Markovian Testing Equivalence and Exponentially Timed Internal Actions

Marco Bernardo

University of Urbino – Italy

Markovian Behavioral Equivalences

- Tools for relating and manipulating formal models with an underlying continuous-time Markov chain (CTMC) semantics.
- Markovian bisimilarity: two processes are equivalent whenever they are able to mimic each other's functional and performance behavior step by step.
- Markovian testing equivalence: two processes are equivalent whenever an external observer is not able to distinguish between them from a functional or performance viewpoint by interacting with them by means of tests and comparing their reactions.
- Markovian trace equivalence: two processes are equivalent whenever they are able to perform computations with the same functional and performance characteristics.

Handling Internal Actions

- When comparing nondeterministic processes, internal actions can be abstracted away.

- When comparing Markovian processes, exponentially timed internal actions *cannot* be abstracted away.

- Markovian bisimilarity smoothly handles them by applying to them the same exit rate equality check that is applied to exponentially timed visible actions.

- *This is not the case with Markovian testing and trace equivalences.*

- Exponentially timed internal actions must be carefully taken into account in order not to equate processes that are distinguishable from a timing viewpoint.

- Example: consider the two Markovian processes:

$$<\tau, \lambda>.\underline{0}$$
$$<\tau, \mu>.\underline{0}$$

where $\lambda, \mu \in \mathbb{R}_{>0}$ such that $\lambda > \mu$.

- They should not be considered equivalent, as the durations of their actions are sampled from *different exponential probability distributions*.

- If they were considered equivalent, then *congruence* with respect to alternative and parallel composition *would not hold*.

- With the current definition of Markovian testing equivalence (comparing the probabilities of passing the same test within the same average time upper bound), there is no way to distinguish between them for any time upper bound (both pass with prob. 1 the test given by the success state and with prob. 0 any other test).

- How to handle exponentially timed internal actions when checking for Markovian testing/trace equivalences?

- *Idea: place an additional constraint on the length of the successful computations to take into account.*

- Example: take a test comprising only the success state and consider successful computations of length 1 and average duration less than $1/\lambda$:
  - Both $<\tau, \lambda>.\underline{0}$ and $<\tau, \mu>.\underline{0}$ are forced to execute their only action and reach success. Is it reached in time?
  - $<\tau, \lambda>.\underline{0}$ reaches success with probability 1, as it has enough time on average to perform its only action.
  - $<\tau, \mu>.\underline{0}$ does not, as it has not enough time on average to perform its only action by the deadline.

Markovian Process Calculus

- *Interested in investigating congruence and axiomatization.*
- Durational actions and asymmetric synchronizations.
- $Name_{\rm v}$: set of visible action names.
- $Name = Name_{\rm v} \cup \{\tau\}$: set of all action names.
- $Rate = \mathbb{R}_{>0} \cup \{*_w \mid w \in \mathbb{R}_{>0}\}$: set of action rates.
- $Act_{\rm M} = Name \times Rate$: set of exponentially timed and passive actions.
- $Relab = \{\varphi : Name \rightarrow Name \mid \varphi^{-1}(\tau) = \{\tau\}\}$: set of visibility-preserv. relabeling functions.
- Var: set of process variables.

- Process term syntax for process language $\mathcal{PL}_{\mathrm{M}}$:

P	$::=$	$\underline{0}$	inactive process	
	$\mid$	$<a, \lambda>.P$	exp. timed action prefix	$(a \in \mathit{Name}, \lambda \in \mathbb{R}_{>0})$
	$\mid$	$<a, *_w>.P$	passive action prefix	$(a \in \mathit{Name}, w \in \mathbb{R}_{>0})$
	$\mid$	$P + P$	alternative composition	
	$\mid$	$P \parallel_S P$	parallel composition	$(S \subseteq \mathit{Name}_{\mathrm{v}})$
	$\mid$	$P \,/\, H$	hiding	$(H \subseteq \mathit{Name}_{\mathrm{v}})$
	$\mid$	$P[\varphi]$	relabeling	$(\varphi \in \mathit{Relab})$
	$\mid$	X	process variable	$(X \in \mathit{Var})$
	$\mid$	$\mathit{rec}\, X : P$	recursion	$(X \in \mathit{Var})$

- $\mathbb{P}_{\mathrm{M}}$: set of closed and guarded process terms.

- State transition graph expressing all computations and branching points and accounting for transition multiplicity ($<a, \lambda>.\underline{0} + <a, \lambda>.\underline{0}$ vs. $<a, \lambda>.\underline{0}$).

- Every $P \in \mathbb{P}_{\mathrm{M}}$ is mapped to a labeled multitransition system $[\![P]\!]_{\mathrm{M}}$:
  - Each state corresponds to a process term into which P can evolve.
  - The initial state corresponds to P.
  - Each transition from a source state to a target state is labeled with the action that determines the corresponding state change.

- Every $P \in \mathbb{P}_{\mathrm{M,pc}}$ is mapped to a CTMC (performance closure if no passive trans.):
  - Dropping action names from all transitions of $[\![P]\!]_{\mathrm{M}}$.
  - Collapsing all the transitions between any two states of $[\![P]\!]_{\mathrm{M}}$ into a single transition by summing up the rates of the original transitions.

$$(\textsc{Pre}_1)\quad \frac{}{<a,\lambda>.P \xrightarrow{a,\lambda}_{\mathrm{M}} P} \qquad (\textsc{Pre}_2)\quad \frac{}{<a,*_w>.P \xrightarrow{a,*w}_{\mathrm{M}} P}$$

$$(\textsc{Alt}_1)\quad \frac{P_1 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'}{P_1 + P_2 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'} \qquad (\textsc{Alt}_2)\quad \frac{P_2 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'}{P_1 + P_2 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'}$$

$$(\textsc{Par}_1)\quad \frac{P_1 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P_1' \qquad a \notin S}{P_1 \parallel_S P_2 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P_1' \parallel_S P_2} \qquad (\textsc{Par}_2)\quad \frac{P_2 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P_2' \qquad a \notin S}{P_1 \parallel_S P_2 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P_1 \parallel_S P_2'}$$

$$(\textsc{Syn}_1)\quad \frac{P_1 \xrightarrow{a,\lambda}_{\mathrm{M}} P_1' \qquad P_2 \xrightarrow{a,*w}_{\mathrm{M}} P_2' \qquad a \in S}{P_1 \parallel_S P_2 \xrightarrow{a,\lambda\cdot\frac{w}{weight(P_2,a)}}_{\mathrm{M}} P_1' \parallel_S P_2'} \qquad (\textsc{Syn}_2)\quad \frac{P_1 \xrightarrow{a,*w}_{\mathrm{M}} P_1' \qquad P_2 \xrightarrow{a,\lambda}_{\mathrm{M}} P_2' \qquad a \in S}{P_1 \parallel_S P_2 \xrightarrow{a,\lambda\cdot\frac{w}{weight(P_1,a)}}_{\mathrm{M}} P_1' \parallel_S P_2'}$$

$$(\textsc{Syn}_3)\quad \frac{P_1 \xrightarrow{a,*w_1}_{\mathrm{M}} P_1' \qquad P_2 \xrightarrow{a,*w_2}_{\mathrm{M}} P_2' \qquad a \in S}{P_1 \parallel_S P_2 \xrightarrow{a,*norm(w_1,w_2,a,P_1,P_2)}_{\mathrm{M}} P_1' \parallel_S P_2'}$$

$$(\textsc{Hid}_1)\quad \frac{P \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P' \qquad a \in H}{P/H \xrightarrow{\tau,\tilde{\lambda}}_{\mathrm{M}} P'/H} \qquad (\textsc{Hid}_2)\quad \frac{P \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P' \qquad a \notin H}{P/H \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'/H}$$

$$(\textsc{Rel})\quad \frac{P \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'}{P[\varphi] \xrightarrow{\varphi(a),\tilde{\lambda}}_{\mathrm{M}} P'[\varphi]} \qquad (\textsc{Rec})\quad \frac{P\{rec\,X : P \hookrightarrow X\} \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'}{rec\,X : P \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'}$$

- Auxiliary functions:

$$
\begin{array}{rcl}
weight(P, a) & = & \sum \{\!|\, w \in \mathbb{R}_{>0} \mid \exists P' \in \mathbb{P}_{\mathrm{M}}.\, P \xrightarrow{a,*_w}_{\mathrm{M}} P' \,|\!\} \\[2ex]
norm(w_1, w_2, a, P_1, P_2) & = & \frac{w_1}{weight(P_1, a)} \cdot \frac{w_2}{weight(P_2, a)} \cdot (weight(P_1, a) + weight(P_2, a)) \\[2ex]
rate_{\mathrm{e}}(P, a, l, D) & = & \begin{cases} \sum \{\!|\, \lambda \in \mathbb{R}_{>0} \mid \exists P' \in D.\, P \xrightarrow{a,\lambda}_{\mathrm{M}} P' \,|\!\} & \text{if } l = 0 \\ \sum \{\!|\, w \in \mathbb{R}_{>0} \mid \exists P' \in D.\, P \xrightarrow{a,*_w}_{\mathrm{M}} P' \,|\!\} & \text{if } l = -1 \end{cases} \\[2ex]
rate_{\mathrm{o}}(P, a, l) & = & rate_{\mathrm{e}}(P, a, l, \mathbb{P}_{\mathrm{M}}) \\[2ex]
rate_{\mathrm{t}}(P, l) & = & \sum\limits_{a \in Name} rate_{\mathrm{o}}(P, a, l)
\end{array}
$$

Probability and Duration of Computations

- A computation of a process term $P \in \mathbb{P}_{\mathrm{M}}$ is a sequence of transitions that can be executed starting from P.
- The *length* of a computation is given by the number of its transitions.
- $\mathcal{C}_{\mathrm{f}}(P)$: multiset of finite-length computations of P.
- Two distinct computations are *independent* of each other iff neither is a proper prefix of the other one.
- Focus on finite multisets of independent, finite-length computations.
- Attributes of a finite-length computation: trace, probability, duration.

- Given a set of sequences, we use:
  - Operator $\_ \circ \_$ for sequence concatenation.
  - Operator $|\_|$ for sequence length.
- The concrete trace associated with the execution of $c \in \mathcal{C}_{\mathrm{f}}(P)$ is the sequence of action names labeling the transitions of c:

$$\boxed{\mathit{trace}_{\mathrm{c}}(c) \; = \; \begin{cases} \varepsilon & \text{if } |c| = 0 \\ a \circ \mathit{trace}_{\mathrm{c}}(c') & \text{if } c \equiv P \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} c' \end{cases}}$$

- We denote by $\mathit{trace}(c)$ the visible part of $\mathit{trace}_{\mathrm{c}}(c)$, i.e., the subsequence of $\mathit{trace}_{\mathrm{c}}(c)$ obtained by removing all the occurrences of τ.

- For the quantitative attributes, we assume $P \in \mathbb{P}_{\mathrm{M,pc}}$.
- The probability of executing $c \in \mathcal{C}_{\mathrm{f}}(P)$ is the product of the execution probabilities of the transitions of c:

$$\boxed{prob(c) \; = \; \begin{cases} 1 & \text{if } |c| = 0 \\ \frac{\lambda}{rate_{\mathrm{t}}(P,0)} \cdot prob(c') & \text{if } c \equiv P \xrightarrow{a,\lambda}_{\mathrm{M}} c' \end{cases}}$$

- Probability of executing a computation in $C \subseteq \mathcal{C}_{\mathrm{f}}(P)$:

$$\boxed{prob(C) \; = \; \sum_{c \in C} prob(c)}$$

assuming that C is finite and all of its computations are independent.

- The stepwise average duration of $c \in \mathcal{C}_{\mathrm{f}}(P)$ is the sequence of average sojourn times in the states traversed by c:

$$\boxed{time_{\mathrm{a}}(c) \;=\; \begin{cases} \varepsilon & \text{if } |c| = 0 \\ \frac{1}{rate_{\mathrm{t}}(P,0)} \circ time_{\mathrm{a}}(c') & \text{if } c \equiv P \xrightarrow{a,\lambda}_{\mathrm{M}} c' \end{cases}}$$

- Multiset of computations in $C \subseteq \mathcal{C}_{\mathrm{f}}(P)$ whose stepwise average duration is not greater than $\theta \in (\mathbb{R}_{>0})^*$:

$$\boxed{C_{\leq \theta} \;=\; \{\!| \, c \in C \mid |c| \leq |\theta| \wedge \forall i = 1, \dots, |c|. \, time_{\mathrm{a}}(c)[i] \leq \theta[i] \, |\!\}}$$

- C^l: multiset of computations in $C \subseteq \mathcal{C}_{\mathrm{f}}(P)$ having length $l \in \mathbb{N}$.

- The stepwise duration of $c \in \mathcal{C}_{\mathrm{f}}(P)$ is the sequence of random variables quantifying the sojourn times in the states traversed by c:

$$\boxed{\mathit{time}_{\mathrm{d}}(c) \;=\; \begin{cases} \varepsilon & \text{if } |c| = 0 \\ \mathit{Exp}_{\mathit{rate}_{\mathrm{t}}(P,0)} \circ \mathit{time}_{\mathrm{d}}(c') & \text{if } c \equiv P \xrightarrow{a,\lambda}_{\mathrm{M}} c' \end{cases}}$$

- Probability distribution of executing a computation in $C \subseteq \mathcal{C}_{\mathrm{f}}(P)$ within a sequence $\theta \in (\mathbb{R}_{>0})^*$ of time units:

$$\boxed{\mathit{prob}_{\mathrm{d}}(C, \theta) \;=\; \sum_{c \in C}^{|c| \leq |\theta|} \mathit{prob}(c) \cdot \prod_{i=1}^{|c|} \Pr\{\mathit{time}_{\mathrm{d}}(c)[i] \leq \theta[i]\}}$$

  assuming that C is finite and all of its computations are independent.

- Factor $\Pr\{\mathit{time}_{\mathrm{d}}(c)[i] \leq \theta[i]\} = 1 - \mathrm{e}^{-\theta[i] / \mathit{time}_{\mathrm{a}}(c)[i]}$ stems from the cumulative distribution function of the exponentially distributed random variable $\mathit{time}_{\mathrm{d}}(c)[i]$ (whose expected value is $\mathit{time}_{\mathrm{a}}(c)[i]$).

- Why not summing up sojourn times? (standard duration instead of stepwise one)
- Consider process terms ($\lambda \neq \mu$, $b \neq d$, identical nonmaximal computations):

$$<g,\gamma>.<a,\lambda>.<b,\mu>.\underline{0} + <g,\gamma>.<a,\mu>.<d,\lambda>.\underline{0}$$
$$<g,\gamma>.<a,\lambda>.<d,\mu>.\underline{0} + <g,\gamma>.<a,\mu>.<b,\lambda>.\underline{0}$$

- Maximal computations of the first term:

$$c_{1,1} \quad \equiv \quad \cdot \xrightarrow{g,\gamma}_{\mathrm{M}} \cdot \xrightarrow{a,\lambda}_{\mathrm{M}} \cdot \xrightarrow{b,\mu}_{\mathrm{M}} \cdot$$
$$c_{1,2} \quad \equiv \quad \cdot \xrightarrow{g,\gamma}_{\mathrm{M}} \cdot \xrightarrow{a,\mu}_{\mathrm{M}} \cdot \xrightarrow{d,\lambda}_{\mathrm{M}} \cdot$$

- Maximal computations of the second term:

$$c_{2,1} \quad \equiv \quad \cdot \xrightarrow{g,\gamma}_{\mathrm{M}} \cdot \xrightarrow{a,\lambda}_{\mathrm{M}} \cdot \xrightarrow{d,\mu}_{\mathrm{M}} \cdot$$
$$c_{2,2} \quad \equiv \quad \cdot \xrightarrow{g,\gamma}_{\mathrm{M}} \cdot \xrightarrow{a,\mu}_{\mathrm{M}} \cdot \xrightarrow{b,\lambda}_{\mathrm{M}} \cdot$$

- Same sum of average sojourn times $\frac{1}{2\cdot\gamma} + \frac{1}{\lambda} + \frac{1}{\mu}$ and $\frac{1}{2\cdot\gamma} + \frac{1}{\mu} + \frac{1}{\lambda}$ but ...
- *. . . an external observer would be able to distinguish between the two terms by taking note of the instants at which the actions are performed.*

Redefining Markovian Testing Equivalence

- Comparing probabilities of passing a test within a time upper bound.
- Syntax of the set $\mathbb{T}_{\mathrm{R}}$ of reactive tests $(a \in \mathit{Name}_{\mathrm{v}},\ w \in \mathbb{R}_{>0})$:

$$\boxed{\begin{array}{rcl} T & ::= & \mathrm{s} \mid T' \\ T' & ::= & <a, *_w>.T \mid T' + T' \end{array}}$$

- Asymmetric action synchronization: only passive actions within tests.
- Performance closure: passive τ-actions not admitted within tests.
- Presence of a time upper bound: recursion not necessary within tests.
- Denoting test passing: zeroary success operator s (success action may interfere).
- Avoiding ambiguous tests like $\mathrm{s} + T$: two-level syntax for tests.

- Interaction system of $P \in \mathbb{P}_{\mathrm{M,pc}}$ and $T \in \mathbb{T}_{\mathrm{R}}$:

$$\boxed{P \,\|_{Name_{\mathrm{v}}}\, T \;\in\; \mathbb{P}_{\mathrm{M,pc}}}$$

- In any of its states, P generates the proposal of an action to be executed by means of a race among the exponentially timed actions enabled in that state.

- If the name of the proposed action is τ, then P advances by itself.

- Otherwise T:

  - ⊙ either reacts by participating in the interaction with P through a passive action having the same name;

  - ⊙ or blocks the interaction if it has no passive actions with the proposed name.

- Consider the interaction system of $P \in \mathbb{P}_{\mathrm{M,pc}}$ and $T \in \mathbb{T}_{\mathrm{R}}$.
- A *configuration* is a state of $[\![P \parallel_{Name_{\mathrm{v}}} T]\!]_{\mathrm{M}}$.
- A *test-driven computation* is a computation of $[\![P \parallel_{Name_{\mathrm{v}}} T]\!]_{\mathrm{M}}$.
- A configuration is formed by a *process projection* and a *test projection*.
- A configuration is *successful* iff its test projection is s.
- A test-driven computation is *successful* iff it traverses a successful configuration.
- $\mathcal{SC}(P, T)$: multiset of successful computations of $P \parallel_{Name_{\mathrm{v}}} T$.

- If P has no exponentially timed τ-actions:
  - ⊙ All the computations in $\mathcal{SC}(P, T)$ have a finite length due to the restrictions imposed on the test syntax.
  - ⊙ All the computations in $\mathcal{SC}(P, T)$ are independent of each other because of their maximality.
  - ⊙ The multiset $\mathcal{SC}(P, T)$ is finite because both P and T are finitely branching.
- Same considerations for $\mathcal{SC}_{\leq \theta}(P, T)$.
- If there are exponentially timed τ-actions:
  - ⊙ Are the computations in $\mathcal{SC}_{\leq \theta}(P, T)$ independent of each other?
  - ⊙ How to distinguish among process terms having only exponentially timed τ-actions, like $<\tau, \lambda>.\underline{0}$ and $<\tau, \mu>.\underline{0}$ with $\lambda > \mu$?

- *Consider subsets of $\mathcal{SC}_{\leq \theta}(P, T)$ including all the successful test-driven computations of the same length.*
- They are $\mathcal{SC}^{l}_{\leq \theta}(P, T)$ for $0 \leq l \leq |\theta|$.
- $\mathcal{SC}^{|\theta|}_{\leq \theta}(P, T)$ is enough as shorter successful test-driven computations can be taken into account when imposing prefixes of θ as time upper bounds.
- Process terms having only exponentially timed τ-actions are compared after giving them the possibility of executing the same number of τ-actions.
- Example:

$$prob(\mathcal{SC}^{1}_{\leq \frac{1}{\lambda}}(<\tau, \lambda>.\underline{0}, \mathrm{s})) = 1 \neq 0 = prob(\mathcal{SC}^{1}_{\leq \frac{1}{\lambda}}(<\tau, \mu>.\underline{0}, \mathrm{s}))$$

- $P_1 \in \mathbb{P}_{\rm M,pc}$ is Markovian testing equivalent to $P_2 \in \mathbb{P}_{\rm M,pc}$, written $P_1 \sim_{\rm MT} P_2$, iff for all reactive tests $T \in \mathbb{T}_{\rm R}$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$\boxed{prob(\mathcal{SC}^{|\theta|}_{\leq \theta}(P_1, T)) \; = \; prob(\mathcal{SC}^{|\theta|}_{\leq \theta}(P_2, T))}$$

- Not defined as the intersection of may- and must-equivalence as the possibility and the necessity of passing a test are qualitative concepts, hence they are not sufficient (probability > 0, probability $= 1$).

- Not defined as the kernel of a Markovian testing preorder as such a preorder would have boiled down to an equivalence relation.

- The presence of time upper bounds makes it possible to decide whether a test is passed or not even if the process term under test can execute infinitely many exponentially timed τ-actions.

Basic Properties and Characterizations

- The new Markovian testing equivalence $\sim_{\mathrm{MT}}$ turns out to be a conservative extension of the old one $\sim_{\mathrm{MT,old}}$.

- The two behavioral equivalences coincide over the set $\mathbb{P}_{\mathrm{M,pc,v}}$ of process terms that contain no exponentially timed τ-actions.

- For all $P_1, P_2 \in \mathbb{P}_{\mathrm{M,pc,v}}$:

$$\boxed{P_1 \sim_{\mathrm{MT}} P_2 \iff P_1 \sim_{\mathrm{MT,old}} P_2}$$

- $\sim_{\mathrm{MT}}$ has the same necessary condition as $\sim_{\mathrm{MT,old}}$.

- $\sim_{\mathrm{MT}}$ has three alternative characterizations, each providing further justifications for the way in which the equivalence has been defined.

- $\sim_{\mathrm{MT}}$ has the same fully abstract characterization as $\sim_{\mathrm{MT,old}}$.

- In order for $P_1 \sim_{\mathrm{MT}} P_2$, it is necessary that for all $c_k \in \mathcal{C}_{\mathrm{f}}(P_k)$, $k \in \{1, 2\}$, there exists $c_h \in \mathcal{C}_{\mathrm{f}}(P_h)$, $h \in \{1, 2\} - \{k\}$, such that:

$$\boxed{\begin{array}{rcl} \mathit{trace}_{\mathrm{c}}(c_k) & = & \mathit{trace}_{\mathrm{c}}(c_h) \\ \mathit{time}_{\mathrm{a}}(c_k) & = & \mathit{time}_{\mathrm{a}}(c_h) \end{array}}$$

and for all $a \in \mathit{Name}$ and $i \in \{0, \dots, |c_k|\}$:

$$\boxed{\mathit{rate}_{\mathrm{o}}(P_k^i, a, 0) \; = \; \mathit{rate}_{\mathrm{o}}(P_h^i, a, 0)}$$

with P_k^i (resp. P_h^i) being the i-th state traversed by c_k (resp. c_h).

- Process terms satisfying the necessary condition that are not Markovian testing equivalent ($\lambda_1 + \lambda_2 = \lambda'_1 + \lambda'_2$ with $\lambda_1 \neq \lambda'_1$, $\lambda_2 \neq \lambda'_2$, and $b \neq c$ or $\mu \neq \gamma$):

$$<a, \lambda_1>.<b, \mu>.\underline{0} + <a, \lambda_2>.<c, \gamma>.\underline{0}$$

$$<a, \lambda'_1>.<b, \mu>.\underline{0} + <a, \lambda'_2>.<c, \gamma>.\underline{0}$$

- The first alternative characterization establishes that the discriminating power does not change if we consider a set $\mathbb{T}_{\mathrm{R,lib}}$ of tests with the following more liberal syntax:

$$\boxed{T \;::=\; \mathrm{s} \mid <a, *_w>.T \mid T + T}$$

- In this setting, a successful configuration is a configuration whose test projection includes s as top-level summand.

- For all $P_1, P_2 \in \mathbb{P}_{\mathrm{M,pc}}$:

$$\boxed{P_1 \sim_{\mathrm{MT,lib}} P_2 \iff P_1 \sim_{\mathrm{MT}} P_2}$$

- The second characterization establishes that the discriminating power does not change if we consider a set $\mathbb{T}_{\mathrm{R},\tau}$ of tests capable of moving autonomously by executing exponentially timed τ-actions:

$$\boxed{\begin{array}{rcl} T & ::= & \mathrm{s} \mid T' \\ T' & ::= & <a, *_w>.T \mid <\tau, \lambda>.T \mid T' + T' \end{array}}$$

- For all $P_1, P_2 \in \mathbb{P}_{\mathrm{M,pc}}$:

$$\boxed{P_1 \sim_{\mathrm{MT},\tau} P_2 \iff P_1 \sim_{\mathrm{MT}} P_2}$$

- The third characterization establishes that the discriminating power does not change if we consider the probability distribution of passing tests within arbitrary sequences of amounts of time.
- Considering the (more accurate) stepwise durations of test-driven computations leads to the same equivalence as considering the (easier to work with) stepwise average durations.
- $P_1 \in \mathbb{P}_{\mathrm{M,pc}}$ is Markovian distribution-testing equivalent to $P_2 \in \mathbb{P}_{\mathrm{M,pc}}$, written $P_1 \sim_{\mathrm{MT,d}} P_2$, iff for all reactive tests $T \in \mathbb{T}_{\mathrm{R}}$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of amounts of time:

$$\boxed{prob_{\mathrm{d}}(\mathcal{SC}^{|\theta|}(P_1, T), \theta) \; = \; prob_{\mathrm{d}}(\mathcal{SC}^{|\theta|}(P_2, T), \theta)}$$

- For all $P_1, P_2 \in \mathbb{P}_{\mathrm{M,pc}}$:

$$\boxed{P_1 \sim_{\mathrm{MT,d}} P_2 \iff P_1 \sim_{\mathrm{MT}} P_2}$$

- $\sim_{\rm MT}$ has another alternative characterization that *fully abstracts* from comparing process term behavior in response to tests.

- Based on traces that are extended at each step with the set of visible action names permitted by the environment at that step.

- An element ξ of $(Name_{\rm v} \times 2^{Name_{\rm v}})^*$ is an extended trace iff either ξ is the empty sequence ε or:

$$\boxed{\xi \; \equiv \; (a_1, \mathcal{E}_1) \circ (a_2, \mathcal{E}_2) \circ \ldots \circ (a_n, \mathcal{E}_n)}$$

  for some $n \in \mathbb{N}_{>0}$ with $a_i \in \mathcal{E}_i$ and $\mathcal{E}_i$ finite for each $i = 1, \ldots, n$.

- $\mathcal{ET}$: set of extended traces.

- Trace associated with $\xi \in \mathcal{ET}$:

$$\boxed{\mathit{trace}_{\mathrm{et}}(\xi) \; = \; \begin{cases} \varepsilon & \text{if } |\xi| = 0 \\ a \circ \mathit{trace}_{\mathrm{et}}(\xi') & \text{if } \xi \equiv (a, \mathcal{E}) \circ \xi' \end{cases}}$$

- $c \in \mathcal{C}_{\mathrm{f}}(P)$ is compatible with $\xi \in \mathcal{ET}$ iff:

$$\boxed{\mathit{trace}(c) \; = \; \mathit{trace}_{\mathrm{et}}(\xi)}$$

- $\mathcal{CC}(P, \xi)$: multiset of computations in $\mathcal{C}_{\mathrm{f}}(P)$ compatible with ξ.

- The probability and the duration of any computation of $\mathcal{CC}(P, \xi)$ have to be calculated by considering only the action names permitted at each step by ξ.

- Probability w.r.t. ξ of executing $c \in \mathcal{CC}(P, \xi)$:

$$\boxed{prob_{\xi}(c) \;=\; \begin{cases} 1 & \text{if } |c| = 0 \\ \frac{\lambda}{rate_{o}(P, \mathcal{E} \cup \{\tau\}, 0)} \cdot prob_{\xi'}(c') & \text{if } c \equiv P \xrightarrow{a, \lambda}_{\mathrm{M}} c' \\ & \quad \text{with } \xi \equiv (a, \mathcal{E}) \circ \xi' \\ \frac{\lambda}{rate_{o}(P, \mathcal{E} \cup \{\tau\}, 0)} \cdot prob_{\xi}(c') & \text{if } c \equiv P \xrightarrow{\tau, \lambda}_{\mathrm{M}} c' \\ & \quad \text{with } \xi \equiv (a, \mathcal{E}) \circ \xi' \\ \frac{\lambda}{rate_{o}(P, \tau, 0)} \cdot prob_{\xi}(c') & \text{if } c \equiv P \xrightarrow{\tau, \lambda}_{\mathrm{M}} c' \wedge \xi \equiv \varepsilon \end{cases}}$$

- Probability w.r.t. ξ of executing a computation in $C \subseteq \mathcal{CC}(P, \xi)$:

$$\boxed{prob_{\xi}(C) \;=\; \sum_{c \in C} prob_{\xi}(c)}$$

assuming that C is finite and all of its computations are independent.

- Stepwise average duration w.r.t. ξ of $c \in \mathcal{CC}(P, \xi)$:

$$\boxed{\mathit{time}_{\mathrm{a},\xi}(c) \;=\; \begin{cases} \varepsilon & \text{if } |c| = 0 \\ \frac{1}{\mathit{rate}_{\mathrm{o}}(P, \mathcal{E} \cup \{\tau\}, 0)} \circ \mathit{time}_{\mathrm{a},\xi'}(c') & \text{if } c \equiv P \xrightarrow{a,\lambda}_{\mathrm{M}} c' \\ & \quad \text{with } \xi \equiv (a, \mathcal{E}) \circ \xi' \\ \frac{1}{\mathit{rate}_{\mathrm{o}}(P, \mathcal{E} \cup \{\tau\}, 0)} \circ \mathit{time}_{\mathrm{a},\xi}(c') & \text{if } c \equiv P \xrightarrow{\tau,\lambda}_{\mathrm{M}} c' \\ & \quad \text{with } \xi \equiv (a, \mathcal{E}) \circ \xi' \\ \frac{1}{\mathit{rate}_{\mathrm{o}}(P, \tau, 0)} \circ \mathit{time}_{\mathrm{a},\xi}(c') & \text{if } c \equiv P \xrightarrow{\tau,\lambda}_{\mathrm{M}} c' \wedge \xi \equiv \varepsilon \end{cases}}$$

- Multiset of computations in $C \subseteq \mathcal{CC}(P, \xi)$ whose stepwise average duration w.r.t. ξ is not greater than $\theta \in (\mathbb{R}_{>0})^*$:

$$\boxed{C_{\leq \theta, \xi} \;=\; \{\!| \, c \in C \mid |c| \leq |\theta| \wedge \forall i = 1, \dots, |c|. \, \mathit{time}_{\mathrm{a},\xi}(c)[i] \leq \theta[i] \, |\!\}}$$

- C^l: multiset of computations in $C \subseteq \mathcal{CC}(P, \xi)$ having length $l \in \mathbb{N}$.

- Consider $\mathcal{CC}^{|\theta|}_{\leq\theta,\xi}(P,\xi)$ in order to ensure independence.

- $P_1 \in \mathbb{P}_{\mathrm{M,pc}}$ is Markovian extended-trace equivalent to $P_2 \in \mathbb{P}_{\mathrm{M,pc}}$, written $P_1 \sim_{\mathrm{MTr,e}} P_2$, iff for all extended traces $\xi \in \mathcal{ET}$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$\boxed{prob_\xi(\mathcal{CC}^{|\theta|}_{\leq\theta,\xi}(P_1,\xi)) \; = \; prob_\xi(\mathcal{CC}^{|\theta|}_{\leq\theta,\xi}(P_2,\xi))}$$

- For all $P_1, P_2 \in \mathbb{P}_{\mathrm{M,pc}}$:

$$\boxed{P_1 \sim_{\mathrm{MTr,e}} P_2 \iff P_1 \sim_{\mathrm{MT}} P_2}$$

- Extended traces identify a set of reactive tests necessary and sufficient in order to establish whether two terms are Markovian testing equivalent.

- Each canonical reactive test admits a main computation leading to success, whose intermediate states can have additional computations each leading to failure in one step.

- Failure is represented through a visible action name z that can occur within tests but not within process terms under test.

- Syntax of the set $\mathbb{T}_{\mathrm{R,c}}$ of canonical reactive tests ($a \in \mathcal{E}$, $\mathcal{E} \subseteq Name_{\mathrm{v}}$ finite):

$$\boxed{T \;::=\; \mathrm{s} \mid <a, *_1>.T + \sum_{b \in \mathcal{E} - \{a\}} <b, *_1>.<\mathrm{z}, *_1>.\mathrm{s}}$$

- $P_1 \sim_{\mathrm{MT}} P_2$ iff for all $T \in \mathbb{T}_{\mathrm{R,c}}$ and $\theta \in (\mathbb{R}_{>0})^*$:

$$\boxed{prob(\mathcal{SC}^{|\theta|}_{\leq \theta}(P_1, T)) \;=\; prob(\mathcal{SC}^{|\theta|}_{\leq \theta}(P_2, T))}$$

Congruence Property

- $\sim_{\rm MT}$ is a congruence over $\mathbb{P}_{\rm M,pc}$ with respect to all operators of MPC (fundamental the additional constraint on the length of successful test-driven computations).

- Let $P_1, P_2 \in \mathcal{P}_{\rm M,pc}$. Whenever $P_1 \sim_{\rm MT} P_2$, then:

$$\boxed{\begin{array}{ccc} & <a,\lambda>.P_1 \;\sim_{\rm MT}\; <a,\lambda>.P_2 & \\ P_1 + P \;\sim_{\rm MT}\; P_2 + P & & P + P_1 \;\sim_{\rm MT}\; P + P_2 \\ P_1 \parallel_S P \;\sim_{\rm MT}\; P_2 \parallel_S P & & P \parallel_S P_1 \;\sim_{\rm MT}\; P \parallel_S P_2 \\ & P_1 \,/\, H \;\sim_{\rm MT}\; P_2 \,/\, H & \\ & P_1[\varphi] \;\sim_{\rm MT}\; P_2[\varphi] & \end{array}}$$

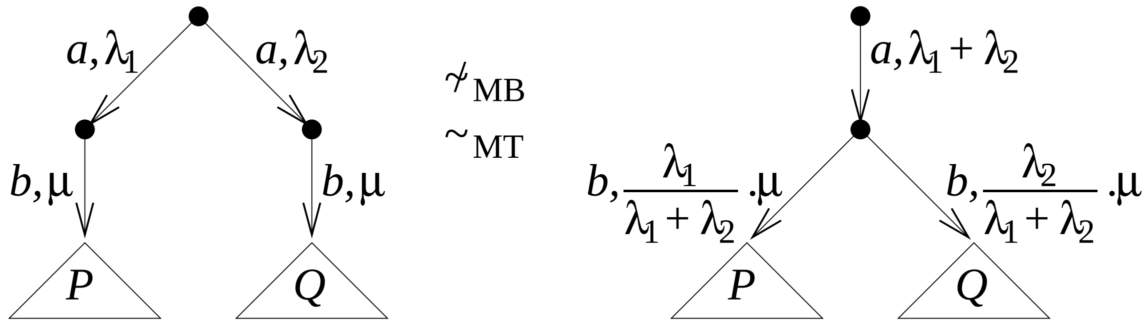
  provided that $P \in \mathcal{P}_{\rm M,pc}$ for the alternative composition operator and $P_1 \parallel_S P, P_2 \parallel_S P \in \mathcal{P}_{\rm M,pc}$ for the parallel composition operator.

- Only a partial congruence result w.r.t. parallel composition for $\sim_{\rm MT,old}$.

Sound and Complete Axiomatization

- $\sim_{\rm MT}$ has a sound and complete axiomatization over the set $\mathbb{P}_{\rm M,pc,nrec}$ of nonrecursive process terms.

- The axioms for $\sim_{\rm MB}$ are sound but not complete for $\sim_{\rm MT}$ ($P \not\sim_{\rm MB} Q$):



- *Possibility of deferring choices related to branches starting with the same action name (see the two a-branches on the left-hand side) that are immediately followed by sets of actions having the same names and total rates (see $\{<b, \mu>\}$ after each of the two a-branches).*

- Basic laws (identical to those for $\sim_{\mathrm{MB}}$):

$$\begin{array}{lrcl}
(\mathcal{X}_{\mathrm{MT},1}) & P_1 + P_2 & = & P_2 + P_1 \\
(\mathcal{X}_{\mathrm{MT},2}) & (P_1 + P_2) + P_3 & = & P_1 + (P_2 + P_3) \\
(\mathcal{X}_{\mathrm{MT},3}) & P + \underline{0} & = & P
\end{array}$$

- Characterizing law (subsumes $\sim_{\mathrm{MB}}$ axiom for race policy):

$$(\mathcal{X}_{\mathrm{MT},4}) \quad \sum_{i \in I} {<}a, \lambda_i{>}. \sum_{j \in J_i} {<}b_{i,j}, \mu_{i,j}{>}.P_{i,j} \quad = \quad {<}a, \sum_{k \in I} \lambda_k{>}. \sum_{i \in I} \sum_{j \in J_i} {<}b_{i,j}, \frac{\lambda_i}{\Sigma_{k \in I} \lambda_k} \cdot \mu_{i,j}{>}.P_{i,j}$$

if: I is a finite index set with $|I| \geq 2$;
for all $i \in I$, index set J_i is finite and its summation is $\underline{0}$ if $J_i = \emptyset$;
for all $i_1, i_2 \in I$ and $b \in \mathit{Name}$:

$$\sum_{j \in J_{i_1}} \{\!| \, \mu_{i_1,j} \mid b_{i_1,j} = b \, |\!\} \quad = \quad \sum_{j \in J_{i_2}} \{\!| \, \mu_{i_2,j} \mid b_{i_2,j} = b \, |\!\}$$

- Expansion law (identical to that for $\sim_{\mathrm{MB}}$):

$$
\begin{array}{lrcl}
(\mathcal{X}_{\mathrm{MT},5}) & \sum_{i \in I} <a_i, \tilde{\lambda}_i>.P_i \parallel_S \sum_{j \in J} <b_j, \tilde{\mu}_j>.Q_j & = & \\
& \multicolumn{3}{l}{\sum_{k \in I, a_k \notin S} <a_k, \tilde{\lambda}_k>.\left(P_k \parallel_S \sum_{j \in J} <b_j, \tilde{\mu}_j>.Q_j\right) \; +} \\
& \multicolumn{3}{l}{\sum_{h \in J, b_h \notin S} <b_h, \tilde{\mu}_h>.\left(\sum_{i \in I} <a_i, \tilde{\lambda}_i>.P_i \parallel_S Q_h\right) \; +} \\
& \multicolumn{3}{l}{\sum_{k \in I, a_k \in S, \tilde{\lambda}_k \in \mathbb{R}_{>0}} \; \sum_{h \in J, b_h = a_k, \tilde{\mu}_h = *_{w_h}} <a_k, \tilde{\lambda}_k \cdot \frac{w_h}{weight(Q, b_h)}>.(P_k \parallel_S Q_h) \; +} \\
& \multicolumn{3}{l}{\sum_{h \in J, b_h \in S, \tilde{\mu}_h \in \mathbb{R}_{>0}} \; \sum_{k \in I, a_k = b_h, \tilde{\lambda}_k = *_{v_k}} <b_h, \tilde{\mu}_h \cdot \frac{v_k}{weight(P, a_k)}>.(P_k \parallel_S Q_h) \; +} \\
& \multicolumn{3}{l}{\sum_{k \in I, a_k \in S, \tilde{\lambda}_k = *_{v_k}} \; \sum_{h \in J, b_h = a_k, \tilde{\mu}_h = *_{w_h}} <a_k, *_{norm(v_k, w_h, a_k, P, Q)}>.(P_k \parallel_S Q_h)} \\
(\mathcal{X}_{\mathrm{MT},6}) & \sum_{i \in I} <a_i, \tilde{\lambda}_i>.P_i \parallel_S \underline{0} & = & \sum_{k \in I, a_k \notin S} <a_k, \tilde{\lambda}_k>.P_k \\
(\mathcal{X}_{\mathrm{MT},7}) & \underline{0} \parallel_S \sum_{j \in J} <b_j, \tilde{\mu}_j>.Q_j & = & \sum_{h \in J, b_h \notin S} <b_h, \tilde{\mu}_h>.Q_h \\
(\mathcal{X}_{\mathrm{MT},8}) & \underline{0} \parallel_S \underline{0} & = & \underline{0}
\end{array}
$$

- Distribution laws (identical to those for $\sim_{\rm MB}$):

$$\begin{array}{|lrcll|}
\hline
(\mathcal{X}_{\rm MT,9}) & \underline{0} \,/\, H & = & \underline{0} & \\
(\mathcal{X}_{\rm MT,10}) & (<a, \tilde{\lambda}>.P) \,/\, H & = & <\tau, \tilde{\lambda}>.(P \,/\, H) & \text{if } a \in H \\
(\mathcal{X}_{\rm MT,11}) & (<a, \tilde{\lambda}>.P) \,/\, H & = & <a, \tilde{\lambda}>.(P \,/\, H) & \text{if } a \notin H \\
(\mathcal{X}_{\rm MT,12}) & (P_1 + P_2) \,/\, H & = & P_1 \,/\, H + P_2 \,/\, H & \\
 & & & & \\
(\mathcal{X}_{\rm MT,13}) & \underline{0}[\varphi] & = & \underline{0} & \\
(\mathcal{X}_{\rm MT,14}) & (<a, \tilde{\lambda}>.P)[\varphi] & = & <\varphi(a), \tilde{\lambda}>.(P[\varphi]) & \\
(\mathcal{X}_{\rm MT,15}) & (P_1 + P_2)[\varphi] & = & P_1[\varphi] + P_2[\varphi] & \\
\hline
\end{array}$$

- Laws dealing with concurrency not available for $\sim_{\rm MT,old}$.

- The deduction system $DED(\mathcal{X}_{\rm MT})$ is sound and complete for $\sim_{\rm MT}$ over $\mathbb{P}_{\rm M,pc,nrec}$; i.e., for all $P_1, P_2 \in \mathbb{P}_{\rm M,pc,nrec}$:

$$\boxed{P_1 \sim_{\rm MT} P_2 \iff \mathcal{X}_{\rm MT} \vdash P_1 = P_2}$$

Modal Logic Characterization

- $\sim_{\mathrm{MT}}$ has a modal logic characterization over $\mathbb{P}_{\mathrm{M,pc}}$ based on a variant of the Hennessy-Milner logic.
- Negation is not included and conjunction is replaced by disjunction (decreased discriminating power with respect to $\sim_{\mathrm{MB}}$).
- Syntax of the modal language $\mathcal{ML}_{\mathrm{MT}}$ ($a \in Name_{\mathrm{v}}$):

$$\boxed{\begin{array}{rcl} \phi & ::= & \mathrm{true} \mid \phi' \\ \phi' & ::= & \langle a \rangle \phi \mid \phi' \vee \phi' \end{array}}$$

where each formula of the form $\phi_1 \vee \phi_2$ satisfies the following constraint (consistent with the name-deterministic nature of canonical reactive tests):

$$init(\phi_1) \cap init(\phi_2) \; = \; \emptyset$$

with $init(\phi)$ being defined as follows:

$$init(\mathrm{true}) = \emptyset \qquad init(\phi_1 \vee \phi_2) = init(\phi_1) \cup init(\phi_2) \qquad init(\langle a \rangle \phi) = \{a\}$$

- No quantitative decorations in the syntax because the focus is on entire computations rather than on step-by-step behavior mimicking, but ...
- ... replacement of the boolean satisfaction relation with a quantitative interpretation function measuring the probability with which a process term satisfies a formula quickly enough on average.
- Interpretation of $\mathcal{ML}_{\mathrm{MT}}$ over $\mathbb{P}_{\mathrm{M,pc}}$:

$$\boxed{[\![\phi]\!]^{|\theta|}_{\mathrm{MT}}(P,\theta) \;=\; \begin{cases} 0 & \text{if } |\theta| = 0 \land \phi \not\equiv \text{true} \\ & \quad \text{or } |\theta| > 0 \land \mathit{rate}_{\mathrm{o}}(P, \mathit{init}(\phi) \cup \{\tau\}, 0) = 0 \\ 1 & \text{if } |\theta| = 0 \land \phi \equiv \text{true} \end{cases}}$$

otherwise:

$$[\![\mathrm{true}]\!]_{\mathrm{MT}}^{|t\circ\theta|}(P, t\circ\theta) \;=\; \begin{cases} \displaystyle\sum_{P \xrightarrow{\tau,\lambda}_{\mathrm{M}} P'} \frac{\lambda}{rate_{\mathrm{o}}(P,\tau,0)} \cdot [\![\mathrm{true}]\!]_{\mathrm{MT}}^{|\theta|}(P',\theta) & \text{if } \frac{1}{rate_{\mathrm{o}}(P,\tau,0)} \leq t \\ 0 & \text{if } \frac{1}{rate_{\mathrm{o}}(P,\tau,0)} > t \end{cases}$$

$$[\![\langle a\rangle\phi]\!]_{\mathrm{MT}}^{|t\circ\theta|}(P, t\circ\theta) \;=\; \begin{cases} \displaystyle\sum_{P \xrightarrow{a,\lambda}_{\mathrm{M}} P'} \frac{\lambda}{rate_{\mathrm{o}}(P,\{a,\tau\},0)} \cdot [\![\phi]\!]_{\mathrm{MT}}^{|\theta|}(P',\theta) \;+ \\ \quad\displaystyle\sum_{P \xrightarrow{\tau,\lambda}_{\mathrm{M}} P'} \frac{\lambda}{rate_{\mathrm{o}}(P,\{a,\tau\},0)} \cdot [\![\langle a\rangle\phi]\!]_{\mathrm{MT}}^{|\theta|}(P',\theta) \\ & \text{if } \frac{1}{rate_{\mathrm{o}}(P,\{a,\tau\},0)} \leq t \\ 0 & \text{if } \frac{1}{rate_{\mathrm{o}}(P,\{a,\tau\},0)} > t \end{cases}$$

$$
\begin{aligned}
\llbracket \phi_1 \vee \phi_2 \rrbracket_{\mathrm{MT}}^{|t \circ \theta|}(P, t \circ \theta) \;=\;& p_1 \cdot \llbracket \phi_1 \rrbracket_{\mathrm{MT}}^{|t_1 \circ \theta|}(P_{\mathit{no\text{-}init\text{-}\tau}}, t_1 \circ \theta) \, + \\
& p_2 \cdot \llbracket \phi_2 \rrbracket_{\mathrm{MT}}^{|t_2 \circ \theta|}(P_{\mathit{no\text{-}init\text{-}\tau}}, t_2 \circ \theta) \, + \\
& \sum_{P \xrightarrow{\tau, \lambda}_{\mathrm{M}} P'} \frac{\lambda}{\mathit{rate}_{\mathrm{o}}(P, \mathit{init}(\phi_1 \vee \phi_2) \cup \{\tau\}, 0)} \cdot \llbracket \phi_1 \vee \phi_2 \rrbracket_{\mathrm{MT}}^{|\theta|}(P', \theta)
\end{aligned}
$$

where:

- $P_{\mathit{no\text{-}init\text{-}\tau}}$ is P without computations starting with a τ-transition.
- For $j \in \{1, 2\}$:

$$
\begin{aligned}
p_j &= \frac{\mathit{rate}_{\mathrm{o}}(P, \mathit{init}(\phi_j), 0)}{\mathit{rate}_{\mathrm{o}}(P, \mathit{init}(\phi_1 \vee \phi_2) \cup \{\tau\}, 0)} \\
t_j &= t + \left(\frac{1}{\mathit{rate}_{\mathrm{o}}(P, \mathit{init}(\phi_j), 0)} - \frac{1}{\mathit{rate}_{\mathrm{o}}(P, \mathit{init}(\phi_1 \vee \phi_2) \cup \{\tau\}, 0)} \right)
\end{aligned}
$$

with p_j representing the conditional probability with which P performs actions whose name is in $\mathit{init}(\phi_j)$ and t_j representing the extra average time granted to P for satisfying ϕ_j.

- The constraint on disjunctions guarantees that their subformulas exercise independent computations of P (correct probability calculation).

- In the absence of p_1 and p_2, the fact that $\phi_1 \vee \phi_2$ offers a set of initial actions at least as large as the ones offered by ϕ_1 alone and by ϕ_2 alone may lead to an overestimate of the probability of satisfying $\phi_1 \vee \phi_2$.

- Considering t instead of t_j in the satisfaction of ϕ_j in isolation may lead to an underestimate of the probability of satisfying $\phi_1 \vee \phi_2$ within the given time upper bound, as P may satisfy $\phi_1 \vee \phi_2$ within $t \circ \theta$ even if P satisfies neither ϕ_1 nor ϕ_2 taken in isolation within $t \circ \theta$.

- For all $P_1, P_2 \in \mathbb{P}_{\mathrm{M,pc}}$:

$$\boxed{P_1 \sim_{\mathrm{MT}} P_2 \iff \forall \phi \in \mathcal{ML}_{\mathrm{MT}}.\, \forall \theta \in (\mathbb{R}_{>0})^*.\, [\![\phi]\!]^{|\theta|}_{\mathrm{MT}}(P_1, \theta) = [\![\phi]\!]^{|\theta|}_{\mathrm{MT}}(P_2, \theta)}$$

Verification Algorithm

- $\sim_{\mathrm{MT}}$ is decidable in polynomial time over the set $\mathbb{P}_{\mathrm{M,pc,fin}}$ of finite-state process terms.
- The reason is that:
  - ⊙ $\sim_{\mathrm{MT}}$ coincides with the Markovian version of ready equivalence.
  - ⊙ Probabilistic ready equivalence can be decided in polynomial time through a suitable reworking of Tzeng algorithm for probabilistic language equivalence.
- Given two process terms, their name-labeled CTMCs are Markovian ready equivalent iff the corresponding embedded name-labeled DTMCs are probabilistic ready equivalent.
- Markovian ready equivalence and probabilistic ready equivalence coincide on corresponding models if the total exit rate of each state of a name-labeled CTMC is encoded inside the names of all transitions departing from that state in the associated embedded DTMC.

- Steps of the algorithm for checking whether $P_1 \sim_{\mathrm{MT}} P_2$:
  1. Transform $[\![P_1]\!]_{\mathrm{M}}$ and $[\![P_2]\!]_{\mathrm{M}}$ into their corresponding embedded discrete-time versions:
     a. Divide the rate of each transition by the total exit rate of its source state.
     b. Augment the name of each transition with the total exit rate of its source state.
  2. Compute the relation $\mathcal{R}$ that equates any two states of the discrete-time versions of $[\![P_1]\!]_{\mathrm{M}}$ and $[\![P_2]\!]_{\mathrm{M}}$ whenever the two sets of augmented action names labeling the transitions departing from the two states coincide.
  3. For each equivalence class R induced by $\mathcal{R}$, consider R as the set of accepting states and check whether the discrete-time versions of $[\![P_1]\!]_{\mathrm{M}}$ and $[\![P_2]\!]_{\mathrm{M}}$ are probabilistic language equivalent.
  4. Return yes/no depending on whether all the checks performed in the previous step have been successful or not.

- Tzeng algorithm for probabilistic language equivalence visits in breadth-first order the tree containing a node for each possible string and studies the linear independence of the state probability vectors associated with a finite subset of the tree nodes.
- Refinement of each iteration of step 3:
  1. Create an empty set V of state probability vectors.
  2. Create a queue whose only element is the empty string ε.
  3. While the queue is not empty:
     a. Remove the first element from the queue, say string ς.
     b. If the state probability vector of the discrete-time versions of $[\![P_1]\!]_{\mathrm{M}}$ and $[\![P_2]\!]_{\mathrm{M}}$ after reading ς does not belong to the vector space generated by V, then:
        i. For each $a \in \mathit{NameReal}_{P_1,P_2}$, add $\varsigma \circ a$ to the queue.
        ii. Add the state probability vector to V.

4. Build a three-valued state vector $\boldsymbol{u}$ whose generic element is:
   a. 0 if it corresponds to a nonaccepting state.
   b. 1 if it corresponds to an accepting state of $[\![P_1]\!]_{\mathrm{M}}$.
   c. -1 if it corresponds to an accepting state of $[\![P_2]\!]_{\mathrm{M}}$.
5. For each $\boldsymbol{v} \in V$, check whether $\boldsymbol{v} \cdot \boldsymbol{u}^{\mathrm{T}} = 0$.
6. Return yes/no depending on whether all the checks performed in the previous step have been successful or not.

- The time complexity of the overall algorithm is $O(n^5)$.

Future Work

- Investigate whether the introduction of $<\tau, \lambda>$ actions within tests makes average time upper bounds useless.
- Find a fully abstract characterization of a more denotational nature; e.g., a suitable variant of acceptance trees.
- Devise a minimization algorithm based on $\sim_{\mathrm{MT}}$.