# On the Expressiveness of Markovian Process Calculi with Durational and Durationless Actions

Marco Bernardo
Dipartimento di Matematica, Fisica e Informatica – Università di Urbino "Carlo Bo" – Italy

Several Markovian process calculi have been proposed in the literature, which differ from each other for various aspects. With regard to the action representation, we distinguish between integrated-time Markovian process calculi, in which every action has an exponentially distributed duration associated with it, and orthogonal-time Markovian process calculi, in which action execution is separated from time passing. Similar to deterministically timed process calculi, we show that these two options are not irreconcilable by exhibiting three mappings from an integrated-time Markovian process calculus to an orthogonal-time Markovian process calculus that preserve the behavioral equivalence of process terms under different interpretations of action execution: eagerness, laziness, and maximal progress. The mappings are limited to classes of process terms of the integrated-time Markovian process calculus with restrictions on parallel composition and do not involve the full capability of the orthogonal-time Markovian process calculus of expressing nondeterministic choices, thus elucidating the only two important differences between the two calculi: their synchronization disciplines and their ways of solving choices.

## 1 Introduction

Communicating concurrent systems are characterized not only by their functional behavior, but also by their quantitative features. A prominent role is played by timing aspects, which express the temporal ordering of system activities and are of paramount importance in the study of the properties of real-time systems as well as shared-resource systems. As witnessed by a rich literature, there are several different options for introducing time and time passing in system descriptions, many of which have been formalized in a process algebraic setting [2].

Starting from the late 80's, a number of deterministically timed process calculi have been proposed – like, e.g., timed CSP [21], temporal CCS [16], timed CCS [24], real-time ACP [3], urgent LOTOS [5], TIC [20], ATP [18], TPL [10], cIPA [1], and PAFAS [8] – in which time and time passing are represented through a dense or discrete time domain – like, e.g., $(\mathbb{N}, +, \leq)$ – equipped with an associative operation with neutral element and a total order defined on the basis of this operation. As observed in [17, 23, 7], the various deterministically timed process calculi differ for a number of time-related options, some of which give rise to the one-phase functioning principle – according to which actions are durational, time is absolute, and several local clocks are present – and the two-phase functioning principle – according to which actions are durationless, time is relative, and there is a single global clock:

- Durational actions versus durationless actions. In the first case, every action takes a fixed amount of time to be performed and time passes only due to action execution; hence, functional behavior and time passing are integrated. In the second case, actions are instantaneous events and time passes in between them; hence, functional behavior and time passing are orthogonal.
- Absolute time versus relative time. Assuming that timestamps are associated with the events observed during system execution, in the first case all timestamps refer to the starting time of the

GANDALF 2010
EPTCS ??, 2010, pp. 1–21, doi:10.4204/EPTCS.??.??

system execution, while in the second case each timestamp refers to the starting time or the completion time of the previously observed event (the two times coincide if events are durationless).

- Local clocks versus global clock. In the first case, there are several clocks associated with the various system parts, which elapse independent of each other although they define a unique notion of global time. In the second case, there is a single clock that governs time passing.

Another degree of freedom is concerned with the different interpretations of action execution, in terms of whether and when it can be delayed. There are at least the following three interpretations:

- Eagerness, which establishes that actions must be performed as soon as they become enabled without any delay, thereby implying that actions are urgent.
- Laziness, which establishes that, once they become enabled, actions can be delayed arbitrarily long before they are executed.
- Maximal progress, which establishes that actions can be delayed arbitrarily long unless they are involved in synchronizations, in which case they are urgent.

In [7], a translating function is defined from a one-phase deterministically timed process calculus inspired by [1] to a two-phase deterministically timed process calculus inspired by [16], which is shown to preserve the behavioral equivalence of process terms based on CCS-like parallel composition and restriction operators [15]. The result holds under eagerness (only for restriction-free terms), laziness, and maximal progress, both when observing the starting time of action execution and when observing the completion time of action execution. This demonstrates that the different choices that can be made about the representation of time and time passing in a deterministically timed framework are not irreconcilable.

Starting from the first half of the 90's, a number of stochastically timed process calculi have been proposed too – like, e.g., TIPP [9, 12], PEPA [13], MPA [6], $\text{EMPA}_{\text{gr}}$ [4], S$\pi$ [19], IMC [11], and PIOA [22] – in which time and time passing are represented by means of exponentially distributed random variables rather than nonnegative numbers. The reason for using only exponential distributions (uniquely identified through their rates, positive real numbers corresponding to the reciprocal of their expected values) is twofold. Firstly, the stochastic process underlying a system description turns out to be a continuous-time Markov chain, which simplifies quantitative analysis without sacrificing expressiveness. Secondly, the memoryless property of exponential distributions fits well with the interleaving view of concurrency.

The time-related options and the action execution interpretations discussed for deterministically timed process calculi apply to a large extent also to stochastically timed process calculi. This is especially true for the difference between durational actions and durationless actions, which results in integrated-time Markovian process calculi like TIPP, PEPA, MPA, $\text{EMPA}_{\text{gr}}$, S$\pi$, and PIOA and orthogonal-time Markovian process calculi like IMC, respectively. By contrast, the distinction between absolute time and relative time and the concept of clock are not important in a Markovian framework. Due to the memoryless property of exponential distributions, only rates of durational actions or time delays matter.

A remarkable difference between deterministically timed process calculi and stochastically timed process calculi is concerned with the way choices among alternative behaviors are solved. In the first case, the choice is nondeterministic precisely as in classical process calculi, which means that time does not solve choices. In an orthogonal-time setting, this is witnessed by the presence of operational semantic rules according to which a process term of the form $(n) \,.\, Q_1 + (n) \,.\, Q_2$ – where $+$ denotes the alternative composition operator – can let $n$ time units pass and then evolves into $Q_1 + Q_2$. In the second case, the choice can instead be probabilistic whenever exponentially distributed delays come into play. In the same orthogonal-time setting, a process term of the form $(\lambda_1) \,.\, Q_1 + (\lambda_2) \,.\, Q_2$ – where $\lambda_1$ and $\lambda_2$ are the rates of two exponentially distributed delays – evolves into $Q_1$ or $Q_2$ with probabilities $\frac{\lambda_1}{\lambda_1+\lambda_2}$ and $\frac{\lambda_2}{\lambda_1+\lambda_2}$.

This has an impact on the expressiveness of Markovian process calculi. In fact, the orthogonal-time ones are more expressive than the integrated-time ones, because the former can represent both action-based nondeterministic choices and time-based probabilistic choices, whereas the latter can represent only probabilistic choices based on action durations. In turn, this has an impact on the expressiveness of the synchronization discipline adopted in the considered calculi. In fact, in the orthogonal-time case the time to the synchronization of two actions can be naturally expressed as the maximum of two exponentially distributed delays, whereas in the integrated-time case the duration of the synchronization of two exponentially timed actions has to be assumed to be exponentially distributed with rate given by the application of an associative and commutative operation to the two original rates.

Another important difference between deterministically timed process calculi and stochastically timed process calculi is concerned with the formalization of the various interpretations of action execution. In the first case, all the three interpretations can be encoded in the operational semantic rules. In the second case, it depends on whether time is integrated with action execution or separated from it, as we show in this paper. On the one hand, observed that the usual operational semantic rules for integrated-time Markovian process calculi encode eagerness as they permit no delay, we recognize that the same rules encode laziness and maximal progress too, because the possibility of delaying the beginning of action execution is inherent in the memoryless property of exponentially distributed durations. On the other hand, since additional operational semantic rules delaying action execution would produce no effect in orthogonal-time Markovian process calculi as time can solve choices, we exploit the behavioral equivalence to express when action execution takes precedence over time passing.

In spite of the different expressiveness they induce, in this paper we show that durational actions and durationless actions are not irreconcilable even in a Markovian setting. Similar to [7], this is accomplished by defining three translating functions from an integrated-time Markovian process calculus to an orthogonal-time Markovian process calculus that preserve the behavioral equivalence of process terms under eagerness, laziness, and maximal progress, respectively. The encodings are limited to classes of process terms of the integrated-time Markovian process calculus with restrictions on parallel composition and do not involve the full capability of the orthogonal-time Markovian process calculus of expressing nondeterministic choices. This formally clarifies the only two important differences between the two calculi, i.e., their different synchronization disciplines and their different ways of solving choices.

This paper is organized as follows. In Sects. 2 and 3, we uniformly present the syntax, the operational semantics, and a bisimulation-based behavioral equivalence for an integrated-time Markovian process calculus and an orthogonal-time Markovian process calculus, respectively, and we discuss how to represent the three different interpretations of action execution. Then, in Sect. 4 we exhibit the three encodings from certain classes of process terms of the integrated-time Markovian process calculus to certain classes of process terms of the orthogonal-time Markovian process calculus and we demonstrate that they preserve the bisimulation-based behavioral equivalence of the considered process terms. Finally, in Sect. 5 we report some concluding remarks.

## 2 Markovian Process Calculus with Durational Actions

In this section, we present a Markovian process calculus inspired by [13, 12] in which every action has associated with it a rate that uniquely identifies its exponentially distributed duration. The presentation of the integrated-time Markovian process calculus – ITMPC for short – consists of the definition of its syntax, its operational semantics, and a bisimulation-based behavioral equivalence. A discussion of the interpretation of action execution accompanies the definition of the operational semantics.

## 2.1 Durational Actions and Behavioral Operators

In ITMPC, an exponentially timed action is represented as a pair $<a, \lambda>$. The first element, $a$, is the name of the action, which is $\tau$ in the case that the action is internal, otherwise it belongs to a set $\mathit{Name}_{\mathrm{v}}$ of visible action names. The second element, $\lambda \in \mathbb{R}_{>0}$, is the rate of the exponentially distributed random variable $RV$ quantifying the duration of the action, i.e., $\Pr\{RV \leq t\} = 1 - \mathrm{e}^{-\lambda \cdot t}$ for $t \in \mathbb{R}_{>0}$. The average duration of the action is equal to the reciprocal of its rate, i.e., $1/\lambda$. If several exponentially timed actions are enabled, the race policy is adopted: the action that is executed is the fastest one.

The sojourn time associated with a process term $P$ is thus the minimum of the random variables quantifying the durations of the exponentially timed actions enabled by $P$. Since the minimum of several exponentially distributed random variables is exponentially distributed and its rate is the sum of the rates of the original variables, the sojourn time associated with $P$ is exponentially distributed with rate equal to the sum of the rates of the actions enabled by $P$. Therefore, the average sojourn time associated with $P$ is the reciprocal of the sum of the rates of the actions it enables. The probability of executing one of those actions is given by the action rate divided by the sum of the rates of all the considered actions.

ITMPC comprises a CSP-like parallel composition operator [14] according to which two exponentially timed actions synchronize iff they have the same visible name belonging to an explicit synchronization set. The resulting action has the same name as the two original actions and its rate is obtained by applying an associative and commutative operator $\otimes$ to the rates of the two original actions.

We denote by $\mathit{Act}_{\mathrm{M,it}} = \mathit{Name} \times \mathbb{R}_{>0}$ the set of actions of ITMPC, where $\mathit{Name} = \mathit{Name}_{\mathrm{v}} \cup \{\tau\}$ is the set of action names – ranged over by $a, b$ – and $\mathbb{R}_{>0}$ is the set of action rates – ranged over by $\lambda, \mu$. We then denote by $\mathit{Relab}$ a set of relabeling functions $\varphi : \mathit{Name} \rightarrow \mathit{Name}$ that preserve action visibility, i.e., such that $\varphi^{-1}(\tau) = \{\tau\}$. Finally, we denote by $\mathit{Var}$ a set of process variables ranged over by $X, Y, Z$.

**Definition 2.1** The process language $\mathscr{PL}_{\mathrm{M,it}}$ is generated by the following syntax:

| | | | |
|---|---|---|---|
| $P$ | $::=$ | $\underline{0}$ | inactive process |
| | $\mid$ | $<a, \lambda>.P$ | exponentially timed action prefix |
| | $\mid$ | $P + P$ | alternative composition |
| | $\mid$ | $P \parallel_S P$ | parallel composition |
| | $\mid$ | $P/H$ | hiding |
| | $\mid$ | $P[\varphi]$ | relabeling |
| | $\mid$ | $X$ | process variable |
| | $\mid$ | $\mathrm{rec}\, X : P$ | recursion |

where $a \in \mathit{Name}$, $\lambda \in \mathbb{R}_{>0}$, $S, H \subseteq \mathit{Name}_{\mathrm{v}}$, $\varphi \in \mathit{Relab}$, and $X \in \mathit{Var}$. We denote by $\mathbb{P}_{\mathrm{M,it}}$ the set of closed and guarded process terms of $\mathscr{PL}_{\mathrm{M,it}}$. ■

## 2.2 Integrated-Time Operational Semantics: Eagerness, Laziness, Maximal Progress

The semantics for ITMPC can be defined in the usual operational style, with an important difference with respect to the nondeterministic case. A process term like $<a, \lambda>.\underline{0} + <a, \lambda>.\underline{0}$ is not the same as $<a, \lambda>.\underline{0}$, because the average sojourn time associated with the latter, i.e., $1/\lambda$, is twice the average sojourn time associated with the former, i.e., $1/(\lambda + \lambda)$. A way of assigning distinct semantic models to terms like the two considered above consists of taking into account the multiplicity of each transition, intended as the number of different proofs for the transition derivation.

The semantic model $[\![P]\!]_{\mathrm{M,it}}$ for a process term $P \in \mathbb{P}_{\mathrm{M,it}}$ is thus a labeled multitransition system. Its multitransition relation is contained in the smallest multiset of elements of $\mathbb{P}_{\mathrm{M,it}} \times Act_{\mathrm{M,it}} \times \mathbb{P}_{\mathrm{M,it}}$ that satisfy the operational semantic rules of Table 1 – where $\{\_ \hookrightarrow \_\}$ denotes syntactical replacement – and keep track of all the possible ways of deriving each transition.

$$(\textsc{Pre}_{\mathrm{M,it}}) \quad \frac{}{<a,\lambda>.P \xrightarrow{a,\lambda}_{\mathrm{M,it}} P}$$

$$(\textsc{Alt}_{\mathrm{M,it},1}) \quad \frac{P_1 \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'}{P_1 + P_2 \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'} \qquad (\textsc{Alt}_{\mathrm{M,it},2}) \quad \frac{P_2 \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'}{P_1 + P_2 \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'}$$

$$(\textsc{Par}_{\mathrm{M,it},1}) \quad \frac{P_1 \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'_1 \qquad a \notin S}{P_1 \,\|_S\, P_2 \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'_1 \,\|_S\, P_2} \qquad (\textsc{Par}_{\mathrm{M,it},2}) \quad \frac{P_2 \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'_2 \qquad a \notin S}{P_1 \,\|_S\, P_2 \xrightarrow{a,\lambda}_{\mathrm{M,it}} P_1 \,\|_S\, P'_2}$$

$$(\textsc{Syn}_{\mathrm{M,it}}) \quad \frac{P_1 \xrightarrow{a,\lambda_1}_{\mathrm{M,it}} P'_1 \qquad P_2 \xrightarrow{a,\lambda_2}_{\mathrm{M,it}} P'_2 \qquad a \in S}{P_1 \,\|_S\, P_2 \xrightarrow{a,\lambda_1 \otimes \lambda_2}_{\mathrm{M,it}} P'_1 \,\|_S\, P'_2}$$

$$(\textsc{Hid}_{\mathrm{M,it},1}) \quad \frac{P \xrightarrow{a,\lambda}_{\mathrm{M,it}} P' \qquad a \in H}{P/H \xrightarrow{\tau,\lambda}_{\mathrm{M,it}} P'/H} \qquad (\textsc{Hid}_{\mathrm{M,it},2}) \quad \frac{P \xrightarrow{a,\lambda}_{\mathrm{M,it}} P' \qquad a \notin H}{P/H \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'/H}$$

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

Table 1: Operational semantic rules for ITMPC

These operational semantic rules encode an eager interpretation of action execution, as they permit no delay between the time at which an exponentially timed action becomes enabled and the time at which the same action starts its execution. This is the standard interpretation adopted by all the integrated-time Markovian process calculi appeared in the literature. However, the operational semantic rules of Table 1 encode laziness and maximal progress too, because the possibility of delaying the beginning of action execution is inherent in the memoryless property of exponentially distributed durations. In fact, if an exponentially timed action does not finish its execution within $t$ time units, the residual execution time has the same distribution as the whole action duration and thus the beginning of the execution of the action can be thought of as being delayed by $t$ time units with respect to the instant in which the action has become enabled.

Recalling that in the durational setting defined in [7] every state is a pair $k \Rightarrow P$ where $k$ is the clock and $P$ is the process, the operational semantic rule for lazy deterministically timed actions is of the form:

$$k \Rightarrow {<}a,n{>}.P \xrightarrow{a,n} (k+t+n) \Rightarrow P \qquad \forall t \in \mathbb{N}$$

where $k \in \mathbb{N}$ is the value of the clock when the action becomes enabled, $n \in \mathbb{N}$ is the fixed duration of the action, and $k+t+n$ is the value of the clock when the action finishes its execution, with $t$ being an arbitrary delay between the time at which the action becomes enabled and the time at which the action starts its execution. In the maximal progress case, the above rule is applied only when $a \in \mathit{Name}_{\mathrm{v}}$, while the rule for deterministically timed $\tau$-actions is still of the form:

$$k \Rightarrow {<}\tau,n{>}.P \xrightarrow{\tau,n} (k+n) \Rightarrow P$$

which enforces an eager interpretation of those actions because they cannot be delayed.

Since in a Markovian framework it is possible to express rates but not fixed durations, the only analogous operational semantic rule for lazy exponentially timed actions would be of the form:

$${<}a,\lambda{>}.P \xrightarrow{a,\lambda'} P \qquad \forall \lambda' \in \mathbb{R}_{]0,\lambda]}$$

However, this would represent an action slowdown rather than delaying the beginning of the action execution by an arbitrary amount of time and then performing the action at its rate. An appropriate semantic treatment of lazy exponentially timed actions should not alter their rates. Therefore, a better option is to add a further operational semantic rule for action prefix of the form:

$${<}a,\lambda{>}.P \xrightarrow{\tau,\lambda'} {<}a,\lambda{>}.P \qquad \forall \lambda' \in \mathbb{R}_{>0}$$

which introduces invisible selfloops each having an arbitrary rate. But these selfloops have no impact on (the transient/stationary state probabilities of) the underlying continuous-time Markov chain. In fact, thanks to the memoryless property of exponential distributions, the time remaining to moving from ${<}a,\lambda{>}.P$ to $P$ after the execution of an arbitrary number of selfloops is still exponentially distributed with rate $\lambda$. As a consequence, the introduction of these selfloops is useless, which means that the operational semantic rules of Table 1 encode also laziness. Since maximal progress is in some sense between eagerness and laziness, it is encoded in those rules as well.

## 2.3 Integrated-Time Markovian Bisimilarity

A behavioral equivalence over $\mathbb{P}_{\mathrm{M,it}}$ can be defined by establishing that, 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. This can be easily formalized through the comparison of the process term exit rates.

The integrated-time exit rate of a process term $P \in \mathbb{P}_{\mathrm{M,it}}$ is the rate at which $P$ can execute actions of a certain name $a \in \mathit{Name}$ that lead to a certain destination $D \subseteq \mathbb{P}_{\mathrm{M,it}}$ and is given by the sum of the rates of those actions due to the race policy:

$$\boxed{\mathit{rate}_{\mathrm{it}}(P,a,D) \,=\, \sum \{\!| \, \lambda \in \mathbb{R}_{>0} \mid \exists P' \in D.\, P \xrightarrow{a,\lambda}_{\mathrm{M,it}} P' \,|\!\}}$$

where $\{\!|$ and $|\!\}$ are multiset delimiters and the summation is taken to be zero if its multiset is empty. By summing up the rates of all the actions of $P$, we obtain the integrated-time total exit rate of $P$:

$$\boxed{\mathit{rate}_{\mathrm{it,t}}(P) \,=\, \sum_{a \in \mathit{Name}} \mathit{rate}_{\mathrm{it}}(P,a,\mathbb{P}_{\mathrm{M,it}})}$$

which coincides with the reciprocal of the average sojourn time associated with $P$.

**Definition 2.2** An equivalence relation $\mathscr{B}$ over $\mathbb{P}_{\rm M,it}$ is an integrated-time Markovian bisimulation iff, whenever $(P_1, P_2) \in \mathscr{B}$, then for all action names $a \in \mathit{Name}$ and equivalence classes $D \in \mathbb{P}_{\rm M,it} / \mathscr{B}$:

$$\mathit{rate}_{\rm it}(P_1, a, D) \; = \; \mathit{rate}_{\rm it}(P_2, a, D)$$

Integrated-time Markovian bisimilarity $\sim_{\rm MB,it}$ is the union of all the integrated-time Markovian bisimulations. ■

$\sim_{\rm MB,it}$ can be shown to be a congruence with respect to all the operators of ITMPC as well as recursion, and to have a sound and complete axiomatization over nonrecursive process terms including typical laws like associativity, commutativity, and neutral element for the alternative composition operator, the expansion law for the parallel composition operator, and distributive laws for hiding and relabeling with respect to alternative composition. Its characterizing law – which replaces the usual idempotency of the alternative composition operator and encodes the race policy – is the following:

$$<a, \lambda_1>.P + <a, \lambda_2>.P \; \sim_{\rm MB,it} \; <a, \lambda_1 + \lambda_2>.P$$

# 3 Markovian Process Calculus with Durationless Actions

In this section, we present a Markovian process calculus inspired by [11] in which actions are durationless and hence action execution is separated from time passing. The presentation of the orthogonal-time Markovian process calculus – OTMPC for short – consists of the definition of its syntax, its operational semantics, and a bisimulation-based behavioral equivalence. A discussion of the interpretation of action execution accompanies the definition of the behavioral equivalence.

## 3.1 Durationless Actions, Time Passing, and Behavioral Operators

In OTMPC, actions are instantaneous and time passes in between them. As a consequence, there are two prefix operators: an action prefix operator $a \, . \, \_$, with $a \in \mathit{Name}$, and a time prefix operator $(\lambda) \, . \, \_$, with $\lambda \in \mathbb{R}_{>0}$. Similar to ITMPC, time delays are governed by exponential distributions and are subject to the race policy. Different from ITMPC, the CSP-like parallel composition operator enforces synchronizations only between two actions that have the same visible name belonging to the synchronization set; hence, time delays are not involved in synchronizations. Moreover, the choice among alternative actions is nondeterministic.

**Definition 3.1** The process language $\mathscr{PL}_{\rm M,ot}$ is generated by the following syntax:

| | | | |
|---|---|---|---|
| $Q$ | $::=$ | $\underline{0}$ | inactive process |
| | $\mid$ | $a \, . \, Q$ | action prefix |
| | $\mid$ | $(\lambda) \, . \, Q$ | time prefix |
| | $\mid$ | $Q + Q$ | alternative composition |
| | $\mid$ | $Q \parallel_S Q$ | parallel composition |
| | $\mid$ | $Q / H$ | hiding |
| | $\mid$ | $Q[\varphi]$ | relabeling |
| | $\mid$ | $X$ | process variable |
| | $\mid$ | $\mathrm{rec} X : Q$ | recursion |

where $a \in \mathit{Name}$, $\lambda \in \mathbb{R}_{>0}$, $S, H \subseteq \mathit{Name}_{\rm v}$, $\varphi \in \mathit{Relab}$, and $X \in \mathit{Var}$. We denote by $\mathbb{P}_{\rm M,ot}$ the set of closed and guarded process terms of $\mathscr{PL}_{\rm M,ot}$. ■

$$(\textsc{Pre})\quad \frac{}{a\,.\,Q \xrightarrow{a} Q}$$

$$(\textsc{Alt}_1)\quad \frac{Q_1 \xrightarrow{a} Q'}{Q_1 + Q_2 \xrightarrow{a} Q'} \qquad (\textsc{Alt}_2)\quad \frac{Q_2 \xrightarrow{a} Q'}{Q_1 + Q_2 \xrightarrow{a} Q'}$$

$$(\textsc{Par}_1)\quad \frac{Q_1 \xrightarrow{a} Q'_1 \qquad a \notin S}{Q_1 \parallel_S Q_2 \xrightarrow{a} Q'_1 \parallel_S Q_2} \qquad (\textsc{Par}_2)\quad \frac{Q_2 \xrightarrow{a} Q'_2 \qquad a \notin S}{Q_1 \parallel_S Q_2 \xrightarrow{a} Q_1 \parallel_S Q'_2}$$

$$(\textsc{Syn})\quad \frac{Q_1 \xrightarrow{a} Q'_1 \qquad Q_2 \xrightarrow{a} Q'_2 \qquad a \in S}{Q_1 \parallel_S Q_2 \xrightarrow{a} Q'_1 \parallel_S Q'_2}$$

$$(\textsc{Hid}_1)\quad \frac{Q \xrightarrow{a} Q' \qquad a \in H}{Q/H \xrightarrow{\tau} Q'/H} \qquad (\textsc{Hid}_2)\quad \frac{Q \xrightarrow{a} Q' \qquad a \notin H}{Q/H \xrightarrow{a} Q'/H}$$

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

$$(\textsc{Pre}_{\mathrm{M}})\quad \frac{}{(\lambda)\,.\,Q \xrightarrow{\lambda}_{\mathrm{M}} Q}$$

$$(\textsc{Alt}_{\mathrm{M},1})\quad \frac{Q_1 \xrightarrow{\lambda}_{\mathrm{M}} Q'}{Q_1 + Q_2 \xrightarrow{\lambda}_{\mathrm{M}} Q'} \qquad (\textsc{Alt}_{\mathrm{M},2})\quad \frac{Q_2 \xrightarrow{\lambda}_{\mathrm{M}} Q'}{Q_1 + Q_2 \xrightarrow{\lambda}_{\mathrm{M}} Q'}$$

$$(\textsc{Par}_{\mathrm{M},1})\quad \frac{Q_1 \xrightarrow{\lambda}_{\mathrm{M}} Q'_1}{Q_1 \parallel_S Q_2 \xrightarrow{\lambda}_{\mathrm{M}} Q'_1 \parallel_S Q_2} \qquad (\textsc{Par}_{\mathrm{M},2})\quad \frac{Q_2 \xrightarrow{\lambda}_{\mathrm{M}} Q'_2}{Q_1 \parallel_S Q_2 \xrightarrow{\lambda}_{\mathrm{M}} Q_1 \parallel_S Q'_2}$$

$$(\textsc{Hid}_{\mathrm{M}})\quad \frac{Q \xrightarrow{\lambda}_{\mathrm{M}} Q'}{Q/H \xrightarrow{\lambda}_{\mathrm{M}} Q'/H}$$

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

Table 2: Operational semantic rules for OTMPC: action transitions and time transitions

### 3.2 Orthogonal-Time Operational Semantics

The semantics for OTMPC relies on two transition relations: one for action execution and one for time passing. Like for nondeterministic processes, the former is defined as the smallest subset of $\mathbb{P}_{\rm M,ot} \times Name \times \mathbb{P}_{\rm M,ot}$ satisfying the operational semantic rules in the upper part of Table 2. Since $(\lambda) \,.\, Q + (\lambda) \,.\, Q$ is not the same as $(\lambda) \,.\, Q$, the latter is defined as the smallest multiset of elements of $\mathbb{P}_{\rm M,ot} \times \mathbb{R}_{>0} \times \mathbb{P}_{\rm M,ot}$ that satisfy the operational semantic rules in the lower part of Table 2 and keep track of all the possible ways of deriving each transition. The semantic model $[\![Q]\!]_{\rm M,ot}$ for a process term $Q \in \mathbb{P}_{\rm M,ot}$ is thus a labeled multitransition system, which can contain both nondeterministic and probabilistic branchings.

### 3.3 Orthogonal-Time Markovian Bisimilarity: Eagerness, Laziness, Maximal Progress

A behavioral equivalence over $\mathbb{P}_{\rm M,ot}$ can be defined by combining classical bisimilarity for action execution with exit rate comparison for time passing. The orthogonal-time exit rate of a process term $Q \in \mathbb{P}_{\rm M,ot}$ is the rate at which $Q$ can let time pass when going to a certain destination $D \subseteq \mathbb{P}_{\rm M,ot}$ and is given by the sum of the rates of $Q$ delays leading to $D$ due to the race policy:

$$\boxed{rate_{\rm ot}(Q, D) \,=\, \sum \{\!| \, \lambda \in \mathbb{R}_{>0} \mid \exists Q' \in D \,.\, Q \xrightarrow{\lambda}_{\rm M} Q' \,|\!\}}$$

By summing up the rates of all the delays of $Q$, we obtain the orthogonal-time total exit rate of $Q$:

$$\boxed{rate_{\rm ot,t}(Q) \,=\, rate_{\rm ot}(Q, \mathbb{P}_{\rm M,ot})}$$

which coincides with the reciprocal of the average sojourn time associated with $Q$.

The behavioral equivalence can be defined in different ways depending on the interpretation of action execution. We observe that the operational semantic rule for action prefix in the upper part of Table 2 encodes an eager interpretation, because it permits no delay between the time at which an action becomes enabled and the time at which the same action starts its execution. In contrast to the integrated time case, a different interpretation of action execution cannot be encoded in the operational semantic rules because time can solve choices due to the adoption of the race policy.

Following the durationless setting defined in [7], an additional operational semantic rule of the form:

$$a \,.\, Q \xrightarrow{t} a \,.\, Q \quad \forall t \in \mathbb{N}$$

has to be introduced to manage lazy actions in a deterministically timed process calculus, with $t$ being an arbitrary delay between the time at which the action becomes enabled and the time at which the action can start its execution. In the maximal progress case, the additional rule is applied only when $a \in Name_{\rm v}$ because $\tau$-actions cannot let time pass. The effect of the additional rule is that a process term can let time pass iff so can all the actions it enables. This is a consequence of some of the operational semantic rules for binary operators, which are of the form:

$$\frac{Q_1 \xrightarrow{t} Q'_1 \qquad Q_2 \xrightarrow{t} Q'_2}{Q_1 + Q_2 \xrightarrow{t} Q'_1 + Q'_2} \qquad\qquad \frac{Q_1 \xrightarrow{t} Q'_1 \qquad Q_2 \xrightarrow{t} Q'_2}{Q_1 \,\|_S\, Q_2 \xrightarrow{t} Q'_1 \,\|_S\, Q'_2}$$

and hence formalize the fact that time does not solve choices.

The analogous additional operational semantic rule for handling lazy actions in a Markovian framework would be of the form:

$$a \,.\, Q \xrightarrow{\lambda} a \,.\, Q \quad \forall \lambda \in \mathbb{R}_{>0}$$

However, the resulting exponentially timed selfloops would have no impact on the underlying continuous-time Markov chain, as already discussed at the end of Sect. 2.2. Most importantly, the additional rule would not produce the desired effect, because in a Markovian framework time can solve choices due to the adoption of the race policy and therefore rules like those above for alternative and parallel composition in the deterministically timed case are not appropriate in the stochastically timed case.

The desired effect is instead obtained by encoding the three different interpretations of action execution into three different variants of orthogonal-time Markovian bisimilarity. All of them work like classical bisimilarity for action execution. As regards time passing, the exit rate comparison is performed: only for pairs of terms that cannot execute any action under eagerness; for all pairs of terms under laziness; only for pairs of terms that cannot execute any $\tau$-action under maximal progress.

**Definition 3.2** An equivalence relation $\mathcal{B}$ over $\mathbb{P}_{\mathrm{M,ot}}$ is an eager/lazy/maximal-progress orthogonal-time Markovian bisimulation iff, whenever $(Q_1, Q_2) \in \mathcal{B}$, then:

- For all action names $a \in \mathit{Name}$:
  - Whenever $Q_1 \xrightarrow{a} Q'_1$, then $Q_2 \xrightarrow{a} Q'_2$ with $(Q'_1, Q'_2) \in \mathcal{B}$.
  - Whenever $Q_2 \xrightarrow{a} Q'_2$, then $Q_1 \xrightarrow{a} Q'_1$ with $(Q'_1, Q'_2) \in \mathcal{B}$.
- For all equivalence classes $D \in \mathbb{P}_{\mathrm{M,ot}}/\mathcal{B}$:
$$\mathit{rate}_{\mathrm{ot}}(Q_1, D) \, = \, \mathit{rate}_{\mathrm{ot}}(Q_2, D)$$
  whenever:
  - $Q_1$ and $Q_2$ cannot perform any action (eagerness).
  - $Q_1$ and $Q_2$ are arbitrary (laziness).
  - $Q_1$ and $Q_2$ cannot perform any $\tau$-action (maximal progress).

Eager/lazy/maximal-progress orthogonal-time Markovian bisimilarity $\sim_{\mathrm{MB,ot,e}}/\sim_{\mathrm{MB,ot,l}}/\sim_{\mathrm{MB,ot,mp}}$ is the union of all the eager/lazy/maximal-progress orthogonal-time Markovian bisimulations. ■

It turns out $\sim_{\mathrm{MB,ot,l}} \subset \sim_{\mathrm{MB,ot,mp}} \subset \sim_{\mathrm{MB,ot,e}}$. In contrast to $\sim_{\mathrm{MB,ot,e}}$, which is not a congruence with respect to parallel composition, $\sim_{\mathrm{MB,ot,l}}$ and $\sim_{\mathrm{MB,ot,mp}}$ can be shown to be congruences with respect to all the operators of OTMPC as well as recursion and to have a sound and complete axiomatization over nonrecursive process terms including typical laws like associativity, commutativity, and neutral element for the alternative composition operator, the expansion law for the parallel composition operator, and distributive laws for hiding and relabeling with respect to alternative composition. In particular, the characterizing laws of $\sim_{\mathrm{MB,ot,mp}}$, which has been proposed and studied in [11], formalize the usual idempotency of the alternative composition operator for action execution, the race policy for time passing, and maximal progress:
$$\begin{array}{rcl} a \,.\, Q + a \,.\, Q & \sim_{\mathrm{MB,ot,mp}} & a \,.\, Q \\ (\lambda_1) \,.\, Q + (\lambda_2) \,.\, Q & \sim_{\mathrm{MB,ot,mp}} & (\lambda_1 + \lambda_2) \,.\, Q \\ \tau \,.\, Q + (\lambda) \,.\, Q' & \sim_{\mathrm{MB,ot,mp}} & \tau \,.\, Q \end{array}$$

## 4 Encoding ITMPC into OTMPC

In this section, we show that a connection can be established between Markovian process calculi with durational actions and Markovian process calculi with durationless actions. First, we single out the classes of process terms of ITMPC and OTMPC for which a translation is possible under eagerness, laziness, and maximal progress. Then, for each of the three interpretations of action execution, we formalize the encoding of the related class of process terms of ITMPC into the related class of process terms of OTMPC and we prove that it preserves the related bisimulation-based behavioral equivalence of the considered process terms.

### 4.1 Classes of Process Terms

In the deterministically timed case, the basic rule of the translating function defined in [7] maps $<a,n>.P$ to $a\,.\,(n)\,.\,Q$ or $(n)\,.\,a\,.\,Q$ depending on whether the starting time – first option – or the completion time – second option – of the execution of timed actions is observed, respectively, where $n$ is a fixed duration and $Q$ is the translation of $P$. As a consequence, a process term like $<a_1,n_1>.P_1 + <a_2,n_2>.P_2$ is mapped to $a_1\,.\,(n_1)\,.\,Q_1 + a_2\,.\,(n_2)\,.\,Q_2$ or $(n_1)\,.\,a_1\,.\,Q_1 + (n_2)\,.\,a_2\,.\,Q_2$, with the choice being nondeterministic in all the three terms as time does not solve choices in this setting.

On the basis of the observation made in Sect. 1, the first option would not work in a stochastically timed setting. In fact, the choice in a process term like $<a_1,\lambda_1>.P_1 + <a_2,\lambda_2>.P_2$ is probabilistic, whereas the choice in the corresponding process term $a_1\,.\,(\lambda_1)\,.\,Q_1 + a_2\,.\,(\lambda_2)\,.\,Q_2$ would be nondeterministic. As a consequence, the basic rule of the translating function from ITMPC to OTMPC should map $<a,\lambda>.P$ to $(\lambda)\,.\,a\,.\,Q$ – with $Q$ being the translation of $P$ – so that a process term like $<a_1,\lambda_1>.P_1 + <a_2,\lambda_2>.P_2$ is mapped to $(\lambda_1)\,.\,a_1\,.\,Q_1 + (\lambda_2)\,.\,a_2\,.\,Q_2$ – with the choice being probabilistic in both terms. In other words, ITMPC process terms can be translated only into OTMPC process terms that do not contain nondeterministic choices. In the following, we denote by $\mathbb{P}_{\mathrm{M,ot,nnd}}$ the set of process terms of $\mathbb{P}_{\mathrm{M,ot}}$ with no nondeterministic choices.

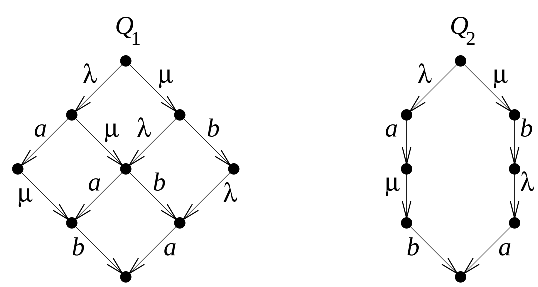
Selecting the appropriate order for action execution and time passing is not enough to achieve an encoding that preserves the bisimulation-based behavioral equivalence of process terms. In fact, consider the following ITMPC process terms:

$$P_1 \;\equiv\; <a,\lambda>.\underline{0} \,\|_{\emptyset}\, <b,\mu>.\underline{0}$$
$$P_2 \;\equiv\; <a,\lambda>.<b,\mu>.\underline{0} + <b,\mu>.<a,\lambda>.\underline{0}$$

and the corresponding OTMPC process terms:

$$Q_1 \;\equiv\; (\lambda)\,.\,a\,.\,\underline{0} \,\|_{\emptyset}\, (\mu)\,.\,b\,.\,\underline{0}$$
$$Q_2 \;\equiv\; (\lambda)\,.\,a\,.\,(\mu)\,.\,b\,.\,\underline{0} + (\mu)\,.\,b\,.\,(\lambda)\,.\,a\,.\,\underline{0}$$

It turns out that $P_1 \sim_{\mathrm{MB,it}} P_2$ because their underlying labeled multitransition systems are isomorphic. By contrast, $Q_1 \not\sim_{\mathrm{MB,ot,l}} Q_2$ because $[\![Q_1]\!]_{\mathrm{M,ot}}$ contains states having both action transitions and time transitions due to interleaving, whereas this is not the case with $[\![Q_2]\!]_{\mathrm{M,ot}}$ as can be seen below:




This shows that a translation of ITMPC into OTMPC is possible under laziness only for sequential process terms, i.e., process terms that do not contain any occurrence of the parallel composition operator. In the following, we denote by $\mathbb{P}_{\mathrm{M,it,seq}}$ (resp. $\mathbb{P}_{\mathrm{M,ot,nnd,seq}}$) the set of sequential process terms of $\mathbb{P}_{\mathrm{M,it}}$ (resp. $\mathbb{P}_{\mathrm{M,ot,nnd}}$).

On the other hand, we have $Q_1 \sim_{\mathrm{MB,ot,e}} Q_2$ because under eagerness action execution always takes precedence over time passing, so that the central state of $[\![Q_1]\!]_{\mathrm{M,ot}}$ and its incoming transitions can be ignored when checking for orthogonal-time Markovian bisimilarity. Similarly, we have $Q_1 \sim_{\mathrm{MB,ot,mp}} Q_2$ whenever $a = \tau = b$. Should this not be the case, it would be enough to add a $\tau$-selfloop to every state of $[\![Q_1]\!]_{\mathrm{M,ot}}$ enabling an action. In other words, under maximal progress the basic rule of the translating function from ITMPC to OTMPC should map $<a,\lambda>.P$ to $(\lambda)\,.\,\mathrm{rec}\,Z : (\tau\,.\,Z + a\,.\,Q)$, where $Q$ is the translation of $P$ and $Z$ does not occur free in $Q$. Note that by doing so we reintroduce nondeterministic choices in a controlled way. In the following, we denote by $\mathbb{P}_{\rm M,ot,cnd}$ the set of process terms of $\mathbb{P}_{\rm M,ot}$ with controlled nondeterministic choices.

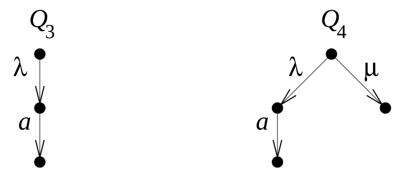
We conclude by showing another issue related to the preservation of the bisimulation-based behavioral equivalence of process terms. Consider the following ITMPC process terms:

$$P_3 \equiv <a,\lambda>.\underline{0}$$
$$P_4 \equiv <a,\lambda>.\underline{0} + <b,\mu>.\underline{0} \parallel_{\{b\}} \underline{0}$$

and the corresponding OTMPC process terms:

$$Q_3 \equiv (\lambda).a.\underline{0}$$
$$Q_4 \equiv (\lambda).a.\underline{0} + (\mu).b.\underline{0} \parallel_{\{b\}} \underline{0}$$

It turns out that $P_3 \sim_{\rm MB,it} P_4$ because their underlying labeled multitransition systems are isomorphic. By contrast, $Q_3 \not\sim_{\rm MB,ot,e} Q_4$ and $Q_3 \not\sim_{\rm MB,ot,mp} Q_4$ as can be seen from their underlying labeled multitransition systems shown below:




Here, the problem is that $[\![Q_4]\!]_{\rm M,ot}$ has a spurious deadlock state deriving from the need of encoding every exponentially timed action as its rate followed by its name. This problem can only arise in the presence of restrictions on the actions that can be executed. According to the syntax of ITMPC, this can only happen in the presence of occurrences of the parallel composition operator whose synchronization set is not empty. Therefore, a translation of ITMPC into OTMPC is possible under eagerness and maximal progress only for synchronization-free process terms. In the following, we denote by $\mathbb{P}_{\rm M,it,sf}$ (resp. $\mathbb{P}_{\rm M,ot,nnd,sf}/\mathbb{P}_{\rm M,ot,cnd,sf}$) the set of synchronization-free process terms of $\mathbb{P}_{\rm M,it}$ (resp. $\mathbb{P}_{\rm M,ot,nnd}/\mathbb{P}_{\rm M,ot,cnd}$).

## 4.2 Translating Function for Laziness

The function $\Gamma_{\rm l} : \mathbb{P}_{\rm M,it,seq} \rightarrow \mathbb{P}_{\rm M,ot,nnd,seq}$ encoding ITMPC into OTMPC under laziness is defined by structural induction as follows:

$$\begin{array}{rcl}
\Gamma_{\rm l}[\![\underline{0}]\!] & = & \underline{0} \\
\Gamma_{\rm l}[\![<a,\lambda>.P]\!] & = & (\lambda).a.\Gamma_{\rm l}[\![P]\!] \\
\Gamma_{\rm l}[\![P_1 + P_2]\!] & = & \Gamma_{\rm l}[\![P_1]\!] + \Gamma_{\rm l}[\![P_2]\!] \\
\Gamma_{\rm l}[\![P/H]\!] & = & \Gamma_{\rm l}[\![P]\!]/H \\
\Gamma_{\rm l}[\![P[\varphi]]\!] & = & \Gamma_{\rm l}[\![P]\!][\varphi] \\
\Gamma_{\rm l}[\![X]\!] & = & X \\
\Gamma_{\rm l}[\![{\rm rec}\, X : P]\!] & = & {\rm rec}\, X : \Gamma_{\rm l}[\![P]\!]
\end{array}$$

We now prove that $\Gamma_{\rm l}$ preserves the bisimulation-based behavioral equivalence of the considered process terms by first demonstrating some useful properties of $\Gamma_{\rm l}$, among which the fact that every ITMPC sequential process term and its $\Gamma_{\rm l}$-translation into OTMPC possess the same total exit rate.

**Lemma 4.1** Let $P \in \mathscr{PL}_{\rm M,it,seq}$, ${\rm rec}\, X : \hat{P} \in \mathbb{P}_{\rm M,it,seq}$, and $Y \in Var$. Then:

$$\Gamma_{\rm l}[\![P\{{\rm rec}\, X : \hat{P} \hookrightarrow Y\}]\!] = \Gamma_{\rm l}[\![P]\!]\{{\rm rec}\, X : \Gamma_{\rm l}[\![\hat{P}]\!] \hookrightarrow Y\}$$ ■

**Lemma 4.2** Let $P \in \mathbb{P}_{\rm M,it,seq}$. Then $\Gamma_{\rm l}[\![P]\!]$ cannot perform any action and:

$$rate_{\rm it,t}(P) = rate_{\rm ot,t}(\Gamma_{\rm l}[\![P]\!])$$ ■

**Lemma 4.3** Let $P \in \mathbb{P}_{\mathrm{M,it,seq}}$. Then $P \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'$ iff $\Gamma_{\mathrm{l}}[\![P]\!] \xrightarrow{\lambda}_{\mathrm{M}} Q$ with the only transition of $Q \in \mathbb{P}_{\mathrm{M,ot,nnd,seq}}$ being $Q \xrightarrow{a} \Gamma_{\mathrm{l}}[\![P']\!]$. ■

**Theorem 4.4** Let $P_1, P_2 \in \mathbb{P}_{\mathrm{M,it,seq}}$. Then:

$$P_1 \sim_{\mathrm{MB,it}} P_2 \iff \Gamma_{\mathrm{l}}[\![P_1]\!] \sim_{\mathrm{MB,ot,l}} \Gamma_{\mathrm{l}}[\![P_2]\!]$$ ■

## 4.3 Translating Function for Eagerness

The function $\Gamma_{\mathrm{e}} : \mathbb{P}_{\mathrm{M,it,sf}} \to \mathbb{P}_{\mathrm{M,ot,nnd,sf}}$ encoding ITMPC into OTMPC under eagerness is defined by structural induction as follows:

$$\begin{array}{rcl}
\Gamma_{\mathrm{e}}[\![\underline{0}]\!] & = & \underline{0} \\
\Gamma_{\mathrm{e}}[\![<a,\lambda>.P]\!] & = & (\lambda).a.\Gamma_{\mathrm{e}}[\![P]\!] \\
\Gamma_{\mathrm{e}}[\![P_1 + P_2]\!] & = & \Gamma_{\mathrm{e}}[\![P_1]\!] + \Gamma_{\mathrm{e}}[\![P_2]\!] \\
\Gamma_{\mathrm{e}}[\![P_1 \parallel_{\emptyset} P_2]\!] & = & \Gamma_{\mathrm{e}}[\![P_1]\!] \parallel_{\emptyset} \Gamma_{\mathrm{e}}[\![P_2]\!] \\
\Gamma_{\mathrm{e}}[\![P/H]\!] & = & \Gamma_{\mathrm{e}}[\![P]\!]/H \\
\Gamma_{\mathrm{e}}[\![P[\varphi]]\!] & = & \Gamma_{\mathrm{e}}[\![P]\!][\varphi] \\
\Gamma_{\mathrm{e}}[\![X]\!] & = & X \\
\Gamma_{\mathrm{e}}[\![\mathrm{rec}\,X : P]\!] & = & \mathrm{rec}\,X : \Gamma_{\mathrm{e}}[\![P]\!]
\end{array}$$

where the only difference with respect to $\Gamma_{\mathrm{l}}$ is the presence of a clause for parallel composition.

**Lemma 4.5** Let $P \in \mathscr{PL}_{\mathrm{M,it,sf}}$, $\mathrm{rec}\,X : \hat{P} \in \mathbb{P}_{\mathrm{M,it,sf}}$, and $Y \in \mathit{Var}$. Then:

$$\Gamma_{\mathrm{e}}[\![P\{\mathrm{rec}\,X : \hat{P} \hookrightarrow Y\}]\!] = \Gamma_{\mathrm{e}}[\![P]\!]\{\mathrm{rec}\,X : \Gamma_{\mathrm{e}}[\![\hat{P}]\!] \hookrightarrow Y\}$$ ■

**Lemma 4.6** Let $P \in \mathbb{P}_{\mathrm{M,it,sf}}$. Then $\Gamma_{\mathrm{e}}[\![P]\!]$ cannot perform any action and:

$$\mathit{rate}_{\mathrm{it,t}}(P) = \mathit{rate}_{\mathrm{ot,t}}(\Gamma_{\mathrm{e}}[\![P]\!])$$ ■

**Lemma 4.7** Let $P \in \mathbb{P}_{\mathrm{M,it,sf}}$. Then $P \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'$ iff $\Gamma_{\mathrm{e}}[\![P]\!] \xrightarrow{\lambda}_{\mathrm{M}} Q$ with the only action transition of $Q \in \mathbb{P}_{\mathrm{M,ot,nnd,sf}}$ being $Q \xrightarrow{a} \Gamma_{\mathrm{e}}[\![P']\!]$. ■

**Theorem 4.8** Let $P_1, P_2 \in \mathbb{P}_{\mathrm{M,it,sf}}$. Then:

$$P_1 \sim_{\mathrm{MB,it}} P_2 \iff \Gamma_{\mathrm{e}}[\![P_1]\!] \sim_{\mathrm{MB,ot,e}} \Gamma_{\mathrm{e}}[\![P_2]\!]$$ ■

## 4.4 Translating Function for Maximal Progress

The function $\Gamma_{\mathrm{mp}} : \mathbb{P}_{\mathrm{M,it,sf}} \to \mathbb{P}_{\mathrm{M,ot,cnd,sf}}$ encoding ITMPC into OTMPC under maximal progress is defined by structural induction as follows:

$$\begin{array}{rcll}
\Gamma_{\mathrm{mp}}[\![\underline{0}]\!] & = & \underline{0} & \\
\Gamma_{\mathrm{mp}}[\![<a,\lambda>.P]\!] & = & (\lambda).\mathrm{rec}\,Z : (\tau.Z + a.\Gamma_{\mathrm{mp}}[\![P]\!]) & Z \text{ not free in } P \\
\Gamma_{\mathrm{mp}}[\![P_1 + P_2]\!] & = & \Gamma_{\mathrm{mp}}[\![P_1]\!] + \Gamma_{\mathrm{mp}}[\![P_2]\!] & \\
\Gamma_{\mathrm{mp}}[\![P_1 \parallel_{\emptyset} P_2]\!] & = & \Gamma_{\mathrm{mp}}[\![P_1]\!] \parallel_{\emptyset} \Gamma_{\mathrm{mp}}[\![P_2]\!] & \\
\Gamma_{\mathrm{mp}}[\![P/H]\!] & = & \Gamma_{\mathrm{mp}}[\![P]\!]/H & \\
\Gamma_{\mathrm{mp}}[\![P[\varphi]]\!] & = & \Gamma_{\mathrm{mp}}[\![P]\!][\varphi] & \\
\Gamma_{\mathrm{mp}}[\![X]\!] & = & X & \\
\Gamma_{\mathrm{mp}}[\![\mathrm{rec}\,X : P]\!] & = & \mathrm{rec}\,X : \Gamma_{\mathrm{mp}}[\![P]\!] &
\end{array}$$

where the only difference with respect to $\Gamma_{\mathrm{e}}$ is the clause for action prefix, which introduces $\tau$-selfloops.

**Lemma 4.9** Let $P \in \mathscr{PL}_{\mathrm{M,it,sf}}$, $\mathrm{rec}\,X : \hat{P} \in \mathbb{P}_{\mathrm{M,it,sf}}$, and $Y \in \mathit{Var}$. Then:
$$\Gamma_{\mathrm{mp}}[\![P\{\mathrm{rec}\,X : \hat{P} \hookrightarrow Y\}]\!] = \Gamma_{\mathrm{mp}}[\![P]\!]\{\mathrm{rec}\,X : \Gamma_{\mathrm{mp}}[\![\hat{P}]\!] \hookrightarrow Y\} \qquad \blacksquare$$

**Lemma 4.10** Let $P \in \mathbb{P}_{\mathrm{M,it,sf}}$. Then $\Gamma_{\mathrm{mp}}[\![P]\!]$ cannot perform any action and:
$$\mathit{rate}_{\mathrm{it,t}}(P) = \mathit{rate}_{\mathrm{ot,t}}(\Gamma_{\mathrm{mp}}[\![P]\!]) \qquad \blacksquare$$

**Lemma 4.11** Let $P \in \mathbb{P}_{\mathrm{M,it,sf}}$. Then $P \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'$ iff $\Gamma_{\mathrm{mp}}[\![P]\!] \xrightarrow{\lambda}_{\mathrm{M}} Q$ with the only action transitions of $Q \in \mathbb{P}_{\mathrm{M,ot,cnd,sf}}$ being $Q \xrightarrow{\tau} Q$ and $Q \xrightarrow{a} \Gamma_{\mathrm{mp}}[\![P']\!]$. $\blacksquare$

**Theorem 4.12** Let $P_1, P_2 \in \mathbb{P}_{\mathrm{M,it,sf}}$. Then:
$$P_1 \sim_{\mathrm{MB,it}} P_2 \iff \Gamma_{\mathrm{mp}}[\![P_1]\!] \sim_{\mathrm{MB,ot,mp}} \Gamma_{\mathrm{mp}}[\![P_2]\!] \qquad \blacksquare$$

## 5 Conclusion

In this paper, we have shown that durational actions and durationless actions are not irreconcilable even in a stochastically timed setting, because we have exhibited suitable semantics-preserving mappings from an integrated-time Markovian process calculus to an orthogonal-time Markovian under eagerness, laziness, and maximal progress. The restrictions on the three mappings emphasize synchronization disciplines and choice resolutions as the only features distinguishing between the two considered calculi.

We have also highlighted a number of differences with respect to the deterministically timed setting examined in [7]. Firstly, due to the adoption of the race policy, time solves choices and hence any exponentially timed action must be translated into an exponentially distributed delay followed by an instantaneous action, rather than the opposite. Secondly, in the integrated-time case the memoryless property of exponential distributions blurs the distinction among eagerness, laziness, and maximal progress. Thirdly, since time solve choices, in the orthogonal-time case the three interpretations of action execution must be formalized through as many variants of the behavioral equivalence, rather than in the operational semantic rules. Fourthly, the mapping for laziness is limited to sequential process terms, rather than being applicable in general. Sixtly, the mapping for maximal progress is limited to synchronization-free process terms and needs the introduction of $\tau$-selfloops, rather than being applicable in general. Seventhly, the three mappings constrain the amount of nondeterminism in the resulting process terms, rather than admitting full nondeterminism.

Orthogonal-time Markovian process calculi turn out to be more expressive as they can represent both probabilistic and nondeterministic choices as well as more natural forms of synchronization. Nevertheless, integrated-time Markovian process calculi should not be neglected. Firstly, they are in general more appropriate for modeling purposes, because it is more natural to think of an action as having a duration rather than expressing a delay followed by an action name. Secondly, unlike orthogonal-time Markovian process calculi they do not incur in spurious deadlock states. Thirdly, they tend to produce system descriptions with no more than half of the states that would result from descriptions of the same systems expressed in orthogonal-time Markovian process calculi.

**Acknowledgment**: This work has been funded by MIUR-PRIN project *PaCo – Performability-Aware Computing: Logics, Models, and Languages*.

## References

[1] L. Aceto and D. Murphy, *“Timing and Causality in Process Algebra”*, in Acta Informatica 33:317–350, 1996.

[2] A. Aldini, M. Bernardo, and F. Corradini, *“A Process Algebraic Approach to Software Architecture Design”*, Springer, 2010.

[3] J.C.M. Baeten and J.A. Bergstra, *“Real Time Process Algebra”*, in Formal Aspects of Computing 3:142–188, 1991.

[4] M. Bernardo and M. Bravetti, *“Performance Measure Sensitive Congruences for Markovian Process Algebras”*, in Theoretical Computer Science 290:117–160, 2003.

[5] T. Bolognesi and F. Lucidi, *“LOTOS-Like Process Algebras with Urgent or Timed Interactions”*, in Proc. of the *4th Int. Conf. on Formal Description Techniques for Distributed Systems and Communication Protocols (FORTE 1991)*, IFIP Trans. C-2:249–264, 1991.

[6] P. Buchholz, *“Markovian Process Algebra: Composition and Equivalence”*, in Proc. of the *2nd Int. Workshop on Process Algebra and Performance Modelling (PAPM 1994)*, pp. 11–30, 1994.

[7] F. Corradini, *“Absolute Versus Relative Time in Process Algebras”*, in Information and Computation 156:122–172, 2000.

[8] F. Corradini, W. Vogler, and L. Jenner, *“Comparing the Worst-Case Efficiency of Asynchronous Systems with PAFAS”*, in Acta Informatica 38:735–792, 2002.

[9] 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)*, Springer, LNCS 729:121–146, 1993.

[10] M. Hennessy and T. Regan, *“A Process Algebra for Timed Systems”*, in Information and Computation 117:221–239, 1995.

[11] H. Hermanns, *“Interactive Markov Chains”*, Springer, LNCS 2428, 2002.

[12] 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)*, pp. 71–87, 1994.

[13] J. Hillston, *“A Compositional Approach to Performance Modelling”*, Cambridge University Press, 1996.

[14] C.A.R. Hoare, *“Communicating Sequential Processes”*, Prentice Hall, 1985.

[15] R. Milner, *“Communication and Concurrency”*, Prentice Hall, 1989.

[16] F. Moller and C. Tofts, *“A Temporal Calculus of Communicating Systems”*, in Proc. of the *1st Int. Conf. on Concurrency Theory (CONCUR 1990)*, Springer, LNCS 458:401–415, 1990.

[17] X. Nicollin and J. Sifakis, *“An Overview and Synthesis on Timed Process Algebras”*, in Proc. of the *REX Workshop on Real Time: Theory in Practice*, Springer, LNCS 600:526–548, 1991.

[18] X. Nicollin and J. Sifakis, *“The Algebra of Timed Processes ATP: Theory and Application”*, in Information and Computation 114:131–178, 1994.

[19] C. Priami, *“Stochastic $\pi$-Calculus”*, in Computer Journal 38:578–589, 1995.

[20] J. Quemada, D. de Frutos, and A. Azcorra, *“TIC: A Timed Calculus”*, in Formal Aspects of Computing 5:224–252, 1993.

[21] G.M. Reed and A.W. Roscoe, *“A Timed Model for Communicating Sequential Processes”*, in Theoretical Computer Science 58:249–261, 1988.

[22] 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)*, Springer, LNCS 2761:189–203, 2003.

[23] I. Ulidowski and S. Yuen, *“Extending Process Languages with Time”*, in Proc. of the *6th Int. Conf. on Algebraic Methodology and Software Technology (AMAST 1997)*, Springer, LNCS 1349:524–538, 1997.

[24] W. Yi, *“CCS + Time = An Interleaving Model for Real Time Systems”*, in Proc. of the *18th Int. Coll. on Automata, Languages and Programming (ICALP 1991)*, Springer, LNCS 510:217–228, 1991.

## Appendix: Proofs of Results of Sects. 4.2, 4.3, and 4.4

**Proof of Lemma 4.1.** We proceed by induction on the syntactical structure of $P \in \mathscr{PL}_{\mathrm{M,it,seq}}$:

- If $P \equiv \underline{0}$ or $P \in Var - \{Y\}$, then:
$$\Gamma_1[\![P\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] = P = \Gamma_1[\![P]\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}$$

- If $P \equiv Y$, then:
$$\Gamma_1[\![P\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] = \Gamma_1[\![\mathrm{rec}X : \hat{P}]\!] = \mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] = \Gamma_1[\![P]\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}$$

- Let $P \equiv <a,\lambda>.P'$ and assume that $\Gamma_1[\![P'\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] = \Gamma_1[\![P']\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}$. Then:
$$\begin{array}{rcll}\Gamma_1[\![P\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] & = & \Gamma_1[\![<a,\lambda>.(P'\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\})]\!] & = \\ & = & (\lambda)\,.\,a\,.\,\Gamma_1[\![P'\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] & = \\ & = & (\lambda)\,.\,a\,.\,(\Gamma_1[\![P']\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}) & = \\ & = & ((\lambda)\,.\,a\,.\,\Gamma_1[\![P']\!])\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\} & = \ \Gamma_1[\![P]\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}\end{array}$$

- Let $P \equiv P_1 + P_2$ and for $k \in \{1,2\}$ assume that $\Gamma_1[\![P_k\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] = \Gamma_1[\![P_k]\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}$. Then:
$$\begin{array}{rcll}\Gamma_1[\![P\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] & = & \Gamma_1[\![P_1\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\} + P_2\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] & = \\ & = & \Gamma_1[\![P_1\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] + \Gamma_1[\![P_2\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] & = \\ & = & \Gamma_1[\![P_1]\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\} + \Gamma_1[\![P_2]\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\} & = \\ & = & (\Gamma_1[\![P_1]\!] + \Gamma_1[\![P_2]\!])\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\} & = \ \Gamma_1[\![P]\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}\end{array}$$

- Let $P \equiv P'/H$ and assume that $\Gamma_1[\![P'\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] = \Gamma_1[\![P']\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}$. Then:
$$\begin{array}{rcll}\Gamma_1[\![P\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] & = & \Gamma_1[\![P'\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}/H]\!] & = \\ & = & \Gamma_1[\![P'\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!]/H & = \\ & = & \Gamma_1[\![P']\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}/H & = \\ & = & (\Gamma_1[\![P']\!]/H)\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\} & = \ \Gamma_1[\![P]\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}\end{array}$$

- The case $P \equiv P'[\varphi]$ is similar to the previous one.
- Let $P \equiv \mathrm{rec}X' : P'$ and assume that $\Gamma_1[\![P'\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] = \Gamma_1[\![P']\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}$. There are two cases.
If $X' \equiv Y$, then:
$$\Gamma_1[\![P\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] = \Gamma_1[\![P]\!] = \Gamma_1[\![P]\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}$$
If $X' \not\equiv Y$, then:
$$\begin{array}{rcll}\Gamma_1[\![P\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] & = & \Gamma_1[\![\mathrm{rec}X' : (P'\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\})]\!] & = \\ & = & \mathrm{rec}X' : \Gamma_1[\![P'\{\mathrm{rec}X : \hat{P} \hookrightarrow Y\}]\!] & = \\ & = & \mathrm{rec}X' : (\Gamma_1[\![P']\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}) & = \\ & = & (\mathrm{rec}X' : \Gamma_1[\![P']\!])\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\} & = \ \Gamma_1[\![P]\!]\{\mathrm{rec}X : \Gamma_1[\![\hat{P}]\!] \hookrightarrow Y\}\end{array}$$ ■

**Proof of Lemma 4.2.** We proceed by induction on the syntactical structure of $P \in \mathbb{P}_{\mathrm{M,it,seq}}$:

- If $P \equiv \underline{0}$, then $\Gamma_1[\![P]\!] = \underline{0}$ and hence cannot perform any action. Moreover:
$$rate_{\mathrm{it,t}}(P) = 0 = rate_{\mathrm{ot,t}}(\Gamma_1[\![P]\!])$$

- If $P \equiv <a,\lambda>.P'$, then $\Gamma_1[\![P]\!] = (\lambda)\,.\,a\,.\,\Gamma_1[\![P']\!]$ and hence cannot perform any action. Moreover:
$$rate_{\mathrm{it,t}}(P) = \lambda = rate_{\mathrm{ot,t}}(\Gamma_1[\![P]\!])$$

- Let $P \equiv P_1 + P_2$ and for $k \in \{1,2\}$ assume that $\Gamma_1[\![P_k]\!]$ cannot perform any action and $rate_{\mathrm{it,t}}(P_k) = rate_{\mathrm{ot,t}}(\Gamma_1[\![P_k]\!])$. Then $\Gamma_1[\![P]\!] = \Gamma_1[\![P_1]\!] + \Gamma_1[\![P_2]\!]$ cannot perform any action and:
$$rate_{\mathrm{it,t}}(P) = rate_{\mathrm{it,t}}(P_1) + rate_{\mathrm{it,t}}(P_2) = rate_{\mathrm{ot,t}}(\Gamma_1[\![P_1]\!]) + rate_{\mathrm{ot,t}}(\Gamma_1[\![P_2]\!]) = rate_{\mathrm{ot,t}}(\Gamma_1[\![P]\!])$$

- Let $P \equiv P'/H$ and assume that $\Gamma_l[\![P']\!]$ cannot perform any action and $rate_{\mathrm{it,t}}(P') = rate_{\mathrm{ot,t}}(\Gamma_l[\![P']\!])$. Then $\Gamma_l[\![P]\!] = \Gamma_l[\![P']\!]/H$ cannot perform any action and:
$$rate_{\mathrm{it,t}}(P) \,=\, rate_{\mathrm{it,t}}(P') \,=\, rate_{\mathrm{ot,t}}(\Gamma_l[\![P']\!]) \,=\, rate_{\mathrm{ot,t}}(\Gamma_l[\![P]\!])$$
- The case $P \equiv P'[\varphi]$ is similar to the previous one.
- Let $P \equiv \mathrm{rec}\,X : P'$ and assume that $\Gamma_l[\![P'\{\mathrm{rec}\,X : P' \hookrightarrow X\}]\!]$ cannot perform any action and $rate_{\mathrm{it,t}}(P'\{\mathrm{rec}\,X : P' \hookrightarrow X\}) = rate_{\mathrm{ot,t}}(\Gamma_l[\![P'\{\mathrm{rec}\,X : P' \hookrightarrow X\}]\!])$. Then $\Gamma_l[\![P]\!] = \mathrm{rec}\,X : \Gamma_l[\![P']\!]$ cannot perform any action because $\Gamma_l[\![P']\!]\{\mathrm{rec}\,X : \Gamma_l[\![P']\!] \hookrightarrow X\} = \Gamma_l[\![P'\{\mathrm{rec}\,X : P' \hookrightarrow X\}]\!]$ by virtue of Lemma 4.1. Moreover:
$$\begin{array}{rcll} rate_{\mathrm{it,t}}(P) & = & rate_{\mathrm{it,t}}(P'\{\mathrm{rec}\,X : P' \hookrightarrow X\}) & = \\ & = & rate_{\mathrm{ot,t}}(\Gamma_l[\![P'\{\mathrm{rec}\,X : P' \hookrightarrow X\}]\!]) & = \\ & = & rate_{\mathrm{ot,t}}(\Gamma_l[\![P']\!]\{\mathrm{rec}\,X : \Gamma_l[\![P']\!] \hookrightarrow X\}) & = \; rate_{\mathrm{ot,t}}(\Gamma_l[\![P]\!]) \end{array}$$ ■

**Proof of Lemma 4.3.** Given $P \in \mathbb{P}_{\mathrm{M,it,seq}}$, the proof is divided into two parts:

$\Rightarrow$ Assuming $P \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'$, we prove that $\Gamma_l[\![P]\!] \xrightarrow{\lambda}_{\mathrm{M}} Q$ with the only transition of $Q \in \mathbb{P}_{\mathrm{M,ot,nnd,seq}}$ being $Q \xrightarrow{a} \Gamma_l[\![P']\!]$ by proceeding by induction on the length of the derivation of $P \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'$, intended as the number of operational semantic rules of Table 1 that have been applied in order to derive the considered transition:

- If the length is 1, then it must be $P \equiv\, <a,\lambda>.P'$ and hence $\Gamma_l[\![P]\!] = (\lambda)\,.\,a\,.\,\Gamma_l[\![P']\!]$. Therefore $\Gamma_l[\![P]\!] \xrightarrow{\lambda}_{\mathrm{M}} a\,.\,\Gamma_l[\![P']\!]$ with the only transition of the reached process term being $a\,.\,\Gamma_l[\![P']\!] \xrightarrow{a} \Gamma_l[\![P']\!]$.
- Let the length be $n > 1$ and assume that the result holds for every transition derivable by applying less than $n$ operational semantic rules of Table 1. There are several cases based on the syntactical structure of $P$:
  - If $P \equiv P_1 + P_2$, then the transition derives from the fact that $P_k \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'$ for some $k \in \{1,2\}$. From the induction hypothesis, it follows that $\Gamma_l[\![P_k]\!] \xrightarrow{\lambda}_{\mathrm{M}} Q$ with the only transition of $Q \in \mathbb{P}_{\mathrm{M,ot,nnd,seq}}$ being $Q \xrightarrow{a} \Gamma_l[\![P']\!]$, and hence $\Gamma_l[\![P]\!] = \Gamma_l[\![P_1]\!] + \Gamma_l[\![P_2]\!] \xrightarrow{\lambda}_{\mathrm{M}} Q$ with the only transition of $Q$ being $Q \xrightarrow{a} \Gamma_l[\![P']\!]$.
  - If $P \equiv \bar{P}/H$, then the transition derives from the fact that $\bar{P} \xrightarrow{b,\lambda}_{\mathrm{M,it}} \bar{P'}$ with $P' \equiv \bar{P'}/H$ and $b \in H \cup \{\tau\}$ if $a = \tau$, $b = a$ otherwise. From the induction hypothesis, it follows that $\Gamma_l[\![\bar{P}]\!] \xrightarrow{\lambda}_{\mathrm{M}} \bar{Q}$ with the only transition of $\bar{Q} \in \mathbb{P}_{\mathrm{M,ot,nnd,sf}}$ being $\bar{Q} \xrightarrow{b} \Gamma_l[\![\bar{P'}]\!]$. Thus $\Gamma_l[\![P]\!] = \Gamma_l[\![\bar{P}]\!]/H \xrightarrow{\lambda}_{\mathrm{M}} \bar{Q}/H$ with the only transition of $\bar{Q}/H$ being $\bar{Q}/H \xrightarrow{a} \Gamma_l[\![\bar{P'}]\!]/H = \Gamma_l[\![P']\!]$.
  - The case $P \equiv \bar{P}[\varphi]$ is similar to the previous one with $\varphi(b) = a$.
  - If $P \equiv \mathrm{rec}\,X : \bar{P}$, then the transition derives from the fact that $\bar{P}\{\mathrm{rec}\,X : \bar{P} \hookrightarrow X\} \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'$. From the induction hypothesis, it follows that $\Gamma_l[\![\bar{P}\{\mathrm{rec}\,X : \bar{P} \hookrightarrow X\}]\!] \xrightarrow{\lambda}_{\mathrm{M}} Q$ with the only transition of $Q \in \mathbb{P}_{\mathrm{M,ot,nnd,sf}}$ being $Q \xrightarrow{a} \Gamma_l[\![P']\!]$, and hence $\Gamma_l[\![P]\!] = \mathrm{rec}\,X : \Gamma_l[\![\bar{P}]\!] \xrightarrow{\lambda}_{\mathrm{M}} Q$ – with the only transition of $Q$ being $Q \xrightarrow{a} \Gamma_l[\![P']\!]$ – because $\Gamma_l[\![\bar{P}]\!]\{\mathrm{rec}\,X : \Gamma_l[\![\bar{P}]\!] \hookrightarrow X\} = \Gamma_l[\![\bar{P}\{\mathrm{rec}\,X : \bar{P} \hookrightarrow X\}]\!]$ by virtue of Lemma 4.1.

⇐ Assuming $\Gamma_{\rm l}[\![P]\!] \xrightarrow{\lambda}_{\rm M} Q$ with the only transition of $Q \in \mathbb{P}_{\rm M,ot,nnd,seq}$ being $Q \xrightarrow{a} \Gamma_{\rm l}[\![P']\!]$, the proof that $P \xrightarrow{a,\lambda}_{\rm M,it} P'$ is similar, in the sense that it proceeds by induction on the number of operational semantic rules of Table 2 applied in order to derive $\Gamma_{\rm l}[\![P]\!] \xrightarrow{\lambda}_{\rm M} Q$ and performs a case analysis of the syntactical structure of $\Gamma_{\rm l}[\![P]\!]$. ■

**Proof of Thm. 4.4.** The proof is divided into two parts:

⇒ Consider the relation $\mathcal{B}_{\rm ot,l} = \mathcal{B}_{\rm ot,l,1} \cup \mathcal{B}_{\rm ot,l,2} \cup \mathcal{B}_{\rm ot,l,3}$ over $\mathbb{P}_{\rm M,ot}$, where:

$$\begin{aligned}
\mathcal{B}_{\rm ot,l,1} &= \{(\Gamma_{\rm l}[\![P_1]\!], \Gamma_{\rm l}[\![P_2]\!]) \mid P_1, P_2 \in \mathbb{P}_{\rm M,it,seq} \wedge P_1 \sim_{\rm MB,it} P_2\} \\
\mathcal{B}_{\rm ot,l,2} &= \{(Q_1, Q_2) \mid Q_1, Q_2 \in \mathbb{P}_{\rm M,ot,nnd,seq} \text{ and the only transition of } Q_k,\ k \in \{1,2\}, \\
&\qquad \text{is } Q_k \xrightarrow{a} \Gamma_{\rm l}[\![P'_k]\!] \text{ with } P'_1 \sim_{\rm MB,it} P'_2\} \\
\mathcal{B}_{\rm ot,l,3} &= \{(Q,Q) \mid Q \notin \mathbb{P}_{\rm M,ot,nnd,seq}\}
\end{aligned}$$

This is an equivalence relation because so is $\sim_{\rm MB,it}$. Moreover, it turns out to be a lazy orthogonal-time Markovian bisimulation for the following reasons:

- If we take $(\Gamma_{\rm l}[\![P_1]\!], \Gamma_{\rm l}[\![P_2]\!]) \in \mathcal{B}_{\rm ot,l,1}$, then for $k \in \{1,2\}$ we have that $\Gamma_{\rm l}[\![P_k]\!]$ cannot perform any action by virtue of Lemma 4.2. Consider an equivalence class $D \in \mathbb{P}_{\rm M,ot}/\mathcal{B}_{\rm ot,l}$. There are two cases:
  - ∗ If $D$ is originated from $\mathcal{B}_{\rm ot,l,1}$ (or $\mathcal{B}_{\rm ot,l,3}$), then by virtue of Lemma 4.3:
  $$rate_{\rm ot}(\Gamma_{\rm l}[\![P_1]\!], D) \,=\, 0 \,=\, rate_{\rm ot}(\Gamma_{\rm l}[\![P_2]\!], D)$$
  - ∗ If $D$ is originated from $\mathcal{B}_{\rm ot,l,2}$ and is characterized by $a \in Name$ and $P \in \mathbb{P}_{\rm M,it,seq}$, then by virtue of Lemma 4.3 and $P_1 \sim_{\rm MB,it} P_2$:
  $$rate_{\rm ot}(\Gamma_{\rm l}[\![P_1]\!], D) = rate_{\rm it}(P_1, a, [P]_{\sim_{\rm MB,it}}) = rate_{\rm it}(P_2, a, [P]_{\sim_{\rm MB,it}}) = rate_{\rm ot}(\Gamma_{\rm l}[\![P_2]\!], D)$$
- If we take $(Q_1, Q_2) \in \mathcal{B}_{\rm ot,l,2}$, then for $k \in \{1,2\}$ it holds that $Q_k$ has no time transitions, so we have to compare only the two action transitions of $Q_1$ and $Q_2$:
  - ∗ When $Q_1 \xrightarrow{a} \Gamma_{\rm l}[\![P'_1]\!]$, then $Q_2$ can respond with $Q_2 \xrightarrow{a} \Gamma_{\rm l}[\![P'_2]\!]$ where $(\Gamma_{\rm l}[\![P'_1]\!], \Gamma_{\rm l}[\![P'_2]\!]) \in \mathcal{B}_{\rm ot,l,1} \subseteq \mathcal{B}_{\rm ot,l}$ because $P'_1 \sim_{\rm MB,it} P'_2$, and vice versa.

⇐ Consider the relation $\mathcal{B}_{\rm it,l} = \mathcal{B}_{\rm it,l,1} \cup \mathcal{B}_{\rm it,l,2}$ over $\mathbb{P}_{\rm M,it}$, where:

$$\begin{aligned}
\mathcal{B}_{\rm it,l,1} &= \{(P_1, P_2) \mid P_1, P_2 \in \mathbb{P}_{\rm M,it,seq} \wedge \Gamma_{\rm l}[\![P_1]\!] \sim_{\rm MB,ot,l} \Gamma_{\rm l}[\![P_2]\!]\} \\
\mathcal{B}_{\rm it,l,2} &= \{(P,P) \mid P \notin \mathbb{P}_{\rm M,it,seq}\}
\end{aligned}$$

This is an equivalence relation because so is $\sim_{\rm MB,ot,l}$. Moreover, it turns out to be an integrated-time Markovian bisimulation. In fact, if we take $(P_1, P_2) \in \mathcal{B}_{\rm it,l,1}$ and we consider $a \in Name$ and $D \in \mathbb{P}_{\rm M,it}/\mathcal{B}_{\rm it,l}$, then there are two cases:

- If $D$ is originated from $\mathcal{B}_{\rm it,l,2}$, then:
$$rate_{\rm it}(P_1, a, D) \,=\, 0 \,=\, rate_{\rm it}(P_2, a, D)$$
- If $D$ is originated from $\mathcal{B}_{\rm it,l,1}$ and is characterized by $\Gamma_{\rm l}[\![P]\!] \in \mathbb{P}_{\rm M,ot,nnd,seq}$, then by virtue of Lemma 4.3 and $\Gamma_{\rm l}[\![P_1]\!] \sim_{\rm MB,ot,l} \Gamma_{\rm l}[\![P_2]\!]$:
$$rate_{\rm it}(P_1, a, D) \,=\, rate_{\rm ot}(\Gamma_{\rm l}[\![P_1]\!], D_{a,P}) \,=\, rate_{\rm ot}(\Gamma_{\rm l}[\![P_2]\!], D_{a,P}) \,=\, rate_{\rm it}(P_2, a, D)$$
where $D_{a,P}$ is the equivalence class of $\sim_{\rm MB,ot,l}$ formed by all the process terms $Q$ whose only transition is $Q \xrightarrow{a} \Gamma_{\rm l}[\![P']\!]$ with $\Gamma_{\rm l}[\![P']\!] \in [\Gamma_{\rm l}[\![P]\!]]_{\sim_{\rm MB,ot,l}}$. ■

**Proof of Lemma 4.5.** Similar to the proof of Lemma 4.1, we proceed by induction on the syntactical structure of $P \in \mathscr{PL}_{\mathrm{M,it,sf}}$. The only new case is $P \equiv P_1 \,\|_{\emptyset}\, P_2$, which is treated in the same way as the case $P \equiv P_1 + P_2$. ■

**Proof of Lemma 4.6.** Similar to the proof of Lemma 4.2, we proceed by induction on the syntactical structure of $P \in \mathbb{P}_{\mathrm{M,it,sf}}$ by exploiting Lemma 4.5. The only new case is $P \equiv P_1 \,\|_{\emptyset}\, P_2$, which is treated in the same way as the case $P \equiv P_1 + P_2$ thanks to the emptiness of the synchronization set. ■

**Proof of Lemma 4.7.** Similar to the proof of Lemma 4.3, the proof is divided into two parts – with the first one exploiting Lemma 4.5 – in both of which we proceed by induction on the length of the transition derivation and we perform a case analysis of the syntactical structure of the process term associated with the state from which the transition departs. Unlike the proof of Lemma 4.3, the transition departing from $Q \in \mathbb{P}_{\mathrm{M,ot,nnd,sf}}$ is not the only transition of $Q$, but the only action transition of $Q$. The only new case with respect to the proof of Lemma 4.3 is the one for parallel composition, which is treated as follows in the first part of the proof:

- ∗ If $P \equiv P_1 \,\|_{\emptyset}\, P_2$, then the transition derives from the fact that $P_k \xrightarrow{a,\lambda}_{\mathrm{M,it}} P'_k$ for some $k \in \{1,2\}$. Assume $k = 1$, so that $P' \equiv P'_1 \,\|_{\emptyset}\, P_2$. From the induction hypothesis, it follows that $\Gamma_{\mathrm{e}}[\![P_1]\!] \xrightarrow{\lambda}_{\mathrm{M}} Q_1$ with the only action transition of $Q_1 \in \mathbb{P}_{\mathrm{M,ot,nnd,sf}}$ being $Q_1 \xrightarrow{a} \Gamma_{\mathrm{e}}[\![P'_1]\!]$. Thus $\Gamma_{\mathrm{e}}[\![P]\!] = \Gamma_{\mathrm{e}}[\![P_1]\!] \,\|_{\emptyset}\, \Gamma_{\mathrm{e}}[\![P_2]\!] \xrightarrow{\lambda}_{\mathrm{M}} Q_1 \,\|_{\emptyset}\, \Gamma_{\mathrm{e}}[\![P_2]\!]$ with the only action transition of $Q_1 \,\|_{\emptyset}\, \Gamma[\![P_2]\!]$ – which may have time transitions as well – being $Q_1 \,\|_{\emptyset}\, \Gamma_{\mathrm{e}}[\![P_2]\!] \xrightarrow{a} \Gamma_{\mathrm{e}}[\![P'_1]\!] \,\|_{\emptyset}\, \Gamma_{\mathrm{e}}[\![P_2]\!] = \Gamma_{\mathrm{e}}[\![P']\!]$ as $\Gamma_{\mathrm{e}}[\![P_2]\!]$ cannot perform any action by virtue of Lemma 4.6. ■

**Proof of Thm. 4.8.** The proof is divided into two parts:

⇒ Consider the relation $\mathscr{B}_{\mathrm{ot,e}} = \mathscr{B}_{\mathrm{ot,e,1}} \cup \mathscr{B}_{\mathrm{ot,e,2}} \cup \mathscr{B}_{\mathrm{ot,e,3}}$ over $\mathbb{P}_{\mathrm{M,ot}}$, where:

$$\begin{array}{rcl}
\mathscr{B}_{\mathrm{ot,e,1}} & = & \{(\Gamma_{\mathrm{e}}[\![P_1]\!], \Gamma_{\mathrm{e}}[\![P_2]\!]) \mid P_1, P_2 \in \mathbb{P}_{\mathrm{M,it,sf}} \wedge P_1 \sim_{\mathrm{MB,it}} P_2\} \\
\mathscr{B}_{\mathrm{ot,e,2}} & = & \{(Q_1, Q_2) \mid Q_1, Q_2 \in \mathbb{P}_{\mathrm{M,ot,nnd,sf}} \text{ and the only action transition of } Q_k,\ k \in \{1,2\}, \\
& & \quad \text{is } Q_k \xrightarrow{a} \Gamma_{\mathrm{e}}[\![P'_k]\!] \text{ with } P'_1 \sim_{\mathrm{MB,it}} P'_2\} \\
\mathscr{B}_{\mathrm{ot,e,3}} & = & \{(Q, Q) \mid Q \notin \mathbb{P}_{\mathrm{M,ot,nnd,sf}}\}
\end{array}$$

This is an equivalence relation because so is $\sim_{\mathrm{MB,it}}$. Moreover, it turns out to be an eager orthogonal-time Markovian bisimulation for the following reasons:

- – If we take $(\Gamma_{\mathrm{e}}[\![P_1]\!], \Gamma_{\mathrm{e}}[\![P_2]\!]) \in \mathscr{B}_{\mathrm{ot,e,1}}$, then for $k \in \{1,2\}$ we have that $\Gamma_{\mathrm{e}}[\![P_k]\!]$ cannot perform any action by virtue of Lemma 4.6. Consider an equivalence class $D \in \mathbb{P}_{\mathrm{M,ot}}/\mathscr{B}_{\mathrm{ot,e}}$. There are two cases:
  - ∗ If $D$ is originated from $\mathscr{B}_{\mathrm{ot,e,1}}$ (or $\mathscr{B}_{\mathrm{ot,e,3}}$), then by virtue of Lemma 4.7:
  $$\mathit{rate}_{\mathrm{ot}}(\Gamma_{\mathrm{e}}[\![P_1]\!], D) \,=\, 0 \,=\, \mathit{rate}_{\mathrm{ot}}(\Gamma_{\mathrm{e}}[\![P_2]\!], D)$$
  - ∗ If $D$ is originated from $\mathscr{B}_{\mathrm{ot,e,2}}$ and is characterized by $a \in \mathit{Name}$ and $P \in \mathbb{P}_{\mathrm{M,it,sf}}$, then by virtue of Lemma 4.7 and $P_1 \sim_{\mathrm{MB,it}} P_2$:
  $$\mathit{rate}_{\mathrm{ot}}(\Gamma_{\mathrm{e}}[\![P_1]\!], D) = \mathit{rate}_{\mathrm{it}}(P_1, a, [P]_{\sim_{\mathrm{MB,it}}}) = \mathit{rate}_{\mathrm{it}}(P_2, a, [P]_{\sim_{\mathrm{MB,it}}}) = \mathit{rate}_{\mathrm{ot}}(\Gamma_{\mathrm{e}}[\![P_2]\!], D)$$
- – If we take $(Q_1, Q_2) \in \mathscr{B}_{\mathrm{ot,e,2}}$, then for $k \in \{1,2\}$ any possible time transition of $Q_k$ is preempted by the only action transition of $Q_k$ in an eager setting, so we have to compare only the two action transitions of $Q_1$ and $Q_2$:

* When $Q_1 \xrightarrow{a} \Gamma_{\rm e}[\![P'_1]\!]$, then $Q_2$ can respond with $Q_2 \xrightarrow{a} \Gamma_{\rm e}[\![P'_2]\!]$ where $(\Gamma_{\rm e}[\![P'_1]\!], \Gamma_{\rm e}[\![P'_2]\!]) \in \mathscr{B}_{\rm ot,e,1} \subseteq \mathscr{B}_{\rm ot,e}$ because $P'_1 \sim_{\rm MB,it} P'_2$, and vice versa.

$\Leftarrow$ Consider the relation $\mathscr{B}_{\rm it,e} = \mathscr{B}_{\rm it,e,1} \cup \mathscr{B}_{\rm it,e,2}$ over $\mathbb{P}_{\rm M,it}$, where:

$$\begin{aligned} \mathscr{B}_{\rm it,e,1} &= \{(P_1,P_2) \mid P_1,P_2 \in \mathbb{P}_{\rm M,it,sf} \wedge \Gamma_{\rm e}[\![P_1]\!] \sim_{\rm MB,ot,e} \Gamma_{\rm e}[\![P_2]\!]\} \\ \mathscr{B}_{\rm it,e,2} &= \{(P,P) \mid P \notin \mathbb{P}_{\rm M,it,sf}\} \end{aligned}$$

This is an equivalence relation because so is $\sim_{\rm MB,ot,e}$. Moreover, it turns out to be an integrated-time Markovian bisimulation. In fact, if we take $(P_1,P_2) \in \mathscr{B}_{\rm it,e,1}$ and we consider $a \in Name$ and $D \in \mathbb{P}_{\rm M,it}/\mathscr{B}_{\rm it,e}$, then there are two cases:

- If $D$ is originated from $\mathscr{B}_{\rm it,e,2}$, then:

$$rate_{\rm it}(P_1,a,D) = 0 = rate_{\rm it}(P_2,a,D)$$

- If $D$ is originated from $\mathscr{B}_{\rm it,e,1}$ and is characterized by $\Gamma_{\rm e}[\![P]\!] \in \mathbb{P}_{\rm M,ot,nnd,sf}$, then by virtue of Lemma 4.7 and $\Gamma_{\rm e}[\![P_1]\!] \sim_{\rm MB,ot,e} \Gamma_{\rm e}[\![P_2]\!]$:

$$rate_{\rm it}(P_1,a,D) = rate_{\rm ot}(\Gamma_{\rm e}[\![P_1]\!],D_{a,P}) = rate_{\rm ot}(\Gamma_{\rm e}[\![P_2]\!],D_{a,P}) = rate_{\rm it}(P_2,a,D)$$

  where $D_{a,P}$ is the equivalence class of $\sim_{\rm MB,ot,e}$ formed by all the process terms $Q$ whose only action transition is $Q \xrightarrow{a} \Gamma_{\rm e}[\![P']\!]$ with $\Gamma_{\rm e}[\![P']\!] \in [\Gamma_{\rm e}[\![P]\!]]_{\sim_{\rm MB,ot,e}}$. ■

**Proof of Lemma 4.9.** Similar to the proof of Lemma 4.5, we proceed by induction on the syntactical structure of $P \in \mathscr{PL}_{\rm M,it,sf}$. The only case that is treated differently is the following:

- Let $P \equiv <a,\lambda>.P'$ and assume that $\Gamma_{\rm mp}[\![P'\{{\rm rec}X : \hat{P} \hookrightarrow Y\}]\!] = \Gamma_{\rm mp}[\![P']\!]\{{\rm rec}X : \Gamma_{\rm mp}[\![\hat{P}]\!] \hookrightarrow Y\}$. Then:

$$\begin{aligned} \Gamma_{\rm mp}[\![P\{{\rm rec}X : \hat{P} \hookrightarrow Y\}]\!] &= \Gamma_{\rm mp}[\![<a,\lambda>.(P'\{{\rm rec}X : \hat{P} \hookrightarrow Y\})]\!] = \\ &= (\lambda)\,.\,{\rm rec}Z : (\tau\,.\,Z + a\,.\,\Gamma_{\rm mp}[\![P'\{{\rm rec}X : \hat{P} \hookrightarrow Y\}]\!]) = \\ &= (\lambda)\,.\,{\rm rec}Z : (\tau\,.\,Z + a\,.\,(\Gamma_{\rm mp}[\![P']\!]\{{\rm rec}X : \Gamma_{\rm mp}[\![\hat{P}]\!] \hookrightarrow Y\})) = \\ &= ((\lambda)\,.\,{\rm rec}Z : (\tau\,.\,Z + a\,.\,\Gamma_{\rm mp}[\![P']\!]))\{{\rm rec}X : \Gamma_{\rm mp}[\![\hat{P}]\!] \hookrightarrow Y\} = \Gamma_{\rm mp}[\![P]\!]\{{\rm rec}X : \Gamma_{\rm mp}[\![\hat{P}]\!] \hookrightarrow Y\} \end{aligned}$$

  We observe that the equality between the process term on the third line and the first process term on the fourth line is correct. In fact, if $Y$ is free in $P'$, then $Y \not\equiv Z$ because $Z$ cannot be free in $P'$, and hence $(\tau\,.\,Z)\{{\rm rec}X : \Gamma_{\rm mp}[\![\hat{P}]\!] \hookrightarrow Y\}$ coincides with $\tau\,.\,Z$. If $Y$ is not free in $P'$, then it might be $Y \equiv Z$, but in that case $({\rm rec}Z : (\tau\,.\,Z + a\,.\,\Gamma_{\rm mp}[\![P']\!]))\{{\rm rec}X : \Gamma_{\rm mp}[\![\hat{P}]\!] \hookrightarrow Y\}$ would be equal to ${\rm rec}Z : (\tau\,.\,Z + a\,.\,\Gamma_{\rm mp}[\![P']\!])$, as well as $\Gamma_{\rm mp}[\![P']\!]\{{\rm rec}X : \Gamma_{\rm mp}[\![\hat{P}]\!] \hookrightarrow Y\} = \Gamma_{\rm mp}[\![P']\!]$. ■

**Proof of Lemma 4.10.** Similar to the proof of Lemma 4.6, we proceed by induction on the syntactical structure of $P \in \mathbb{P}_{\rm M,it,sf}$ by exploiting Lemma 4.9. The only case that is treated differently is the following:

- If $P \equiv <a,\lambda>.P'$, then $\Gamma_{\rm mp}[\![P]\!] = (\lambda)\,.\,{\rm rec}Z : (\tau\,.\,Z + a\,.\,\Gamma_{\rm mp}[\![P']\!])$ – with $Z$ not free in $P'$ – and hence cannot perform any action. Moreover:

$$rate_{\rm it,t}(P) = \lambda = rate_{\rm ot,t}(\Gamma_{\rm mp}[\![P]\!])$$

■

**Proof of Lemma 4.11.** Similar to the proof of Lemma 4.7, the proof is divided into two parts – with the first one exploiting Lemmas 4.9 and 4.10 – in both of which we proceed by induction on the length of the transition derivation and we perform a case analysis of the syntactical structure of the process term associated with the state from which the transition departs. Unlike the proof of Lemma 4.7, the intermediate process term $Q \in \mathbb{P}_{\rm M,ot,cnd,sf}$ has not only one action transition, but only two action transitions one of which is a $\tau$-selfloop. This is a consequence of the different treatment of the basic case of the induction, which is as follows in the first part of the proof:

– If the length is 1, then it must be $P \equiv {<}a,\lambda{>}.P'$ and hence $\Gamma_{\rm mp}[\![P]\!] = (\lambda)\,.\,{\rm rec}\,Z : (\tau\,.\,Z + a\,.\,\Gamma_{\rm mp}[\![P']\!])$ with $Z$ not free in $P'$. Therefore $\Gamma_{\rm mp}[\![P]\!] \xrightarrow{\lambda}_{\rm M} {\rm rec}\,Z : (\tau\,.\,Z + a\,.\,\Gamma_{\rm mp}[\![P']\!])$ with the only action transitions of the reached process term being ${\rm rec}\,Z : (\tau\,.\,Z + a\,.\,\Gamma_{\rm mp}[\![P']\!]) \xrightarrow{\tau} {\rm rec}\,Z : (\tau\,.\,Z + a\,.\,\Gamma_{\rm mp}[\![P']\!])$ and ${\rm rec}\,Z : (\tau\,.\,Z + a\,.\,\Gamma_{\rm mp}[\![P']\!]) \xrightarrow{a} \Gamma_{\rm mp}[\![P']\!]$. ■

**Proof of Thm. 4.12.** The proof is divided into two parts:

$\Rightarrow$ Consider the relation $\mathscr{B}_{\rm ot,mp} = \mathscr{B}_{\rm ot,mp,1} \cup \mathscr{B}_{\rm ot,mp,2} \cup \mathscr{B}_{\rm ot,mp,3}$ over $\mathbb{P}_{\rm M,ot}$, where:

$$\begin{array}{rcl} \mathscr{B}_{\rm ot,mp,1} & = & \{(\Gamma_{\rm mp}[\![P_1]\!], \Gamma_{\rm mp}[\![P_2]\!]) \mid P_1, P_2 \in \mathbb{P}_{\rm M,it,sf} \wedge P_1 \sim_{\rm MB,it} P_2\} \\ \mathscr{B}_{\rm ot,mp,2} & = & \{(Q_1, Q_2) \mid Q_1, Q_2 \in \mathbb{P}_{\rm M,ot,cnd,sf} \text{ and the only action transitions of } Q_k,\ k \in \{1,2\}, \\ & & \qquad \text{are } Q_k \xrightarrow{\tau} Q_k \text{ and } Q_k \xrightarrow{a} \Gamma_{\rm mp}[\![P'_k]\!] \text{ with } P'_1 \sim_{\rm MB,it} P'_2\} \\ \mathscr{B}_{\rm ot,mp,3} & = & \{(Q, Q) \mid Q \notin \mathbb{P}_{\rm M,ot,cnd,sf}\} \end{array}$$

This is an equivalence relation because so is $\sim_{\rm MB,it}$. Moreover, it turns out to be a maximal-progress orthogonal-time Markovian bisimulation for the following reasons:

– If we take $(\Gamma_{\rm mp}[\![P_1]\!], \Gamma_{\rm mp}[\![P_2]\!]) \in \mathscr{B}_{\rm ot,mp,1}$, then for $k \in \{1,2\}$ we have that $\Gamma_{\rm mp}[\![P_k]\!]$ cannot perform any action by virtue of Lemma 4.10. Consider an equivalence class $D \in \mathbb{P}_{\rm M,ot}/\mathscr{B}_{\rm ot,mp}$. There are two cases:

  ∗ If $D$ is originated from $\mathscr{B}_{\rm ot,mp,1}$ (or $\mathscr{B}_{\rm ot,mp,3}$), then by virtue of Lemma 4.11:
  $$rate_{\rm ot}(\Gamma_{\rm mp}[\![P_1]\!], D) \;=\; 0 \;=\; rate_{\rm ot}(\Gamma_{\rm mp}[\![P_2]\!], D)$$

  ∗ If $D$ is originated from $\mathscr{B}_{\rm ot,mp,2}$ and is characterized by $a \in Name$ and $P \in \mathbb{P}_{\rm M,it,sf}$, then by virtue of Lemma 4.11 and $P_1 \sim_{\rm MB,it} P_2$:
  $$rate_{\rm ot}(\Gamma_{\rm mp}[\![P_1]\!], D) = rate_{\rm it}(P_1, a, [P]_{\sim_{\rm MB,it}}) = rate_{\rm it}(P_2, a, [P]_{\sim_{\rm MB,it}}) = rate_{\rm ot}(\Gamma_{\rm mp}[\![P_2]\!], D)$$

– If we take $(Q_1, Q_2) \in \mathscr{B}_{\rm ot,mp,2}$, then for $k \in \{1,2\}$ any possible time transition of $Q_k$ is pre-empted by the $\tau$-selfloop of $Q_k$ in a maximal-progress setting, so we have to compare only the four action transitions of $Q_1$ and $Q_2$:

  ∗ When $Q_1 \xrightarrow{\tau} Q_1$, then $Q_2$ can respond with $Q_2 \xrightarrow{\tau} Q_2$ where $(Q_1, Q_2) \in \mathscr{B}_{\rm ot,mp,2} \subseteq \mathscr{B}_{\rm ot,mp}$, and vice versa.

  ∗ When $Q_1 \xrightarrow{a} \Gamma_{\rm mp}[\![P'_1]\!]$, then $Q_2$ can respond with $Q_2 \xrightarrow{a} \Gamma_{\rm mp}[\![P'_2]\!]$ where $(\Gamma_{\rm mp}[\![P'_1]\!], \Gamma_{\rm mp}[\![P'_2]\!]) \in \mathscr{B}_{\rm ot,mp,1} \subseteq \mathscr{B}_{\rm ot,mp}$ because $P'_1 \sim_{\rm MB,it} P'_2$, and vice versa.

$\Leftarrow$ Consider the relation $\mathscr{B}_{\rm it,mp} = \mathscr{B}_{\rm it,mp,1} \cup \mathscr{B}_{\rm it,mp,2}$ over $\mathbb{P}_{\rm M,it}$, where:

$$\begin{array}{rcl} \mathscr{B}_{\rm it,mp,1} & = & \{(P_1, P_2) \mid P_1, P_2 \in \mathbb{P}_{\rm M,it,sf} \wedge \Gamma_{\rm mp}[\![P_1]\!] \sim_{\rm MB,ot,mp} \Gamma_{\rm mp}[\![P_2]\!]\} \\ \mathscr{B}_{\rm it,mp,2} & = & \{(P, P) \mid P \notin \mathbb{P}_{\rm M,it,sf}\} \end{array}$$

This is an equivalence relation because so is $\sim_{\rm MB,ot,mp}$. Moreover, it turns out to be an integrated-time Markovian bisimulation. In fact, if we take $(P_1, P_2) \in \mathscr{B}_{\rm it,mp,1}$ and we consider $a \in Name$ and $D \in \mathbb{P}_{\rm M,it}/\mathscr{B}_{\rm it,mp}$, then there are two cases:

– If $D$ is originated from $\mathscr{B}_{\rm it,mp,2}$, then:
$$rate_{\rm it}(P_1, a, D) \;=\; 0 \;=\; rate_{\rm it}(P_2, a, D)$$

– If $D$ is originated from $\mathscr{B}_{\rm it,mp,1}$ and is characterized by $\Gamma_{\rm mp}[\![P]\!] \in \mathbb{P}_{\rm M,ot,cnd,sf}$, then by virtue of Lemma 4.11 and $\Gamma_{\rm mp}[\![P_1]\!] \sim_{\rm MB,ot,mp} \Gamma_{\rm mp}[\![P_2]\!]$:
$$rate_{\rm it}(P_1, a, D) \;=\; rate_{\rm ot}(\Gamma_{\rm mp}[\![P_1]\!], D_{a,P}) \;=\; rate_{\rm ot}(\Gamma_{\rm mp}[\![P_2]\!], D_{a,P}) \;=\; rate_{\rm it}(P_2, a, D)$$
where $D_{a,P}$ is the equivalence class of $\sim_{\rm MB,ot,mp}$ formed by all the process terms $Q$ whose only action transitions are $Q \xrightarrow{\tau} Q$ and $Q \xrightarrow{a} \Gamma_{\rm mp}[\![P']\!]$ with $\Gamma_{\rm mp}[\![P']\!] \in [\Gamma_{\rm mp}[\![P]\!]]_{\sim_{\rm MB,ot,mp}}$.

■