

Performance preserving equivalences for dtsPBC*

I. V. Tarasyuk

Abstract. For a discrete time stochastic extension dtsPBC of finite Petri box calculus (PBC) enriched with iteration, we define a number of stochastic equivalences. They allow one to identify processes with similar behaviour which are differentiated by the too discriminate semantic equivalence of the calculus. We investigate which is the weakest equivalence that guarantees a coincidence of performance indices for stochastic systems and can be used to simplify their performance evaluation.

Keywords: stochastic process algebra, Petri box calculus, iteration, discrete time, stochastic equivalence, stationary behaviour, performance evaluation.

1. Introduction

Process algebras (PAs) are a widely used formal model designed to specify concurrent systems and analyze their behavioural properties. In such calculi, processes are specified by compositional formulas constructed with operations from symbols of actions, and verification of properties is accomplished syntactically by means of algebraic laws and equivalences. In the last decades, stochastic extensions of PAs were proposed. Stochastic process algebras (SPAs) do not just specify actions which can happen (qualitative features) like standard PAs, but they associate random delays with actions (quantitative features). The well-known SPAs are PEPA [4] and EMPA [2].

Petri box calculus (PBC) [1, 3] is an expressive process algebra intended to provide support for compositional translation from high level concurrent programming languages into Petri nets (PNs). In the calculus, synchronization is separated from the parallelism operator, and there is a multiway synchronization capability. PBC has a step operational semantics in terms of labeled transition systems based on rules in the Structured Operational Semantics (SOS) style. A denotational semantics of PBC was proposed via a subclass of PNs equipped with an interface called Petri boxes.

A continuous time stochastic extension of a finite part of PBC called stochastic Petri box calculus (sPBC) was proposed in [6]. The algebra sPBC in its former version had neither refinement nor recursion nor iteration operations and thus specified finite processes only. An interleaving operational semantics of the calculus was constructed in terms of labeled probabilistic transition systems. A denotational semantics of sPBC was defined via a subclass of labeled continuous time stochastic PNs (LCTSPNs) called stochastic

* Partially supported by Deutsche Forschungsgemeinschaft under Grant 436 RUS 113/1002/01, and Russian Foundation for Basic Research under grant 09-01-91334.

Petri boxes (s-boxes). The iteration operation was added to sPBC in [5] to specify infinite processes.

A discrete time stochastic extension dtsPBC of finite PBC was presented in [7, 9]. A step operational semantics of the algebra was constructed with the use of labeled probabilistic transition systems. dtsPBC has a denotational semantics based on a subclass of labeled discrete time stochastic PNs (LDTSPNs) called discrete time stochastic Petri boxes (dts-boxes). A number of stochastic equivalences were proposed to identify processes with similar behaviour which are differentiated by the semantic equivalence of dtsPBC. The interrelations of all the introduced equivalences were studied. In [8], the syntax of dtsPBC was supplemented by the iteration operator.

Since dtsPBC has a discrete time semantics and geometrically distributed delays in the states unlike sPBC with continuous time semantics and exponentially distributed delays, the calculi apply two different approaches to the stochastic extension of PBC, in spite of some similarity of their syntax and semantics inherited from PBC. The main advantage of dtsPBC is that concurrency is treated naturally, like in PBC, whereas in sPBC parallelism is simulated by interleaving obliging one to collect the information on causal independence of activities before constructing the semantics.

In this paper, a problem of performance preservation by the equivalence notions is discussed within dtsPBC enriched with iteration. First, we present syntax of the calculus. Second, we describe its operational semantics in terms of labeled transition systems and the denotational semantics based on a subclass of LDTSPNs. Further, we propose a number of stochastic equivalences based on transition systems without empty behaviour. We investigate which equivalences guarantee identity of stationary behaviour, and which of them is the weakest one. The mentioned property implies a coincidence of performance indices based on steady-state probabilities of the modeled stochastic systems. It can be used for performance analysis simplification due to the state space reduction modulo the equivalences.

The paper is organized as follows. The syntax of dtsPBC is presented in Section 2. Section 3 describes the operational semantics of the calculus and Section 4 presents its denotational semantics. Stochastic equivalences are defined in Section 5. In Section 6 the relations are applied to the stationary behaviour comparison. Section 7 summarizes the results obtained.

2. Syntax

In this section, we propose the syntax of the discrete time stochastic extension of finite PBC enriched with iteration and called dtsPBC.

We denote the *set of all finite multisets* over X by IN_f^X . Let $Act = \{a, b, \dots\}$ be the set of *elementary actions*. Then $\widehat{Act} = \{\hat{a}, \hat{b}, \dots\}$ is the set of *conjugated actions (conjugates)* s.t. $a \neq \hat{a}$ and $\hat{\hat{a}} = a$. Let $\mathcal{A} = Act \cup \widehat{Act}$

be the set of *all actions*, and $\mathcal{L} = \mathbb{N}_f^{\mathcal{A}}$ be the set of *all multiactions*. Note that $\emptyset \in \mathcal{L}$, this corresponds to an internal activity, i.e., the execution of a multiaction containing no visible actions. The *alphabet* of $\alpha \in \mathcal{L}$ is $\mathcal{A}(\alpha) = \{x \in \mathcal{A} \mid \alpha(x) > 0\}$.

An *activity (stochastic multiaction)* is a pair (α, ρ) , where $\alpha \in \mathcal{L}$ and $\rho \in (0; 1)$ is the probability of the multiaction α . The multiaction probabilities are used to calculate those of state changes (steps) at discrete time moments. Let \mathcal{SL} be the set of *all activities*. Let us note that the same multiaction $\alpha \in \mathcal{L}$ may have different probabilities in the same specification. The *alphabet* of $(\alpha, \rho) \in \mathcal{SL}$ is defined as $\mathcal{A}(\alpha, \rho) = \mathcal{A}(\alpha)$. For $(\alpha, \rho) \in \mathcal{SL}$, we define its *multiaction part* as $\mathcal{L}(\alpha, \rho) = \alpha$ and its *probability part* as $\Omega(\alpha, \rho) = \rho$.

Activities are combined into formulas by the following operations: *sequential execution* $;$, *choice* $[]$, *parallelism* $||$, *relabeling* $[f]$ of actions, *restriction* rs over an action, *synchronization* sy on an action and its conjugate and *iteration* $[**]$ with three arguments: initialization, body and termination.

Sequential execution and choice have the standard interpretation, but parallelism does not include synchronization. Relabeling functions $f : \mathcal{A} \rightarrow \mathcal{A}$ are bijections preserving conjugates, i.e., $\forall x \in \mathcal{A} f(\hat{x}) = \widehat{f(x)}$. Relabeling is extended to multiactions in a usual way: for $\alpha \in \mathcal{L}$ let $f(\alpha) = \sum_{x \in \mathcal{A}} f(x)$. Restriction over an action a means that for a given expression any process behaviour containing a or its conjugate \hat{a} is not allowed. In the iteration, the initialization subprocess is executed first, then the body is performed zero or more times, and, finally, the termination subprocess is executed.

Let $\alpha, \beta \in \mathcal{L}$ be multiactions s.t. for some $a \in Act$ we have $a \in \alpha$ and $\hat{a} \in \beta$ or $\hat{a} \in \alpha$ and $a \in \beta$. The synchronization of α and β by a is defined as $\alpha \oplus_a \beta = \gamma$, where $\gamma(x) = \begin{cases} \alpha(x) + \beta(x) - 1, & x = a \text{ or } x = \hat{a}; \\ \alpha(x) + \beta(x), & \text{otherwise.} \end{cases}$

Static expressions specify the structure of processes. They correspond to unmarked LDTSPNs.

Definition 1. Let $(\alpha, \rho) \in \mathcal{SL}$, $a \in Act$. A *static expression* of dtsPBC is

$$E ::= (\alpha, \rho) \mid E; E \mid E[]E \mid E||E \mid E[f] \mid E \text{ rs } a \mid E \text{ sy } a \mid [E * E * E].$$

StatExpr denotes the set of *all static expressions* of dtsPBC.

To avoid inconsistency of the iteration operator, we should not allow any concurrency in the highest level of the second argument of iteration. This is not a severe restriction though, since we can always prefix parallel expressions by an activity with the empty multiaction.

Definition 2. Let $(\alpha, \rho) \in \mathcal{SL}$, $a \in Act$. A *regular static expression* of dtsPBC is

$$E ::= (\alpha, \rho) \mid E; E \mid E[]E \mid E\|E \mid E[f] \mid E \text{ rs } a \mid E \text{ sy } a \mid [E * D * E],$$

where $D ::= (\alpha, \rho) \mid D; E \mid D[]D \mid D[f] \mid D \text{ rs } a \mid D \text{ sy } a \mid [D * D * E]$.

RegStatExpr denotes the set of *all regular static expressions* of dtsPBC. Dynamic expressions specify the states of processes. They correspond to (marked) LDTSPNs. \overline{E} denotes the *initial* and \underline{E} denotes the *final* state of the process specified by a static expression E . The *underlying static expression* is obtained by removing the upper and lower bars from a dynamic one.

Definition 3. Let $a \in Act$, $E \in StatExpr$. A *dynamic expression* of dtsPBC is

$$G ::= \overline{E} \mid \underline{E} \mid G; E \mid E; G \mid G[]E \mid E[]G \mid G\|G \mid G[f] \mid G \text{ rs } a \mid G \text{ sy } a \mid [G * E * E] \mid [E * G * E] \mid [E * E * G].$$

DynExpr denotes the set of *all dynamic expressions* of dtsPBC. A dynamic expression is *regular* if it is so for its underlying static expression. *RegDynExpr* denotes the set of *all regular dynamic expressions* of dtsPBC. We shall consider regular expressions only and we omit the word “regular”.

3. Operational semantics

In this section, we define the step operational semantics in terms of labeled probabilistic transition systems with transitions representing steps, i.e., the executions of multisets of activities.

3.1. Inaction rules

Inaction rules describe expression transformations due to the execution of the empty multiset of activities. The rules will be used to define the *empty loop* transitions reflecting a non-zero probability to stay in the current state at the next moment, an essential feature of discrete time stochastic processes.

First, in Table 1, we define inaction rules for overlined and underlined static expressions. In this table, $E, F, K \in RegStatExpr$ and $a \in Act$.

Second, in Table 2, we propose inaction rules for arbitrary dynamic expressions. In this table, $E, F \in RegStatExpr$, $G, H, \tilde{G}, \tilde{H} \in RegDynExpr$ and $a \in Act$.

A regular dynamic expression G is *operative* if no inaction rule can be applied to it, with the exception of $G \xrightarrow{\emptyset} G$. *OpRegDynExpr* denotes the set of *all operative regular dynamic expressions* of dtsPBC. Any dynamic expression can be transformed into a (not always unique) operative one by the inaction rules.

Table 1. Inaction rules for overlined and underlined static expressions

$\overline{E}; \overline{F} \xrightarrow{\emptyset} \overline{E}; \overline{F}$	$\underline{E}; \underline{F} \xrightarrow{\emptyset} \underline{E}; \underline{F}$	$E; \underline{F} \xrightarrow{\emptyset} E; \underline{F}$
$\overline{E}[] \overline{F} \xrightarrow{\emptyset} \overline{E}[] \overline{F}$	$\underline{E}[] \underline{F} \xrightarrow{\emptyset} \underline{E}[] \underline{F}$	$\underline{E}[] \underline{F} \xrightarrow{\emptyset} \underline{E}[] \underline{F}$
$E[] \underline{F} \xrightarrow{\emptyset} E[] \underline{F}$	$\overline{E}[] \overline{F} \xrightarrow{\emptyset} \overline{E}[] \overline{F}$	$\underline{E}[] \underline{F} \xrightarrow{\emptyset} \underline{E}[] \underline{F}$
$\overline{E}[f] \xrightarrow{\emptyset} \overline{E}[f]$	$\underline{E}[f] \xrightarrow{\emptyset} \underline{E}[f]$	$\overline{E} \text{ rs } a \xrightarrow{\emptyset} \overline{E} \text{ rs } a$
$\underline{E} \text{ rs } a \xrightarrow{\emptyset} \underline{E} \text{ rs } a$	$\overline{E} \text{ sy } a \xrightarrow{\emptyset} \overline{E} \text{ sy } a$	$\underline{E} \text{ sy } a \xrightarrow{\emptyset} \underline{E} \text{ sy } a$
$\overline{[E * F * K]} \xrightarrow{\emptyset} \overline{[E * F * K]}$	$\underline{[E * F * K]} \xrightarrow{\emptyset} \underline{[E * F * K]}$	$[E * \overline{F} * K] \xrightarrow{\emptyset} [E * \overline{F} * K]$
$[E * \underline{F} * K] \xrightarrow{\emptyset} [E * \underline{F} * K]$	$[E * \overline{F} * K] \xrightarrow{\emptyset} [E * \overline{F} * K]$	$[E * \underline{F} * K] \xrightarrow{\emptyset} [E * \underline{F} * \overline{K}]$
	$[E * \underline{F} * \underline{K}] \xrightarrow{\emptyset} [E * \underline{F} * \underline{K}]$	

Table 2. Inaction rules for arbitrary dynamic expressions

$G \xrightarrow{\emptyset} G$	$\frac{G \xrightarrow{\emptyset} \tilde{G}, \text{ o} \in \{::, []\}}{G \circ E \xrightarrow{\emptyset} \tilde{G} \circ E}$	$\frac{G \xrightarrow{\emptyset} \tilde{G}, \text{ o} \in \{::, []\}}{E \circ G \xrightarrow{\emptyset} E \circ \tilde{G}}$	$\frac{G \xrightarrow{\emptyset} \tilde{G}}{G \ H \xrightarrow{\emptyset} \tilde{G} \ H}$
$\frac{H \xrightarrow{\emptyset} \tilde{H}}{G \ H \xrightarrow{\emptyset} G \ \tilde{H}}$	$\frac{G \xrightarrow{\emptyset} \tilde{G}}{G[f] \xrightarrow{\emptyset} \tilde{G}[f]}$	$\frac{G \xrightarrow{\emptyset} \tilde{G}, \text{ o} \in \{\text{rs}, \text{sy}\}}{G \circ a \xrightarrow{\emptyset} \tilde{G} \circ a}$	$\frac{G \xrightarrow{\emptyset} \tilde{G}}{[G * E * F] \xrightarrow{\emptyset} [\tilde{G} * E * F]}$
	$\frac{G \xrightarrow{\emptyset} \tilde{G}}{[E * G * F] \xrightarrow{\emptyset} [E * \tilde{G} * F]}$	$\frac{G \xrightarrow{\emptyset} \tilde{G}}{[E * F * G] \xrightarrow{\emptyset} [E * F * \tilde{G}]}$	

Definition 4. Let $\approx = (\xrightarrow{\emptyset} \cup \xleftarrow{\emptyset})^*$ be structural equivalence of dynamic expressions in dtsPBC. Thus, two dynamic expressions G and G' are *structurally equivalent*, denoted by $G \approx G'$, if they can be reached from each other by applying the inaction rules in forward or backward direction.

The rule $G \xrightarrow{\emptyset} G$ was intentionally included in the set of inaction rules to define later the empty loop transitions for the states corresponding to the dynamic expressions like $\overline{(\alpha, \rho)}$ to which no different structurally equivalent ones exist. The rule has no prototype in PBC.

3.2. Action rules

Action rules describe expression transformations due to the execution of non-empty multisets of activities. The rules will be used to define transitions representing the state changes when some non-empty multisets of activities are executed. Let $\Gamma \in \mathcal{N}_f^{S\mathcal{L}}$. Relabeling is extended to the multisets of activities as follows: $f(\Gamma) = \sum_{(\alpha, \rho) \in \Gamma} (f(\alpha), \rho)$. The *alphabet* of Γ is defined as $\mathcal{A}(\Gamma) = \cup_{(\alpha, \rho) \in \Gamma} \mathcal{A}(\alpha)$.

In Table 3, we define the action rules. In this table, $(\alpha, \rho), (\beta, \chi) \in \mathcal{SL}$, $E, F \in \text{RegStatExpr}$, $G, H \in \text{OpRegDynExpr}$, $\tilde{G}, \tilde{H} \in \text{RegDynExpr}$ and $a \in \text{Act}$. Moreover, $\Gamma, \Delta \in \mathbb{N}_f^{\mathcal{SL}} \setminus \{\emptyset\}$.

Table 3. Action rules

B $\frac{}{(\alpha, \rho) \xrightarrow{\{(\alpha, \rho)\}} (\alpha, \rho)}$	SC1 $\frac{G \xrightarrow{\Gamma} \tilde{G}, \circ \in \{:, []\}}{G \circ E \xrightarrow{\Gamma} \tilde{G} \circ E}$	SC2 $\frac{G \xrightarrow{\Gamma} \tilde{G}, \circ \in \{:, []\}}{E \circ G \xrightarrow{\Gamma} E \circ \tilde{G}}$
P1 $\frac{G \xrightarrow{\Gamma} \tilde{G}}{G \parallel H \xrightarrow{\Gamma} \tilde{G} \parallel H}$	P2 $\frac{H \xrightarrow{\Gamma} \tilde{H}}{G \parallel H \xrightarrow{\Gamma} G \parallel \tilde{H}}$	P3 $\frac{G \xrightarrow{\Gamma} \tilde{G}, H \xrightarrow{\Delta} \tilde{H}}{G \parallel H \xrightarrow{\Gamma + \Delta} \tilde{G} \parallel \tilde{H}}$
L $\frac{G \xrightarrow{\Gamma} \tilde{G}}{G[f] \xrightarrow{f(\Gamma)} \tilde{G}[f]}$	Rs $\frac{G \xrightarrow{\Gamma} \tilde{G}, a, \hat{a} \notin \mathcal{A}(\Gamma)}{G \text{ rs } a \xrightarrow{\Gamma} \tilde{G} \text{ rs } a}$	I1 $\frac{G \xrightarrow{\Gamma} \tilde{G}}{[G * E * F] \xrightarrow{\Gamma} [\tilde{G} * E * F]}$
I2 $\frac{G \xrightarrow{\Gamma} \tilde{G}}{[E * G * F] \xrightarrow{\Gamma} [E * \tilde{G} * F]}$	I3 $\frac{G \xrightarrow{\Gamma} \tilde{G}}{[E * F * G] \xrightarrow{\Gamma} [E * F * \tilde{G}]}$	Sy1 $\frac{G \xrightarrow{\Gamma} \tilde{G}}{G \text{ sy } a \xrightarrow{\Gamma} \tilde{G} \text{ sy } a}$
Sy2 $\frac{G \text{ sy } a \xrightarrow{\Gamma + \{(\alpha, \rho)\} + \{(\beta, \chi)\}} \tilde{G} \text{ sy } a, a \in \alpha, \hat{a} \in \beta}{G \text{ sy } a \xrightarrow{\Gamma + \{(\alpha \oplus a \beta, \rho \cdot \chi)\}} \tilde{G} \text{ sy } a}$		

In the rule **Sy2**, we multiply the probabilities of synchronized multiactions, since it corresponds to the probability of event intersection.

3.3. Transition systems

Now we define labeled probabilistic step transition systems associated with dynamic expressions and used to define their operational semantics.

The expressions of dtsPBC can contain identical activities. To avoid technical difficulties, such as calculation of the state change probabilities for multiple transitions, we must enumerate coinciding activities, e.g., from left to right in the syntax of expressions. The new activities resulted from synchronization will be annotated with concatenation of the numbering of the activities they come, hence, the numbering should have a tree structure to reflect the effect of multiple synchronizations. We define the numbering which encodes a binary tree with the leaves labeled by natural numbers.

Definition 5. Let $\iota \in \mathbb{N}$. The *numbering* of expressions is defined as $\iota ::= \iota \mid (\iota)(\iota)$. *Num* denotes the set of *all numberings* of expressions.

The new activities resulting from applications of the second rule for synchronization **Sy2** in different orders should be considered up to permutation of their numbering. In this way, we shall recognize different instances of the same activity. If we compare the contents of different numberings, i.e., the sets of natural numbers in them, we shall be able to identify the mentioned instances. The *content* of a numbering $\iota \in \text{Num}$ is

$$\text{Cont}(\iota) = \begin{cases} \{\iota\}, & \iota \in \mathbb{N}; \\ \text{Cont}(\iota_1) \cup \text{Cont}(\iota_2), & \iota = (\iota_1)(\iota_2). \end{cases}$$

We suppose that the identical activities are enumerated to avoid ambiguity.

Definition 6. Let G be a dynamic expression. Then $[G]_{\approx} = \{H \mid G \approx H\}$ is the equivalence class of G w.r.t. the structural equivalence. The *derivation set* of a dynamic expression G , denoted by $DR(G)$, is the minimal set s.t.

- $[G]_{\approx} \in DR(G)$;
- if $[H]_{\approx} \in DR(G)$ and $\exists \Gamma H \xrightarrow{\Gamma} \tilde{H}$ then $[\tilde{H}]_{\approx} \in DR(G)$.

Let G be a dynamic expression and $s, \tilde{s} \in DR(G)$.

The set of *all multisets of activities executable in s* is defined as

$$Exec(s) = \{\Gamma \mid \exists H \in s \exists \tilde{H} H \xrightarrow{\Gamma} \tilde{H}\}.$$

Let $\Gamma \in Exec(s) \setminus \{\emptyset\}$. The *probability that the multiset of activities Γ is ready for execution in s* is

$$PF(\Gamma, s) = \prod_{(\alpha, \rho) \in \Gamma} \rho \cdot \prod_{\{(\beta, \chi)\} \in Exec(s) \mid (\beta, \chi) \notin \Gamma} (1 - \chi).$$

In the case $\Gamma = \emptyset$ we define

$$PF(\emptyset, s) = \begin{cases} \prod_{\{(\beta, \chi)\} \in Exec(s)} (1 - \chi), & Exec(s) \neq \{\emptyset\}; \\ 1, & \text{otherwise.} \end{cases}$$

Let $\Gamma \in Exec(s)$. The *probability to execute the multiset of activities $\Gamma \in Exec(s)$ in s* is

$$PT(\Gamma, s) = \frac{PF(\Gamma, s)}{\sum_{\Delta \in Exec(s)} PF(\Delta, s)}.$$

The *probability to move from s to \tilde{s} by executing any multiset of activities* is

$$PM(s, \tilde{s}) = \sum_{\{\Gamma \mid \exists H \in s \exists \tilde{H} \in \tilde{s} H \xrightarrow{\Gamma} \tilde{H}\}} PT(\Gamma, s).$$

Definition 7. Let G be a dynamic expression. The *(labeled probabilistic step) transition system* of G is a quadruple $TS(G) = (S_G, L_G, \mathcal{T}_G, s_G)$, where

- the set of *states* is $S_G = DR(G)$;
- the set of *labels* is $L_G \subseteq \mathcal{N}_f^{S_{\mathcal{L}}} \times (0; 1]$;
- the set of *transitions* is $\mathcal{T}_G = \{(s, (\Gamma, PT(\Gamma, s)), \tilde{s}) \mid s \in DR(G), \exists H \in s \exists \tilde{H} \in \tilde{s} H \xrightarrow{\Gamma} \tilde{H}\}$;
- the *initial state* is $s_G = [G]_{\approx}$.

The transition system $TS(G)$ associated with a dynamic expression G describes all steps that happen at moments of discrete time with some (one-step) probability and consist of multisets of activities. Every step happens instantaneously after one discrete time unit delay, and the step can change the current state to another one. The states are the structural equivalence classes of dynamic expressions obtained by application of action rules starting from the expressions belonging to $[G]_{\approx}$. A transition $(s, (\Gamma, \mathcal{P}), \tilde{s}) \in \mathcal{T}_G$ will be written as $s \xrightarrow{\mathcal{P}}_{\Gamma} \tilde{s}$. The interpretation is: the probability to change the state s to \tilde{s} in the result of executing Γ is \mathcal{P} . Note that Γ can be the empty multiset, and its execution does not change the current state (i.e., the equivalence class), since we have a loop transition $s \xrightarrow{\emptyset}_{\emptyset} s$ from a state s to itself in the result of executing the empty multiset. This corresponds to application of inaction rules to the expressions from the equivalence class. We have to keep track of such executions, called *empty loops*, because they have nonzero probabilities. This follows from the definition of $PF(\emptyset, s)$ and the fact that multiaction probabilities cannot be equal to 1 as they belong to the interval $(0; 1)$. The step probabilities belong to the interval $(0; 1]$. The step probability is 1 in the case when we cannot leave a state s , hence, there exists the only transition from it, namely, the empty loop one $s \xrightarrow{\emptyset}_1 s$.

We write $s \xrightarrow{\Gamma} \tilde{s}$ if $\exists \mathcal{P} \ s \xrightarrow{\mathcal{P}}_{\Gamma} \tilde{s}$ and $s \rightarrow \tilde{s}$ if $\exists \Gamma \ s \xrightarrow{\Gamma} \tilde{s}$.

Isomorphism is a coincidence of systems up to renaming of their components or states. \simeq denotes isomorphism between transition systems that relates their initial states.

Definition 8. Two dynamic expressions G and G' are *equivalent w.r.t. transition systems*, denoted by $G =_{ts} G'$, if $TS(G) \simeq TS(G')$.

Definition 9. The *underlying discrete time Markov chain (DTMC)* of a dynamic expression G , denoted by $DTMC(G)$, has the state space $DR(G)$ and the transitions $s \rightarrow_{\mathcal{P}} \tilde{s}$, if $s \rightarrow \tilde{s}$ and $\mathcal{P} = PM(s, \tilde{s})$.

4. Denotational semantics

In this section, we construct the denotational semantics in terms of a subclass of LDTSPNs called discrete time stochastic Petri boxes (dts-boxes).

Definition 10. A *discrete time stochastic Petri box (dts-box)* is a tuple $N = (P_N, T_N, W_N, \Lambda_N)$, where

- P_N and T_N are finite sets of *places* and *transitions*, respectively, with $P_N \cup T_N \neq \emptyset$ and $P_N \cap T_N = \emptyset$;
- $W_N : (P_N \times T_N) \cup (T_N \times P_N) \rightarrow \mathbb{N}$ is a function describing the *weights of arcs* between places and transitions and vice versa;

- Λ_N is the *place and transition labeling* function s.t. $\Lambda_N|_{P_N} : P_N \rightarrow \{\mathbf{e}, \mathbf{i}, \mathbf{x}\}$ (it specifies *entry*, *internal* and *exit* places, respectively) and $\Lambda_N|_{T_N} : T_N \rightarrow \{\varrho \mid \varrho \subseteq \mathbb{N}_f^{\mathcal{S}\mathcal{L}} \times \mathcal{S}\mathcal{L}\}$ (it associates transitions with the *relabeling relations*).

Let $t \in T_N$, $U \in \mathbb{N}_f^{T_N}$. The *precondition* $\bullet t$ and the *postcondition* $t\bullet$ of t are the multisets of places defined as $(\bullet t)(p) = W_N(p, t)$ and $(t\bullet)(p) = W_N(t, p)$. The *precondition* $\bullet U$ and the *postcondition* $U\bullet$ of U are the multisets of places defined as $\bullet U = \sum_{t \in U} \bullet t$ and $U\bullet = \sum_{t \in U} t\bullet$. We require that $\forall t \in T_N \bullet t \neq \emptyset \neq t\bullet$. In addition, for the set of *entry* places of N defined as ${}^\circ N = \{p \in P_N \mid \Lambda_N(p) = \mathbf{e}\}$ and the set of *exit* places of N defined as $N^\circ = \{p \in P_N \mid \Lambda_N(p) = \mathbf{x}\}$, we require that ${}^\circ N \neq \emptyset \neq N^\circ$ and $\bullet({}^\circ N) = \emptyset = (N^\circ)\bullet$.

A dts-box is *plain* if $\forall t \in T_N \Lambda_N(t) \in \mathcal{S}\mathcal{L}$, i.e., $\Lambda_N(t)$ is the constant relabeling that will be defined later. A *marked plain dts-box* is a pair (N, M_N) , where N is a plain dts-box and $M_N \in \mathbb{N}_f^{P_N}$ is the *initial marking*. We denote $\bar{N} = (N, {}^\circ N)$ and $\underline{N} = (N, N^\circ)$. A marked plain dts-box $(P_N, T_N, W_N, \Lambda_N, M_N)$ can be interpreted as the LDTSPN $(P_N, T_N, W_N, \Omega_N, L_N, M_N)$, where $\forall t \in T_N \Omega_N(t) = \Omega(\Lambda_N(t))$ (*transition probability function*) and $L_N(t) = \mathcal{L}(\Lambda_N(t))$ (*transition labeling function*).

To define a semantic function that associates a plain dts-box with every static expression of dtsPBC, we need to propose the *enumeration* function $Enu : T_N \rightarrow Num$. It associates numberings with transitions of a plain dts-box N according to those of activities. In the case of synchronization, the function associates the concatenation of the parenthesized numberings of the synchronized transitions with a resulting new transition. The structure of the plain dts-box corresponding to a static expression is constructed like in PBC [3]. I.e., we use a simultaneous refinement and relabeling meta-operator (net refinement) in addition to the *operator dts-boxes* corresponding to algebraic operations of dtsPBC and featuring transformational transition relabelings. In the definition of the denotational semantics, we apply standard constructions used for PBC. Let ϱ , Θ and u denote *relabeling*, *operator box* and *transition name* from PBC setting, respectively.

The relabeling relations $\varrho \subseteq \mathbb{N}_f^{\mathcal{S}\mathcal{L}} \times \mathcal{S}\mathcal{L}$ are defined as follows:

- $\varrho_{id} = \{(\{(\alpha, \rho)\}, (\alpha, \rho)) \mid (\alpha, \rho) \in \mathcal{S}\mathcal{L}\}$ is the *identity relabeling* keeping the interface as it is;
- $\varrho_{(\alpha, \rho)} = \{(\emptyset, (\alpha, \rho))\}$ is the *constant relabeling* that can be identified with $(\alpha, \rho) \in \mathcal{S}\mathcal{L}$ itself;
- $\varrho_{[f]} = \{(\{(\alpha, \rho)\}, (f(\alpha), \rho)) \mid (\alpha, \rho) \in \mathcal{S}\mathcal{L}\}$;
- $\varrho_{rs\ a} = \{(\{(\alpha, \rho)\}, (\alpha, \rho)) \mid (\alpha, \rho) \in \mathcal{S}\mathcal{L}, a, \hat{a} \notin \alpha\}$;

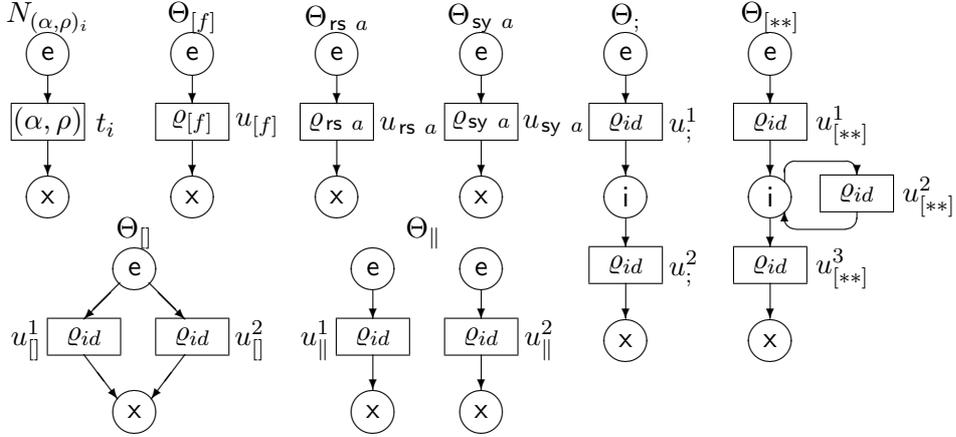


Figure 1. The plain and operator dts-boxes

- $\varrho_{sy a}$ is the least relabeling relation containing in ϱ_{id} s.t. if $(\Gamma, (\alpha, \rho)), (\Delta, (\beta, \chi)) \in \varrho_{sy a}$ and $a \in \alpha, \hat{a} \in \beta$, then $(\Gamma + \Delta, (\alpha \oplus_a \beta, \rho \cdot \chi)) \in \varrho_{sy a}$.

The plain and operator dts-boxes are presented in Figure 1. The symbol i is usually omitted.

Now we define the enumeration function Enu . Let $Box_{dts}(E) = (P_E, T_E, W_E, \Lambda_E)$ be the plain dts-box of a static expression E , and Enu_E be the enumeration function for T_E . The notation for F and K is similar.

- $Box_{dts}(E \circ F) = \Theta_{\circ}(Box_{dts}(E), Box_{dts}(F))$, $\circ \in \{;, [], \parallel\}$. Since we do not introduce new transitions, we preserve the initial numbering:

$$Enu(t) = \begin{cases} Enu_E(t), & t \in T_E; \\ Enu_F(t), & t \in T_F. \end{cases}$$
- $Box_{dts}(E[f]) = \Theta_{[f]}(Box_{dts}(E))$. Since we only replace the labels of some multiactions by a bijection, we preserve the initial numbering:
 $Enu(t) = Enu_E(t)$, $t \in T_E$.
- $Box_{dts}(E \text{ rs } a) = \Theta_{rs a}(Box_{dts}(E))$. Since we remove all transitions labeled with multiactions containing a or \hat{a} , this does not change the numbering of the remaining transitions:
 $Enu(t) = Enu_E(t)$, $t \in T_E$, $a, \hat{a} \notin \mathcal{L}(\Lambda_E(t))$.
- $Box_{dts}(E \text{ sy } a) = \Theta_{sy a}(Box_{dts}(E))$. Then $\forall v, w \in T_E$ s.t. $\Lambda_E(v) = (\alpha, \rho)$, $\Lambda_E(w) = (\beta, \chi)$, $a \in \alpha, \hat{a} \in \beta$, the new transition t resulting from synchronization of v and w has the label $\Lambda(t) = (\alpha \oplus_a \beta, \rho \cdot \chi)$ and the numbering $Enu(t) = (Enu_E(v))(Enu_E(w))$. The enumeration is

$$Enu(t) = \begin{cases} Enu_E(t), & t \in T_E; \\ (Enu_E(v))(Enu_E(w)), & t \text{ results from synchronization} \\ & \text{of } v \text{ and } w. \end{cases}$$
- $Box_{dts}([E * F * K]) = \Theta_{[**]}(Box_{dts}(E), Box_{dts}(F), Box_{dts}(K))$. Since

we do not introduce new transitions, we preserve the initial numbering:

$$Enu(t) = \begin{cases} Enu_E(t), & t \in T_E; \\ Enu_F(t), & t \in T_F; \\ Enu_K(t), & t \in T_K. \end{cases}$$

Definition 11. Let $(\alpha, \rho) \in \mathcal{SL}$, $a \in Act$ and $E, F, K \in RegStatExpr$. The *denotational semantics* of dtsPBC is a mapping Box_{dts} from $RegStatExpr$ into the area of plain dts-boxes defined as follows:

1. $Box_{dts}((\alpha, \rho)_i) = N_{(\alpha, \rho)_i}$;
2. $Box_{dts}(E \circ F) = \Theta_{\circ}(Box_{dts}(E), Box_{dts}(F))$, $\circ \in \{;, [], \|\}$;
3. $Box_{dts}(E[f]) = \Theta_{[f]}(Box_{dts}(E))$;
4. $Box_{dts}(E \circ a) = \Theta_{\circ a}(Box_{dts}(E))$, $\circ \in \{rs, sy\}$;
5. $Box_{dts}([E * F * K]) = \Theta_{[**]}(Box_{dts}(E), Box_{dts}(F), Box_{dts}(K))$.

For $E \in RegStatExpr$, let $Box_{dts}(\overline{E}) = \overline{Box_{dts}(E)}$ and $Box_{dts}(\underline{E}) = \underline{Box_{dts}(E)}$. The definition of dts-boxes is compositional in the sense that any dynamic expression can be decomposed in some inner dynamic and static ones for which we can apply the definition again.

Let \simeq denote isomorphism between transition systems or DTMCs and reachability graphs. For a dts-box N , we denote its *reachability graph* by $RG(N)$ and its *underlying DTMC* by $DTMC(N)$.

Theorem 1. [8] For any static expression E

$$TS(\overline{E}) \simeq RG(Box_{dts}(\overline{E})).$$

Proposition 1. [8] For any static expression E

$$DTMC(\overline{E}) \simeq DTMC(Box_{dts}(\overline{E})).$$

5. Stochastic equivalences

Now we propose stochastic equivalences of expressions. Since the semantic equivalence $=_{ts}$ is often too discriminate, we need some weaker relations.

To identify processes with intuitively similar behaviour and to apply standard constructions and techniques, we should abstract from infinite internal behaviour. Since dtsPBC is a stochastic extension of a finite part of PBC with iteration, the only source of infinite silent behaviour are empty loops, i.e., the transitions which are labeled by the empty multiset of activities and do not change states. During such an abstraction, we should collect the probabilities of empty loops. Note that the resulting probabilities are

those defined for an infinite number of empty steps. In the following, we explain how to abstract from the empty loops both in the algebraic setting of dtsPBC and in the net one of LDTSPNs.

5.1. Empty loops

Let G be a dynamic expression. A transition system $TS(G)$ can have loops going from a state to itself which are labeled by the empty multiset and have non-zero probability. Such *empty loops* $s \xrightarrow{\emptyset} s$ appear when no activities occur at a time step, and this happens with some positive probability. Obviously, in this case the current state remains unchanged.

Let G be a dynamic expression and $s \in DR(G)$. The *probability to stay in s due to k ($k \geq 1$) empty loops* is $(PT(\emptyset, s))^k$. Let $\Gamma \in Exec(s) \setminus \{\emptyset\}$. The *probability to execute the non-empty multiset of activities Γ in s after possible empty loops* is

$$PT^*(\Gamma, s) = PT(\Gamma, s) \sum_{k=0}^{\infty} (PT(\emptyset, s))^k = \frac{PT(\Gamma, s)}{1 - PT(\emptyset, s)}.$$

The value $k = 0$ in the summation above is for the case when no empty loops occur. We have $PT^*(\Gamma, s) \leq 1$, hence, it is really a probability, since $PT(\emptyset, s) + PT(\Gamma, s) \leq PT(\emptyset, s) + \sum_{\Delta \in Exec(s) \setminus \{\emptyset\}} PT(\Delta, s) = \sum_{\Delta \in Exec(s)} PT(\Delta, s) = 1$. Moreover, $PT^*(\Gamma, s)$ defines a probability distribution, i.e., $\forall s \in DR(G) \sum_{\Gamma \in Exec(s) \setminus \{\emptyset\}} PT^*(\Gamma, s) = 1$.

Definition 12. The (*labeled probabilistic*) transition system without empty loops $TS^*(G)$ has the state space $DR(G)$ and the transitions $s \xrightarrow{\Gamma} \tilde{s}$, if $s \xrightarrow{\Gamma} \tilde{s}$, $\Gamma \neq \emptyset$ and $\mathcal{P} = PT^*(\Gamma, s)$.

We write $s \xrightarrow{\Gamma} \tilde{s}$ if $\exists \mathcal{P} s \xrightarrow{\mathcal{P}} \tilde{s}$ and $s \rightarrow \tilde{s}$ if $\exists \Gamma s \xrightarrow{\Gamma} \tilde{s}$. For a one-element multiset of activities $\Gamma = \{(\alpha, \rho)\}$, we write $s \xrightarrow{(\alpha, \rho)} \tilde{s}$ and $s \xrightarrow{(\alpha, \rho)} \tilde{s}$.

Definition 13. Two dynamic expressions G and G' are *equivalent w.r.t. transition systems without empty loops*, denoted by $G =_{ts^*} G'$, if $TS^*(G) \simeq TS^*(G')$.

Definition 14. The *underlying DTMC without empty loops* $DTMC^*(G)$ has the state space $DR(G)$ and the transitions $s \xrightarrow{\mathcal{P}} \tilde{s}$, if $s \rightarrow \tilde{s}$, where $\mathcal{P} = PM^*(s, \tilde{s})$ is the *probability to move from s to \tilde{s} by executing any non-empty multiset of activities after possible empty loops* defined as

$$PM^*(s, \tilde{s}) = \sum_{\{\Gamma | s \xrightarrow{\Gamma} \tilde{s}\}} PT^*(\Gamma, s).$$

Let $N = (P_N, T_N, W_N, \Omega_N, L_N, M_N)$ be an LDTSPN and $M, \tilde{M} \in \mathbb{N}_f^{P_N}$, $U \subseteq T_N$. The probabilities $PT^*(U, M)$, $PM^*(M, \tilde{M})$, the transition relations $M \xrightarrow{U} \tilde{M}$, $M \xrightarrow{U} \tilde{M}$, $M \rightarrow \tilde{M}$, $M \rightarrow_P \tilde{M}$, the *reachability graph without empty loops* $RG^*(N)$ and the *underlying DTMC without empty loops* $DTMC^*(N)$ are defined like those for dynamic expressions.

Theorem 2. *For any static expression E*

$$TS^*(\bar{E}) \simeq RG^*(Box_{dts}(\bar{E})).$$

Proof. For the qualitative behaviour, we have the same isomorphism as in PBC. The quantitative behaviour is the same, since the activities of an expression have probability parts coinciding with the probabilities of the transitions belonging to the corresponding dts-box and, both in stochastic processes specified by expressions and in dts-boxes, conflicts are resolved via similar probability functions.

Proposition 2. *For any static expression E*

$$DTMC^*(\bar{E}) \simeq DTMC^*(Box_{dts}(\bar{E})).$$

Proof. By Theorem 2 and definitions of underlying DTMC for dynamic expressions and LDTSPNs, since transition probabilities of the associated DTMCs are the sums of those from transition systems or reachability graphs.

Theorem 2 guarantees that the net versions of algebraic equivalences could be defined. For every equivalence on the transition system without empty loops of a dynamic expression, a similarly defined analogue exists on the reachability graph without empty loops of the corresponding dts-box.

5.2. Stochastic bisimulation equivalences

Bisimulation equivalences respect completely the particular points of choice in the behaviour of a modeled system. We intend to present a definition of stochastic bisimulation equivalences parameterized for the cases of interleaving and step semantics.

When concurrency aspects are not relevant, the interleaving semantics is considered which abstracts from steps with more than one element. For this, a special *interleaving transition relation* is proposed. Let G be a dynamic expression, $s, \tilde{s} \in DR(G)$ and $s \xrightarrow{(\alpha, \rho)} \tilde{s}$. We write $s \xrightarrow{(\alpha, \rho)}_{\mathcal{P}} \tilde{s}$, where $\mathcal{P} =$

$pt^*((\alpha, \rho), s)$ is the probability to execute the activity (α, ρ) in s after possible empty loops, when only one-element steps are allowed, defined as

$$pt^*((\alpha, \rho), s) = \frac{PT^*({(\alpha, \rho)}, s)}{\sum_{\{(\beta, \chi)\} \in Exec(s)} PT^*({(\beta, \chi)}, s)}.$$

Let G be a dynamic expression and $\mathcal{H} \subseteq DR(G)$. Then for some $s \in DR(G)$ and $A \in \mathcal{N}_f^{\mathcal{L}} \setminus \{\emptyset\}$ we write $s \xrightarrow{A} \mathcal{P} \mathcal{H}$, where $\mathcal{P} = PM_A^*(s, \mathcal{H})$ is the overall probability to move from s into the set of states \mathcal{H} via non-empty steps with the multiaction part A after possible empty loops defined as

$$PM_A^*(s, \mathcal{H}) = \sum_{\{\Gamma \mid \exists \bar{s} \in \mathcal{H} \ s \xrightarrow{\Gamma} \bar{s}, \mathcal{L}(\Gamma) = A\}} PT^*(\Gamma, s).$$

We propose the corresponding interleaving transition relation $s \xrightarrow{\alpha} \mathcal{P} \mathcal{H}$, where $\mathcal{P} = pm_\alpha^*(s, \mathcal{H})$ is the overall probability to move from s into the set of states \mathcal{H} via steps with the multiaction part $\{\alpha\}$ after possible empty loops when only one-element steps are allowed defined as

$$pm_\alpha^*(s, \mathcal{H}) = \sum_{\{(\alpha, \rho) \mid \exists \bar{s} \in \mathcal{H} \ s \xrightarrow{(\alpha, \rho)} \bar{s}\}} pt^*((\alpha, \rho), s).$$

Definition 15. An equivalence relation $\mathcal{R} \subseteq (DR(G) \cup DR(G'))^2$ is a \star -stochastic bisimulation between dynamic expressions G and G' , $\star \in \{\text{interleaving, step}\}$, denoted by $\mathcal{R} : G \xleftrightarrow{\star} G'$, $\star \in \{i, s\}$, if:

1. $([G]_{\approx}, [G']_{\approx}) \in \mathcal{R}$.
2. $(s_1, s_2) \in \mathcal{R} \Rightarrow \forall \mathcal{H} \in (DR(G) \cup DR(G'))/\mathcal{R}$
 - $\forall x \in \mathcal{L}$ and $\hookrightarrow \Rightarrow \dashrightarrow$, if $\star = i$;
 - $\forall x \in \mathcal{N}_f^{\mathcal{L}} \setminus \{\emptyset\}$ and $\hookrightarrow \Rightarrow \dashrightarrow$, if $\star = s$;

$$s_1 \xrightarrow{x} \mathcal{H} \Leftrightarrow s_2 \xrightarrow{x} \mathcal{H}.$$

Two dynamic expressions G and G' are \star -stochastic bisimulation equivalent, $\star \in \{\text{interleaving, step}\}$, denoted by $G \xleftrightarrow{\star} G'$, if $\exists \mathcal{R} : G \xleftrightarrow{\star} G'$, $\star \in \{i, s\}$.

5.3. Interrelations of the stochastic equivalences

Now we compare the introduced stochastic equivalences and obtain the lattice of their interrelations.

Theorem 3. For any dynamic expressions G and G' the following strict implications and the identity hold:

$$G \approx G' \Rightarrow G =_{ts} G' \Leftrightarrow G =_{ts*} G' \Rightarrow G \xleftrightarrow{ss} G' \Rightarrow G \xleftrightarrow{is} G'.$$

Proof. Like the analogous theorem in [9], but for infinite processes as well.

6. Stationary behaviour

Let us examine how the proposed equivalences can be used to compare behaviour of stochastic processes in their steady states. We shall consider only formulas specifying stochastic processes with infinite behavior, i.e., expressions with the iteration operator. We have observed that, for every such expression, the parts of its underlined DTMC corresponding to the execution of the iteration bodies are ergodic, hence, the unique steady state exists.

6.1. Theoretical background

Let G be a dynamic expression. The elements \mathcal{P}_{ij}^* ($1 \leq i, j \leq n = |DR(G)|$) of the transition probability matrix (TPM) \mathbf{P}^* for $DTMC^*(G)$ are defined as

$$\mathcal{P}_{ij}^* = \begin{cases} PM^*(s_i, s_j), & s_i \twoheadrightarrow s_j; \\ 0, & \text{otherwise.} \end{cases}$$

The transient (k -step, $k \in \mathbb{N}$) probability mass function (PMF) $\psi^*[k] = (\psi_1^*[k], \dots, \psi_n^*[k])$ for $DTMC^*(G)$ is the solution of the equation system

$$\psi^*[k] = \psi^*[0](\mathbf{P}^*)^k,$$

s.t. $\psi^*[0] = (\psi_1^*[0], \dots, \psi_n^*[0])$ is the initial PMF: $\psi_i^*[0] = \begin{cases} 1, & s_i = [G]_{\approx}; \\ 0, & \text{otherwise.} \end{cases}$

Note also that $\psi^*[k+1] = \psi^*[k]\mathbf{P}^*$ ($k \in \mathbb{N}$).

The steady-state PMF $\psi^* = (\psi_1^*, \dots, \psi_n^*)$ for $DTMC^*(G)$ is the solution of the equation system

$$\begin{cases} \psi^*(\mathbf{P}^* - \mathbf{E}) = \mathbf{0} \\ \psi^* \mathbf{1}^T = 1 \end{cases},$$

where \mathbf{E} is the unitary matrix of size n and $\mathbf{0}$ is a vector with n values 0, $\mathbf{1}$ is that with n values 1.

If $DTMC^*(G)$ has the single steady state, then $\psi^* = \lim_{k \rightarrow \infty} \psi^*[k]$.

For $s \in DR(G)$ with $s = s_i$ ($1 \leq i \leq n$) we define $\psi^*[k](s) = \psi_i^*[k]$ ($k \in \mathbb{N}$) and $\psi^*(s) = \psi_i^*$.

The following standard *performance indices* can be calculated based on the steady-state PMF ψ^* .

- The *average recurrence (return) time in the state s* is $\frac{1}{\psi^*(s)}$.

- The fraction of residence time in the state s is $\psi^*(s)$.
- The relative fraction of residence time in the state s_1 w.r.t. that in the state s_2 is $\frac{\psi^*(s_1)}{\psi^*(s_2)}$.
- The fraction of residence time in the set of states $S \subseteq DR(G)$ or the probability of the event determined by a condition that is true for all states from S is $\sum_{s \in S} \psi^*(s)$.
- The steady-state probability to perform a step with an activity (α, ρ) is $\sum_{s \in DR(G)} \psi^*(s) \sum_{\{\Gamma | (\alpha, \rho) \in \Gamma\}} PT^*(\Gamma, s)$.
- The probability of the event determined by a reward function r on the states is $\sum_{s \in DR(G)} \psi^*(s) r(s)$.

6.2. Steady state and equivalences

The following proposition demonstrates that for two dynamic expressions related by \leftrightarrow_{ss} the steady-state probabilities to come in an equivalence class coincide. One can also interpret the result stating that the mean recurrence time for an equivalence class is the same for both expressions.

Proposition 3. *Let G, G' be dynamic expressions with $\mathcal{R} : G \leftrightarrow_{ss} G'$. Then $\forall \mathcal{H} \in (DR(G) \cup DR(G'))/\mathcal{R}$*

$$\sum_{s \in \mathcal{H} \cap DR(G)} \psi^*(s) = \sum_{s' \in \mathcal{H} \cap DR(G')} \psi'^*(s').$$

Proof. See Appendix A.

The expression $\text{Stop} = (\{c\}, \frac{1}{2}) \text{ rs } c$ specifies the process that is only able to perform empty loops with probability 1 and never terminates. The following example demonstrates that Proposition 3 does not hold for \leftrightarrow_{is} .

Example 1. Let $E = [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2}) \| (\{d\}, \frac{1}{2}))) * \text{Stop}]$ and $E' = [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2})_1; (\{d\}, \frac{1}{2})_1) \| ((\{d\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2))] * \text{Stop}$. We have $\overline{E} \leftrightarrow_{is} \overline{E'}$.

$DR(\overline{E})$ consists of the equivalence classes

$$\begin{aligned} s_1 &= [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2}) \| (\{d\}, \frac{1}{2}))) * \text{Stop}] \approx, \\ s_2 &= [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2}) \| (\{d\}, \frac{1}{2}))) * \text{Stop}] \approx, \\ s_3 &= [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2}) \| (\{d\}, \frac{1}{2}))) * \text{Stop}] \approx, \\ s_4 &= [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2}) \| (\{d\}, \frac{1}{2}))) * \text{Stop}] \approx, \\ s_5 &= [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2}) \| (\{d\}, \frac{1}{2}))) * \text{Stop}] \approx. \end{aligned}$$

$DR(\overline{E'})$ consists of the equivalence classes

$$s'_1 = [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2})_1; (\{d\}, \frac{1}{2})_1) \| ((\{d\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2))] * \text{Stop} \approx,$$

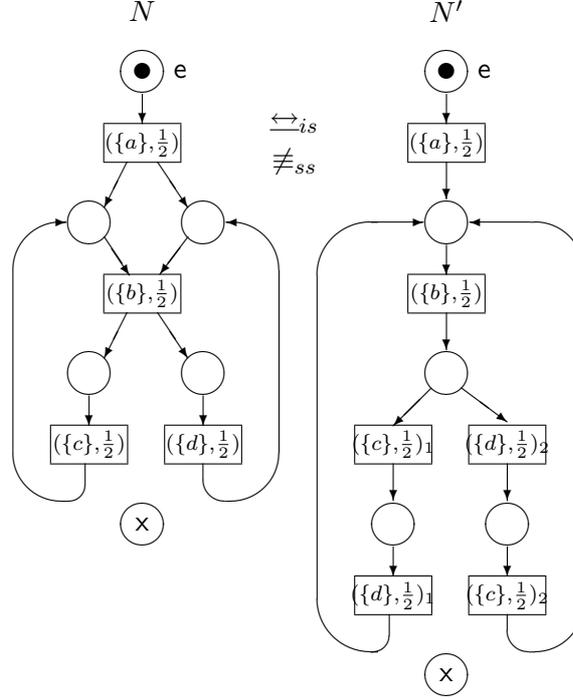


Figure 2. \leftrightarrow_{is} does not guarantee a coincidence of steady-state probabilities to come in an equivalence class

$$s'_2 = [[[\{a\}, \frac{1}{2}] * \overline{(\{b\}, \frac{1}{2})}; ((\{c\}, \frac{1}{2})_1; (\{d\}, \frac{1}{2})_1) \square ((\{d\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2)]] * \text{Stop}] \approx,$$

$$s'_3 = [[[\{a\}, \frac{1}{2}] * \overline{(\{b\}, \frac{1}{2})}; ((\{c\}, \frac{1}{2})_1; (\{d\}, \frac{1}{2})_1) \square ((\{d\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2)]] * \text{Stop}] \approx,$$

$$s'_4 = [[[\{a\}, \frac{1}{2}] * \overline{(\{b\}, \frac{1}{2})}; ((\{c\}, \frac{1}{2})_1; (\{d\}, \frac{1}{2})_1) \square ((\{d\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2)]] * \text{Stop}] \approx,$$

$$s'_5 = [[[\{a\}, \frac{1}{2}] * \overline{(\{b\}, \frac{1}{2})}; ((\{c\}, \frac{1}{2})_1; (\{d\}, \frac{1}{2})_1) \square ((\{d\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2)]] * \text{Stop}] \approx.$$

The steady-state PMFs ψ^* for $DTMC^*(\bar{E})$ and ψ'^* for $DTMC^*(\bar{E}')$ are

$$\psi^* = \left(0, \frac{3}{8}, \frac{3}{8}, \frac{1}{8}, \frac{1}{8} \right), \quad \psi'^* = \left(0, \frac{1}{3}, \frac{1}{3}, \frac{1}{6}, \frac{1}{6} \right).$$

For the equivalence class (w.r.t. \leftrightarrow_{is}) $\mathcal{H} = \{s_3, s'_3\}$ we have $\sum_{s \in \mathcal{H} \cap DR(\bar{E})} \psi^*(s) = \psi^*(s_3) = \frac{3}{8}$, whereas $\sum_{s' \in \mathcal{H} \cap DR(\bar{E}')} \psi'^*(s') = \psi'^*(s'_3) = \frac{1}{3}$. Thus, \leftrightarrow_{is} does not guarantee a coincidence of steady-state probabilities to come in an equivalence class.

In Figure 2, the marked dts-boxes of the dynamic expressions above are presented, i.e., $N = \text{Box}_{dts}(\bar{E})$ and $N' = \text{Box}_{dts}(\bar{E}')$. In addition, we depict the net analogues of the algebraic equivalences.

By Proposition 3, \leftrightarrow_{ss} preserves the quantitative properties of the stationary behaviour. Now we intend to demonstrate that the qualitative properties of the stationary behaviour based of the multiaction labels are pre-

served as well.

Definition 16. A *step trace* of a dynamic expression G is $\Sigma = A_1 \cdots A_n \in (\mathcal{N}_f^c \setminus \{\emptyset\})^*$ s.t. $\exists s \in DR(G)$ $s \xrightarrow{\Gamma_1} s_1 \xrightarrow{\Gamma_2} \cdots \xrightarrow{\Gamma_n} s_n$, $\mathcal{L}(\Gamma_i) = A_i$ ($1 \leq i \leq n$). The *probability to execute the step trace Σ in s* is

$$PT^*(\Sigma, s) = \sum_{\{\Gamma_1, \dots, \Gamma_n | s \xrightarrow{\Gamma_1} s_1 \xrightarrow{\Gamma_2} \cdots \xrightarrow{\Gamma_n} s_n, \mathcal{L}(\Gamma_i) = A_i \ (1 \leq i \leq n)\}} \prod_{i=1}^n PT^*(\Gamma_i, s_{i-1}).$$

The following theorem demonstrates that for two dynamic expressions related by \leftrightarrow_{ss} the steady-state probabilities to come in an equivalence class and start a step trace from it coincide.

Theorem 4. Let G, G' be dynamic expressions with $\mathcal{R} : G \leftrightarrow_{ss} G'$ and Σ be a step trace. Then $\forall \mathcal{H} \in (DR(G) \cup DR(G'))/\mathcal{R}$

$$\sum_{s \in \mathcal{H} \cap DR(G)} \psi^*(s) PT^*(\Sigma, s) = \sum_{s' \in \mathcal{H} \cap DR(G')} \psi'^*(s') PT^*(\Sigma, s').$$

Proof. See Appendix B.

Example 2. Let $E = [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2})_1 \square (\{c\}, \frac{1}{2})_2)) * \text{Stop}]$ and $E' = [(\{a\}, \frac{1}{2}) * (((\{b\}, \frac{1}{2})_1; (\{c\}, \frac{1}{2})_1) \square ((\{b\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2)) * \text{Stop}]$. We have $\overline{E} \leftrightarrow_{ss} \overline{E'}$.

$DR(\overline{E})$ consists of the equivalence classes

$$s_1 = [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2})_1 \square (\{c\}, \frac{1}{2})_2)) * \text{Stop}]_{\approx},$$

$$s_2 = [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2})_1 \square (\{c\}, \frac{1}{2})_2)) * \text{Stop}]_{\approx},$$

$$s_3 = [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2})_1 \square (\{c\}, \frac{1}{2})_2)) * \text{Stop}]_{\approx}.$$

$DR(\overline{E'})$ consists of the equivalence classes

$$s'_1 = [(\{a\}, \frac{1}{2}) * (((\{b\}, \frac{1}{2})_1; (\{c\}, \frac{1}{2})_1) \square ((\{b\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2)) * \text{Stop}]_{\approx},$$

$$s'_2 = [(\{a\}, \frac{1}{2}) * (((\{b\}, \frac{1}{2})_1; (\{c\}, \frac{1}{2})_1) \square ((\{b\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2)) * \text{Stop}]_{\approx},$$

$$s'_3 = [(\{a\}, \frac{1}{2}) * (((\{b\}, \frac{1}{2})_1; (\{c\}, \frac{1}{2})_1) \square ((\{b\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2)) * \text{Stop}]_{\approx},$$

$$s'_4 = [(\{a\}, \frac{1}{2}) * (((\{b\}, \frac{1}{2})_1; (\{c\}, \frac{1}{2})_1) \square ((\{b\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2)) * \text{Stop}]_{\approx}.$$

The steady-state PMFs ψ^* for $DTMC^*(\overline{E})$ and ψ'^* for $DTMC^*(\overline{E'})$ are

$$\psi^* = \left(0, \frac{1}{2}, \frac{1}{2}\right), \quad \psi'^* = \left(0, \frac{1}{2}, \frac{1}{4}, \frac{1}{4}\right).$$

One can see that the steady-state probabilities for the equivalence class (w.r.t. \leftrightarrow_{ss}) $\mathcal{H} = \{s_3, s'_3, s'_4\}$ coincide: $\sum_{s \in \mathcal{H} \cap DR(\overline{E})} \psi^*(s) = \psi^*(s_3) =$

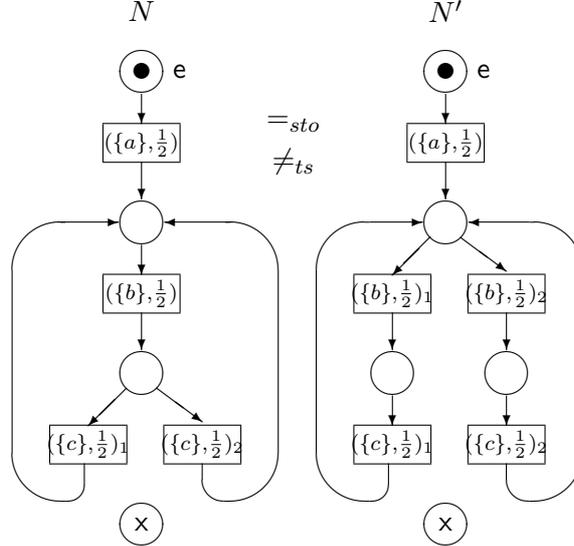


Figure 3. \xleftrightarrow{ss} implies a coincidence of the steady-state probabilities to come in an equivalence class and start a trace from it

$\frac{1}{2} = \frac{1}{4} + \frac{1}{4} = \psi'^*(s'_3) + \psi'^*(s'_4) = \sum_{s' \in \mathcal{H} \cap DR(\overline{E'})} \psi'^*(s')$. Let $\Sigma = \{\{c\}\}$. The steady-state probabilities to come in the equivalence class \mathcal{H} and start the step trace Σ from it coincide as well: $\psi^*(s_3)(PT^*({(\{c\}, \frac{1}{2})_1}, s_3) + PT^*({(\{c\}, \frac{1}{2})_2}, s_3)) = \frac{1}{2} (\frac{1}{2} + \frac{1}{2}) = \frac{1}{2} = \frac{1}{4} \cdot 1 + \frac{1}{4} \cdot 1 = \psi'^*(s'_3)PT^*({(\{c\}, \frac{1}{2})_1}, s'_3) + \psi'^*(s'_4)PT^*({(\{c\}, \frac{1}{2})_2}, s'_4)$.

In Figure 3, the marked dts-boxes of the dynamic expressions above are presented, i.e., $N = Box_{dts}(\overline{E})$ and $N' = Box_{dts}(\overline{E'})$. In addition, we depict the net analogues of the algebraic equivalences.

6.3. Performance preservation and simplification of its analysis

Many performance indices are based on the steady-state probabilities to come in a set of similar states or, after coming in, to start a step trace from this set. The similarity of states is usually captured by an equivalence relation, hence, the sets are often the equivalence classes. For example, the average system run-through, the common memory utilization for the shared memory system and the time fractions when nobody, one or two persons eat for the dining philosophers system are based on the steady-state probabilities to come in an equivalence class. On the other hand, the steady-state probability of the common memory request from a processor for the shared memory system and that of the beginning of eating for the dining philosophers system are based on steady-state probabilities to come in an equivalence class and to start a step trace from it. Performance indices of the shared memory system were calculated in [10].

Proposition 3 and Theorem 4 guarantee a coincidence of the mentioned indices for the expressions related by \leftrightarrow_{ss} . Thus, \leftrightarrow_{ss} preserves performance of stochastic systems modeled by expressions of dtsPBC. Moreover, Example 1 demonstrates that it is the weakest relation we considered that has the performance preservation property. The equivalence \leftrightarrow_{ss} can be used to reduce transition systems and, hence, DTMCs of expressions. It allows one to simplify the performance analysis, since the reduced DTMCs have less states, the dimension of the transition probability matrix is smaller, and we solve less equations to calculate steady-state probabilities.

7. Conclusion

In this paper, within dtsPBC with iteration, we have considered the stochastic algebraic equivalences having natural net analogues on LDTSPNs. The equivalences abstract from empty loops in transition systems of dynamic expressions. We have investigated which of the equivalences we proposed guarantee identity of the stationary behaviour and can be used for simplification of performance analysis. We have proved that the weakest of the relations having this property is the step stochastic bisimulation equivalence. The advantage of our framework is twofold. First, one can specify concurrent composition and synchronization of (multi)actions unlike Markov chains. Second, algebraic formulas represent processes in a more compact way than Petri nets.

In the future, we plan to apply the equivalence notions to reduction of expressions and boxes with the intention to propose a method of performance analysis simplification.

References

- [1] Best E., Devillers R., Hall J.G. The box calculus: a new causal algebra with multi-label communication // Lect. Notes Comput. Sci. – 1992. – Vol. 609. – P. 21–69.
- [2] Bernardo M., Gorrieri R. A tutorial on EMPA: a theory of concurrent processes with nondeterminism, priorities, probabilities and time // Theor. Comput. Sci. – 1998. – Vol. 202. – P. 1–54.
- [3] Best E., Koutny M. A refined view of the box algebra // Lect. Notes Comp. Sci. – 1995. – Vol. 935. – P. 1–20.
- [4] Hillston J. A Compositional Approach to Performance Modelling. – Cambridge University Press, Great Britain, 1996.
- [5] Macià H., Valero V., Cazorla D., Cuartero F. Introducing the iteration in sPBC // Lect. Notes Comp. Sci. – 2004. – Vol. 3235. – P. 292–308. – <http://www.info-ab.uclm.es/retics/publications/2004/forte04.pdf>

- [6] Macià H., Valero V., de Frutos D. sPBC: a Markovian extension of finite Petri box calculus // Proc. of 9th IEEE Internat. Workshop on Petri Nets and Performance Models - 01 (PNPM'01). – Aachen, Germany: IEEE Computer Society Press, 2001. – P. 207–216. – <http://www.info-ab.uclm.es/retics/publications/2001/pnpm01.ps>
- [7] Tarasyuk I.V. Discrete Time Stochastic Petri Box Calculus. – Carl von Ossietzky Universität Oldenburg, Germany, 2005. – 25 p. – (Berichte aus dem Department für Informatik; Vol. 3/05). – http://db.iis.nsk.su/persons/itar/dtspbcib_cov.pdf
- [8] Tarasyuk I.V. Iteration in discrete time stochastic Petri box calculus // Bull. Novosibirsk Comp. Center. Ser. Computer Science. – Novosibirsk, 2006. – Iss. 24. – P. 129–148. – <http://db.iis.nsk.su/persons/itar/dtsitncc.pdf>
- [9] Tarasyuk I.V. Stochastic Petri box calculus with discrete time // Fundamenta Informaticae. – IOS Press, Amsterdam, The Netherlands, 2007. – Vol. 76, No. 1–2. – P. 189–218. – <http://db.iis.nsk.su/persons/itar/dtspbcfi.pdf>
- [10] Tarasyuk I.V. Performance evaluation in dtsPBC // Proc. of 18th Workshop on Concurrency, Specification and Programming - 09 (CS&P'09). – Warsaw University, Poland, 2009. – P. 602–615. – <http://db.iis.nsk.su/persons/itar/dtsshmcsp.pdf>

A. Proof of Proposition 3

It is sufficient to prove the statement of the proposition for transient PMFs only, since $\psi^* = \lim_{k \rightarrow \infty} \psi^*[k]$ and $\psi'^* = \lim_{k \rightarrow \infty} \psi'^*[k]$. We proceed by induction on k .

- $k = 0$

The only nonzero values of the initial PMFs of $DTMC^*(G)$ and $DTMC^*(G')$ are $\psi^*[0]([G]_{\approx})$ and $\psi'^*[0]([G']_{\approx})$. The only equivalence class containing $[G]_{\approx}$ or $[G']_{\approx}$ is $\mathcal{H}_0 = \{[G]_{\approx}, [G']_{\approx}\}$. Thus,

$$\sum_{s \in \mathcal{H}_0 \cap DR(G)} \psi^*[0](s) = \psi^*[0]([G]_{\approx}) = 1 = \psi'^*[0]([G']_{\approx}) = \sum_{s' \in \mathcal{H}_0 \cap DR(G')} \psi'^*[0](s').$$

As for other equivalence classes, $\forall \mathcal{H} \in ((DR(G) \cup DR(G'))/\mathcal{R}) \setminus \mathcal{H}_0$ we have $\sum_{s \in \mathcal{H} \cap DR(G)} \psi^*[0](s) = 0 = \sum_{s' \in \mathcal{H} \cap DR(G')} \psi'^*[0](s')$.

- $k \rightarrow k + 1$

Let $\mathcal{H} \in (DR(G) \cup DR(G'))/\mathcal{R}$ and $s_1, s_2 \in \mathcal{H}$. We have $\forall \tilde{\mathcal{H}} \in (DR(G) \cup DR(G'))/\mathcal{R} \ \forall A \in \mathcal{N}_f^{\mathcal{L}} \setminus \{\emptyset\} \ s_1 \xrightarrow{A}_{\mathcal{P}} \tilde{\mathcal{H}} \Leftrightarrow s_2 \xrightarrow{A}_{\mathcal{P}} \tilde{\mathcal{H}}$. Therefore, $PM^*(s_1, \tilde{\mathcal{H}}) = \sum_{\{\Gamma \mid \exists \tilde{s}_1 \in \tilde{\mathcal{H}} \ s_1 \xrightarrow{\Gamma} \tilde{s}_1\}} PT^*(\Gamma, s_1) = \sum_{A \in \mathcal{N}_f^{\mathcal{L}} \setminus \{\emptyset\}} \sum_{\{\Gamma \mid \exists \tilde{s}_1 \in \tilde{\mathcal{H}} \ s_1 \xrightarrow{\Gamma} \tilde{s}_1, \mathcal{L}(\Gamma) = A\}} PT^*(\Gamma, s_1) =$

$\sum_{A \in \mathcal{N}_f^c \setminus \{\emptyset\}} PM_A^*(s_1, \tilde{\mathcal{H}}) = \sum_{A \in \mathcal{N}_f^c \setminus \{\emptyset\}} PM_A^*(s_2, \tilde{\mathcal{H}}) =$
 $\sum_{A \in \mathcal{N}_f^c \setminus \{\emptyset\}} \sum_{\{\Gamma | \exists \tilde{s}_2 \in \tilde{\mathcal{H}} \ s_2 \xrightarrow{\Gamma} \tilde{s}_2, \mathcal{L}(\Gamma) = A\}} PT^*(\Gamma, s_2) =$
 $\sum_{\{\Gamma | \exists \tilde{s}_2 \in \tilde{\mathcal{H}} \ s_2 \xrightarrow{\Gamma} \tilde{s}_2\}} PT^*(\Gamma, s_2) = PM^*(s_2, \tilde{\mathcal{H}})$. Since we have the previous equality for all $s_1, s_2 \in \mathcal{H}$, we can denote $PM^*(\mathcal{H}, \tilde{\mathcal{H}}) = PM^*(s_1, \tilde{\mathcal{H}}) = PM^*(s_2, \tilde{\mathcal{H}})$. Note that transitions from the states of $DR(G)$ always lead to those from the same set, hence, $\forall s \in DR(G)$ $PM^*(s, \tilde{\mathcal{H}}) = PM^*(s, \tilde{\mathcal{H}} \cap DR(G))$. The same is true for $DR(G')$.

By induction hypothesis, $\sum_{s \in \mathcal{H} \cap DR(G)} \psi^*[k](s) =$
 $\sum_{s' \in \mathcal{H} \cap DR(G')} \psi'^*[k](s')$. Further, $\sum_{\tilde{s} \in \tilde{\mathcal{H}} \cap DR(G)} \psi^*[k+1](\tilde{s}) =$
 $\sum_{\tilde{s} \in \tilde{\mathcal{H}} \cap DR(G)} \sum_{s \in DR(G)} \psi^*[k](s) PM^*(s, \tilde{s}) =$
 $\sum_{s \in DR(G)} \sum_{\tilde{s} \in \tilde{\mathcal{H}} \cap DR(G)} \psi^*[k](s) PM^*(s, \tilde{s}) =$
 $\sum_{s \in DR(G)} \psi^*[k](s) \sum_{\tilde{s} \in \tilde{\mathcal{H}} \cap DR(G)} PM^*(s, \tilde{s}) =$
 $\sum_{\mathcal{H}} \sum_{s \in \mathcal{H} \cap DR(G)} \psi^*[k](s) \sum_{\tilde{s} \in \tilde{\mathcal{H}} \cap DR(G)} PM^*(s, \tilde{s}) =$
 $\sum_{\mathcal{H}} \sum_{s \in \mathcal{H} \cap DR(G)} \psi^*[k](s) \sum_{\tilde{s} \in \tilde{\mathcal{H}} \cap DR(G)} \sum_{\{\Gamma | s \xrightarrow{\Gamma} \tilde{s}\}} PT^*(\Gamma, s) =$
 $\sum_{\mathcal{H}} \sum_{s \in \mathcal{H} \cap DR(G)} \psi^*[k](s) \sum_{\{\Gamma | \exists \tilde{s} \in \tilde{\mathcal{H}} \cap DR(G) \ s \xrightarrow{\Gamma} \tilde{s}\}} PT^*(\Gamma, s) =$
 $\sum_{\mathcal{H}} \sum_{s \in \mathcal{H} \cap DR(G)} \psi^*[k](s) PM^*(s, \tilde{\mathcal{H}}) =$
 $\sum_{\mathcal{H}} \sum_{s \in \mathcal{H} \cap DR(G)} \psi^*[k](s) PM^*(\mathcal{H}, \tilde{\mathcal{H}}) =$
 $\sum_{\mathcal{H}} PM^*(\mathcal{H}, \tilde{\mathcal{H}}) \sum_{s \in \mathcal{H} \cap DR(G)} \psi^*[k](s) =$
 $\sum_{\mathcal{H}} PM^*(\mathcal{H}, \tilde{\mathcal{H}}) \sum_{s' \in \mathcal{H} \cap DR(G')} \psi'^*[k](s') =$
 $\sum_{\mathcal{H}} \sum_{s' \in \mathcal{H} \cap DR(G')} \psi'^*[k](s') PM^*(\mathcal{H}, \tilde{\mathcal{H}}) =$
 $\sum_{\mathcal{H}} \sum_{s' \in \mathcal{H}' \cap DR(G')} \psi'^*[k](s') PM^*(s', \tilde{\mathcal{H}}) =$
 $\sum_{\mathcal{H}} \sum_{s' \in \mathcal{H} \cap DR(G')} \psi'^*[k](s') \sum_{\{\Gamma | \exists \tilde{s}' \in \tilde{\mathcal{H}} \cap DR(G') \ s' \xrightarrow{\Gamma} \tilde{s}'\}} PT^*(\Gamma, s') =$
 $\sum_{\mathcal{H}} \sum_{s' \in \mathcal{H} \cap DR(G')} \psi'^*[k](s') \sum_{\tilde{s}' \in \tilde{\mathcal{H}} \cap DR(G')} \sum_{\{\Gamma | \exists \tilde{s}' \ s' \xrightarrow{\Gamma} \tilde{s}'\}} PT^*(\Gamma, s') =$
 $\sum_{\mathcal{H}} \sum_{s' \in \mathcal{H} \cap DR(G')} \psi'^*[k](s') \sum_{\tilde{s}' \in \tilde{\mathcal{H}} \cap DR(G')} PM^*(s', \tilde{s}') =$
 $\sum_{s' \in DR(G')} \psi'^*[k](s') \sum_{\tilde{s}' \in \tilde{\mathcal{H}} \cap DR(G')} PM^*(s', \tilde{s}') =$
 $\sum_{s' \in DR(G')} \sum_{\tilde{s}' \in \tilde{\mathcal{H}} \cap DR(G')} \psi'^*[k](s') PM^*(s', \tilde{s}') =$
 $\sum_{\tilde{s}' \in \tilde{\mathcal{H}} \cap DR(G')} \sum_{s' \in DR(G')} \psi'^*[k](s') PM^*(s', \tilde{s}') =$
 $\sum_{\tilde{s}' \in \tilde{\mathcal{H}} \cap DR(G')} \psi'^*[k+1](\tilde{s}')$.

B. Proof of Theorem 4

Let $\mathcal{H} \in (DR(G) \cup DR(G'))/\mathcal{R}$ and $s, \bar{s} \in \mathcal{H}$. We have $\forall \tilde{\mathcal{H}} \in (DR(G) \cup DR(G'))/\mathcal{R} \ \forall A \in \mathcal{N}_f^c \setminus \{\emptyset\} \ s \xrightarrow{A} \tilde{\mathcal{H}} \Leftrightarrow \bar{s} \xrightarrow{A} \tilde{\mathcal{H}}$. The previous equality is valid for all $s, \bar{s} \in \mathcal{H}$, hence, we can rewrite it as $\mathcal{H} \xrightarrow{A} \tilde{\mathcal{H}}$ and denote $PM_A^*(\mathcal{H}, \tilde{\mathcal{H}}) = PM_A^*(s, \tilde{\mathcal{H}}) = PM_A^*(\bar{s}, \tilde{\mathcal{H}})$. The transitions from

the states of $DR(G)$ always lead to those from the same set, hence, $\forall s \in DR(G) PM_A^*(s, \mathcal{H}) = PM_A^*(s, \tilde{\mathcal{H}} \cap DR(G))$. The same is true for $DR(G')$.

Let $\Sigma = A_1 \cdots A_n$ be a step trace of G and G' . We have $\exists \mathcal{H}_0, \dots, \exists \mathcal{H}_n \in (DR(G) \cup DR(G'))/\mathcal{R}$ $\mathcal{H}_0 \xrightarrow{A_1}_{\mathcal{P}_1} \mathcal{H}_1 \xrightarrow{A_2}_{\mathcal{P}_2} \dots \xrightarrow{A_n}_{\mathcal{P}_n} \mathcal{H}_n$. Let us prove that the sum of probabilities of all the paths starting in every $s_0 \in \mathcal{H}_0$ and going through the states from $\mathcal{H}_1, \dots, \mathcal{H}_n$ is equal to the product of $\mathcal{P}_1, \dots, \mathcal{P}_n$:

$$\sum_{\{\Gamma_1, \dots, \Gamma_n | s_0 \xrightarrow{\Gamma_1} \dots \xrightarrow{\Gamma_n} s_n, \mathcal{L}(\Gamma_i) = A_i, s_i \in \mathcal{H}_i (1 \leq i \leq n)\}} \prod_{i=1}^n PT^*(\Gamma_i, s_{i-1}) = \prod_{i=1}^n PM_{A_i}^*(\mathcal{H}_{i-1}, \mathcal{H}_i).$$

We prove this equality by induction on the step trace length n .

- $n = 1$

$$\sum_{\{\Gamma_1 | s_0 \xrightarrow{\Gamma_1} s_1, \mathcal{L}(\Gamma_1) = A_1, s_1 \in \mathcal{H}_1\}} PT^*(\Gamma_1, s_0) = PM_{A_1}^*(s_0, \mathcal{H}_1) = PM_{A_1}^*(\mathcal{H}_0, \mathcal{H}_1).$$

- $n \rightarrow n + 1$

$$\begin{aligned} & \sum_{\{\Gamma_1, \dots, \Gamma_n, \Gamma_{n+1} | s_0 \xrightarrow{\Gamma_1} \dots \xrightarrow{\Gamma_n} s_n \xrightarrow{\Gamma_{n+1}} s_{n+1}, \mathcal{L}(\Gamma_i) = A_i, s_i \in \mathcal{H}_i (1 \leq i \leq n+1)\}} \prod_{i=1}^{n+1} PT^*(\Gamma_i, s_{i-1}) = \\ & \sum_{\{\Gamma_{n+1} | s_n \xrightarrow{\Gamma_{n+1}} s_{n+1}, \mathcal{L}(\Gamma_{n+1}) = A_{n+1}, s_n \in \mathcal{H}_n, s_{n+1} \in \mathcal{H}_{n+1}\}} \prod_{i=1}^{n+1} PT^*(\Gamma_i, s_{i-1}) PT^*(\Gamma_{n+1}, s_n) = \\ & \sum_{\{\Gamma_1, \dots, \Gamma_n | s_0 \xrightarrow{\Gamma_1} \dots \xrightarrow{\Gamma_n} s_n, \mathcal{L}(\Gamma_i) = A_i, s_i \in \mathcal{H}_i (1 \leq i \leq n)\}} \left[\prod_{i=1}^n PT^*(\Gamma_i, s_{i-1}) PT^*(\Gamma_{n+1}, s_n) \right] = \\ & \sum_{\{\Gamma_1, \dots, \Gamma_n | s_0 \xrightarrow{\Gamma_1} \dots \xrightarrow{\Gamma_n} s_n, \mathcal{L}(\Gamma_i) = A_i, s_i \in \mathcal{H}_i (1 \leq i \leq n)\}} \prod_{i=1}^n PT^*(\Gamma_i, s_{i-1}) PM_{A_{n+1}}^*(s_n, \mathcal{H}_{n+1}) = \\ & \sum_{\{\Gamma_1, \dots, \Gamma_n | s_0 \xrightarrow{\Gamma_1} \dots \xrightarrow{\Gamma_n} s_n, \mathcal{L}(\Gamma_i) = A_i, s_i \in \mathcal{H}_i (1 \leq i \leq n)\}} \prod_{i=1}^n PT^*(\Gamma_i, s_{i-1}) PM_{A_{n+1}}^*(\mathcal{H}_n, \mathcal{H}_{n+1}) = \\ & PM_{A_{n+1}}^*(\mathcal{H}_n, \mathcal{H}_{n+1}) \sum_{\{\Gamma_1, \dots, \Gamma_n | s_0 \xrightarrow{\Gamma_1} \dots \xrightarrow{\Gamma_n} s_n, \mathcal{L}(\Gamma_i) = A_i, s_i \in \mathcal{H}_i (1 \leq i \leq n)\}} \prod_{i=1}^n PT^*(\Gamma_i, s_{i-1}) = \\ & PM_{A_{n+1}}^*(\mathcal{H}_n, \mathcal{H}_{n+1}) \prod_{i=1}^n PM_{A_i}^*(\mathcal{H}_{i-1}, \mathcal{H}_i) = \prod_{i=1}^{n+1} PM_{A_i}^*(\mathcal{H}_{i-1}, \mathcal{H}_i). \end{aligned}$$

Let $s_0, \bar{s}_0 \in \mathcal{H}_0$. We have $PM^*(A_1 \cdots A_n, s_0) =$

$$\begin{aligned} & \sum_{\{\Gamma_1, \dots, \Gamma_n | s_0 \xrightarrow{\Gamma_1} \dots \xrightarrow{\Gamma_n} s_n, \mathcal{L}(\Gamma_i) = A_i, (1 \leq i \leq n)\}} \prod_{i=1}^n PT^*(\Gamma_i, s_{i-1}) = \\ & \sum_{\mathcal{H}_1, \dots, \mathcal{H}_n} \sum_{\{\Gamma_1, \dots, \Gamma_n | s_0 \xrightarrow{\Gamma_1} \dots \xrightarrow{\Gamma_n} s_n, \mathcal{L}(\Gamma_i) = A_i, s_i \in \mathcal{H}_i (1 \leq i \leq n)\}} \prod_{i=1}^n PT^*(\Gamma_i, s_{i-1}) = \end{aligned}$$

$$\begin{aligned}
& \sum_{\mathcal{H}_1, \dots, \mathcal{H}_n} \prod_{i=1}^n PM_{A_i}^*(\mathcal{H}_{i-1}, \mathcal{H}_i) = \\
& \sum_{\mathcal{H}_1, \dots, \mathcal{H}_n} \sum_{\{\bar{\Gamma}_1, \dots, \bar{\Gamma}_n | \bar{s}_0 \xrightarrow{\bar{\Gamma}_1} \dots \xrightarrow{\bar{\Gamma}_n} \bar{s}_n, \mathcal{L}(\bar{\Gamma}_i) = A_i, \bar{s}_i \in \mathcal{H}_i \ (1 \leq i \leq n)\}} \prod_{i=1}^n PT^*(\bar{\Gamma}_i, \bar{s}_{i-1}) = \\
& \sum_{\{\bar{\Gamma}_1, \dots, \bar{\Gamma}_n | \bar{s}_0 \xrightarrow{\bar{\Gamma}_1} \dots \xrightarrow{\bar{\Gamma}_n} \bar{s}_n, \mathcal{L}(\bar{\Gamma}_i) = A_i, \ (1 \leq i \leq n)\}} \prod_{i=1}^n PT^*(\bar{\Gamma}_i, \bar{s}_{i-1}) = \\
& PM^*(A_1 \cdots A_n, \bar{s}_0).
\end{aligned}$$

Since we have the previous equality for all $s_0, \bar{s}_0 \in \mathcal{H}_0$, we can denote $PM^*(A_1 \cdots A_n, \mathcal{H}_0) = PM^*(A_1 \cdots A_n, s_0) = PM^*(A_1 \cdots A_n, \bar{s}_0)$.

By Proposition 3, $\sum_{s \in \mathcal{H} \cap DR(G)} \psi^*(s) = \sum_{s' \in \mathcal{H} \cap DR(G')} \psi'^*(s')$. Now we can complete the proof: $\sum_{s \in \mathcal{H} \cap DR(G)} \psi^*(s) PT^*(\Sigma, s) = \sum_{s \in \mathcal{H} \cap DR(G)} \psi^*(s) PT^*(\Sigma, \mathcal{H}) = PT^*(\Sigma, \mathcal{H}) \sum_{s \in \mathcal{H} \cap DR(G)} \psi^*(s) = PT^*(\Sigma, \mathcal{H}) \sum_{s' \in \mathcal{H} \cap DR(G')} \psi'^*(s') = \sum_{s' \in \mathcal{H} \cap DR(G')} \psi'^*(s') PT^*(\Sigma, \mathcal{H}) = \sum_{s' \in \mathcal{H} \cap DR(G')} \psi'^*(s') PT^*(\Sigma, s')$.