot b \quad \text{lift}(\trianglelefteq_S) \quad \frac{1}{4} \cdot a + \frac{1}{2} \cdot (a + b) + \frac{1}{4} \cdot b \]

Because:
\[ a \trianglelefteq_S \frac{1}{2} \cdot a + \frac{1}{2} \cdot (a + b) \]
\[ b \trianglelefteq_S \frac{1}{2} \cdot b + \frac{1}{2} \cdot (a + b) \]

Moral:
\[ \trianglelefteq_S \text{ must have type } S \times D(S) \]
\[ \text{NOT type } S \times S \]
Example simulation

\[ d \cdot (a \frac{1}{2} \oplus b) \triangleleft_s d \cdot ((a \frac{1}{2} \oplus b) \frac{1}{2} \oplus (a + b)) \] because

\[ a \frac{1}{2} \oplus b \quad \text{lift}(\triangleleft_s) \quad (a \frac{1}{2} \oplus b) \frac{1}{2} \oplus (a + b) \]

Because:

\[ a \triangleleft_s \frac{1}{2} \cdot a + \frac{1}{2} \cdot (a + b) \]
\[ b \triangleleft_s \frac{1}{2} \cdot b + \frac{1}{2} \cdot (a + b) \]

Moral:

\[ \triangleleft_s \text{ must have type } S \times \mathcal{D}(S) \]
\[ \text{ NOT type } S \times S \]
Example simulation

\[ d.(a \frac{1}{2} \oplus b) \triangleleft_S d.((a \frac{1}{2} \oplus b) \frac{1}{2} \oplus (a + b)) \text{ because} \]

\[ a \frac{1}{2} \oplus b \quad \text{lift}(\triangleleft_S) \quad (a \frac{1}{2} \oplus b) \frac{1}{2} \oplus (a + b) \]

\[ \frac{1}{2} \cdot a + \frac{1}{2} \cdot b \quad \text{lift}(\triangleleft_S) \quad \frac{1}{4} \cdot a + \frac{1}{2} \cdot (a + b) + \frac{1}{4} \cdot b \]

Because:

- \[ a \triangleleft_S \frac{1}{2} \cdot a + \frac{1}{2} \cdot (a + b) \]
- \[ b \triangleleft_S \frac{1}{2} \cdot b + \frac{1}{2} \cdot (a + b) \]

Moral:

- \[ \triangleleft_S \text{ must have type } S \times D(S) \]
- \[ \text{NOT type } S \times S \]
Example simulation

\[ d \cdot (a \cheduling b) \preceq_s d \cdot ((a \ scheduling b) \ scheduling (a + b)) \] because

\[ a \ scheduling b \ scheduling (a + b) \]

\[ \frac{1}{2} \cdot a + \frac{1}{2} \cdot b \ scheduling (a + b) \]

Because:

\[ a \preceq_s \frac{1}{2} \cdot a + \frac{1}{2} \cdot (a + b) \]

\[ b \preceq_s \frac{1}{2} \cdot b + \frac{1}{2} \cdot (a + b) \]

Moral:

\[ \preceq_s \text{ must have type } S \times D(S) \]

\[ \text{NOT type } S \times S \]
Example simulation

\[
d.(a \ 2 \oplus b) \triangleleft_S d.\left((a\ 2 \oplus b)\ 2 \oplus (a + b)\right) \text{ because}
\]

\[
a \ 2 \oplus b \quad \text{lift}(\triangleleft_S) \quad (a \ 2 \oplus b) \ 2 \oplus (a + b)
\]

\[
\frac{1}{2} \cdot a \ + \ \frac{1}{2} \cdot b \quad \text{lift}(\triangleleft_S) \quad \frac{1}{4} \cdot a \ + \ \frac{1}{2} \cdot (a + b) \ + \ \frac{1}{4} \cdot b
\]

Because:

- \( a \triangleleft_S \frac{1}{2} \cdot a + \frac{1}{2} \cdot (a + b) \)
- \( b \triangleleft_S \frac{1}{2} \cdot b + \frac{1}{2} \cdot (a + b) \)

Moral:

- \( \triangleleft_S \text{ must have type } S \times D(S) \)
- NOT type \( S \times S \)
Example simulation

\[ d \cdot (a \odot b) \triangleleft_s d \cdot ((a \odot b) \odot (a + b)) \text{ because} \]

\[ a \odot b \quad \text{lift}(\triangleleft_s) \quad (a \odot b) \odot (a + b) \]

\[ \frac{1}{2} \cdot a + \frac{1}{2} \cdot b \quad \text{lift}(\triangleleft_s) \quad \frac{1}{4} \cdot a + \frac{1}{2} \cdot (a + b) + \frac{1}{4} \cdot b \]

Because:

\[ a \triangleleft_s \frac{1}{2} \cdot a + \frac{1}{2} \cdot (a + b) \]
\[ b \triangleleft_s \frac{1}{2} \cdot b + \frac{1}{2} \cdot (a + b) \]

Moral:

\[ \triangleleft_s \text{ must have type } S \times D(S) \]
\[ \text{NOT type } S \times S \]
Second problem

\[ a.B \]

\[ a \]

\[ B \]

\[ \tau \]

\[ a.b \sqsubseteq_{pmay} a.B \]

\[ a.b \not\cong_s a.B \]

because \( a.B \not\xrightarrow{a} b \) because \( a.B \not\xrightarrow{\tau}^* \frac{3}{4} \xrightarrow{\tau}^* b \)

Moral:

weak internal actions must include limiting behaviour

\( B \) reaches state \( s_2 \) with probability 1
Second problem

\[ a.B \]

\[ s_1 \xrightarrow{\tau} s_2 \]

\[ s_2 \xrightarrow{a} s_1 \]

\[
\begin{align*}
\text{Moral:} & \\
& \text{weak internal actions must include limiting behaviour} \\
& B \text{ reaches state } s_2 \text{ with probability } 1
\end{align*}
\]

\[ a.b \subseteq_{\text{p may}} a.B \]
\[ a.b \not\subset_{\text{s}} a.B \]

because \[ a.B \xrightarrow{a} b \] because
\[ a.B \xrightarrow{\tau} \ast \xrightarrow{a} \xrightarrow{\tau} \ast b \]
Second problem

\[
\begin{align*}
& a.B \\
& \quad \downarrow a \\
& B \\
& \quad \downarrow \tau \\
& s_1 \xrightarrow{\frac{3}{4}} s_2 \\
& \quad \downarrow b \\
& B \xrightarrow{\frac{1}{4}} B \\
& \quad \uparrow \tau \\
& a.b \sqsubseteq_{\text{p may}} a.B \\
& a.b \not\sqsupseteq_S a.B \\
& \text{because } a.B \not
\xrightarrow{a} b \text{ because } \\
& a.B \xrightarrow{\tau} * \not\xrightarrow{a} \tau \xrightarrow{\tau} * b
\end{align*}
\]

Moral:

weak internal actions must include limiting behaviour

\( B \) reaches state \( s_2 \) with probability 1
Second problem

\[ a.B \]

\[ s_1 \]

\[ B \]

\[ \tau \]

\[ s_2 \]

\[ a \]

\[ a.b \sqsubseteq_{\text{pmay}} a.B \]

\[ a.b \not\sqsubseteq_S a.B \]

because \[ a.B \not\xrightarrow{a} b \] because

\[ a.B \xrightarrow{\tau} \not\xrightarrow{\tau} \xrightarrow{a} \xrightarrow{\tau} \not\xrightarrow{\tau} \]

Moral:

weak internal actions must include limiting behaviour

\( B \) reaches state \( s_2 \) with probability 1
Weak internal actions in a pLTS

Idea: internal computation is a partial execution

\[ \Delta \overset{\tau}{\rightarrow} \Theta \]

\[ \Delta = \Delta_{\text{go}} + \Delta_{\text{stay}} \]

\[ \Delta_{\text{go}} = \Delta_{0} + \Delta_{1} + \ldots \]

\[ \Delta_{\text{go}} = \Delta_{(k+1)} + \Delta_{(k+1)} + \ldots \]

Total: \[ \Theta = \sum_{k=0}^{\infty} \Delta_{k} \]

\[ \Delta_{\text{stay}}: \text{any subdistribution} \]

\[ \Delta_{\text{go}}: \text{any subdistribution which can perform } \tau \]

Note: use of subdistributions
Weak internal actions in a pLTS

\[ \Delta \implies \Theta \]

Idea: internal computation is a partial execution

\[
\begin{align*}
\Delta & = \Delta_0 + \Delta^\text{go}_0 \\
\Delta^\text{go}_0 & \xrightarrow{\tau} \Delta_0 + \Delta^\text{go}_0 \\
\ldots & \xrightarrow{\tau} \ldots \\
\Delta^\text{go}_k & \xrightarrow{\tau} \Delta^\text{go}_{(k+1)} + \Delta^\text{stay}_{(k+1)} \\
\ldots & \xrightarrow{\tau} \ldots \\
\ldots & \\
\Delta^\text{stay}_k & \\
\end{align*}
\]

Total:
\[ \Theta = \sum_{k=0}^{\infty} \Delta^\text{stay}_k \]

\( \Delta^\text{go} \): any subdistribution which can perform \( \tau \)

\( \Delta^\text{stay} \): any subdistribution

Note: use of subdistributions
Weak internal actions in a pLTS

\[ \Delta \implies \Theta \]

Idea: internal computation is a partial execution

\[
\begin{align*}
\Delta &= \Delta_0 + \Delta_{\text{go}} + \Delta_{\text{stay}} \\
\Delta_{\text{go}}^0 &\xrightarrow{\tau} \Delta_{\text{go}}^0 + \Delta_0 + \Delta_{\text{stay}}^1 \\
\Delta_{\text{go}}^k &\xrightarrow{\tau} \Delta_{\text{go}}^{(k+1)} + \Delta_{\text{stay}}^{(k+1)} \\
\text{Total: } \Theta &= \sum_{k=0}^{\infty} \Delta_{\text{stay}}^k
\end{align*}
\]

\(\Delta_{\text{stay}}^k\): any subdistribution

\(\Delta_{\text{go}}^k\): any subdistribution which can perform \(\tau\)

Note: use of subdistributions
Weak internal actions in a pLTS

Idea: internal computation is a partial execution

\[
\Delta \mapsto \Theta
\]

\[
\begin{align*}
\Delta & = \Delta_0^\text{go} + \Delta_0^\text{stay} + \\
\Delta_0^\text{go} & \xrightarrow{\tau} \Delta_0^\text{go} + \Delta_1^\text{stay} \\
\ldots & \quad \ldots \\
\Delta_k^\text{go} & \xrightarrow{\tau} \Delta_{k+1}^\text{go} + \Delta_{k+1}^\text{stay} \\
\ldots & \quad \ldots \\
\ldots & \quad \ldots \\
\text{Total:} & \quad \Theta = \sum_{k=0}^{\infty} \Delta_k^\text{stay}
\end{align*}
\]

\(\Delta^\text{stay}\): any subdistribution

\(\Delta^\text{go}\): any subdistribution which can perform \(\tau\)

Note: use of subdistributions
Example

\[ B \xrightarrow{a} B \]

\[ B \xrightarrow{\tau} \frac{3}{4} \cdot s_2 \xrightarrow{\tau} \frac{3}{4} \cdot B \xrightarrow{\tau} (\frac{3}{4})^2 \cdot s_1 \]

\[ \vdots \]

\[ (\frac{3}{4})^k \cdot B \xrightarrow{\tau} (\frac{3}{4})^{k+1} \cdot B \]

\[ \frac{1}{4} \]

\[ s_1 \]

\[ s_2 \]

\[ a := \text{go} \]

\[ B + \frac{3}{4} \cdot s_1 + \frac{3}{4} \cdot B + (\frac{3}{4})^2 \cdot s_1 + \]

\[ \vdots \]

\[ (\frac{3}{4})^k \cdot s_1 + (\frac{3}{4})^{k+1} \cdot B + (\frac{3}{4})^k \cdot s_2 \]

\[ \text{Total:} \quad s_2 = \sum_{k=0}^{\infty} (\frac{3}{4})^k \frac{1}{4} \cdot s_2 \]
Example

\[ B \quad \Rightarrow \quad s_2 \]

\[ \begin{align*}
    B & \xrightarrow{\tau} B + \frac{3}{4} \cdot s_2 + \frac{3}{4} \cdot B + \left(\frac{3}{4}\right)^2 \cdot s_1 + \ldots + \left(\frac{3}{4}\right)^k \cdot s_2 + \ldots \\
    & \quad + \text{empDist} \quad + \frac{1}{4} \cdot s_2 \\
    s_1 & \xrightarrow{1/4} \frac{3}{4} \cdot B + \ldots + \left(\frac{3}{4}\right)^k \cdot B + \ldots \\
    s_2 & \xrightarrow{1/4} \frac{3}{4} \cdot s_2 + \ldots + \left(\frac{3}{4}\right)^k \frac{1}{4} \cdot s_2 + \ldots \\
\end{align*} \]

Total:

\[ s_2 = \sum_{k=0}^{\infty} \left(\frac{3}{4}\right)^k \frac{1}{4} \cdot s_2 \]
Simulations in a pLTS finally

Largest relation $\triangleleft_s \subseteq S \times \mathcal{D}(S)$ satisfying:

\[
\begin{align*}
  s \quad \triangleleft_s \quad \Theta \\
  \mu \\
  \Delta 
\end{align*}
\]

implies

\[
\begin{align*}
  s \quad \triangleleft_s \\
  \mu \\
  \Delta 
\end{align*}
\]

lift($\triangleleft_s$)

\[
\begin{align*}
  s \quad \triangleleft_s \quad \Theta \\
  \mu \\
  \Delta 
\end{align*}
\]

\[
\begin{align*}
  \Theta \xrightarrow{a} \Theta' \quad \text{now means } \Theta \Longrightarrow \Theta_1 \xrightarrow{a} \Theta_2 \Longrightarrow \Theta \\
  \Theta \xrightarrow{\tau} \Theta' \quad \text{now means } \Theta \Longrightarrow \Theta'
\end{align*}
\]
Simulations in a pLTS

Largest relation $\triangleleft_s \subseteq S \times D(S)$ satisfying:

$$s \triangleleft_s \Theta$$

implies

$$s \triangleleft_s \Theta$$

$$\mu$$

$$\Delta$$

$$\Theta$$

$$\mu$$

$$\Delta$$

$\text{lift}(\triangleleft_s)$

$\Theta'$

$\Theta$ $\Rightarrow$ $\Theta'$: now means $\Theta \Rightarrow \Theta_1 \overset{a}{\Rightarrow} \Theta_2 \Rightarrow \Theta$

$\Theta$ $\Rightarrow$ $\Theta'$: now means $\Theta \Rightarrow \Theta'$
Example simulation

\[ a.B \leadsto S a.B \]

because \[ a.B \rightarrow^a b \]

Also:

\[ a.B \leq_S a.b \]
Example simulation

Also:

\[ a.B \triangleleft_S a.b \]

because \[ a.B \xrightarrow{a} b \]
Outline

Background  why bother ?

Probabilistic Labelled Transition Systems

Testing Theory

Simulations

Results  some
Simulations and testing

Soundness:

\[ s \prec_s \Theta \implies s \sqsubseteq_{p\text{may}} \Theta \]

proof is straightforward

Completeness:

In a finitary pLTS \[ s \sqsubseteq_{p\text{may}} \Theta \implies s \prec_s \Theta \]

difficult proof

Algebra:

Complete set of algebraic axioms for finite pCSP

Must testing:

Similar results using Failure simulation

add divergence and deadlocks/failures
Simulations and testing

Soundness:
\[ s \triangleleft_S \Theta \implies s \sqsubseteq_{\text{pmay}} \Theta \]
Proof is straightforward

Completeness:
In a finitary pLTS
\[ s \sqsubseteq_{\text{pmay}} \Theta \implies s \triangleleft_S \Theta \]
Difficult proof

Algebra:
Complete set of algebraic axioms for finite pCSP

Must testing:
Similar results using Failure simulation
Add divergence and deadlocks/failures
Simulations and testing

Soundness:

\[ s \not\prec_s \Theta \text{ implies } s \preceq_{\text{pmay}} \Theta \]

proof is straightforward

Completeness:

In a finitary pLTS \( s \preceq_{\text{pmay}} \Theta \) implies \( s \not\prec_s \Theta \)

difficult proof

Algebra:

Complete set of algebraic axioms for finite pCSP

Must testing:

Similar results using *Failure simulation*

add divergence and deadlocks/failures
Simulations and testing

Soundness:

\[ s \prec_s \Theta \implies s \preceq_{p\text{may}} \Theta \]

proof is straightforward

Completeness:

In a finitary pLTS \[ s \preceq_{p\text{may}} \Theta \implies s \prec_s \Theta \]
difficult proof

Algebra:

Complete set of algebraic axioms for finite pCSP

Must testing:

Similar results using Failure simulation

add divergence and deadlocks/failures
Simulations and testing

Soundness:
\[ s \triangleleft \Theta \implies s \sqsubseteq_{\text{pmay}} \Theta \]
proof is straightforward

Completeness:
In a finitary pLTS
\[ s \sqsubseteq_{\text{pmay}} \Theta \implies s \triangleleft \Theta \]
difficult proof

Algebra:
Complete set of algebraic axioms for finite pCSP

Must testing:
Similar results using *Failure simulation*
add divergence and deadlocks/failures
Are these distinguishable? Tests v. formulae

Formula: $\langle d \rangle (\langle a \rangle \text{tt} \frac{1}{2} \oplus (\langle b \rangle \text{tt} \land \langle c \rangle \text{tt}))$

Test: $d.(\tau.a.(\omega \frac{1}{2} \oplus \emptyset) \oplus \tau.(b.\omega \frac{1}{2} \oplus c.\omega))$

Proof of completeness generates test from formulae
Are these distinguishable? Tests v. formulae

Formula: $\langle d \rangle (\langle a \rangle \text{tt} \frac{1}{2} \oplus (\langle b \rangle \text{tt} \land \langle c \rangle \text{tt}))$

Test: $d.(\tau.a.(\omega \frac{1}{2} \oplus 0) + \tau.(b.\omega \frac{1}{2} \oplus c.\omega))$

Proof of completeness generates test from formulae
Are these distinguishable? Tests v. formulae

Formula: $\langle d \rangle (\langle a \rangle \text{tt} \frac{1}{2} \oplus (\langle b \rangle \text{tt} \land \langle c \rangle \text{tt}))$ easy

Test: $d.(\tau.a.(\omega \frac{1}{2} \oplus 0) + \tau.(b.\omega \frac{1}{2} \oplus c.\omega))$ hard

Proof of completeness generates test from formulae
Are these distinguishable? Tests v. formulae

Formula: \[ \langle d \rangle (\langle a \rangle \text{tt} \frac{1}{2} \oplus (\langle b \rangle \text{tt} \land \langle c \rangle \text{tt}) \]

Test: \[ d.(\tau.a.(\omega \frac{1}{2} \oplus 0) + \tau.(b.\omega \frac{1}{2} \oplus c.\omega)) \]

Proof of completeness generates test from formulae