# Markovian Behavioral Equivalences:
# A Comparative Survey

Marco Bernardo

University of Urbino – Italy

# Table of Contents

# *Part I:*
# Introduction

# Performance-Oriented Notations

- Building performance-aware system models:
  - Predicting the satisfiability of QoS requirements.
  - Choosing among alternative designs based on their expected QoS.
- *Theory*:
  - Queueing networks.
  - Stochastic Petri nets.
  - Stochastic process algebras.
- *Practice*:
  - Formal modeling languages (MODEST).
  - Architectural description languages (ÆMILIA).
  - Coordination languages (STOKLAIM).
  - Object-oriented modeling languages (UML SPT/MARTE).

- Performance-oriented notations usually produce behavioral models.
- Class of uniform models consisting of state-transition graphs.
- The current state represents:
  - the current number of customers in each service center;
  - the current Petri net marking;
  - the current process term.
- Each transition represents a state change due to:
  - the execution of certain activities;
  - the occurrence of certain events.
- Interleaving view of concurrency.

# Behavioral Equivalences

- Models are equivalent if they represent systems that behave the same.
- Useful for theoretical and applicative purposes:
  - ⊙ Comparing models that are syntactically different on the basis of the behavior they exhibit.
  - ⊙ Relating models of the same system at different abstraction levels (top-down modeling).
  - ⊙ Manipulating models in a way that system properties are preserved (state space reduction before analysis).

- Several different approaches to the definition of behavioral equivalences developed in a purely functional framework:
  - ⊙ Bisimilarity [Milner]: two models are equivalent if they are able to *mimic* each other's behavior *stepwise*.
  - ⊙ Testing [De Nicola, Hennessy]: two models are equivalent if an *external observer* cannot distinguish between them by interacting with them by means of *tests* and looking at their reactions.
  - ⊙ Trace [Hoare]: two models are equivalent if they are able to perform the same *sequences* of activities.
- How to include performance aspects in behavioral equivalences?

# Markovian Framework

- Quantitative description of system evolution over time.
- A Markov chain is a discrete-state stochastic process $\{X(t) \mid t \in \mathbb{R}_{\geq 0}\}$ such that for all $n \in \mathbb{N}$, time instants $t_0 < t_1 < \ldots < t_n < t_{n+1}$, and states $s_0, s_1, \ldots, s_n, s_{n+1} \in S$:

$$\boxed{\begin{aligned}&\Pr\{X(t_{n+1}) = s_{n+1} \mid X(t_0) = s_0 \wedge X(t_1) = s_1 \wedge \ldots \wedge X(t_n) = s_n\} = \\ &\qquad \Pr\{X(t_{n+1}) = s_{n+1} \mid X(t_n) = s_n\}\end{aligned}}$$

- The past history is completely summarized by the current state.
- Equivalently, the stochastic process has no memory of the past.
- Time homogeneity: probabilities independent of state change times.
- The solution of a Markov chain is its state probability distribution $\boldsymbol{\pi}()$ at an arbitrary time instant.

- In the continuous-time case (CTMC):
  - State transitions are described by a rate matrix $\mathbf{Q}$.
  - The sojourn time in any state is exponentially distributed.
  - Given $\boldsymbol{\pi}(0)$, the transient solution $\boldsymbol{\pi}(t)$ is obtained by solving:

$$\boxed{\boldsymbol{\pi}(t) \cdot \mathbf{Q} = \frac{d\boldsymbol{\pi}(t)}{dt}}$$

  - The stationary solution $\boldsymbol{\pi} = \lim_{t \to \infty} \boldsymbol{\pi}(t)$ is obtained (if any) by solving:

$$\boxed{\begin{array}{rcl} \boldsymbol{\pi} \cdot \mathbf{Q} & = & \mathbf{0} \\ \sum_{s \in S} \pi[s] & = & 1 \end{array}}$$

- Exponentially distributed random variables are the only continuous random variables satisfying the memoryless property:

$$\boxed{\Pr\{X \leq t + t' \mid X > t'\} = \Pr\{X \leq t\}}$$

- Every CTMC has an embedded DTMC:
  - State transitions are described by a probability matrix $\mathbf{P}$.
  - $\mathbf{P}$ is obtained from $\mathbf{Q}$ by dividing the rate of each transition by the sum of the rates of the transitions that depart from the source state and do not return to it.
  - The sojourn time in any state is geometrically distributed.
  - Given $\boldsymbol{\pi}(0)$, the transient solution $\boldsymbol{\pi}(n)$ is computed as follows:

$$\boxed{\boldsymbol{\pi}(n) = \boldsymbol{\pi}(0) \cdot \mathbf{P}^n}$$

  - The stationary solution $\boldsymbol{\pi} = \lim_{n \to \infty} \boldsymbol{\pi}(n)$ is obtained (if any) by solving:

$$\boxed{\begin{aligned} \boldsymbol{\pi} &= \boldsymbol{\pi} \cdot \mathbf{P} \\ \sum_{s \in S} \pi[s] &= 1 \end{aligned}}$$

# Markovian Behavioral Equivalences

- The memoryless property of the exponential distribution results in a simpler mathematical treatment ...
  - $\triangleleft$ easy calculation of *state sojourn times* and *transition probabilities*;
  - $\triangleleft$ compliance with *interleaving semantics* of state-transition graphs;

  ... without sacrificing expressiveness
  - $\triangleright$ adequate for modeling the timing of many *real-life phenomena* like arrival processes, failure events, and chemical reactions;
  - $\triangleright$ most appropriate stochastic approximation in the case in which only the *average duration* of an activity is known;
  - $\triangleright$ proper combinations (phase-type distributions) can approximate most of *general distributions* arbitrarily closely.
- How to define Markovian behavioral equivalences?

- Comparison criteria for Markovian behavioral equivalences:

  1. Discriminating power:
     which of them is finer/coarser than the others?
  2. Exactness (of their induced CTMC-level aggregations):
     do they make sense from the performance viewpoint?
  3. Congruence:
     do they support compositional reasoning?
  4. Axiomatization:
     what are their fundamental equational laws?
  5. Modal logic characterization:
     what behavioral properties do they preserve?
  6. Complexity (of their verification algorithms):
     can they be checked for efficiently?

- Exactness: the probability of being in a macrostate of an aggregated CTMC is the sum of the probabilities of being in one of the constituent microstates of the original CTMC (transient/stationary).
- Exactness guarantees the preservation of performance characteristics when going from the original CTMC to the aggregated one.
- Congruence enables the compositional minimization of models.
- Axioms can be used as rewriting rules that syntactically manipulate models in a way that is consistent with the equivalence.
- The modal logic characterization provides diagnostic information in the form of distinguishing formulas that explain model inequivalence.

# *Part II:* Markovian Process Algebra

# Process Algebraic Markovian Modeling

- Behavioral equivalences abstract from specific kinds of models but ...
- ... are better investigated and understood in a process algebraic setting.
- Action-based modeling relaying on a set of behavioral operators.
- Performance-oriented process calculi with CTMC semantics:
  - ⊙ TIPP [Götz, Herzog, Rettelbach].
  - ⊙ PEPA [Hillston].
  - ⊙ MPA [Buchholz].
  - ⊙ $\mathrm{EMPA}_{\mathrm{gr}}$ [Bernardo, Bravetti, Gorrieri].
  - ⊙ S$\pi$ [Priami].
  - ⊙ IMC [Hermanns].
  - ⊙ PIOA [Stark, Cleaveland, Smolka].

- Markovian process calculi differ for the action representation.
- Durational actions (integrated time):
  - An action is executed while time passes.
  - Single action prefix operator comprising the name $a$ of the action and the rate $\lambda \in \mathbb{R}_{>0}$ of the exponentially distributed random variable quantifying the duration of the action: $<a, \lambda>.\_$
  - The choice among several actions is probabilistic.
  - TIPP, PEPA, MPA, $\mathrm{EMPA}_{\mathrm{gr}}$, $\mathrm{S}\pi$, PIOA.
- Action names separated from time (orthogonal time):
  - An action is instantaneously executed after some time has elapsed.
  - Two action prefix operators: $(\lambda).\_$ and $a.\_$
  - The choice among several actions is nondeterministic.
  - IMC.

- Markovian process calculi also differ for the discipline adopted for action synchronization.
- In the orthogonal time case action synchronization is governed as in nondeterministic process calculi.
- In the integrated time case action synchronization can be handled in different ways.
- The more natural choice for deciding the duration of the synchronization of two exponentially timed actions would be to take the maximum of their durations.
- *The maximum of two exponentially distributed random variables is not exponentially distributed* (phase-type: IMC).

- Symmetric synchronizations:
  - The synchronization of two exponentially timed actions is assumed to be exponentially timed.
  - Its rate is defined through an associative and commutative operator applied to the two original rates (multiplication, min, max).
  - TIPP, PEPA, MPA, S$\pi$.
- Asymmetric synchronizations:
  - Passive actions of the form $<a, *_w>$ whose duration is undefined.
  - An exponentially timed action can synchronize only with a passive action, thus determining the duration of the synchronization.
  - PEPA, EMPA$_{\mathrm{gr}}$, PIOA.
- The rate of an action should not increase when synchronizing that action with other actions (bounded capacity assumption).

# Markovian Process Calculus

- Basic design choices: durational actions (more natural modeling style) and asymmetric synchronizations (reintroducing nondeterminism).
- $\mathit{Name}_{\mathrm{v}}$: set of visible action names.
- $\mathit{Name} = \mathit{Name}_{\mathrm{v}} \cup \{\tau\}$: set of all action names.
- $\mathit{Rate} = \mathbb{R}_{>0} \cup \{*_w \mid w \in \mathbb{R}_{>0}\}$: set of action rates.
- $\mathit{Act}_{\mathrm{M}} = \mathit{Name} \times \mathit{Rate}$: set of exponentially timed and passive actions.
- $\mathit{Relab} = \{\varphi : \mathit{Name} \rightarrow \mathit{Name} \mid \varphi^{-1}(\tau) = \{\tau\}\}$: set of visibility-preserving relabeling functions.
- $\mathit{Var}$: set of process variables.
- Behavioral operators: dynamic vs. static.

- Syntax of the language $\mathcal{L}_{\mathrm{M}}$ for MPC:

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

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

- The duration of $<a, \lambda>$ is given by exponentially distributed random variable $Exp_\lambda$: $\Pr\{Exp_\lambda \leq t\} = 1 - \mathrm{e}^{-\lambda \cdot t}$ and $\mathrm{E}\{Exp_\lambda\} = 1/\lambda$.

- The duration of $<a, *_w>$ is undefined.

- The choice among exp. timed actions is generative (prob. over arbitrary names) and is solved by applying the race policy (exec. prob. proportional to action rates).

- The choice among passive actions is reactive (prob. restricted to same name):
  - probabilistic for passive actions with the same name and solved by applying the preselection policy (exec. prob. proportional to action weights);
  - nondeterministic for passive actions with different names.

- The choice between an exponentially timed action and a passive action is nondeterministic.

- Applying the race policy to the exponentially timed actions enabled by a process term means executing the *fastest* action.
- The sojourn time in that term is thus the *minimum* of the random variables quantifying the durations of the enabled exp. timed actions.
- *The minimum of a set of* $n \in \mathbb{N}_{>0}$ *exponentially distributed random variables with rates* $\lambda_1, \lambda_2, \ldots, \lambda_n \in \mathbb{R}_{>0}$ *is an exponentially distributed random variable with rate* $\lambda_1 + \lambda_2 + \ldots + \lambda_n$.
- The sojourn time associated with a term is exponentially distributed with rate given by the sum of the rates of the enabled exp. timed actions.
- The average sojourn time is the reciprocal of the sum of those rates.
- The execution probability of one of the enabled exp. timed action turns out to be the ratio of its rate to the sum of the rates of all those actions.

- Process term semantics based on state-transition models where:
  - states correspond to process terms;
  - transitions are labeled with actions.
- Recording transition multiplicities to distinguish between terms like $<a, \lambda>.\underline{0} + <a, \lambda>.\underline{0}$ and $<a, \lambda>.\underline{0}$.
- Labeled multitransition system $[\![ P ]\!]_{\mathrm{M}} = (\mathcal{P}_{\mathrm{M}}, Act_{\mathrm{M}}, \longrightarrow_{\mathrm{M}}, P)$.
- CTMC derivation if there are no passive transitions:
  - Drop action names from all transitions of $[\![ P ]\!]_{\mathrm{M}}$.
  - Collapse all the transitions between any two states of $[\![ P ]\!]_{\mathrm{M}}$ into a single transition by summing up the rates of the original transitions.
- $\mathcal{P}_{\mathrm{M,pc}}$: set of performance closed process terms of $\mathcal{P}_{\mathrm{M}}$.

- Derivation of one single transition at a time by applying operational semantic rules inductively defined on the syntactical structure of the process term associated with the source state of the transition.

- Operational semantic rules for dynamic operators and recursion:

$$<a, \lambda>.P \xrightarrow{a,\lambda}_{\mathrm{M}} P \qquad\qquad <a, *_w>.P \xrightarrow{a,*_w}_{\mathrm{M}} P$$

$$\frac{P_1 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'}{P_1 + P_2 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'} \qquad\qquad \frac{P_2 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'}{P_1 + P_2 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'}$$

$$\frac{P\{(rec\, X : P)/X\} \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'}{rec\, X : P \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'}$$

- Operational semantic rules for unary static operators:

$$\frac{P \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P' \qquad a \in H}{P/H \xrightarrow{\tau,\tilde{\lambda}}_{\mathrm{M}} P'/H} \qquad \frac{P \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P' \qquad a \notin H}{P/H \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'/H}$$

$$\frac{P \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P' \qquad a \notin L}{P \backslash L \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P' \backslash L}$$

$$\frac{P \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P'}{P[\varphi] \xrightarrow{\varphi(a),\tilde{\lambda}}_{\mathrm{M}} P'[\varphi]}$$

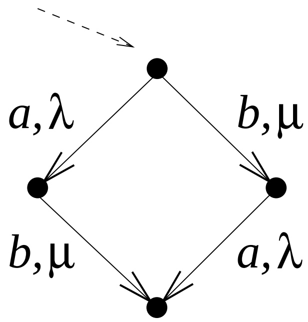
- Classical interleaving semantics for parallel composition:
  - ⊙ *Due to the memoryless property of the exponential distribution, the execution of an exponentially timed action can be thought of as being started in the last state in which the action is enabled.*
  - ⊙ *Due to the infinite support of the exponential distribution, the prob. of simultaneous termination of two concurrent exponentially timed actions is zero.*
- Operational semantic rules for parallel composition in the case of non-synchronizing actions:

$$\frac{P_1 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P_1' \qquad a \notin S}{P_1 \parallel_S P_2 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P_1' \parallel_S P_2} \qquad\qquad \frac{P_2 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P_2' \qquad a \notin S}{P_1 \parallel_S P_2 \xrightarrow{a,\tilde{\lambda}}_{\mathrm{M}} P_1 \parallel_S P_2'}$$

- Syntactically and structurally different process terms like:

$$<a, \lambda>.\underline{0} \parallel_{\emptyset} <b, \mu>.\underline{0}$$
$$<a, \lambda>.<b, \mu>.\underline{0} + <b, \mu>.<a, \lambda>.\underline{0}$$

yield the same labeled multitransition system:




- Interleave concurrent exp. timed actions without adjusting their rates inside transition labels.

- Synchronization admitted only between two actions with the same name, provided that at most one of them is exponentially timed.

- The synchronization of two exponentially timed actions is forbidden.

- Generative-reactive or reactive-reactive synchronizations.

- The rate of the synchronization of an exponentially timed action with a passive action is given by the rate of the former multiplied by the execution probability of the latter (complies with the bounded capacity assumption).

- Multiway synchronizations among actions with the same name are allowed only if they involve at most one exponentially timed action.

- Weight of a process term $P$ with respect to passive actions of name $a$:

$$\boxed{weight(P, a) \;=\; \sum \{\!\mid w \in \mathbb{R}_{>0} \mid \exists P' \in \mathcal{P}_{\mathrm{M}} .\, P \xrightarrow{a, *_w}_{\mathrm{M}} P' \mid\!\}}$$

- Operational semantic rules for generative-reactive synchronizations:

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

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

- Operational semantic rule for reactive-reactive synchronizations:

$$\frac{P_1 \xrightarrow{a,*_{w_1}}_{\mathrm{M}} P_1' \qquad P_2 \xrightarrow{a,*_{w_2}}_{\mathrm{M}} P_2' \qquad a \in S}{P_1 \parallel_S P_2 \xrightarrow{a,*_{\frac{w_1}{weight(P_1,a)}\cdot\frac{w_2}{weight(P_2,a)}\cdot(weight(P_1,a)+weight(P_2,a))}}_{\mathrm{M}} P_1' \parallel_S P_2'}$$

# *Part III:* Markovian Bisimilarity

# Equivalence Definition

- Two process terms are equivalent if they are able to mimic each other's functional and performance behavior stepwise.

- Whenever a process term can perform actions with a certain name that reach a certain set of terms at a certain speed, then any process term equivalent to the given one has to be able to respond with actions with the same name that reach an equivalent set of terms at the same speed.

- Comparison of process term exit rates rather than individual transitions (different from classical bisimilarity).

- High sensitivity to branching points in process term behavior.

- The exit rate of a process term is the rate at which the process term can execute actions of a given name that lead to a given set of terms.
- Two-level definition as there are two kinds of actions.
- Exit rate at which $P \in \mathcal{P}_{\mathrm{M}}$ executes actions of name $a \in \mathit{Name}$ and level $l \in \{0, -1\}$ that lead to $C \subseteq \mathcal{P}_{\mathrm{M}}$:

$$\boxed{\mathit{rate}_{\mathrm{e}}(P, a, l, C) \;=\; \begin{cases} \sum\{\!| \, \lambda \in \mathbb{R}_{>0} \mid \exists P' \in C.\, P \xrightarrow{a,\lambda}_{\mathrm{M}} P' \, |\!\} & \text{if } l = 0 \\ \sum\{\!| \, w \in \mathbb{R}_{>0} \mid \exists P' \in C.\, P \xrightarrow{a,*_w}_{\mathrm{M}} P' \, |\!\} & \text{if } l = -1 \end{cases}}$$

- Overall exit rate of $P$ with respect to $a$ at level $l$ (it is $\mathit{weight}(P, a)$ when $l = -1$):

$$\boxed{\mathit{rate}_{\mathrm{o}}(P, a, l) \;=\; \mathit{rate}_{\mathrm{e}}(P, a, l, \mathcal{P}_{\mathrm{M}})}$$

- Total exit rate of $P$ at level $l$ (inverse of avg sojourn time for $l = 0$ when $P \in \mathcal{P}_{\mathrm{M,pc}}$):

$$\boxed{\mathit{rate}_{\mathrm{t}}(P, l) \;=\; \sum_{a \in \mathit{Name}} \mathit{rate}_{\mathrm{o}}(P, a, l)}$$

- The exit probability of a process term is the probability with which the process term can execute actions of a given name and level that lead to a given set of terms.
- Generative probability for exponentially timed actions.
- Reactive probability for passive actions.
- Exit probability with which $P \in \mathcal{P}_{\mathrm{M}}$ executes actions of name $a \in \mathit{Name}$ and level $l \in \{0, -1\}$ that lead to $C \subseteq \mathcal{P}_{\mathrm{M}}$:

$$\boxed{\mathit{prob}_{\mathrm{e}}(P, a, l, C) \; = \; \begin{cases} \mathit{rate}_{\mathrm{e}}(P, a, l, C) / \mathit{rate}_{\mathrm{t}}(P, l) & \text{if } l = 0 \\ \mathit{rate}_{\mathrm{e}}(P, a, l, C) / \mathit{rate}_{\mathrm{o}}(P, a, l) & \text{if } l = -1 \end{cases}}$$

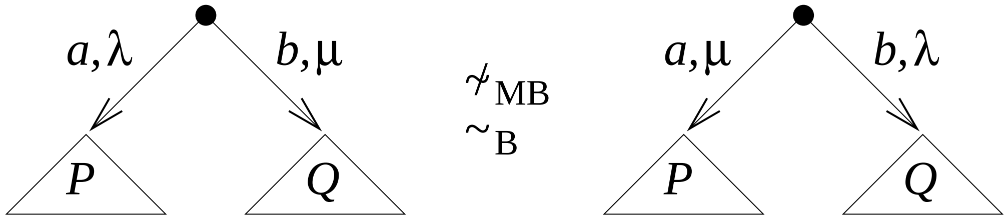
- An equivalence relation $\mathcal{B} \subseteq \mathcal{P}_{\mathrm{M}} \times \mathcal{P}_{\mathrm{M}}$ is a Markovian bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then for all action names $a \in \mathit{Name}$, levels $l \in \{0, -1\}$, and equivalence classes $C \in \mathcal{P}_{\mathrm{M}}/\mathcal{B}$:

$$\boxed{\mathit{rate}_{\mathrm{e}}(P_1, a, l, C) \; = \; \mathit{rate}_{\mathrm{e}}(P_2, a, l, C)}$$

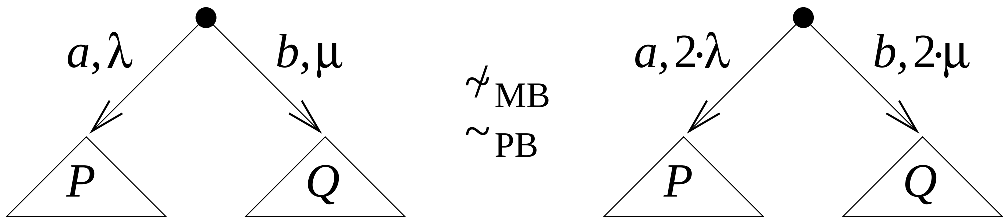
- Markovian bisimilarity, denoted by $\sim_{\mathrm{MB}}$, is the union of all the Markovian bisimulations.

- *A consequence of the coinductive nature of Markovian bisimilarity is that the derivatives of two equivalent terms are still equivalent.*

- $\sim_{\mathrm{MB}}$ is a strict refinement of classical bisimilarity ($\lambda \neq \mu$ and $P \not\sim_{\mathrm{MB}} Q$):




- $\sim_{\mathrm{MB}}$ is a strict refinement of probabilistic bisimilarity:



# Conditions and Characterizations

- In order for $P_1 \sim_{\rm MB} P_2$, it is necessary that for all $a \in \mathit{Name}$ and $l \in \{0, -1\}$:

$$\boxed{\mathit{rate}_{\rm o}(P_1, a, l) \; = \; \mathit{rate}_{\rm o}(P_2, a, l)}$$

- A relation $\mathcal{B} \subseteq \mathcal{P}_{\rm M} \times \mathcal{P}_{\rm M}$ is a Markovian bisimulation up to $\sim_{\rm MB}$ iff, whenever $(P_1, P_2) \in \mathcal{B}$, then for all action names $a \in \mathit{Name}$, levels $l \in \{0, -1\}$, and equivalence classes $C \in \mathcal{P}_{\rm M} / (\mathcal{B} \cup \mathcal{B}^{-1} \cup \sim_{\rm MB})^{+}$:

$$\boxed{\mathit{rate}_{\rm e}(P_1, a, l, C) \; = \; \mathit{rate}_{\rm e}(P_2, a, l, C)}$$

- In order for $P_1 \sim_{\rm MB} P_2$, it is sufficient to find a Markovian bisimulation up to $\sim_{\rm MB}$, say $\mathcal{B}$, such that $(P_1, P_2) \in \mathcal{B}$.

- $\sim_{\rm MB}$ has an alternative characterization in which time and probability are kept separate (instead of being both subsumed by rates).

- An equivalence relation $\mathcal{B} \subseteq \mathcal{P}_{\rm M} \times \mathcal{P}_{\rm M}$ is a separate Markovian bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then for all action names $a \in Name$ and levels $l \in \{0, -1\}$:

$$\boxed{rate_{\rm o}(P_1, a, l) \; = \; rate_{\rm o}(P_2, a, l)}$$

and for all equivalence classes $C \in \mathcal{P}_{\rm M}/\mathcal{B}$:

$$\boxed{prob_{\rm e}(P_1, a, l, C) \; = \; prob_{\rm e}(P_2, a, l, C)}$$

- Separate Markovian bisimilarity, denoted by $\sim_{\rm MB,s}$, is the union of all the separate Markovian bisimulations.

- For all $P_1, P_2 \in \mathcal{P}_{\rm M}$:

$$\boxed{P_1 \sim_{\rm MB,s} P_2 \iff P_1 \sim_{\rm MB} P_2}$$

# Equivalence Properties

- The CTMC-level aggregation induced by $\sim_{\mathrm{MB}}$ is an ordinary lumping.
- A partition $\mathcal{O}$ of the state space of a CTMC is an ordinary lumping iff, whenever $s_1, s_2 \in O$ for some $O \in \mathcal{O}$, then for all $O' \in \mathcal{O}$:

$$\boxed{\begin{aligned} &\sum\{\!|\, \lambda \in \mathbb{R}_{>0} \mid \exists s' \in O'.\, s_1 \xrightarrow{\lambda} s' \,|\!\} \;= \\ &\qquad\qquad\qquad \sum\{\!|\, \lambda \in \mathbb{R}_{>0} \mid \exists s' \in O'.\, s_2 \xrightarrow{\lambda} s' \,|\!\} \end{aligned}}$$

- Ordinary lumping is an exact CTMC-level aggregation.
- The probability of being in a macrostate of an ordinarily lumped CTMC is the sum of the probabilities of being in one of its constituent microstates of the original CTMC.
- Two Markovian bisimilar terms in $\mathcal{P}_{\mathrm{M,pc}}$ are guaranteed to possess the same performance characteristics.

- $\sim_{\mathrm{MB}}$ is a congruence with respect to all the operators of MPC.
- Let $P_1, P_2 \in \mathcal{P}_{\mathrm{M}}$. Whenever $P_1 \sim_{\mathrm{MB}} P_2$, then:

$$\boxed{\begin{array}{c}
<a,\tilde{\lambda}>.P_1 \;\sim_{\mathrm{MB}}\; <a,\tilde{\lambda}>.P_2 \\
P_1 + P \;\sim_{\mathrm{MB}}\; P_2 + P \qquad\qquad P + P_1 \;\sim_{\mathrm{MB}}\; P + P_2 \\
P_1 \,\|_S\, P \;\sim_{\mathrm{MB}}\; P_2 \,\|_S\, P \qquad\qquad P \,\|_S\, P_1 \;\sim_{\mathrm{MB}}\; P \,\|_S\, P_2 \\
P_1/H \;\sim_{\mathrm{MB}}\; P_2/H \qquad\qquad P_1 \backslash L \;\sim_{\mathrm{MB}}\; P_2 \backslash L \qquad\qquad P_1[\varphi] \;\sim_{\mathrm{MB}}\; P_2[\varphi]
\end{array}}$$

- Recursion: extend $\sim_{\mathrm{MB}}$ to open process terms by replacing all variables freely occurring outside *rec* binders with every closed process term.
- Let $P_1, P_2 \in \mathcal{L}_{\mathrm{M}}$ contain a free process variable $X$. We define $P_1 \sim_{\mathrm{MB}} P_2$ iff $P_1\{Q/X\} \sim_{\mathrm{MB}} P_2\{Q/X\}$ for all $Q \in \mathcal{P}_{\mathrm{M}}$.
- Let $P_1, P_2 \in \mathcal{L}_{\mathrm{M}}$ contain a free process variable $X$. Whenever $P_1 \sim_{\mathrm{MB}} P_2$, then:

$$\boxed{rec\,X : P_1 \;\sim_{\mathrm{MB}}\; rec\,X : P_2}$$

- $\sim_{\mathrm{MB}}$ has a sound and complete axiomatization.
- Basic laws (commutativity, associativity, and neutral element of $+$):

$$\boxed{\begin{array}{lrcl}
(\mathcal{A}_1^{\mathrm{MB}}) & P_1 + P_2 & = & P_2 + P_1 \\
(\mathcal{A}_2^{\mathrm{MB}}) & (P_1 + P_2) + P_3 & = & P_1 + (P_2 + P_3) \\
(\mathcal{A}_3^{\mathrm{MB}}) & P + \underline{0} & = & P
\end{array}}$$

- Characterizing laws (race policy and preselection policy, instead of $+$ idempotency):

$$\boxed{\begin{array}{lrcl}
(\mathcal{A}_4^{\mathrm{MB}}) & <a, \lambda_1>.P + <a, \lambda_2>.P & = & <a, \lambda_1 + \lambda_2>.P \\
(\mathcal{A}_5^{\mathrm{MB}}) & <a, *_{w_1}>.P + <a, *_{w_2}>.P & = & <a, *_{w_1 + w_2}>.P
\end{array}}$$

- Expansion law (interleaving view of concurrency supported by memoryless property):

$$(\mathcal{A}_6^{\mathrm{MB}}) \quad \sum_{i \in I} <a_i, \tilde{\lambda}_i>.P_{1,i} \parallel_S \sum_{j \in J} <b_j, \tilde{\mu}_j>.P_{2,j} \; = $$

$$\sum_{k \in I, a_k \notin S} <a_k, \tilde{\lambda}_k>. \left( P_{1,k} \parallel_S \sum_{j \in J} <b_j, \tilde{\mu}_j>.P_{2,j} \right) \; +$$

$$\sum_{h \in J, b_h \notin S} <b_h, \tilde{\mu}_h>. \left( \sum_{i \in I} <a_i, \tilde{\lambda}_i>.P_{1,i} \parallel_S P_{2,h} \right) \; +$$

$$\sum_{k \in I, a_k \in S, \tilde{\lambda}_k \in \mathbb{R}_{>0}} \; \sum_{h \in J, b_h = a_k, \tilde{\mu}_h = *_{w_h}} <a_k, \tilde{\lambda}_k \cdot \frac{w_h}{weight(P_2, b_h)}>.(P_{1,k} \parallel_S P_{2,h}) \; +$$

$$\sum_{h \in J, b_h \in S, \tilde{\mu}_h \in \mathbb{R}_{>0}} \; \sum_{k \in I, a_k = b_h, \tilde{\lambda}_k = *_{v_k}} <b_h, \tilde{\mu}_h \cdot \frac{v_k}{weight(P_1, a_k)}>.(P_{1,k} \parallel_S P_{2,h}) \; +$$

$$\sum_{k \in I, a_k \in S, \tilde{\lambda}_k = *_{v_k}} \; \sum_{h \in J, b_h = a_k, \tilde{\mu}_h = *_{w_h}}$$

$$<a_k, *_{\frac{v_k}{weight(P_1, a_k)} \cdot \frac{w_h}{weight(P_2, b_h)} \cdot (weight(P_1, a_k) + weight(P_2, b_h))}>.(P_{1,k} \parallel_S P_{2,h})$$

- Distribution laws (for unary static operators):

$$
\begin{array}{|lrcll|}
\hline
(\mathcal{A}_{7}^{\text{MB}}) & \underline{0}/H & = & \underline{0} & \\
(\mathcal{A}_{8}^{\text{MB}}) & (<a,\tilde{\lambda}>.P)/H & = & <\tau,\tilde{\lambda}>.(P/H) & \text{if } a \in H \\
(\mathcal{A}_{9}^{\text{MB}}) & (<a,\tilde{\lambda}>.P)/H & = & <a,\tilde{\lambda}>.(P/H) & \text{if } a \notin H \\
(\mathcal{A}_{10}^{\text{MB}}) & (P_1 + P_2)/H & = & P_1/H + P_2/H & \\
 & & & & \\
(\mathcal{A}_{11}^{\text{MB}}) & \underline{0}\backslash L & = & \underline{0} & \\
(\mathcal{A}_{12}^{\text{MB}}) & (<a,\tilde{\lambda}>.P)\backslash L & = & \underline{0} & \text{if } a \in L \\
(\mathcal{A}_{13}^{\text{MB}}) & (<a,\tilde{\lambda}>.P)\backslash L & = & <a,\tilde{\lambda}>.(P\backslash L) & \text{if } a \notin L \\
(\mathcal{A}_{14}^{\text{MB}}) & (P_1 + P_2)\backslash L & = & P_1\backslash L + P_2\backslash L & \\
 & & & & \\
(\mathcal{A}_{15}^{\text{MB}}) & \underline{0}[\varphi] & = & \underline{0} & \\
(\mathcal{A}_{16}^{\text{MB}}) & (<a,\tilde{\lambda}>.P)[\varphi] & = & <\varphi(a),\tilde{\lambda}>.(P[\varphi]) & \\
(\mathcal{A}_{17}^{\text{MB}}) & (P_1 + P_2)[\varphi] & = & P_1[\varphi] + P_2[\varphi] & \\
\hline
\end{array}
$$

- Recursion laws:

$$\boxed{\begin{array}{llcll}
(\mathcal{A}_{18}^{\mathrm{MB}}) & \mathit{rec}\, X : P & = & P\{(\mathit{rec}\, X : P)/X\} & \\
(\mathcal{A}_{19}^{\mathrm{MB}}) & Q = P\{Q/X\} & \Rightarrow & Q = \mathit{rec}\, X : P & \\
(\mathcal{A}_{20}^{\mathrm{MB}}) & \mathit{rec}\, X : P & = & \mathit{rec}\, Y : P\{Y/X\} & \text{if } Y \text{ not free in } P
\end{array}}$$

- The deduction system $\mathit{DED}(\mathcal{A}^{\mathrm{MB}})$ is sound and complete for $\sim_{\mathrm{MB}}$, i.e. for all $P_1, P_2 \in \mathcal{P}_{\mathrm{M}}$:

$$\boxed{P_1 \sim_{\mathrm{MB}} P_2 \iff \mathcal{A}^{\mathrm{MB}} \vdash P_1 = P_2}$$

- $\sim_{\rm MB}$ has a modal logic characterization.

- Variant of HML in which the diamond operator is decorated with a lower bound on the rate (resp. weight) with which exponentially timed (resp. passive) actions with a given name should be executed.

- Decorating individual action-based modal operators is consistent with the fact that bisimulation captures step-by-step behavior mimicking.

- Syntax of $\mathcal{ML}_{\rm MB}$ ($a \in Name$, $\lambda, w \in \mathbb{R}_{>0}$):

$$\boxed{\begin{array}{lcl} \phi & ::= & \mathrm{true} \\ & \mid & \neg \phi \\ & \mid & \phi \wedge \phi \\ & \mid & \langle a \rangle_{\lambda} \phi \\ & \mid & \langle a \rangle_{*_w} \phi \end{array}}$$

- Interpretation of $\mathcal{ML}_{\mathrm{MB}}$:

$$\boxed{\begin{array}{llll} P & \models_{\mathrm{MB}} & \mathrm{true} & \\ P & \models_{\mathrm{MB}} & \neg\phi & \text{if } P \not\models_{\mathrm{MB}} \phi \\ P & \models_{\mathrm{MB}} & \phi_1 \wedge \phi_2 & \text{if } P \models_{\mathrm{MB}} \phi_1 \text{ and } P \models_{\mathrm{MB}} \phi_2 \\ P & \models_{\mathrm{MB}} & \langle a \rangle_{\lambda} \phi & \text{if } \mathit{rate}_{\mathrm{e}}(P, a, 0, \mathit{sat}(\phi)) \geq \lambda \\ P & \models_{\mathrm{MB}} & \langle a \rangle_{*_w} \phi & \text{if } \mathit{rate}_{\mathrm{e}}(P, a, -1, \mathit{sat}(\phi)) \geq w \end{array}}$$

where:

$$\boxed{\mathit{sat}(\phi) \; = \; \{P' \in \mathcal{P}_{\mathrm{M}} \mid P' \models_{\mathrm{MB}} \phi\}}$$

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

$$\boxed{P_1 \sim_{\mathrm{MB}} P_2 \iff (\forall \phi \in \mathcal{ML}_{\mathrm{MB}}.\, P_1 \models_{\mathrm{MB}} \phi \iff P_2 \models_{\mathrm{MB}} \phi)}$$

- $\sim_{\rm MB}$ can be decided in polynomial time through an algorithm inspired by Paige-Tarjan partition refinement algorithm.

- $\sim_{\rm MB}$ characterized as a fixed point of successively finer relations:

$$\boxed{\sim_{\rm MB} \; = \; \bigcap_{i \in \mathbb{N}} \sim_{{\rm MB},i}}$$

- $\sim_{{\rm MB},0} \; = \; \mathcal{P}_{\rm M} \times \mathcal{P}_{\rm M}$ hence it induces the trivial partition $\{\mathcal{P}_{\rm M}\}$.

- Let $i \geq 1$. Whenever $(P_1, P_2) \in \sim_{{\rm MB},i}$, then for all $a \in \mathit{Name}$, $l \in \{0, -1\}$, and $C \in \mathcal{P}_{\rm M} / \sim_{{\rm MB},i-1}$:

$$\mathit{rate}_{\rm e}(P_1, a, l, C) \; = \; \mathit{rate}_{\rm e}(P_2, a, l, C)$$

- $\sim_{{\rm MB},1}$ refines the partition induced by $\sim_{{\rm MB},0}$ by creating an equivalence class for each set of terms that satisfy the necessary condition for $\sim_{\rm MB}$.

- Steps of the algorithm to check whether $P_1 \sim_{\mathrm{MB}} P_2$:
  1. Build a partition with a single class including all the states of the disjoint union of $[\![P_1]\!]_{\mathrm{M}}$ and $[\![P_2]\!]_{\mathrm{M}}$, then initialize a list of splitters with this class.
  2. Refine the current partition by splitting each of its classes according to the exit rates towards one of the splitters, then remove this splitter from the list.
  3. For each split class, insert into the list of splitters all the resulting subclasses except for the largest one.
  4. If the list of splitters is empty, return yes/no depending on whether the initial state of $[\![P_1]\!]_{\mathrm{M}}$ and the initial state of $[\![P_2]\!]_{\mathrm{M}}$ belong to the same class or not, otherwise go back to the refinement step.
- The time complexity is $O(m \cdot \log n)$ if a *splay tree* is used for representing the subclasses arising from the splitting of a class.

***Part IV:***
**Variants of Markovian Bisimilarity**

# Markovian Bisimilarity and Rewards

- Specific performance measures may distinguish between ordinarily lumpable states by ascribing them a different meaning.
- Make $\sim_{\mathrm{MB}}$ sensitive to performance measures.
- Specification of performance measures for CTMC-based models via reward structures:
  - ⊙ A *yield reward* $yr \in \mathbb{R}$ expresses the rate at which a gain/loss is accumulated while sojourning in the related state.
  - ⊙ A *bonus reward* $br \in \mathbb{R}$ expresses the instantaneous gain/loss implied by the execution of the related transition.

- Instant-of-time value of a reward-based performance measure:

$$\boxed{\sum_{s \in S} yr(s) \cdot \pi[s] + \sum_{s \xrightarrow{\lambda} s'} br(s, \lambda, s') \cdot \phi(s, \lambda, s')}$$

where:

  - $yr(s)$ is the yield reward associated with state $s$.
  - $\pi[s]$ is the probability of being in state $s$ at the considered instant of time.
  - $br(s, \lambda, s')$ is the bonus reward associated with transition $s \xrightarrow{\lambda} s'$.
  - $\phi(s, \lambda, s')$ is the frequency of transition $s \xrightarrow{\lambda} s'$ at the considered instant of time: $\phi(s, \lambda, s') = \pi[s] \cdot \lambda$.

- *Ascribing a different meaning to ordinarily lumpable states amounts to giving different rewards to such states or their outgoing transitions.*
- How to specify rewards at the process algebraic level?
- Bonus rewards can naturally be associated with actions.
- Yield rewards are problematic, as in process calculi the concept of state is implicit.
- Process calculi are action-based, hence associate yield rewards with actions too.
- *Additivity assumption*: the yield reward of a state corresponding to a process term is given by the sum of the yield rewards associated with the actions enabled by that term.

- $\mathrm{MPC_r}$: Markovian process calculus with rewards.
- New action syntax $(yr, br \in \mathbb{R})$:

$$\boxed{\begin{array}{c} <a, \lambda, yr, br> \\ <a, *_w, *, *> \end{array}}$$

- $\mathcal{P}_{\mathrm{M,r}}$: set of closed and guarded process terms of $\mathrm{MPC_r}$.
- New semantic rule for action prefix:

$$\boxed{<a, \tilde{\lambda}, \tilde{yr}, \tilde{br}>.P \xrightarrow{a, \tilde{\lambda}, \tilde{yr}, \tilde{br}}_{\mathrm{M,r}} P}$$

- The other semantic rules are modified accordingly.

- In particular, here are the semantic rules for synchronization:

$$\frac{P_1 \xrightarrow{a,\lambda,yr,br}_{\mathrm{M,r}} P_1' \qquad P_2 \xrightarrow{a,*_w,*,*}_{\mathrm{M,r}} P_2' \qquad a \in S}{P_1 \parallel_S P_2 \xrightarrow{a,\lambda\cdot\frac{w}{weight(P_2,a)},yr\cdot\frac{w}{weight(P_2,a)},br}_{\mathrm{M,r}} P_1' \parallel_S P_2'}$$

$$\frac{P_1 \xrightarrow{a,*_w,*,*}_{\mathrm{M,r}} P_1' \qquad P_2 \xrightarrow{a,\lambda,yr,br}_{\mathrm{M,r}} P_2' \qquad a \in S}{P_1 \parallel_S P_2 \xrightarrow{a,\lambda\cdot\frac{w}{weight(P_1,a)},yr\cdot\frac{w}{weight(P_1,a)},br}_{\mathrm{M,r}} P_1' \parallel_S P_2'}$$

$$\frac{P_1 \xrightarrow{a,*_{w_1},*,*}_{\mathrm{M,r}} P_1' \qquad P_2 \xrightarrow{a,*_{w_2},*,*}_{\mathrm{M,r}} P_2' \qquad a \in S}{P_1 \parallel_S P_2 \xrightarrow{a,*_{\frac{w_1}{weight(P_1,a)}\cdot\frac{w_2}{weight(P_2,a)}\cdot(weight(P_1,a)+weight(P_2,a))},*,*}_{\mathrm{M,r}} P_1' \parallel_S P_2'}$$

- Yield rewards normalized in the same way as rates.

- Exit reward with which $P \in \mathcal{P}_{\mathrm{M,r}}$ executes actions of name $a \in \mathit{Name}$ and level $l \in \{0, -1\}$ that lead to $C \subseteq \mathcal{P}_{\mathrm{M,r}}$:

$$\boxed{\begin{array}{rcl} \mathit{reward}_{\mathrm{e}}(P, a, 0, C) & = & \sum\{\!|\, \mathit{yr} + \lambda \cdot \mathit{br} \in \mathbb{R} \mid \exists P' \in C.\, P \xrightarrow{a, \lambda, \mathit{yr}, \mathit{br}}_{\mathrm{M,r}} P' \,|\!\} \\ \mathit{reward}_{\mathrm{e}}(P, a, -1, C) & = & 0 \end{array}}$$

- An equivalence relation $\mathcal{B} \subseteq \mathcal{P}_{\mathrm{M,r}} \times \mathcal{P}_{\mathrm{M,r}}$ is a reward Markovian bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then for all action names $a \in \mathit{Name}$, levels $l \in \{0, -1\}$, and equivalence classes $C \in \mathcal{P}_{\mathrm{M,r}}/\mathcal{B}$:

$$\boxed{\begin{array}{rcl} \mathit{rate}_{\mathrm{e}}(P_1, a, l, C) & = & \mathit{rate}_{\mathrm{e}}(P_2, a, l, C) \\ \mathit{reward}_{\mathrm{e}}(P_1, a, l, C) & = & \mathit{reward}_{\mathrm{e}}(P_2, a, l, C) \end{array}}$$

- Reward Markovian bisimilarity, denoted by $\sim_{\mathrm{MB,r}}$, is the union of all the reward Markovian bisimulations.

- $\sim_{\mathrm{MB,r}}$ enjoys the same properties as $\sim_{\mathrm{MB}}$.

- Axioms characterizing $\sim_{\mathrm{MB,r}}$:

$$\boxed{\begin{aligned} <a, \lambda_1, yr_1, br_1>.P + <a, \lambda_2, yr_2, br_2>.P \quad &= \\ &<a, \lambda_1 + \lambda_2, yr_1 + yr_2, \tfrac{\lambda_1}{\lambda_1+\lambda_2} \cdot br_1 + \tfrac{\lambda_2}{\lambda_1+\lambda_2} \cdot br_2>.P \\ <a, *_{w_1}, *, *>.P + <a, *_{w_2}, *, *>.P \quad &= \quad <a, *_{w_1+w_2}, *, *>.P \end{aligned}}$$

- Yield rewards summed up in the same way as rates (additivity assumption).

- Bonus rewards summed up by considering execution probabilities too.

- Equivalent characterizing axioms in yield-normal-form:

$$\begin{aligned}
<a, \lambda, yr, br>.P &= <a, \lambda, yr + \lambda \cdot br, 0>.P \\
<a, \lambda_1, yr_1, 0>.P + <a, \lambda_2, yr_2, 0>.P &= <a, \lambda_1 + \lambda_2, yr_1 + yr_2, 0>.P \\
<a, *_{w_1}, *, *>.P + <a, *_{w_2}, *, *>.P &= <a, *_{w_1+w_2}, *, *>.P
\end{aligned}$$

- Equivalent characterizing axioms in bonus-normal-form:

$$\begin{aligned}
<a, \lambda, yr, br>.P &= <a, \lambda, 0, br + \tfrac{yr}{\lambda}>.P \\
<a, \lambda_1, 0, br_1>.P + <a, \lambda_2, 0, br_2>.P &= \\
& \quad <a, \lambda_1 + \lambda_2, 0, \tfrac{\lambda_1}{\lambda_1+\lambda_2} \cdot br_1 + \tfrac{\lambda_2}{\lambda_1+\lambda_2} \cdot br_2>.P \\
<a, *_{w_1}, *, *>.P + <a, *_{w_2}, *, *>.P &= <a, *_{w_1+w_2}, *, *>.P
\end{aligned}$$

# Markovian Bisimilarity and Nondeterminism

- Nondeterminism is a useful abstraction whenever not all the details of a model are known in the early design stages.
- Combine nondeterministic process calculi and CTMCs.
- Separate exponential delays from interacting actions (orthogonal time).
- Markovian branchings and nondeterministic branchings.
- Inter-process communication implemented through the synchronization of visible interacting actions.
- Interacting actions take no time.
- *Maximal progress: $\tau$-actions take precedence over time passing.*

- $\mathrm{MPC_i}$: interactive Markovian process calculus.
- New syntax for prefixing:

$$\boxed{\begin{array}{c} a.P \\ (\lambda).P \end{array}}$$

- $\mathcal{P}_{\mathrm{M,i}}$: set of closed and guarded process terms of $\mathrm{MPC_i}$.
- Two transition relations are necessary: actions and delays.
- New semantic rules for prefixing:

$$\boxed{\begin{array}{c} a.P \xrightarrow{a}_{\mathrm{M,a}} P \\ (\lambda).P \xrightarrow{\lambda}_{\mathrm{M,d}} P \end{array}}$$

- The other semantic rules are modified accordingly.

- In particular, here are the semantic rules for parallel composition:

$$\frac{P_1 \xrightarrow{a}_{\mathrm{M,a}} P'_1 \qquad a \notin S}{P_1 \parallel_S P_2 \xrightarrow{a}_{\mathrm{M,a}} P'_1 \parallel_S P_2} \qquad \frac{P_2 \xrightarrow{a}_{\mathrm{M,a}} P'_2 \qquad a \notin S}{P_1 \parallel_S P_2 \xrightarrow{a}_{\mathrm{M,a}} P_1 \parallel_S P'_2}$$

$$\frac{P_1 \xrightarrow{a}_{\mathrm{M,a}} P'_1 \qquad P_2 \xrightarrow{a}_{\mathrm{M,a}} P'_2 \qquad a \in S}{P_1 \parallel_S P_2 \xrightarrow{a}_{\mathrm{M,a}} P'_1 \parallel_S P'_2}$$

$$\frac{P_1 \xrightarrow{\lambda}_{\mathrm{M,d}} P'_1}{P_1 \parallel_S P_2 \xrightarrow{\lambda}_{\mathrm{M,d}} P'_1 \parallel_S P_2} \qquad \frac{P_2 \xrightarrow{\lambda}_{\mathrm{M,d}} P'_2}{P_1 \parallel_S P_2 \xrightarrow{\lambda}_{\mathrm{M,d}} P_1 \parallel_S P'_2}$$

- CMTC derivation if there are no states with several outgoing action transitions or a non-$\tau$-action transition alternative to delay transitions, by superposing source and destination state of each action transition (absence of nondeterminism).

- Exit rate of $P \in \mathcal{P}_{\mathrm{M,i}}$ towards $C \subseteq \mathcal{P}_{\mathrm{M,i}}$:

$$\boxed{rate_{\mathrm{e,d}}(P, C) \; = \; \sum \{\!| \, \lambda \in \mathbb{R}_{>0} \mid \exists P' \in C. \, P \xrightarrow{\lambda}_{\mathrm{M,d}} P' \, |\!\}}$$

- An equivalence relation $\mathcal{B} \subseteq \mathcal{P}_{\mathrm{M,i}} \times \mathcal{P}_{\mathrm{M,i}}$ is an interactive Markovian bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
  - ⊙ For all action names $a \in \mathit{Name}$, $P_1 \xrightarrow{a}_{\mathrm{M,a}} P'_1$ implies $P_2 \xrightarrow{a}_{\mathrm{M,a}} P'_2$ for some $P'_2$ with $(P'_1, P'_2) \in \mathcal{B}$.
  - ⊙ For all equivalence classes $C \in \mathcal{P}_{\mathrm{M,i}}/\mathcal{B}$, $P_1 \stackrel{\tau}{\not\longrightarrow}_{\mathrm{M,a}}$ implies $P_2 \stackrel{\tau}{\not\longrightarrow}_{\mathrm{M,a}}$ with:

$$\boxed{rate_{\mathrm{e,d}}(P_1, C) \; = \; rate_{\mathrm{e,d}}(P_2, C)}$$

- Interactive Markovian bisimilarity, denoted by $\sim_{\mathrm{MB,i}}$, is the union of all the interactive Markovian bisimulations.

- $\sim_{\mathrm{MB,i}}$ enjoys properties similar to those of $\sim_{\mathrm{MB}}$.

- Axioms characterizing $\sim_{\mathrm{MB,i}}$:

$$\boxed{\begin{array}{rcl} a.P + a.P & = & a.P \\ (\lambda_1).P + (\lambda_2).P & = & (\lambda_1 + \lambda_2).P \\ \tau.P + (\lambda).Q & = & \tau.P \end{array}}$$

- Idempotency of + like in nondeterministic process calculi.

- Race policy like in CTMCs.

- Maximal progress too.

- $\tau$-actions should be ignored when playing the bisimulation game.
- They are invisible and take no time.
- Weak variant of $\sim_{\mathrm{MB,i}}$.
- After any non-pre-emptable exponential delay, skip all the states that can evolve via a finite sequence of $\tau$-transitions to a given class.
- Internal backward closure of $C \subseteq \mathcal{P}_{\mathrm{M,i}}$:

$$\boxed{C_\tau \;=\; \{P' \in \mathcal{P}_{\mathrm{M,i}} \mid \exists P \in C.\, P' \xRightarrow{\tau^*}_{\mathrm{M,a}} P\}}$$

- An equivalence relation $\mathcal{B} \subseteq \mathcal{P}_{\mathrm{M,i}} \times \mathcal{P}_{\mathrm{M,i}}$ is a weak interactive Markovian bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then:
  - For all visible action names $a \in \mathit{Name}_{\mathrm{v}}$, $P_1 \xrightarrow{a}_{\mathrm{M,a}} P_1'$ implies $P_2 \xRightarrow{\tau^* a \tau^*}_{\mathrm{M,a}} P_2'$ for some $P_2'$ with $(P_1', P_2') \in \mathcal{B}$.
  - $P_1 \xrightarrow{\tau}_{\mathrm{M,a}} P_1'$ implies $P_2 \xRightarrow{\tau^*}_{\mathrm{M,a}} P_2'$ for some $P_2'$ with $(P_1', P_2') \in \mathcal{B}$.
  - For all equivalence classes $C \in \mathcal{P}_{\mathrm{M,i}}/\mathcal{B}$, $P_1 \xRightarrow{\tau^*}_{\mathrm{M,a}} P_1' \stackrel{\tau}{\not\longrightarrow}_{\mathrm{M,a}}$ implies $P_2 \xRightarrow{\tau^*}_{\mathrm{M,a}} P_2' \stackrel{\tau}{\not\longrightarrow}_{\mathrm{M,a}}$ for some $P_2'$ with:

$$\boxed{\mathit{rate}_{\mathrm{e,d}}(P_1', C_\tau) \;=\; \mathit{rate}_{\mathrm{e,d}}(P_2', C_\tau)}$$

- Weak interactive Markovian bisimilarity, denoted by $\approx_{\mathrm{MB,i}}$, is the union of all the weak interactive Markovian bisimulations.

- The first part of the third clause distinguishes $\tau$-divergent processes from non-$\tau$-divergent processes that cannot execute any visible action (congruence w.r.t. parallel composition would be lost in a timed setting with maximal progress).

- $\approx_{\mathrm{MB,i}}$ is strictly coarser than $\sim_{\mathrm{MB,i}}$ but it is not a congruence with respect to alternative composition.
- Initial $\tau$-actions need a different treatment (as for classical weak bisimilarity).
- $P_1 \in \mathcal{P}_{\mathrm{M,i}}$ is weakly interactive Markovian bisimulation congruent to $P_2 \in \mathcal{P}_{\mathrm{M,i}}$, written $P_1 \simeq_{\mathrm{MB,i}} P_2$, iff:
  - ⊙ For all action names $a \in \mathit{Name}$, $P_1 \xrightarrow{a}_{\mathrm{M,a}} P_1'$ implies $P_2 \xRightarrow{\tau^* a \tau^*}_{\mathrm{M,a}} P_2'$ for some $P_2'$ with $P_1' \approx_{\mathrm{MB,i}} P_2'$.
  - ⊙ For all action names $a \in \mathit{Name}$, $P_2 \xrightarrow{a}_{\mathrm{M,a}} P_2'$ implies $P_1 \xRightarrow{\tau^* a \tau^*}_{\mathrm{M,a}} P_1'$ for some $P_1'$ with $P_1' \approx_{\mathrm{MB,i}} P_2'$.
  - ⊙ $P_1 \xrightarrow{\tau}\!\!\!\!\!\!\!/\;\;_{\mathrm{M,a}}$ iff $P_2 \xrightarrow{\tau}\!\!\!\!\!\!\!/\;\;_{\mathrm{M,a}}$ .
  - ⊙ For all equivalence classes $C \in \mathcal{P}_{\mathrm{M,i}} / \approx_{\mathrm{MB,i}}$, $P_1 \xrightarrow{\tau}\!\!\!\!\!\!\!/\;\;_{\mathrm{M,a}}$ implies:

$$\boxed{\mathit{rate}_{\mathrm{e,d}}(P_1, C) \; = \; \mathit{rate}_{\mathrm{e,d}}(P_2, C)}$$

- $\sim_{\rm MB,i} \subset \simeq_{\rm MB,i} \subset \approx_{\rm MB,i}$ with $\simeq_{\rm MB,i}$ having the same properties as $\sim_{\rm MB,i}$.
- Additional axioms characterizing $\simeq_{\rm MB,i}$:

$$\boxed{\begin{array}{rcl} a.\tau.P & = & a.P \\ P + \tau.P & = & \tau.P \\ a.(P + \tau.Q) + \tau.Q & = & a.(P + \tau.Q) \\ (\lambda).\tau.P & = & (\lambda).P \end{array}}$$

- $\tau$-laws witnessing the capability of $\simeq_{\rm MB,i}$ of abstracting from $\tau$-actions that are non-initial.

## Markovian Bisimilarity and Immediate Actions

- Combinations of exponential distributions approximate many general distributions arbitrarily closely, still some useful durations cannot be represented in the integrated time case, specially zero durations.

- Performance abstraction mechanism for integrated time, useful for handling systems with activities that are several orders of magnitude faster than those important for certain performance measures.

- Necessary to manage situations with which no timing can be associated like choices among logical events (e.g., the reception of a message vs. its loss).

- Zero durations implemented through immediate actions à la GSPN.

- Markovian branchings and prioritized/probabilistic branchings.

- *Pre-emption: immediate $\tau$-actions take precedence over all the lower priority actions.*

- $\mathrm{MPC_x}$: Markovian process calculus extended with immediate actions.
- New action syntax ($l \in \mathbb{N}_{>0}$, $l' \in \mathbb{N}$):

$$\boxed{\begin{array}{c} <a, \lambda>.P \\ <a, \infty_{l,w}>.P \\ <a, *_{w}^{l'}>.P \end{array}}$$

- $\mathcal{P}_{\mathrm{M,x}}$: set of closed and guarded process terms of $\mathrm{MPC_x}$.
- *Preselection policy*: each of the highest priority immediate actions that are enabled is given an execution probability proportional to its weight.
- *Priority constraints* to control process priority interrelation:
  - ⊙ An exponentially timed action can synchronize only with a passive action with priority constraint $l' = 0$.
  - ⊙ An immediate action with priority level $l$ can synchronize only with a passive action with priority constraint $l' = l$.

- Additional semantic rules for immediate actions:

$$<a, \infty_{l,w}>.P \xrightarrow{a, \infty_{l,w}}_{\mathrm{M,x}} P$$

$$\frac{P_1 \xrightarrow{a, \infty_{l,w}}_{\mathrm{M,x}} P_1' \qquad P_2 \xrightarrow{a, *_v^l}_{\mathrm{M,x}} P_2' \qquad a \in S}{P_1 \parallel_S P_2 \xrightarrow{a, \infty_{l,w \cdot \frac{v}{weight(P_2, a, l)}}}_{\mathrm{M,x}} P_1' \parallel_S P_2'}$$

$$\frac{P_1 \xrightarrow{a, *_v^l}_{\mathrm{M,x}} P_1' \qquad P_2 \xrightarrow{a, \infty_{l,w}}_{\mathrm{M,x}} P_2' \qquad a \in S}{P_1 \parallel_S P_2 \xrightarrow{a, \infty_{l,w \cdot \frac{v}{weight(P_1, a, l)}}}_{\mathrm{M,x}} P_1' \parallel_S P_2'}$$

- The other semantic rules are modified by taking into account priority constraints associated with passive actions.

- $\mathcal{P}_{\mathrm{M,x,pc}}$: set of performance closed process terms of $\mathcal{P}_{\mathrm{M,x}}$.
- CTMC derivation for performance closed process terms by eliminating all *vanishing states* (those with outgoing immediate transitions, hence zero sojourn time):
  - ⊙ Make as many copies of every transition entering a vanishing state as there are highest priority immediate transitions departing from the vanishing state.
  - ⊙ Connect each copy to the destination state of one of the highest priority immediate transitions leaving the vanishing state.
  - ⊙ Assign as rate of each copy the rate of the original incoming transition multiplied by the execution probability of the highest priority immediate transition corresponding to the copy.

- Exit rate at which $P \in \mathcal{P}_{\mathrm{M,x}}$ executes actions of name $a \in \mathit{Name}$ and level $l \in \mathbb{Z}$ that lead to $C \subseteq \mathcal{P}_{\mathrm{M,x}}$:

$$\boxed{\mathit{rate}_{\mathrm{e}}(P, a, l, C) = \begin{cases} \sum \{\!| \lambda \in \mathbb{R}_{>0} \mid \exists P' \in C.\, P \xrightarrow{a,\lambda}_{M,x} P' |\!\} & \text{if } l = 0 \\ \sum \{\!| w \in \mathbb{R}_{>0} \mid \exists P' \in C.\, P \xrightarrow{a,\infty_{l,w}}_{M,x} P' |\!\} & \text{if } l > 0 \\ \sum \{\!| w \in \mathbb{R}_{>0} \mid \exists P' \in C.\, P \xrightarrow{a,*_w^{-l-1}}_{M,x} P' |\!\} & \text{if } l < 0 \end{cases}}$$

- $\mathit{pri}^{\tau}_{\infty}(P)$: priority level of the highest priority immediate $\tau$-action enabled by $P \in \mathcal{P}_{\mathrm{M,x}}$.

- $\mathit{pri}^{\tau}_{\infty}(P) = 0$ if $P$ does not enable any immediate $\tau$-action.

- $\mathit{no\text{-}pre}(l, P)$ if no action of level $l$ can be pre-empted in $P$:

$$\boxed{\mathit{no\text{-}pre}(l, P) \iff l \geq \mathit{pri}^{\tau}_{\infty}(P) \vee -l - 1 \geq \mathit{pri}^{\tau}_{\infty}(P)}$$

- The exit rate comparison should be conducted only when no pre-emption can be exercised.

- An equivalence relation $\mathcal{B} \subseteq \mathcal{P}_{\mathrm{M,x}} \times \mathcal{P}_{\mathrm{M,x}}$ is an extended Markovian bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then for all action names $a \in \mathit{Name}$, levels $l \in \mathbb{Z}$ such that $\mathit{no\text{-}pre}(l, P_1)$ and $\mathit{no\text{-}pre}(l, P_2)$, and equivalence classes $C \in \mathcal{P}_{\mathrm{M,x}}/\mathcal{B}$:

$$\boxed{\mathit{rate}_{\mathrm{e}}(P_1, a, l, C) \; = \; \mathit{rate}_{\mathrm{e}}(P_2, a, l, C)}$$

- Extended Markovian bisimilarity, denoted by $\sim_{\mathrm{MB,x}}$, is the union of all the extended Markovian bisimulations.

- $\sim_{\mathrm{MB,x}}$ enjoys the same properties as $\sim_{\mathrm{MB}}$.
- Axioms characterizing $\sim_{\mathrm{MB,x}}$:

$$
\boxed{
\begin{array}{rcll}
<a,\lambda_1>.P + <a,\lambda_2>.P & = & <a,\lambda_1+\lambda_2>.P & \\
<a,\infty_{l,w_1}>.P + <a,\infty_{l,w_2}>.P & = & <a,\infty_{l,w_1+w_2}>.P & \\
<a,*^{l}_{w_1}>.P + <a,*^{l}_{w_2}>.P & = & <a,*^{l}_{w_1+w_2}>.P & \\
& & & \\
<\tau,\infty_{l,w}>.P + <a,\lambda>.Q & = & <\tau,\infty_{l,w}>.P & \\
<\tau,\infty_{l,w}>.P + <a,\infty_{l',w'}>.Q & = & <\tau,\infty_{l,w}>.P & \text{if } l > l' \\
<\tau,\infty_{l,w}>.P + <a,*^{l'}_{w'}>.Q & = & <\tau,\infty_{l,w}>.P & \text{if } l > l'
\end{array}
}
$$

- The last three axioms encode pre-emption exercised by immediate $\tau$-actions over lower priority actions.

- Immediate $\tau$-actions should be ignored in the bisimulation game.
- They are invisible and take no time.
- Weak variant of $\sim_{\mathrm{MB,x}}$.
- After any non-pre-emptable action, skip all the states that can evolve via a finite sequence of immediate $\tau$-transitions to a given class.
- Harder than weakening $\sim_{\mathrm{MB,i}}$:
  - ⊙ Need to keep track of quantitative information associated with the actions to be abstracted away.
  - ⊙ Need to take into account the degree of observability of classes of terms to be reached.

- Process term $P \in \mathcal{P}_{\mathrm{M,x}}$ is $l$-unobservable, with $l \in \mathbb{N}_{>0}$, iff $pri_{\infty}^{\tau}(P) = l$ and $P$ does not enable any immediate non-$\tau$-action with priority level $l' \geq l$, nor any passive action with priority constraint $l' \geq l$.

- A path $h$ of length $n \in \mathbb{N}_{>0}$:
$$P_1 \xrightarrow{\tau,\infty_{l_1,w_1}}_{\mathrm{M,x}} P_2 \xrightarrow{\tau,\infty_{l_2,w_2}}_{\mathrm{M,x}} \dots \xrightarrow{\tau,\infty_{l_n,w_n}}_{\mathrm{M,x}} P_{n+1}$$
is unobservable iff for all $i = 1, \dots, n$ process term $P_i$ is $l_i$-unobservable.

- The probability of executing the unobservable path $h$ is given by:
$$\boxed{prob_{\mathrm{p}}(h) \; = \; \prod_{i=1}^{n} \frac{w_i}{rate_{\mathrm{o}}(P_i, \tau, l_i)}}$$

- Weak exit rate at which $P \in \mathcal{P}_{\mathrm{M,x}}$ executes actions of name $a \in \mathit{Name}$ and level $l \in \mathbb{Z}$ that lead to $C \subseteq \mathcal{P}_{\mathrm{M,x}}$:

$$\boxed{\mathit{rate}_{\mathrm{e,w}}(P, a, l, C) \;=\; \sum_{P' \in C_{\mathrm{w}}} \mathit{rate}_{\mathrm{e}}(P, a, l, \{P'\}) \cdot \mathit{prob}_{\mathrm{w}}(P', C)}$$

  where:

  - $C_{\mathrm{w}}$ is the weak backward closure of $C$:

$$\boxed{C_{\mathrm{w}} \;=\; C \cup \{Q \in \mathcal{P}_{\mathrm{M,x}} - C \mid Q \text{ can reach } C \text{ via unobservable paths}\}}$$

  - $\mathit{prob}_{\mathrm{w}}(P', C)$ is the sum of the probabilities of all the unobservable paths from a term in $C_{\mathrm{w}}$ to $C$:

$$\boxed{\mathit{prob}_{\mathrm{w}}(P', C) = \begin{cases} 1 & \text{if } P' \in C \\ \sum \{\!| \, \mathit{prob}_{\mathrm{p}}(h) \mid h \text{ unobservable path from } P' \text{ to } C \, |\!\} & \\ & \text{if } P' \in C_{\mathrm{w}} - C \end{cases}}$$

- The weak exit rate comparison should be conducted only with respect to certain classes of terms.
- An observable term is a term that enables a non-$\tau$-action that cannot be pre-empted by any enabled immediate $\tau$-action.
- An initially unobservable term is a term in which all the enabled non-$\tau$-actions are pre-empted by some enabled immediate $\tau$-action, but at least one of the paths starting at this term with one of the higher priority enabled immediate $\tau$-actions reaches an observable term.
- A fully unobservable term is a term in which all the enabled non-$\tau$-actions are pre-empted by some enabled immediate $\tau$-action, and all the paths starting at this term with one of the higher priority enabled immediate $\tau$-actions are unobservable.
- $\mathcal{P}_{\mathrm{M,x,fu}}$: set of fully unobservable process terms of $\mathcal{P}_{\mathrm{M,x}}$.

- The weak exit rate comparison with respect to observable and fully unobservable classes must obviously be performed.
- The comparison should be made with respect to all fully unobservable classes together, in order to maximize the abstraction power despite the quantitative information attached to immediate $\tau$-actions.
- The comparison with respect to initially unobservable classes should be skipped, otherwise terms like:

$$<a, \lambda>.<\tau, \infty_{l_1, w_1}>.<b, \mu>.\underline{0}$$

$$<a, \lambda>.<\tau, \infty_{l_2, w_2}>.<b, \mu>.\underline{0}$$

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

could not be considered equivalent to each other.

- An equivalence relation $\mathcal{B} \subseteq \mathcal{P}_{\mathrm{M,x}} \times \mathcal{P}_{\mathrm{M,x}}$ is a weak extended Markovian bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then for all action names $a \in \mathit{Name}$ and levels $l \in \mathbb{Z}$ such that $\mathit{no\text{-}pre}(l, P_1)$ and $\mathit{no\text{-}pre}(l, P_2)$:

$$\boxed{\begin{array}{rcll} \mathit{rate}_{\mathrm{e,w}}(P_1, a, l, C) & = & \mathit{rate}_{\mathrm{e,w}}(P_2, a, l, C) & \forall\, C \in \mathcal{P}_{\mathrm{M,x}}/\mathcal{B} \text{ obs.} \\ \mathit{rate}_{\mathrm{e,w}}(P_1, a, l, \mathcal{P}_{\mathrm{M,x,fu}}) & = & \mathit{rate}_{\mathrm{e,w}}(P_2, a, l, \mathcal{P}_{\mathrm{M,x,fu}}) & \end{array}}$$

- Weak extended Markovian bisimilarity, denoted by $\approx_{\mathrm{MB,x}}$, is the union of all the weak extended Markovian bisimulations.

- $\approx_{\mathrm{MB,x}}$ enjoys the same properties as $\sim_{\mathrm{MB,x}}$ except for congruence with respect to parallel composition.

- Need to restrict to a well-prioritized subset of $\mathcal{P}_{\mathrm{M,x,nd}}$, the set of non-divergent process terms of $\mathcal{P}_{\mathrm{M,x}}$.

- A fully unobservable process term like $<\tau, \infty_{l,w}>.\underline{0}$ allows concurrent exp. timed actions to be executed, while the equivalent divergent process term $\mathit{rec}\, X : <\tau, \infty_{l,w}>.X$ prevents time from passing.

- State observability and pre-emption schemes for two equivalent terms may change differently when composing each of them in parallel with some term, thus exposing parts of their behavior not compared before.

- A set of terms of $\mathcal{P}_{\mathrm{M,x}}$ is well-prioritized if, taken two arbitrary terms $P_1$ and $P_2$ in the set, any immediate/passive transition of each of $[\![P_1]\!]_{\mathrm{M,x}}$ and $[\![P_2]\!]_{\mathrm{M,x}}$ has priority level/constraint less than the priority level of any highest priority immediate $\tau$-transition departing from an unobservable state of the other one.

- Additional axioms characterizing $\approx_{\mathrm{MB,x}}$:

$$<a, \lambda>. \sum_{i \in I} <\tau, \infty_{l,w_i}>.P_i \quad = \quad \sum_{i \in I} <a, \lambda \cdot w_i / \Sigma_{k \in I}\, w_k>.P_i$$

$$<a, \infty_{l',w'}>. \sum_{i \in I} <\tau, \infty_{l,w_i}>.P_i \quad = \quad \sum_{i \in I} <a, \infty_{l', w' \cdot w_i / \Sigma_{k \in I}\, w_k}>.P_i$$

$$<a, *^{l'}_{w'}>. \sum_{i \in I} <\tau, \infty_{l,w_i}>.P_i \quad = \quad \sum_{i \in I} <a, *^{l'}_{w' \cdot w_i / \Sigma_{k \in I}\, w_k}>.P_i$$

- $\tau, \infty$-laws showing the ability of $\approx_{\mathrm{MB,x}}$ of abstracting from immediate $\tau$-actions and encoding the procedure for removing vanishing states.

- No abstraction from initial immediate $\tau$-actions, hence $\approx_{\mathrm{MB,x}}$ does not incur the congruence problem with respect to alternative composition found in $\approx_{\mathrm{MB,i}}$ (a consequence of the way the weak exit rate is defined).

## *Part V:*
## Markovian Testing Equivalence

## Equivalence Definition

- Two process terms are equivalent if an external observer cannot distinguish between them, with the only way for the observer to infer information about their functional and performance behavior being to interact with them by means of tests and look at their reactions.
- Was the test passed? If so, with which probability? And how long did it take to pass the test?
- Tests formalized as process terms.
- Interaction formalized as parallel composition of process term and test with synchronization enforced on any action name.
- Comparison of process term probabilities of performing a successful test-driven computation within a given amount of time.

- A computation of a process term is a *sequence of transitions* that can be executed starting from the state corresponding to the term.
- The *length* of a computation is the number of its transitions.
- Two distinct computations are *independent* of each other if neither is a proper prefix of the other one.
- $\mathcal{C}_{\mathrm{f}}(P)$: multiset of finite-length computations of $P \in \mathcal{P}_{\mathrm{M}}$.
- $\mathcal{I}_{\mathrm{f}}(P)$: multiset of finite-length independent computations of $P \in \mathcal{P}_{\mathrm{M}}$.

- Attributes of a finite-length computation:
  - *trace*;
  - *probability*;
  - *duration*.
- The trace associated with the execution of $c \in \mathcal{C}_{\mathrm{f}}(P)$ is the sequence of action names labeling the transitions of $c$:

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

- The probability of executing $c \in \mathcal{C}_{\mathrm{f}}(P)$ – with $P \in \mathcal{P}_{\mathrm{M,pc}}$ – is the product of the execution probabilities of the transitions of $c$:

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

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

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

- The above probability would not be well defined if set $C$ contained computations that are not indepedent of each other.

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

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

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

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

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

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

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

$$\boxed{prob_{\mathrm{d}}(C, \theta) \;=\; \sum_{c \in C}^{length(c) \leq length(\theta)} prob(c) \cdot \prod_{i=1}^{length(c)} \Pr(time_{\mathrm{d}}(c)[i] \leq \theta[i])}$$

- The exponential random variable $time_{\mathrm{d}}(c)[i]$ has cumulative distribution function $\Pr\{time_{\mathrm{d}}(c)[i] \leq \theta[i]\} = 1 - \mathrm{e}^{-\theta[i]/time_{\mathrm{a}}(c)[i]}$ and expected value $time_{\mathrm{a}}(c)[i]$.

- Why not summing up sojourn times? (standard duration instead of stepwise)
- Process terms with identical non-maximal computations ($\lambda \neq \mu$, $b \neq d$):

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

- Maximal computations of the first term:

$$c_{1,1} \equiv \; . \xrightarrow{g,\gamma} . \xrightarrow{a,\lambda} . \xrightarrow{b,\mu} .$$
$$c_{1,2} \equiv \; . \xrightarrow{g,\gamma} . \xrightarrow{a,\mu} . \xrightarrow{d,\lambda} .$$

- Maximal computations of the second term:

$$c_{2,1} \equiv \; . \xrightarrow{g,\gamma} . \xrightarrow{a,\lambda} . \xrightarrow{d,\mu} .$$
$$c_{2,2} \equiv \; . \xrightarrow{g,\gamma} . \xrightarrow{a,\mu} . \xrightarrow{b,\lambda} .$$

- Same average durations $\frac{1}{2 \cdot \gamma} + \frac{1}{\lambda} + \frac{1}{\mu}$ 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.*

- From now on, we assume that $\tau$ is removed from $Name$, hence we consider only visible actions.

- Syntax of the set $\mathcal{T}$ of tests ($I$ non-empty finite index set):

$$\boxed{\begin{array}{lcl} T & ::= & \mathrm{f} \\ & \mid & \mathrm{s} \\ & \mid & \sum\limits_{i \in I} {<} a_i, *_{w_i} {>} . T_i \end{array}}$$

- Asymmetric action synchronization: only passive actions.

- Passing tests within a finite amount of time: no recursion.

- No ambiguous tests like $\mathrm{s} + \mathrm{f}$: guarded alternative composition.

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

$$\boxed{P \,\|_{Name}\, T}$$

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

- The test:

  - either reacts by participating in the interaction with the process term through a passive action having the same name,

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

- Any interaction system is finite state, acyclic, and performance closed.

- Its computations have finite length.

- A *configuration* is a state of $[\![P \parallel_{Name} T]\!]_{\mathrm{M}}$.
- A configuration is formed by a *process part* and a *test part*.
- A configuration is *successful* (resp. *failed*) iff its test part is "s" (resp. "f").
- A computation is *successful* (resp. *failed*) iff so is the last configuration it reaches.
- A computation that is neither successful nor failed is *interrupted*.
- $\mathcal{SC}(P,T)$: multiset of successful computations of $\mathcal{C}_{\mathrm{f}}(P \parallel_{Name} T)$.
- $\mathcal{SC}(P,T)$ is finite because of the finitely-branching structure of the considered terms.
- $\mathcal{SC}(P,T) \subseteq \mathcal{I}_{\mathrm{f}}(P \parallel_{Name} T)$ because of the maximality of successful test-driven computations.

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

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

- $\sim_{\mathrm{MT}}$ is strictly finer than classical and probabilistic testing equivalences.
- $\sim_{\mathrm{MT}}$ is strictly coarser than $\sim_{\mathrm{MB}}$ as it is less sensitive to branching points.
- The derivatives of two Markovian testing equivalent terms are not necessarily related by $\sim_{\mathrm{MT}}$.

# Conditions and Characterizations

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

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

and for all $a \in \mathit{Name}$:

$$\boxed{\mathit{rate}_{\mathrm{o}}(P_{k,\mathrm{last}}, a, 0) \; = \; \mathit{rate}_{\mathrm{o}}(P_{h,\mathrm{last}}, a, 0)}$$

- Considering the (more accurate) stepwise durations of the test-driven computations leads to the same equivalence as considering the (easier to work with) stepwise average durations of the test-driven computations.

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

$$\boxed{\mathit{prob}_{\mathrm{d}}(\mathcal{SC}(P_1, T), \theta) \; = \; \mathit{prob}_{\mathrm{d}}(\mathcal{SC}(P_2, T), \theta)}$$

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

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

- $\sim_{\mathrm{MT}}$ has a *fully abstract alternative characterization* that avoids analyzing the process term behavior in response to tests.

- *Extended traces*: traces that are suitably extended with the sets of action names permitted at each step by the environment.

- An element $\sigma$ of $(Name \times 2^{Name})^*$ is an extended trace iff either $\sigma$ is the empty sequence or:

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

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

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

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

$$\boxed{\mathit{trace}(\sigma) \;=\; \begin{cases} \varepsilon & \text{if } \mathit{length}(\sigma) = 0 \\ a \circ \mathit{trace}(\sigma') & \text{if } \sigma \equiv (a, \mathcal{E}) \circ \sigma' \end{cases}}$$

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

$$\boxed{\mathit{trace}(c) \;=\; \mathit{trace}(\sigma)}$$

- $\mathcal{CC}(P, \sigma)$: multiset of computations of $\mathcal{C}_{\mathrm{f}}(P)$ compatible with $\sigma$.
- $\mathcal{CC}(P, \sigma)$ is finite because of the finitely-branching structure of the considered terms.
- $\mathcal{CC}(P, \sigma) \subseteq \mathcal{I}_{\mathrm{f}}(P)$ because of the compatibility of the computations with the same extended trace $\sigma$.

- Execution probability of $c \in \mathcal{CC}(P, \sigma)$ with respect to $\sigma$:

$$prob^{\sigma}(c) \;=\; \begin{cases} 1 & \text{if } \mathit{length}(c) = 0 \\[2ex] \dfrac{\lambda}{\sum\limits_{b \in \mathcal{E}} \mathit{rate}_{\mathrm{o}}(P, b, 0)} \cdot prob^{\sigma'}(c') & \text{if } c \equiv P \xrightarrow{a,\lambda}_{\mathrm{M}} c' \\ & \quad \text{with } \sigma \equiv (a, \mathcal{E}) \circ \sigma' \end{cases}$$

- Stepwise average duration of $c \in \mathcal{CC}(P, \sigma)$ with respect to $\sigma$:

$$time_{\mathrm{a}}^{\sigma}(c) \;=\; \begin{cases} \varepsilon & \text{if } \mathit{length}(c) = 0 \\[2ex] \dfrac{1}{\sum\limits_{b \in \mathcal{E}} \mathit{rate}_{\mathrm{o}}(P, b, 0)} \circ time_{\mathrm{a}}^{\sigma'}(c') & \text{if } c \equiv P \xrightarrow{a,\lambda}_{\mathrm{M}} c' \\ & \quad \text{with } \sigma \equiv (a, \mathcal{E}) \circ \sigma' \end{cases}$$

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

$$\boxed{prob^{\sigma}(\mathcal{CC}^{\sigma}_{\leq \theta}(P_1, \sigma)) \; = \; prob^{\sigma}(\mathcal{CC}^{\sigma}_{\leq \theta}(P_2, \sigma))}$$

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

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

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

- Single computation leading to success, whose states can have additional computations each leading to failure in one step.

- Syntax of the set $\mathcal{T}_{\rm c}$ of canonical tests $_{(a \,\in\, \mathcal{E})}$:

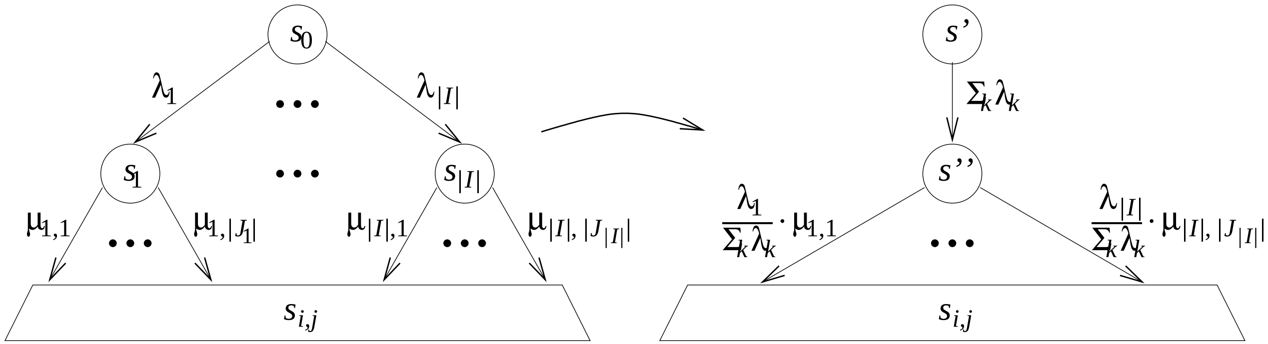
$$\boxed{\begin{array}{lll} T & ::= & \mathrm{s} \\ & | & <\!a, *_1\!>.T + \sum\limits_{b \in \mathcal{E} - \{a\}} <\!b, *_1\!>.\mathrm{f} \end{array}}$$

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

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

# Equivalence Properties

- $\sim_{\mathrm{MT}}$ induces a CTMC-level aggregation called T-lumping, which is strictly coarser than ordinary lumping:




where for all $i_1, i_2 \in I$:

$$\boxed{\sum_{j \in J_{i_1}} \mu_{i_1,j} = \sum_{j \in J_{i_2}} \mu_{i_2,j}}$$

- Exact aggregation not previously known in the CTMC field, but entirely characterizable in a process algebraic framework like ordinary lumping.

- Two Markovian testing equivalent terms in $\mathcal{P}_{\mathrm{M,pc}}$ are guaranteed to possess the same performance characteristics.

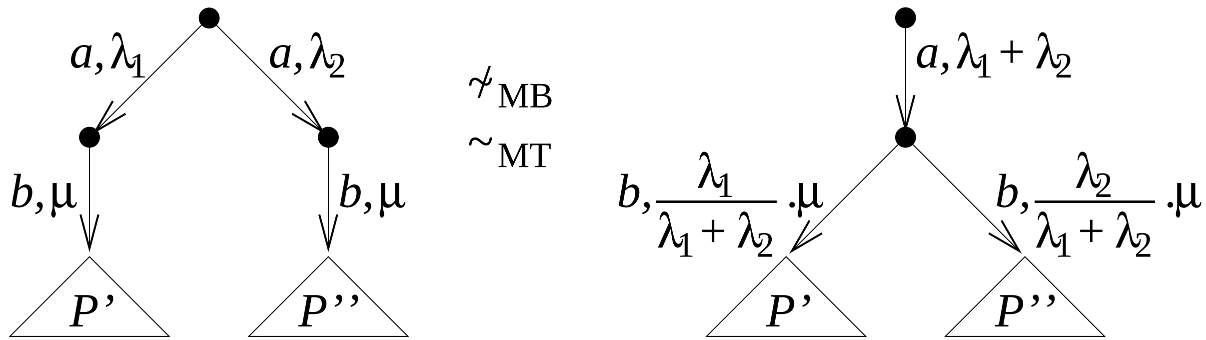
- $\sim_{\mathrm{MT}}$ is a congruence with respect to all the dynamic operators and parallel composition.

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

$$\boxed{\begin{array}{ccc} & {<}a, \lambda{>}.P_1 \;\sim_{\mathrm{MT}}\; {<}a, \lambda{>}.P_2 & \\ P_1 + P \;\sim_{\mathrm{MT}}\; P_2 + P & & P + P_1 \;\sim_{\mathrm{MT}}\; P + P_2 \\ P_1 \parallel_S P \;\sim_{\mathrm{MT}}\; P_2 \parallel_S P & & P \parallel_S P_1 \;\sim_{\mathrm{MT}}\; P \parallel_S P_2 \end{array}}$$

- Parallel composition: $P \in \mathcal{P}_{\mathrm{M}}$ containing only passive actions such that $P_1 \parallel_S P, P_2 \parallel_S P \in \mathcal{P}_{\mathrm{M,pc}}$.

- The axioms for $\sim_{\mathrm{MB}}$ are sound but not complete for $\sim_{\mathrm{MT}}$:




where $P' \not\sim_{\mathrm{MB}} P''$.

- *Choices among actions with the same name can be deferred whenever such actions are followed by actions having the same names and the same cumulative rates in all the branches.*

- Axiom schema characterizing $\sim_{\mathrm{MT}}$:

$$\boxed{\begin{aligned} &\sum_{i \in I} {<}a, \lambda_i{>}. \sum_{j \in J_i} {<}b_{i,j}, \mu_{i,j}{>}.P_{i,j} \quad = \\ &\qquad {<}a, \sum_{k \in I} \lambda_k{>}. \sum_{i \in I} \sum_{j \in J_i} {<}b_{i,j}, \frac{\lambda_i}{\Sigma_{k \in I}\, \lambda_k} \cdot \mu_{i,j}{>}.P_{i,j} \end{aligned}}$$

where for all $i_1, i_2 \in I$ and $b \in \mathit{Name}$:

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

- Subsumes ${<}a, \lambda_1{>}.P + {<}a, \lambda_2{>}.P \;=\; {<}a, \lambda_1 + \lambda_2{>}.P$.

- $\sim_{\rm MT}$ has a modal logic characterization over $\mathcal{P}_{\rm M,pc}$.

- Variant of HML in which negation is ruled out and conjunction is replaced by disjunction.

- No quantitative decorations in the syntax but ...

- ... quantitative interpretation establishing the probability with which a process term satisfies a formula quickly enough on average.

- Syntax of $\rm HML_{MT}$ $(a \in \mathit{Name})$:

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

where each formula of the form $\phi_1 \vee \phi_2$ obeys (independent initial action names):

$$\boxed{\mathit{init}(\phi_1) \cap \mathit{init}(\phi_2) \; = \; \emptyset}$$

- Interpretation of $\mathrm{HML_{MT}}$ over $\mathcal{P}_{\mathrm{M,pc}} \times (\mathbb{R}_{>0})^*$ is zero for $\phi \neq \mathrm{true}$ whenever $\mathit{init}(P) \cap \mathit{init}(\phi) = \emptyset$ or $\theta = \varepsilon$, otherwise:

$$\begin{aligned}
[\![\mathrm{true}]\!]_{\mathrm{MT}}(P, \theta) &= 1 \\
[\![\phi_1 \vee \phi_2]\!]_{\mathrm{MT}}(P, t \circ \theta) &= p_1 \cdot [\![\phi_1]\!]_{\mathrm{MT}}(P, t_1 \circ \theta) + p_2 \cdot [\![\phi_2]\!]_{\mathrm{MT}}(P, t_2 \circ \theta) \\
[\![\langle a \rangle \phi]\!]_{\mathrm{MT}}(P, t \circ \theta) &= \begin{cases} \displaystyle\sum_{P \xrightarrow{a,\lambda}_{\mathrm{M}} P'} \frac{\lambda}{\mathit{rate}_{\mathrm{o}}(P,a,0)} \cdot [\![\phi]\!]_{\mathrm{MT}}(P', \theta) & \text{if } \frac{1}{\mathit{rate}_{\mathrm{o}}(P,a,0)} \leq t \\ 0 & \text{if } \frac{1}{\mathit{rate}_{\mathrm{o}}(P,a,0)} > t \end{cases}
\end{aligned}$$

where for $j \in \{1, 2\}$:

$$\begin{aligned}
p_j &= \Sigma\{\!|\, \mathit{rate}_{\mathrm{o}}(P, b, 0) \mid b \in \mathit{init}(\phi_j) \,|\!\} \;/\; \Sigma\{\!|\, \mathit{rate}_{\mathrm{o}}(P, b, 0) \mid b \in \mathit{init}(\phi_1 \vee \phi_2) \,|\!\} \\
t_j &= t + \left(\frac{1}{\Sigma\{\!|\, \mathit{rate}_{\mathrm{o}}(P,b,0) \mid b \in \mathit{init}(\phi_j) \,|\!\}} - \frac{1}{\Sigma\{\!|\, \mathit{rate}_{\mathrm{o}}(P,b,0) \mid b \in \mathit{init}(\phi_1 \vee \phi_2) \,|\!\}}\right)
\end{aligned}$$

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

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

- $\sim_{\mathrm{MT}}$ can be decided in polynomial time through an algorithm inspired by Tzeng algorithm for probabilistic language equivalence.

- Two action-labeled CTMCs are Markovian testing equivalent iff their corresponding embedded action-labeled DTMCs are probabilistic testing equivalent, with the latter coinciding with probabilistic ready equivalence.

- The name occurring in the label of each transition of the embedded action-labeled DTMCs must be enriched with the total exit rate of the transition source state.

- Steps of the algorithm to check 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 equivalence $\mathcal{R}$ that relates any two states of the disjoint union of $[\![P_1]\!]_{\mathrm{M}}$ and $[\![P_2]\!]_{\mathrm{M}}$ such that their two sets of (original) action names labeling their outgoing transitions coincide.
  3. For each equivalence class $R$ induced by $\mathcal{R}$, apply Tzeng algorithm to check the embedded discrete-time versions of $[\![P_1]\!]_{\mathrm{M}}$ and $[\![P_2]\!]_{\mathrm{M}}$ for probabilistic language equivalence by considering $R$ as the set of accepting states.
- The time complexity is $O(n^5)$.

# *Part VI:*
# Markovian Trace Equivalence

## Equivalence Definition

- Two process terms are equivalent if they can perform computations with the same functional and performance characteristics.
- Comparison of process term probabilities of performing a computation within a given amount of time.
- Branching points are completely ignored.
- We keep considering only visible actions.

- $c \in \mathcal{C}_{\mathrm{f}}(P)$ is compatible with $\alpha \in \mathit{Name}^*$ iff:

$$\boxed{\mathit{trace}(c) \;=\; \alpha}$$

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

- $\mathcal{CC}(P, \alpha)$ is finite because of the finitely-branching structure of the considered terms.

- $\mathcal{CC}(P, \alpha) \subseteq \mathcal{I}_{\mathrm{f}}(P)$ because of the compatibility of the computations with the same trace $\alpha$.

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

$$\boxed{\mathit{prob}(\mathcal{CC}_{\leq\theta}(P_1, \alpha)) \;=\; \mathit{prob}(\mathcal{CC}_{\leq\theta}(P_2, \alpha))}$$

- $\sim_{\mathrm{MTr}}$ is strictly finer than classical and probabilistic trace equivalences.

- $\sim_{\mathrm{MTr}}$ is strictly coarser than $\sim_{\mathrm{MT}}$ as it is insensitive to branching points.

- The derivatives of two Markovian trace equivalent terms are not necessarily related by $\sim_{\mathrm{MTr}}$.

## Conditions and Characterizations

- In order for $P_1 \sim_{\mathrm{MTr}} P_2$, given $\alpha \in \mathit{Name}^*$ it is necessary that for all $c_k \in \mathcal{CC}(P_k, \alpha)$ with $k \in \{1, 2\}$ there exists $c_h \in \mathcal{CC}(P_h, \alpha)$ with $h \in \{1, 2\} - \{k\}$ such that:

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

and:

$$\boxed{\mathit{rate}_{\mathrm{t}}(P_{k,\mathrm{last}}, 0) = \mathit{rate}_{\mathrm{t}}(P_{h,\mathrm{last}}, 0)}$$

- Considering the (more accurate) stepwise durations of the computations leads to the same equivalence as considering the (easier to work with) stepwise average durations of the computations.

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

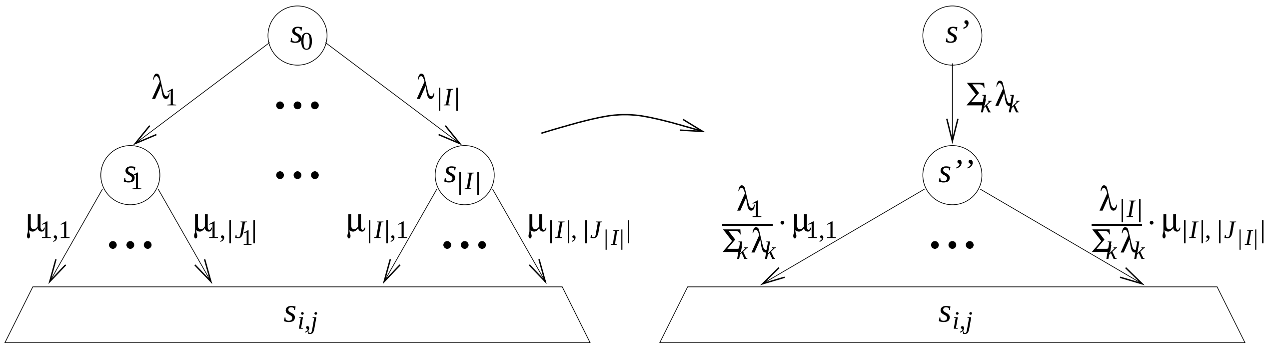
$$\boxed{\mathit{prob}_{\mathrm{d}}(\mathcal{CC}(P_1, \alpha), \theta) \; = \; \mathit{prob}_{\mathrm{d}}(\mathcal{CC}(P_2, \alpha), \theta)}$$

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

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

# Equivalence Properties

- $\sim_{\mathrm{MTr}}$ induces the same exact CTMC-level aggregation as $\sim_{\mathrm{MT}}$ (T-lumping):




  where for all $i_1, i_2 \in I$:

$$\boxed{\sum_{j \in J_{i_1}} \mu_{i_1,j} = \sum_{j \in J_{i_2}} \mu_{i_2,j}}$$

- Two Markovian trace equivalent terms in $\mathcal{P}_{\mathrm{M,pc}}$ are guaranteed to possess the same performance characteristics.

- $\sim_{\mathrm{MTr}}$ is a congruence with respect to all the dynamic operators.
- Let $P_1, P_2 \in \mathcal{P}_{\mathrm{M,pc}}$. Whenever $P_1 \sim_{\mathrm{MTr}} P_2$, then:

$$\boxed{\begin{array}{c} <a,\lambda>.P_1 \;\sim_{\mathrm{MTr}}\; <a,\lambda>.P_2 \\ P_1 + P \;\sim_{\mathrm{MTr}}\; P_2 + P \qquad\qquad P + P_1 \;\sim_{\mathrm{MTr}}\; P + P_2 \end{array}}$$

- *Not a congruence with respect to parallel composition:*
  - Markovian trace equivalent process terms $(b \neq c)$:

$$\begin{aligned} P_1 &\equiv <a,\lambda_1>.<b,\mu>.P' + <a,\lambda_2>.<c,\mu>.P'' \\ P_2 &\equiv <a,\lambda_1+\lambda_2>.(<b,\tfrac{\lambda_1}{\lambda_1+\lambda_2}\cdot\mu>.P' + <c,\tfrac{\lambda_2}{\lambda_1+\lambda_2}\cdot\mu>.P'') \end{aligned}$$

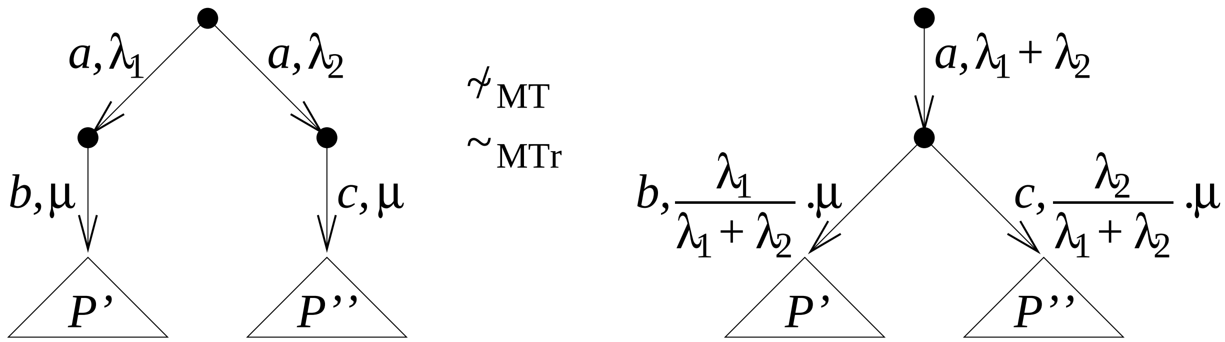
  - but:

$$\begin{array}{c} P_1 \;\|_{\{a,b,c\}}\; <a,*_1>.<b,*_1>.\underline{0} \\ P_2 \;\|_{\{a,b,c\}}\; <a,*_1>.<b,*_1>.\underline{0} \end{array}$$

  are distinguished by the following trace:

$$\alpha \;\equiv\; a \circ b$$

- The axioms for $\sim_{\mathrm{MT}}$ are sound but not complete for $\sim_{\mathrm{MTr}}$:




  where $b \neq c$.

- Action prefix tends to become left-distributive with respect to alternative composition.

- *Choices among actions with the same name can be deferred whenever such actions are followed by terms having the same total exit rate.*

- The names and the total rates of the initial actions of such derivative terms can be different in the various branches.

- Axiom schema characterizing $\sim_{\mathrm{MTr}}$:

$$\boxed{\begin{aligned}\sum_{i \in I} <a, \lambda_i>. \sum_{j \in J_i} <b_{i,j}, \mu_{i,j}>.P_{i,j} \quad = \\ <a, \sum_{k \in I} \lambda_k>. \sum_{i \in I} \sum_{j \in J_i} <b_{i,j}, \frac{\lambda_i}{\Sigma_{k \in I} \lambda_k} \cdot \mu_{i,j}>.P_{i,j}\end{aligned}}$$

where for all $i_1, i_2 \in I$:

$$\boxed{\sum_{j \in J_{i_1}} \mu_{i_1,j} = \sum_{j \in J_{i_2}} \mu_{i_2,j}}$$

- Subsumes the axiom schema characterizing $\sim_{\mathrm{MT}}$.

- $\sim_{\rm MTr}$ has a modal logic characterization over $\mathcal{P}_{\rm M,pc}$.
- Variant of HML in which both negation and conjunction are ruled out.
- No quantitative decorations; quantitative interpretation.
- Syntax of $\mathrm{HML}_{\rm MTr}$ $_{(a \, \in \, Name)}$:

$$\boxed{\begin{array}{lcl} \phi & ::= & \mathrm{true} \\ & | & \langle a \rangle \phi \end{array}}$$

- Interpretation of $\mathrm{HML}_{\mathrm{MTr}}$ over $\mathcal{P}_{\mathrm{M,pc}} \times (\mathbb{R}_{>0})^*$ is zero for $\phi \neq \mathrm{true}$ whenever $\theta = \varepsilon$, otherwise:

$$\boxed{\begin{array}{rcl} [\![\mathrm{true}]\!]_{\mathrm{MTr}}(P, \theta) & = & 1 \\ [\![\langle a \rangle \phi]\!]_{\mathrm{MTr}}(P, t \circ \theta) & = & \begin{cases} \sum\limits_{P \xrightarrow{a,\lambda}_{\mathrm{M}} P'} \frac{\lambda}{rate_{\mathrm{t}}(P,0)} \cdot [\![\phi]\!]_{\mathrm{MTr}}(P', \theta) & \text{if } \frac{1}{rate_{\mathrm{t}}(P,0)} \leq t \\ 0 & \text{if } \frac{1}{rate_{\mathrm{t}}(P,0)} > t \end{cases} \end{array}}$$

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

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

- $\sim_{\mathrm{MTr}}$ can be decided in polynomial time through an algorithm inspired by Tzeng algorithm for probabilistic language equivalence.
- Two action-labeled CTMCs are Markovian trace equivalent iff their corresponding embedded action-labeled DTMCs are probabilistic trace equivalent.
- The name occurring in the label of each transition of the embedded action-labeled DTMCs must be enriched with the total exit rate of the transition source state.

- Steps of the algorithm to check whether $P_1 \sim_{\mathrm{MTr}} P_2$:
  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. Apply Tzeng algorithm to check the embedded discrete-time versions of $[\![P_1]\!]_{\mathrm{M}}$ and $[\![P_2]\!]_{\mathrm{M}}$ for probabilistic language equivalence by viewing each of their states as being an accepting state.
- The time complexity is $O(n^4)$.

# *Part VII:* Conclusion

# Markovian Spectrum

- Variants of $\sim_{\mathrm{MTr}}$ based on:
  - *Completed trace*: trace ending up in a deadlock state.
  - *Failure set*: set of names of actions that cannot be executed in a certain state.
  - *Failure trace*: trace extended at each step with a failure set.
  - *Ready set*: set of the names of all the actions that must be executable in a certain state.
  - *Ready trace*: trace extended at each step with a ready set.
- Enhancements with respect to traces in the nondeterministic setting:
  - completed traces for gaining deadlock sensitivity;
  - failures for reasoning about safety;
  - readies for reasoning about liveness.
- Less variability in the Markovian setting.

- $c \in \mathcal{C}_{\rm f}(P)$ is a maximal computation compatible with $\alpha \in \mathit{Name}^*$ iff $c \in \mathcal{CC}(P, \alpha)$ and the last configuration of $c$ is deadlocked.

- $\mathcal{MCC}(P, \alpha)$: multiset of maximal computations of $\mathcal{C}_{\rm f}(P)$ compatible with $\alpha$.

- $P_1 \in \mathcal{P}_{\rm M,pc}$ is Markovian completed-trace equivalent to $P_2 \in \mathcal{P}_{\rm M,pc}$, written $P_1 \sim_{\rm MCTr} P_2$, iff for all traces $\alpha \in \mathit{Name}^*$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$\boxed{\begin{array}{rcl} \mathit{prob}(\mathcal{CC}_{\leq \theta}(P_1, \alpha)) & = & \mathit{prob}(\mathcal{CC}_{\leq \theta}(P_2, \alpha)) \\ \mathit{prob}(\mathcal{MCC}_{\leq \theta}(P_1, \alpha)) & = & \mathit{prob}(\mathcal{MCC}_{\leq \theta}(P_2, \alpha)) \end{array}}$$

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

$$\boxed{P_1 \sim_{\rm MCTr} P_2 \iff P_1 \sim_{\rm MTr} P_2}$$

- $c \in \mathcal{C}_{\mathrm{f}}(P)$ is a failure computation compatible with $\varphi \equiv (\alpha, \mathcal{F}) \in \mathit{Name}^* \times 2^{\mathit{Name}}$ iff $c \in \mathcal{CC}(P, \alpha)$ and the last configuration of $c$ cannot execute any action whose name belongs to the failure set $\mathcal{F}$.

- $\mathcal{FCC}(P, \varphi)$: multiset of failure computations of $\mathcal{C}_{\mathrm{f}}(P)$ compatible with $\varphi$.

- $P_1 \in \mathcal{P}_{\mathrm{M,pc}}$ is Markovian failure equivalent to $P_2 \in \mathcal{P}_{\mathrm{M,pc}}$, written $P_1 \sim_{\mathrm{MF}} P_2$, iff for all traces with final failure set $\varphi \in \mathit{Name}^* \times 2^{\mathit{Name}}$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$\boxed{\mathit{prob}(\mathcal{FCC}_{\leq \theta}(P_1, \varphi)) = \mathit{prob}(\mathcal{FCC}_{\leq \theta}(P_2, \varphi))}$$

- $c \in \mathcal{C}_{\mathrm{f}}(P)$ is a ready computation compatible with $\rho \equiv (\alpha, \mathcal{R}) \in Name^* \times 2^{Name}$ iff $c \in \mathcal{CC}(P, \alpha)$ and the set of names of all the actions executable by the last configuration of $c$ coincides with the ready set $\mathcal{R}$.

- $\mathcal{RCC}(P, \rho)$: multiset of ready computations of $\mathcal{C}_{\mathrm{f}}(P)$ compatible with $\rho$.

- $P_1 \in \mathcal{P}_{\mathrm{M,pc}}$ is Markovian ready equivalent to $P_2 \in \mathcal{P}_{\mathrm{M,pc}}$, written $P_1 \sim_{\mathrm{MR}} P_2$, iff for all traces with final ready set $\rho \in Name^* \times 2^{Name}$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$\boxed{prob(\mathcal{RCC}_{\leq \theta}(P_1, \rho)) \; = \; prob(\mathcal{RCC}_{\leq \theta}(P_2, \rho))}$$

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

$$\boxed{P_1 \sim_{\mathrm{MR}} P_2 \iff P_1 \sim_{\mathrm{MF}} P_2}$$

- $c \in \mathcal{C}_{\rm f}(P)$ is a failure-trace computation compatible with $\phi \in (\mathit{Name} \times 2^{\mathit{Name}})^*$ iff $c$ is compatible with the trace component of $\phi$ and each configuration of $c$ cannot execute any action whose name belongs to the corresponding failure set in the failure component of $\phi$.

- $\mathcal{FTCC}(P, \phi)$: multiset of failure-trace computations of $\mathcal{C}_{\rm f}(P)$ compatible with $\phi$.

- $P_1 \in \mathcal{P}_{\rm M,pc}$ is Markovian failure-trace equivalent to $P_2 \in \mathcal{P}_{\rm M,pc}$, written $P_1 \sim_{\rm MFTr} P_2$, iff for all failure traces $\phi \in (\mathit{Name} \times 2^{\mathit{Name}})^*$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$\boxed{\mathit{prob}(\mathcal{FTCC}_{\leq \theta}(P_1, \phi)) \; = \; \mathit{prob}(\mathcal{FTCC}_{\leq \theta}(P_2, \phi))}$$

- $c \in \mathcal{C}_{\rm f}(P)$ is a ready-trace computation compatible with $\varrho \in (Name \times 2^{Name})^*$ iff $c$ is compatible with the trace component of $\varrho$ and the sets of names of all the actions executable by the configurations of $c$ coincide with the corresponding ready sets in the ready component of $\varrho$.

- $\mathcal{RTCC}(P, \varrho)$: multiset of ready-trace computations of $\mathcal{C}_{\rm f}(P)$ compatible with $\varrho$.

- $P_1 \in \mathcal{P}_{\rm M,pc}$ is Markovian ready-trace equivalent to $P_2 \in \mathcal{P}_{\rm M,pc}$, written $P_1 \sim_{\rm MRTr} P_2$, iff for all ready traces $\varrho \in (Name \times 2^{Name})^*$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$\boxed{prob(\mathcal{RTCC}_{\leq \theta}(P_1, \varrho)) \; = \; prob(\mathcal{RTCC}_{\leq \theta}(P_2, \varrho))}$$

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

$$\boxed{P_1 \sim_{\rm MRTr} P_2 \iff P_1 \sim_{\rm MFTr} P_2}$$

- Markovian linear-time/branching-time spectrum:

$$
\begin{array}{l}
\sim_{\mathrm{MB}} \subset \\
\qquad \sim_{\mathrm{MRTr}} = \sim_{\mathrm{MFTr}} \subset \\
\qquad\qquad \sim_{\mathrm{MR}} = \sim_{\mathrm{MT}} = \sim_{\mathrm{MF}} \subset \\
\qquad\qquad\qquad \sim_{\mathrm{MCTr}} = \sim_{\mathrm{MTr}}
\end{array}
$$

- More condensed than the nondeterministic spectrum.
- Similar to the probabilistic spectrum.

# Summary of Results

- Comparing Markovian behavioral equivalences:

| | *exact aggreg.* | *congr. property* | *sound & compl. axiomatization* | *modal logic charact.* | *verification complexity* |
|---|---|---|---|---|---|
| $\sim_{\mathrm{MB}}$ | $\surd$ | $\surd$ | $\surd$ | $\surd$ | $O(m \cdot \log n)$ |
| $\sim_{\mathrm{MT}}$ | $\surd$ | $\surd$ | $\surd$ | $\surd$ | $O(n^5)$ |
| $\sim_{\mathrm{MTr}}$ | $\surd$ | *no* | $\surd$ | $\surd$ | $O(n^4)$ |

- Bisimilarity, testing, and trace approaches are not only intuitively appropriate from the functional viewpoint, but also meaningful for performance evaluation purposes:
  - Aggregate the state space of a model by exploiting symmetries.
  - Reduce the state space of a model before analysis takes place.
  - No alteration of the performance properties to be assessed.

# Open Problems

- *Markovian behavioral equivalence inducing the coarsest exact non-trivial CTMC-level aggregation?*

- *Weaker versions of* $\sim_{\mathrm{MB}}$, $\sim_{\mathrm{MT}}$, *and* $\sim_{\mathrm{MTr}}$ *that abstract from invisible exponentially timed actions while preserving non-trivial exactness?*

- *Minimization algorithms for* $\sim_{\mathrm{MT}}$ *and* $\sim_{\mathrm{MTr}}$ *(and T-lumping)?*

# References


[1] M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis, *"Modelling with Generalized Stochastic Petri Nets"*, John Wiley & Sons, 1995.

[2] J.C.M. Baeten and W.P. Weijland, *"Process Algebra"*, Cambridge University Press, 1990.

[3] C. Baier, J.-P. Katoen, H. Hermanns, and V. Wolf, *"Comparative Branching-Time Semantics for Markov Chains"*, in Information and Computation 200:149-214, 2005.

[4] S. Balsamo, M. Bernardo, and M. Simeoni, *"Performance Evaluation at the Software Architecture Level"*, in *Formal Methods for Software Architectures*, LNCS 2804:207-258, 2003.

[5] J.A. Bergstra, A. Ponse, and S.A. Smolka (eds.), *"Handbook of Process Algebra"*, Elsevier, 2001.

[6] M. Bernardo, *"Non-Bisimulation-Based Markovian Behavioral Equivalences"*, in Journal of Logic and Algebraic Programming 72:3-49, 2007.

[7] M. Bernardo, *"Towards State Space Reduction Based on T-Lumpability-Consistent Relations"*, in Proc. of the *5th European Performance Engineering Workshop (EPEW 2008)*, LNCS 5261:64-78, Palma de Mallorca (Spain), 2008.

[8] M. Bernardo, *"Uniform Logical Characterizations of Testing Equivalences for Nondeterministic, Probabilistic and Markovian Processes"*, submitted for publication.

[9] M. Bernardo and A. Aldini, *"Weak Markovian Bisimilarity: Abstracting from Prioritized/Weighted Internal Immediate Actions"*, in Proc. of the 10th Italian Conf. on Theoretical Computer Science (ICTCS 2007), World Scientific, pp. 39-56, Roma (Italy), 2007.

[10] M. Bernardo and S. Botta, *"A Survey of Modal Logics Characterizing Behavioral Equivalences for Nondeterministic and Stochastic Systems"*, in Mathematical Structures in Computer Science 18:29-55, 2008.

[11] M. Bernardo and M. Bravetti, *"Performance Measure Sensitive Congruences for Markovian Process Algebras"*, in Theoretical Computer Science 290:117-160, 2003.

[12] H. Bohnenkamp, P.R. D'Argenio, H. Hermanns, and J.-P. Katoen, *"Modest: A Compositional Modeling Formalism for Hard and Softly Timed Systems"*, in IEEE Trans. on Software Engineering 32:812-830, 2006.

[13] M. Bravetti, *"Revisiting Interactive Markov Chains"*, in Proc. of the *3rd Int. Workshop on Models for Time-Critical Systems (MTCS 2002)*, ENTCS 68(5):1-20, Brno (Czech Republic), 2002.

[14] M. Bravetti, M. Bernardo, and R. Gorrieri, *"A Note on the Congruence Proof for Recursion in Markovian Bisimulation Equivalence"*, in Proc. of the *6th Int. Workshop on Process Algebra and Performance Modelling (PAPM 1998)*, pp. 153-164, Nice (France), 1998.

[15] P. Buchholz, *"Exact and Ordinary Lumpability in Finite Markov Chains"*, in Journal of Applied Probability 31:59-75, 1994.

[16] P. Buchholz, *"Markovian Process Algebra: Composition and Equivalence"*, in Proc. of the *2nd Int. Workshop on Process Algebra and Performance Modelling (PAPM 1994)*, Technical Report 27-4, pp. 11-30, Erlangen (Germany), 1994.

[17] I. Christoff, *"Testing Equivalences and Fully Abstract Models for Probabilistic Processes"*, in Proc. of the *1st Int. Conf. on Concurrency Theory (CONCUR 1990)*, LNCS 458:126-140, Amsterdam (The Netherlands), 1990.

[18] G. Clark, S. Gilmore, and J. Hillston, *"Specifying Performance Measures for PEPA"*, in Proc. of the *5th AMAST Int. Workshop on Formal Methods for Real Time and Probabilistic Systems (ARTS 1999)*, LNCS 1601:211-227, Bamberg (Germany), 1999.

[19] R. Cleaveland, Z. Dayar, S.A. Smolka, and S. Yuen, *"Testing Preorders for Probabilistic Processes"*, in Information and Computation 154:93-148, 1999.

[20] T.M. Cover and J.A. Thomas, *"Elements of Information Theory"*, John Wiley & Sons, 1991.

[21] R. De Nicola and M. Hennessy, *"Testing Equivalences for Processes"*, in Theoretical Computer Science 34:83-133, 1983.

[22] R. De Nicola, D. Latella, and M. Massink, *"Formal Modeling and Quantitative Analysis of* KLAIM-*Based Mobile Systems"*, in Proc. of the *20th ACM Symp. on Applied Computing (SAC 2005)*, ACM Press, pp. 428-435, Santa Fe (NM), 2005.

[23] S. Derisavi, H. Hermanns, and W.H. Sanders, *"Optimal State-Space Lumping in Markov Chains"*, in Information Processing Letters 87:309-315, 2003.

[24] S. Gilmore, J. Hillston, and M. Ribaudo, *"An Efficient Algorithm for Aggregating PEPA Models"*, in IEEE Trans. on Software Engineering 27:449-464, 2001.

[25] R.J. van Glabbeek, *"The Linear Time - Branching Time Spectrum I"*, in [5], pp. 3-99, 2001.

[26] R.J. van Glabbeek, S.A. Smolka, and B. Steffen, *"Reactive, Generative and Stratified Models of Probabilistic Processes"*, in Information and Computation 121:59-80, 1995.

[27] N. Götz, U. Herzog, and M. Rettelbach, *"Multiprocessor and Distributed System Design: The Integration of Functional Specification and Performance Analysis Using Stochastic Process Algebras"*, in Proc. of the *16th Int. Symp. on Computer Performance Modelling, Measurement and Evaluation (PERFORMANCE 1993)*, LNCS 729:121-146, Roma (Italy), 1993.

[28] M. Hennessy and R. Milner, *"Algebraic Laws for Nondeterminism and Concurrency"*, in Journal of the ACM 32:137-162, 1985.

[29] H. Hermanns, *"Interactive Markov Chains"*, LNCS 2428, 2002.

[30] H. Hermanns and M. Rettelbach, *"Syntax, Semantics, Equivalences, and Axioms for MTIPP"*, in Proc. of the *2nd Int. Workshop on Process Algebra and Performance Modelling (PAPM 1994)*, Technical Report 27-4, pp. 71-87, Erlangen (Germany), 1994.

[31] J. Hillston, *"The Nature of Synchronisation"*, in Proc. of the *2nd Int. Workshop on Process Algebra and Performance Modelling (PAPM 1994)*, Technical Report 27-4, pp. 51-70, Erlangen (Germany), 1994.

[32] J. Hillston, *"A Compositional Approach to Performance Modelling"*, Cambridge University Press, 1996.

[33] C.A.R. Hoare, *"Communicating Sequential Processes"*, Prentice Hall, 1985.

[34] R.A. Howard, *"Dynamic Probabilistic Systems"*, John Wiley & Sons, 1971.

[35] D.T. Huynh and L. Tian, *"On Some Equivalence Relations for Probabilistic Processes"*, in Fundamenta Informaticae 17:211-234, 1992.

[36] C.-C. Jou and S.A. Smolka, *"Equivalences, Congruences, and Complete Axiomatizations for Probabilistic Processes"*, in Proc. of the *1st Int. Conf. on Concurrency Theory (CONCUR 1990)*, LNCS 458:367-383, Amsterdam (The Netherlands), 1990.

[37] L. Kleinrock, *"Queueing Systems"*, John Wiley & Sons, 1975.

[38] M.Z. Kwiatkowska and G.J. Norman, *"A Testing Equivalence for Reactive Probabilistic Processes"*, in Proc. of the *2nd Int. Workshop on Expressiveness in Concurrency (EXPRESS 1998)*, ENTCS 16(2):114-132, Nice (France), 1998.

[39] K.G. Larsen and A. Skou, *"Bisimulation through Probabilistic Testing"*, in Information and Computation 94:1-28, 1991.

[40] R. Milner, *"Communication and Concurrency"*, Prentice Hall, 1989.

[41] R. Paige and R.E. Tarjan, *"Three Partition Refinement Algorithms"*, in SIAM Journal on Computing 16:973-989, 1987.

[42] C. Priami, *"Stochastic $\pi$-Calculus"*, in Computer Journal 38:578-589, 1995.

[43] M. Rettelbach, *"Probabilistic Branching in Markovian Process Algebras"*, in Computer Journal 38:590-599, 1995.

[44] J. Sproston and S. Donatelli, *"Backward Bisimulation in Markov Chain Model Checking"*, in IEEE Trans. on Software Engineering 32:531-546, 2006.

[45] E.W. Stark, R. Cleaveland, and S.A. Smolka, *"A Process-Algebraic Language for Probabilistic I/O Automata"*, in Proc. of the *14th Int. Conf. on Concurrency Theory (CONCUR 2003)*, LNCS 2761:189-203, Marseille (France), 2003.

[46] W.J. Stewart, *"Introduction to the Numerical Solution of Markov Chains"*, Princeton University Press, 1994.

[47] W.-G. Tzeng, *"A Polynomial-Time Algorithm for the Equivalence of Probabilistic Automata"*, in SIAM Journal on Computing 21:216-227, 1992.

[48] V. Wolf, C. Baier, and M. Majster-Cederbaum, *"Trace Machines for Observing Continuous-Time Markov Chains"*, in Proc. of the *3rd Int. Workshop on Quantitative Aspects of Programming Languages (QAPL 2005)*, ENTCS 153(2):259-277, Edinburgh (UK), 2005.

[49] M. Woodside, *"From Annotated Software Designs (UML SPT/MARTE) to Model Formalisms"*, in *Formal Methods for Performance Evaluation*, LNCS 4486:429-467, 2007.