CARL VON OSSIETZKY universität OLDENBURG

BERICHTE

AUS DEM DEPARTMENT FÜR INFORMATIK
der Fakultät II - Informatik, Wirtschafts- und Rechtswissenschaften

Herausgeber: Die Professorinnen und Professoren des Departments für Informatik

Equivalence relations for behaviour-preserving reduction and modular performance evaluation in *dtsPBC*

Dr. Igor Tarasyuk

Bericht

Nummer 01/10 – April 2010

ISSN 1867-9218

CARL VON OSSIETZKY universität OLDENBURG

BERICHTE

AUS DEM DEPARTMENT FÜR INFORMATIK

der Fakultät II - Informatik, Wirtschafts- und Rechtswissenschaften

Herausgeber: Die Professorinnen und Professoren
des Departments für Informatik

Equivalence relations for behaviour-preserving reduction and modular performance evaluation in *dtsPBC*

Dr. Igor Tarasyuk

Bericht

Nummer 01/10 – April 2010

ISSN 1867-9218

Equivalence relations for behaviour-preserving reduction and modular performance evaluation in $dtsPBC$ *

Igor V. Tarasyuk

A.P. Ershov Institute of Informatics Systems
Siberian Branch of the Russian Academy of Sciences
6, Acad. Lavrentiev ave., 630090 Novosibirsk, Russia
itar@iis.nsk.su

Abstract

In the last decades, a number of stochastic enrichments of process algebras was constructed to specify stochastic processes within the well-developed framework of algebraic calculi. In 2003, a continuous time stochastic extension $sPBC$ of finite Petri box calculus (PBC) was enriched with iteration operator by H.S. Macià, V.R. Valero, D.L. Cazorla and F.G. Cuartero. In 2006, the author added iteration to the discrete time stochastic extension $dtsPBC$ of finite PBC. In this paper, in the framework of the $dtsPBC$ with iteration, we define a variety of stochastic equivalences. They allow one to identify stochastic processes with similar behaviour that are however differentiated by the standard semantic equivalence of the calculus. The interrelations of all the introduced equivalences are investigated. It is explained how the equivalences we propose can be used to reduce transition systems of expressions. A logical characterization of the equivalences is presented via formulas of the new probabilistic modal logics. We demonstrate how to apply the equivalences to compare stationary behaviour. A problem of preservation of the equivalences by algebraic operations is discussed. As a result, we define an equivalence that is a congruence relation. At last, two case studies of performance evaluation in the algebra are presented.

Keywords: stochastic Petri net, stochastic process algebra, Petri box calculus, iteration, discrete time, transition system, operational semantics, dts-box, denotational semantics, empty loop, stochastic equivalence, reduction, modal logic, stationary behaviour, congruence relation, performance evaluation.

Contents

*This work was supported in part by Deutsche Forschungsgemeinschaft (DFG), grant 436 RUS 113/1002/01, and Russian Foundation for Basic Research (RFBR), grant 09-01-91334.

1 Introduction

Stochastic Petri nets (SPNs) are a well-known model for quantitative analysis of discrete dynamic event systems proposed initially in [57]. Essentially, SPNs are a high level language for specification and performance analysis of concurrent systems. A stochastic process corresponding to this formal model is a Markov chain generated and analyzed by well-developed algorithms and methods. Firing probabilities distributed along continuous or discrete time scale are associated with transitions of an SPN. Thus, there exist SPNs with continuous [57, 35] and discrete [58] time. Markov chains of the corresponding types are associated with the SPNs. As a rule, for SPNs with continuous time (CTSPNs), exponential or phase distributions of transition probabilities are used. For SPNs with discrete time (DTSPNs), geometric or combinations of geometric distributions are usually used. Transitions of CTSPNs fire one by one at continuous time moments. Hence, the semantics of this model is an interleaving one. In this semantics, parallel computations are modeled by all possible execution sequences of their components. Transitions of DTSPNs fire concurrently in steps at discrete time moments. Hence, this model has a step semantics. In this semantics, parallel computations are modeled by sequences of concurrent occurrences (steps) of their components. In [26, 27], a labeling for transitions of CTSPNs with action names was proposed. The labeling allows SPNs to model processes with functionally similar components: the transitions corresponding to the similar components are labeled by the same action. Moreover, one can compare labeled SPNs by different behavioural equivalences, and this makes possible to check stochastic processes specified by labeled SPNs for functional similarity. Therefore, one can compare both functional and performance properties, and labeled SPNs turn into a formalism for quantitative and qualitative analysis.

Algebraic calculi occupy a special place among formal models for specification of concurrent systems and analysis of their behavioral properties. In such process algebras (PAs), a system or a process is specified by an algebraic formula. Verification of the properties is accomplished at a syntactic level by means of well-developed systems of equivalences, axioms and inference rules. The most well-known of the first PAs are Theory of Communicating Sequential Processes ($TCSP$) [41] and Calculus of Communicating Systems (CCS) [56]. Process algebras have been acknowledged to be very suitable formalism to operate with real time and stochastic systems as well. In the last years, stochastic extensions of PAs, called stochastic process algebras (SPAs), became very popular as a modeling framework. SPAs do not just specify actions which can happen (qualitative features) as usual process algebras, but they associate some quantitative parameters with actions (quantitative characteristics). The most popular SPAs proposed so far are Markovian Timed Processes for Performance Evaluation ($MTIPP$) [42], Performance Evaluation Process Algebra ($PEPA$) [39] and Extended Markovian Process Algebra ($EMPA$) [13].

In $MTIPP$, every activity is a pair consisting of the action name (including the symbol τ for the *internal*, invisible action) and the parameter of exponential distribution of the activity duration (the *rate*). The operations are *prefix*, *choice*, *parallel* composition including *synchronization* on the specified action set and *recursion*. It is possible to specify processes by recursive equations as well. The interleaving semantics is defined on the basis of Markovian (i.e., extended with the specification of rates) labeled transition systems. Note that we have the interleaving behaviour here because the exponential probability distribution function is a continuous one, and a simultaneous firing of any two activities has zero probability according to the properties of continuous time distributions. The continuous time Markov chains (CTMCs) can be derived from the mentioned transition systems to analyze the performance issues.

In $PEPA$, activities are the pairs consisting of action types (including the *unknown*, unimportant type τ) and activity rates. The rate is either the parameter of exponential distribution of the activity duration or it is *unspecified*, denoted by $\top$. An activity with unspecified rate is *passive* by its action type. The set of operations includes *prefix*, *choice*, *cooperation*, *hiding* and constants whose meaning is given by the defining equations including the *recursive* ones. The cooperation is accomplished on the set of action types (the cooperation set) on which the components must *synchronize* or cooperate. If the cooperation set is empty, the cooperation operator turns into the *parallel* combinator. The semantics is interleaving, it is defined via the extension of labeled transition systems with a possibility to specify activity rates. Based on the transition systems, the continuous time Markov processes (CTMPs) are generated which are used for performance evaluation with the help of the embedded continuous time Markov chains (ECTMCs).

In $EMPA$, each action is a pair consisting of its type and rate. Actions can be *external* or *internal* (denoted by τ) according to types. There are three kinds of actions according to rates: *timed* ones with exponentially distributed durations (essentially, the actions from $MTIPP$ and $PEPA$), *immediate* ones with priorities and weights (the actions analogous to immediate transitions of generalized SPNs, GSPNs) and *passive* ones (similar to passive actions of $PEPA$). Timed actions specify activities that are relevant for performance analysis. Immediate actions model logical events and the activities that are irrelevant from the performance viewpoint or much faster than others. Passive actions model activities waiting for the synchronization with timed of immediate ones, and express nondeterministic choice. The set of operators consist of *prefix*, functional *abstraction*, functional *relabeling*, *alternative* composition and *parallel* composition ones. Parallel composition includes *synchronization* on the set of action types like in $TCSP$. The syntax also includes *recursive* definitions given by means of constants. The semantics is interleaving and based on the labeled transition systems enriched with the information on action rates. For the exponentially timed kernel of the algebra (the sublanguage including only exponentially timed and passive actions), it is possible to construct CTMCs from the transition systems of the process terms to analyze the performance.

An extension of CCS with probabilities and time, called $TPCCS$, was defined in [38]. An enrichment of Basic Process Algebra (BPA) with probabilistic choice, $prBPA$, as well as extension of the latter with parallel composition operator named ACP_π^+ have been proposed in [1]. A stochastic process calculus Priced Process Algebra (PPA) based on CCS was constructed in [71, 74]. The papers [24, 32, 76, 16] propose a variety of other SPAs. A standard way for probabilistic extension of process algebras into the calculi of probabilistic transition systems was described in [43].

One can see that the stochastic process calculi proposed in the literature are based on interleaving, as a rule. As a semantic domain, the interleaving formalism of transition systems is often used. Therefore, investigation of a stochastic extension for more expressive and powerful algebraic calculi is an important issue. At present, the development of step or "true concurrency" (such that parallelism is considered as a causal independence) SPAs is in the very beginning. At the same time, there does not yet exist an algebra of infinite concurrent stochastic processes.

Process algebras allow one to specify processes in a compositional way via an expressive formal syntax. On the other hand, Petri nets provide one with an ability for visual representation of a process structure and execution. Hence, the relationship between SPNs and SPAs is of particular interest. To combine advantages of both models, a semantics of algebraic formulas in terms of Petri nets is usually defined. In the stochastic case, the Markov chain of the stochastic process specified by an SPA formula is built based on the state transition graph of the corresponding SPN.

Petri box calculus (PBC) is a flexible and expressive process algebra based on calculi CCS [56]. Note that some operations of PBC are similar to those of the algebra Algebra of Finite Processes (AFP_0) [47]. PBC was proposed fifteen years ago [5], and it was well explored since that time [4, 19, 29, 46, 48, 17, 18, 30, 31, 33, 40, 6, 7, 44, 8, 9, 10, 11]. Its goal was to propose a compositional semantics for high level constructs of concurrent programming languages in terms of elementary Petri nets. Thus, PBC serves as a bridge between theory and applications. Formulas of PBC are combined not from single actions (including the invisible one) and variables only, like in CCS, but from multisets of elementary actions and their conjugates, called multiactions (*basic formulas*) as well. The empty multiset of actions is allowed that is considered as the silent multiaction specifying some invisible or internal activity. In contrast to CCS, concurrency and synchronization are different operations (*concurrent constructs*). Synchronization is defined as a unary multi-way stepwise operation based on communication of actions and their conjugates. The CCS approach with conjugate matching labels was extended to define synchronization in PBC. This approach was preferred as being more flexible and compositional than that of the process algebras $TCSP$ and $COSY$ [12] where synchronization is accomplished over common action names. Moreover, synchronization operation of PBC is asynchronous in contrast to the approach of Synchronous CCS ($SCCS$) [56] where it is synchronous. The other fundamental operations are sequence and choice (*sequential constructs*). The calculus includes also restriction and relabeling (*abstraction constructs*). To specify infinite processes, refinement, recursion and iteration operations were added (*hierarchical constructs*). Thus, unlike CCS, the algebra PBC has an additional iteration construction to specify infiniteness in the cases when finite Petri nets can be used as the semantic interpretation. For PBC, a denotational semantics was proposed in terms of a subclass of Petri nets equipped with interface and considered up to isomorphism. This subclass is called Petri boxes. The calculus PBC has a step operational semantics in terms of labeled transition systems based on the structural operational semantics (SOS) rules. Pomset operational semantics of PBC was defined in [48] such that the partial order information was extracted from "decorated" step traces. In these step sequences, multiactions were annotated with an information on the relative position of the expression part they were derived from. More detailed comparison of PBC with other well-known process algebras can be found in [5, 8]. Last years, several extensions of PBC were presented.

A time extension of PBC, called time Petri box calculus ($tPBC$), was proposed in [49]. In $tPBC$, timing information is added by combining instantaneous multiactions and time delays. Its denotational semantics was defined in terms of a subclass of labeled time Petri nets (tPNs), called time Petri boxes (ct-boxes). $tPBC$ has an interleaving time operational semantics in terms of labeled transition systems. Another time enrichment of PBC, called Timed Petri box calculus ($TPBC$), was defined in [54, 55]. In contrast to $tPBC$, multiactions of $TPBC$ are not instantaneous but have time durations. Additionally, in $TPBC$ there exist no "illegal" multiaction occurrences unlike $tPBC$. The complexity of "illegal" occurrences mechanism was one of the main intentions to construct $TPBC$ though the calculus appeared to be more complicated than $tPBC$. The denotational semantics of $TPBC$ was defined in terms of a subclass of labeled Timed Petri nets (TPNs), called Timed Petri boxes (T-boxes). $TPBC$ has a step timed operational semantics in terms of labeled transition systems. Note that $tPBC$ and $TPBC$ differ in ways they capture time information, and they are not in competition but complement each other. The third time extension of PBC, called arc time Petri box calculus ($atPBC$), was constructed in [73, 72]. In $atPBC$, multiactions are associated with time delay intervals. Its denotational semantics was defined on a subclass of arc time Petri nets (atPNs), where time restrictions associated with the arcs, called arc time Petri boxes (at-boxes). $atPBC$ possesses a step operational semantics in terms of labeled transition systems.

A stochastic extension of PBC, called stochastic Petri box calculus ($sPBC$), was proposed in [68, 69, 70, 59, 64, 65, 66, 52]. In $sPBC$, multiactions have stochastic durations that follow negative exponential distribution. Each multiaction is instantaneous and equipped with a rate that is a parameter of the corresponding exponential distribution. The execution of a multiaction is possible only after the corresponding stochastic time delay. Only a finite part of PBC was used for the stochastic enrichment. This means that $sPBC$ has neither refinement nor recursion nor iteration operations. Its denotational semantics was defined in terms of a subclass of labeled CTSPNs, called stochastic Petri boxes (s-boxes). Calculus $sPBC$ has an interleaving operational semantics in terms of labeled transition systems. Current research in this branch has an aim to extend the specification abilities of $sPBC$ and to define appropriate congruence relation over algebraic formulas. The results on constructing the iteration for $sPBC$ were reported in [61, 62]. In the papers [60, 63], a number of new equivalence relations were proposed for regular terms of $sPBC$ to choose later a suitable candidate for a congruence. In [67], the special multiactions with zero time delay were added to $sPBC$. A denotational semantics of such an $sPBC$ extension was defined via a subclass of labeled generalized SPNs (GSPNs). The subclass is called generalized stochastic Petri boxes (gs-boxes).

An ambient extension of PBC, called Ambient Petri box calculus ($APBC$), was proposed in [34]. Ambient calculus is used to model behaviour of mobile systems. Ambient is a named environment delimited by a boundary. The ambients can be moved to a new location thus modeling mobility. The algebra $APBC$ includes ambients and mobility capabilities. Hence, it could be interpreted as an extension of the Ambient Calculus with the operations of PBC. Basic actions of $APBC$ are capabilities defined over ambient names and standard multiactions of PBC. Only finite part of PBC was taken for the ambient enrichment. Moreover, only concurrency and sequence were transferred into $APBC$ from the set of PBC operations in [34]. This reduced algebra was called Simple Ambient Petri box calculus ($SAPBC$). A denotational semantics was defined in terms of

Elementary Object Systems (EOSs) that are two-level net systems composed from a system net and object nets. Object nets could be interpreted as high-level tokens of the system net modeling the execution of mobilie processes. The calculus $SAPBC$ has a step operational semantics in terms of labeled transition systems.

Nevertheless, there were no stochastic extension of PBC with step semantics until recent times. It can be done with the use of labeled DTSPNs as a semantic area, since discrete time models allow for concurrent action occurrences. The enrichment based on DTSPNs is natural because PBC has a step operational semantics.

A notion of equivalence is very important in formal theory of computing processes and systems. Behavioural equivalences are applied during verification stage both to compare behaviour of systems and reduce their structure. At present time, there exists a great diversity of different equivalence notions for concurrent systems, and their interrelations were well explored in the literature. The most popular and widely used one is bisimulation. Unfortunately, the mentioned behavioural equivalences take into account only functional (qualitative) but not performance (quantitative) aspects of system behaviour. Additionally, the equivalences are often interleaving ones, and they do not respect concurrency. SPAs inherited from untimed PAs a possibility to apply equivalences for comparison of specified processes. Like equivalences for other stochastic models, the relations for SPAs have special requirements due to the summation of probabilities. The states from which similar future behaviours start have to be grouped into equivalence classes. The classes form elements of the aggregated state space, and they are defined a posteriori while searching for equivalences on state space of a model. Interleaving probabilistic weak trace equivalence was proposed in [28] on labeled probabilistic transition systems and in [89] it was defined on labeled CTMCs. Interleaving probabilistic strong bisimulation equivalence was proposed in [20, 50, 51] on labeled probabilistic transition systems, in [42] on labeled CTMCs and in [39] on probabilistic process algebras. Interleaving probabilistic equivalences were defined for probabilistic processes in [37]. Interleaving probabilistic weak bisimulation equivalence was introduced in [24] on Markovian process algebras, in [25] on stochastic automata, in [26] on labeled CTSPNs and in [27] on GSPNs. Interleaving probabilistic weak and strong bisimulation equivalences were proposed in [14] on labeled probabilistic transition systems and in [15] they were defined on labeled DTMCs and CTMCs. In [16], a notion of interleaving stochastic weak bisimulation equivalence for process terms was introduced. The authors proved that the equivalence is preserved by formula composition within SPAs considered in the paper, i.e., the relation is a congruence. At the same time, no appropriate equivalence notion was defined for concurrent SPAs so far. Thus, it is desirable to propose an equivalence relation for parallel SPAs that relates formulas specifying processes with similar behavior and differentiates those having non-similar one from a certain viewpoint. It would be fine to find a relation that is a congruence with respect to the algebraic operations. In this case, the formulas combined by algebraic operations from equivalent subformulas will be equivalent as well. This is very significant property while bottom-up design of processes.

We did some work on the development of concurrent discrete time SPNs and SPAs as well as on defining a variety of concurrent probabilistic equivalences. In [21], labeled weighted DTSPNs (LWDTSPNs) were proposed that is a modification of DTSPNs by transition labeling and weights. In [23, 83], labeled DTSPNs (LDTSPNs) were introduced. Transitions of LWDTSPNs and LDTSPNs are labeled by actions which represent elementary activities and can be visible or invisible to an external observer. For these two net classes, a number of new probabilistic τ-trace and τ-bisimulation equivalences were defined that abstract from invisible actions (denoted by τ) and respect concurrency in different degrees (interleaving and step relations). In addition, probabilistic relations that require back or back-forth simulation were introduced. An application of the probabilistic back-forth τ-bisimulation equivalences to compare stationary behaviour of the LWDTSPNs or LDTSPNs was demonstrated. In [78, 23], a logical characterization was presented for interleaving and step probabilistic τ-bisimulation equivalences via formulas of the new probabilistic modal logics. The characterization means that two LWDTSPNs or LDTSPNs are (interleaving or step) probabilistic τ-bisimulation equivalent if they satisfy the same formulas of the corresponding probabilistic modal logic. Thus, instead of comparing nets operationally, one have to check the corresponding satisfaction relation only applying standard verification techniques. The new interleaving and step logics are modifications of that, called PML, was proposed in [50] on probabilistic transition systems with visible actions. In [22, 23, 83], a stochastic algebra of finite nondeterministic processes $StAFP_0$ was proposed with semantics in terms of a subclass of LWDTSPNs and LDTSPNs, called stochastic acyclic nets (SANs). The calculus defined is a stochastic extension of the algebra AFP_0 introduced in [45]. $StAFP_0$ specifies concurrent stochastic processes. Another feature of the algebra is a net semantics allowing one to preserve the level of parallelism, since Petri nets is a classical "true concurrency" model. Usually, transition systems are used for this purpose, but they are not able to respect concurrency completely. An axiomatization for the semantic equivalence of $StAFP_0$ was proposed. It was proved that any algebraic formula could be reduced to the "fully stratified" one with the use of the axiom system. This simplifies semantic comparison of formulas. In [79, 83], we considered different classes of stochastic Petri nets. We explored how transition labeling could be defined to compare SPNs by equivalences. An suitability of the SPN classes for modeling and analysis of different kinds of dynamic systems was investigated. In [80, 82], a discrete time stochastic extension $dtsPBC$ of finite PBC was constructed. A step operational and a net denotational semantics of $dtsPBC$ were defined, and their consistency was demonstrated. In addition, a variety of probabilistic equivalences were proposed to identify stochastic processes with similar behaviour which are differentiated by the semantic equivalence. The interrelations of all the introduced equivalences were studied. In [81, 84, 85, 86, 87], we constructed an enrichment of $dtsPBC$ with the iteration operator used to specify infinite processes. In [88], we presented the extension $dtsiPBC$ of the latter calculus with immediate multiactions.

Let us consider the difference between $dtsPBC$ and the classical SPAs $MTIPP$, $PEPA$ and $EMPA$. In $dtsPBC$, every activity is a pair consisting of the multiaction (not just an action, as in the classical SPAs) and its (conditional) probability (not the rate, as in the mentioned SPAs) to be executed under condition that no other multiaction can happen at the current discrete time moment. Algebra $dtsPBC$ has sequence operator in contrast to prefix one in the three SPAs which we compare with. One can combine arbitrary expressions with sequence operation, i.e., it is more flexible than the prefix one, where the first argument should be a single activity. Choice operator in $dtsPBC$ is analogous to that in $MTIPP$ and $PEPA$ as well as to the alternative composition in $EMPA$, in the sense that choice is determined by the first activity that appears, i.e., by the activity with maximal execution probability. On the other hand, concurrency and synchronization in $dtsPBC$ are the different operations (this feature is inherited from PBC) unlike the situation in the classical SPAs where parallel composition (combinator) has a synchronization capability. Relabeling in $dtsPBC$ is analogous to that in $EMPA$, but it is additionally extended to conjugated actions. Restriction operation in $dtsPBC$ is similar to hiding in $PEPA$ and functional abstraction in $EMPA$, but it is extended to conjugated actions too. The synchronization on an elementary action collects all the pairs consisting of this elementary action and its conjugate which are contained in the multiactions from the synchronized activities. The operation produces new activities such that the first element of every resulting activity is the union of the multiactions from which all the mentioned pairs of conjugated actions are removed, and the second element is the product of the probabilities of the activities involved in the synchronization. Thus, there is a difference with the way synchronization is applied in the mentioned SPAs where it is accomplished over identical action names, and every resulting activity consist of the same action name and the sum of the rates of the initial activities. Algebra $dtsPBC$ has no recursion operation or a possibility for recursive definitions, but it includes iteration operation that gives an ability to specify infinite behaviour with the explicitly defined start and termination. Iteration allows for a syntactic description of many realistic processes with loops. Calculus $dtsPBC$ has a discrete time semantics, and time delays in the states are geometrically distributed unlike the mentioned SPAs with continuous time semantics and exponentially distributed activity delays. As a consequence, the semantics of $dtsPBC$ is the step one in contrast to the interleaving semantics of the three SPAs mentioned above. The performance issues can be investigated based on the discrete time Markov chain (DTMC) extracted from the labeled probabilistic transition system associated with each expression of $dtsPBC$. Note that in the classical SPAs we generate CTMCs from the transition systems. In addition, $dtsPBC$ has a denotational semantics in terms of LDTSPNs from which the corresponding DTMCs can be derived as well. Thus, the multiaction labels and the set of very flexible and powerful operations, as well as a step operational and a Petri net denotational semantics allowing for really concurrent execution of activities (or transitions) are the main advantages of $dtsPBC$ with respect to other well-known SPAs like $MTIPP$, $PEPA$ and $EMPA$.

In this paper, we investigate equivalence notions for $dtsPBC$ with iteration. First, we present the syntax of the extended $dtsPBC$. Each multiaction of the initial calculus PBC is associated with a probability. Such a pair is called stochastic multiaction or activity. Second, we propose semantics of $dtsPBC$. The step operational semantics is constructed in terms of labeled probabilistic transition systems based on action and inaction rules. The denotational semantics is defined in terms of a subclass of LDTSPNs, called discrete time stochastic Petri boxes (dts-boxes). Consistency of operational and denotational semantics is proved. Further, we define a number of stochastic equivalences in the algebraic setting based on transition systems without empty behaviour. These relations are weaker than the semantic equivalence of $dtsPBC$. They are used to identify stochastic processes with similar behaviour which are differentiated by the semantic equivalence that is too discriminate in many cases. The interrelations diagram of all the introduced equivalences is built. We describe how the stochastic equivalences can be used to reduce transition systems of expressions and the related formalisms. We present a characterization of the stochastic bisimulation equivalences via two new probabilistic modal logics based on PML. It is demonstrated how to compare stochastic processes in their steady states with the use of the relations. Moreover, a problem of preservation of the equivalence notions by algebraic operations is discussed. The proposed equivalences are used to construct a congruence relation. At the end, we present two case studies explaining how to analyze performance of systems within the calculus. We consider algebraic models of shared memory system and dining philosophers one.

The paper is organized as follows. In the next Section 2, the syntax of the extended calculus $dtsPBC$ is presented. Then, in Section 3, we construct the operational semantics of the algebra in terms of labeled transition systems. In Section 4, we propose the denotational semantics based on a subclass of LDTSPNs.

Section 5 is devoted to the construction and the interrelations of stochastic algebraic equivalences based on transition systems without empty loops. In Section 6 we explain how one can reduce transition systems and the related formalisms modulo the equivalences. A logical characterization of the equivalences is presented in Section 7. In Section 8, an application of the relations to comparison of stationary behaviour is investigated. Preservation of the equivalences by the algebraic operations, i.e., a congruence problem is discussed in Section 9. Section 10 contains two examples of performance evaluation for systems specified by the algebraic expressions. The concluding Section 11 summarizes the results obtained and outlines research perspectives in this area.

2 Syntax

In this section, we propose the syntax of the discrete time stochastic extension of finite PBC enriched with iteration, called *discrete time stochastic Petri box calculus* ($dtsPBC$).

First, we recall a definition of multiset that is an extension of the set notion by allowing several identical elements.

Definition 2.1 *Let X be a set. A finite* multiset (bag) *M over X is a mapping $M : X \to I\!N$ such that $|\{x \in X \mid M(x) > 0\}| < \infty$, i.e., it can contain a finite number of elements only.*

We denote the *set of all finite multisets* over X by $I\!N_f^X$. The *cardinality* of a multiset M is defined as $|M| = \sum_{x\in X} M(x)$. We write $x \in M$ if $M(x) > 0$ and $M \subseteq M'$ if $\forall x \in X\ M(x) \leq M'(x)$. We define $(M + M')(x) = M(x) + M'(x)$ and $(M - M')(x) = \max\{0, M(x) - M'(x)\}$. When $\forall x \in X\ M(x) \leq 1$, M is a proper set such that $M \subseteq X$. The *set of all subsets* of X is denoted by 2^X.

Let $Act = \{a, b, \ldots\}$ be the set of *elementary actions*. Then $\widehat{Act} = \{\hat{a}, \hat{b}, \ldots\}$ is the set of *conjugated actions (conjugates)* such that $a \neq \hat{a}$ and $\hat{\hat{a}} = a$. Let $\mathcal{A} = Act \cup \widehat{Act}$ be the set of *all actions*, and $\mathcal{L} = I\!N_f^{\mathcal{A}}$ 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 that contains no visible action names. The *alphabet* of $\alpha \in \mathcal{L}$ is defined as $\mathcal{A}(\alpha) = \{x \in \mathcal{A} \mid \alpha(x) > 0\}$.

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 the probabilities of state changes (steps) at discrete time moments. The multiaction probabilities are required not to be equal to 1, since otherwise, the multiactions with probability 1 always happen in a step (i.e., they are *instantaneous*, since they have zero time delay) and all other with the less probabilities do not. In this case, technical difficulties appear with conflicts resolving, see [58]. In this version of the algebra, we do not allow instantaneous multiactions. On the other hand, there is no sense to allow zero probabilities of multiactions, since they would never be performed in this case. 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 a single 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 like in other process algebras, but parallelism does not include synchronization unlike the corresponding operation in CCS.

Relabeling functions $f : \mathcal{A} \to \mathcal{A}$ 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 $\alpha \in \mathcal{L}$ we define $f(\alpha) = \sum_{x\in\alpha} f(x)$.

Restriction over an action a means that for a given expression any process behaviour containing a or its conjugate $\hat{a}$ is not allowed.

Let $\alpha, \beta \in \mathcal{L}$ be two multiactions such that for some action $a \in Act$ we have $a \in \alpha$ and $\hat{a} \in \beta$ or $\hat{a} \in \alpha$ and $a \in \beta$. Then synchronization of α and β by a is defined as $\alpha \oplus_a \beta = \gamma$, where

$$\gamma(x) = \begin{cases} \alpha(x) + \beta(x) - 1, & x = a \text{ or } x = \hat{a}; \\ \alpha(x) + \beta(x), & \text{otherwise.} \end{cases}$$

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.

Static expressions specify the structure of processes. As we shall see, the expressions correspond to unmarked LDTSPNs (note that LDTSPNs are marked by definition).

Definition 2.2 *Let $(\alpha, \rho) \in \mathcal{SL}$ and $a \in Act$. A* static expression *of $dtsPBC$ is defined as*

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

Let *StatExpr* denote the set of *all static expressions* of *dtsPBC*.

To make the grammar above unambiguous, one can add parentheses in the productions with binary operations: $(E; E)$, $(E[]E)$, $(E\|E)$ or to associate priorities with operations. However, here and further we prefer the *PBC* approach: we add parentheses to resolve ambiguities only and we assume no priorities.

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

$$\begin{array}{rl} E ::= & (\alpha,\rho) \mid E;E \mid E[]E \mid E\|E \mid E[f] \mid E \text{ rs } a \mid E \text{ sy } a \mid [E*D*E], \\ \textit{where } D ::= & (\alpha,\rho) \mid D;E \mid D[]D \mid D[f] \mid D \text{ rs } a \mid D \text{ sy } a \mid [D*D*E]. \end{array}$$

Let *RegStatExpr* denote the set of *all regular static expressions* of *dtsPBC*.

Dynamic expressions specify the states of processes. Dynamic expressions are combined from static ones which are annotated with upper or lower bars and specify active components of the system at the current instant of time. As we shall see, dynamic expressions correspond to LDTSPNs (which are marked by default). The dynamic expression with upper bar (the overlined one) $\overline{E}$ denotes the *initial*, and that with lower bar (the underlined one) $\underline{E}$ denotes the *final* state of the process specified by a static expression E. The *underlying static expression* of a dynamic one is obtained by removing all upper and lower bars from it.

Definition 2.4 *Let $E \in StatExpr$ and $a \in Act$. A* dynamic expression *of dtsPBC is defined as*

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

Let *DynExpr* denote the set of *all dynamic expressions* of *dtsPBC*.

Note that if the underlying static expression of a dynamic one is not regular, the corresponding LDTSPN can be non-safe (though, it is 2-bounded in the worst case, see [8]). A dynamic expression is *regular* if its underlying static expression is regular.

Let *RegDynExpr* denote the set of *all regular dynamic expressions* of *dtsPBC*.

In the following, we shall consider regular static and dynamic expressions only, hence, we can omit the word "regular".

3 Operational semantics

In this section, we define the step operational semantics in terms of labeled transition systems.

3.1 Inaction rules

Inaction rules describe expression transformations due to the execution of the empty multiset of activities. The rules will be used later to define the *empty loop* transitions which reflect a non-zero probability to stay in the current state at the next time moment, which is an essential feature of discrete time stochastic processes. As we shall see, for every empty loop transition, its net analog in an LDTSPN does not change the current marking corresponding to the initial dynamic expression from the applied inaction rule.

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

Table 1: Inaction rules for overlined and underlined static expressions

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

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

Table 2: Inaction rules for arbitrary dynamic expressions

$G \stackrel{\emptyset}{\rightarrow} G$	$\frac{G\stackrel{\emptyset}{\rightarrow}\widetilde{G},\ \circ\in\{;,[]\}}{G\circ E\stackrel{\emptyset}{\rightarrow}\widetilde{G}\circ E}$	$\frac{G\stackrel{\emptyset}{\rightarrow}\widetilde{G},\ \circ\in\{;,[]\}}{E\circ G\stackrel{\emptyset}{\rightarrow}E\circ\widetilde{G}}$	$\frac{G\stackrel{\emptyset}{\rightarrow}\widetilde{G}}{G\parallel H\stackrel{\emptyset}{\rightarrow}\widetilde{G}\parallel H}$	$\frac{H\stackrel{\emptyset}{\rightarrow}\widetilde{H}}{G\parallel H\stackrel{\emptyset}{\rightarrow}G\parallel\widetilde{H}}$
$\frac{G\stackrel{\emptyset}{\rightarrow}\widetilde{G}}{G[f]\stackrel{\emptyset}{\rightarrow}\widetilde{G}[f]}$	$\frac{G\stackrel{\emptyset}{\rightarrow}\widetilde{G},\ \circ\in\{\mathsf{rs},\mathsf{sy}\}}{G\circ a\stackrel{\emptyset}{\rightarrow}\widetilde{G}\circ a}$	$\frac{G\stackrel{\emptyset}{\rightarrow}\widetilde{G}}{[G*E*F]\stackrel{\emptyset}{\rightarrow}[\widetilde{G}*E*F]}$	$\frac{G\stackrel{\emptyset}{\rightarrow}\widetilde{G}}{[E*G*F]\stackrel{\emptyset}{\rightarrow}[E*\widetilde{G}*F]}$	$\frac{G\stackrel{\emptyset}{\rightarrow}\widetilde{G}}{[E*F*G]\stackrel{\emptyset}{\rightarrow}[E*F*\widetilde{G}]}$

A regular dynamic expression G is *operative* if no inaction rule can be applied to it, with the exception of $G \stackrel{\emptyset}{\rightarrow} G$.

Let *OpRegDynExpr* denote the set of *all operative regular dynamic expressions* of *dtsPBC*.

Note that any dynamic expression can be always transformed into a (not necessarily unique) operative one by using the inaction rules.

Definition 3.1 *Let* $\approx = (\stackrel{\emptyset}{\rightarrow} \cup \stackrel{\emptyset}{\leftarrow})^*$ *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.*

Note that the rule $G \stackrel{\emptyset}{\rightarrow} G$ 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, hence, the corresponding equivalence class is the singleton one. This is a new rule that has no prototype among inaction rules of *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 later to define transitions representing the state changes when some non-empty multisets of activities are executed. As we shall see, for every such transition, its net analog in an LDTSPN changes the current marking corresponding to the initial dynamic expression from the applied action rule, unless there is a self-loop produced by the iterative execution of a non-empty multiset (which should be additionally the one-element one, i.e., the single activity, since we do not allow concurrency in the highest level of the second argument of iteration).

Let $\Gamma \in I\!N_f^{\mathcal{SL}}$. 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 action rules. In this table, $(\alpha, \rho), (\beta, \chi) \in \mathcal{SL}$, $E, F \in$ *RegStatExpr*, $G, H \in$ *OpRegDynExpr*, $\widetilde{G}, \widetilde{H} \in$ *RegDynExpr* and $a \in Act$. Moreover, $\Gamma, \Delta \in I\!N_f^{\mathcal{SL}} \setminus \{\emptyset\}$.

Table 3: Action rules

B $\overline{(\alpha,\rho)} \stackrel{\{(\alpha,\rho)\}}{\longrightarrow} \underline{(\alpha,\rho)}$	**SC1** $\frac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G},\ \circ\in\{;,[]\}}{G\circ E\stackrel{\Gamma}{\rightarrow}\widetilde{G}\circ E}$	**SC2** $\frac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G},\ \circ\in\{;,[]\}}{E\circ G\stackrel{\Gamma}{\rightarrow}E\circ\widetilde{G}}$	**P1** $\frac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G}}{G\parallel H\stackrel{\Gamma}{\rightarrow}\widetilde{G}\parallel H}$	**P2** $\frac{H\stackrel{\Gamma}{\rightarrow}\widetilde{H}}{G\parallel H\stackrel{\Gamma}{\rightarrow}G\parallel\widetilde{H}}$
P3 $\frac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G},\ H\stackrel{\Delta}{\rightarrow}\widetilde{H}}{G\parallel H\stackrel{\Gamma+\Delta}{\longrightarrow}\widetilde{G}\parallel\widetilde{H}}$	**L** $\frac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G}}{G[f]\stackrel{f(\Gamma)}{\longrightarrow}\widetilde{G}[f]}$	**Rs** $\frac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G},\ a,\hat{a}\not\in\mathcal{A}(\Gamma)}{G \text{ rs } a\stackrel{\Gamma}{\rightarrow}\widetilde{G} \text{ rs } a}$	**I1** $\frac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G}}{[G*E*F]\stackrel{\Gamma}{\rightarrow}[\widetilde{G}*E*F]}$	**I2** $\frac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G}}{[E*G*F]\stackrel{\Gamma}{\rightarrow}[E*\widetilde{G}*F]}$
I3 $\frac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G}}{[E*F*G]\stackrel{\Gamma}{\rightarrow}[E*F*\widetilde{G}]}$	**Sy1** $\frac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G}}{G \text{ sy } a\stackrel{\Gamma}{\rightarrow}\widetilde{G} \text{ sy } a}$	**Sy2** $\frac{G \text{ sy } a\stackrel{\Gamma+\{(\alpha,\rho)\}+\{(\beta,\chi)\}}{\longrightarrow}\widetilde{G} \text{ sy } a,\ a\in\alpha,\ \hat{a}\in\beta}{G \text{ sy } a\stackrel{\Gamma+\{(\alpha\oplus_a\beta,\rho\cdot\chi)\}}{\longrightarrow}\widetilde{G} \text{ sy } a}$		

Note that in the second rule for synchronization **Sy2** we multiply the probabilities of synchronized multiactions, since this corresponds to the probability of the events intersection. This is a new rule that has no analogous action rule in *PBC*.

Observe also that we do not allow a self-synchronization, i.e., a synchronization of an activity with itself. The purpose of this restriction is to avoid rather cumbersome and unexpected behaviour as well as many technical difficulties.

3.3 Transition systems

Now we intend to construct labeled probabilistic transition systems associated with dynamic expressions. The transition systems will be used to define operational semantics of expressions of *dtsPBC*.

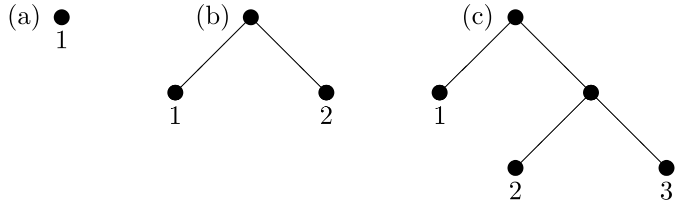

Figure 1: The binary trees encoded with the numberings 1, (1)(2) and (1)((2)(3))

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

Definition 3.2 *Let $\iota \in I\!\!N$. The* numbering *of expressions is defined as*

$$\iota ::= \ \iota \mid (\iota)(\iota).$$

Let Num denote the set of *all numberings* of expressions.

Example 3.1 *The numbering* 1 *encodes the binary tree depicted in Figure 1(a) with the root labeled by* 1*. The numbering* (1)(2) *corresponds to the binary tree depicted in Figure 1(b) without internal nodes and with two leaves labeled by* 1 *and* 2*. The numbering* (1)((2)(3)) *represents the binary tree depicted in Figure 1(c) with one internal node, which is the root for the subtree* (2)(3)*, and three leaves labeled by* 1, 2 *and* 3*.*

The new activities resulting from the applications of the second rule for synchronization **Sy2** in different orders should be considered up to the permutation of their numbering. In this way, we shall recognize the 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 $\iota \in Num$ is

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

After we apply the enumeration, the multisets of activities from the expressions will be the proper sets. In the following, we suppose that the identical activities are enumerated when it is needed to avoid ambiguity. This enumeration is considered to be implicit.

Let X be some set. We denote the cartesian product $X \times X$ by X^2. Let $\mathcal{E} \subseteq X^2$ be an equivalence relation on X. Then the *equivalence class* (with respect to $\mathcal{E}$) of an element $x \in X$ is defined by $[x]_{\mathcal{E}} = \{y \in X \mid (x, y) \in \mathcal{E}\}$. The equivalence $\mathcal{E}$ partitions X into the *set of equivalence classes* $X/_{\mathcal{E}} = \{[x]_{\mathcal{E}} \mid x \in X\}$.

Definition 3.3 *Let G be a dynamic expression. Then $[G]_\approx = \{H \mid G \approx H\}$ is the equivalence class of G with respect to the structural equivalence. The* derivation set *of a dynamic expression G, denoted by $DR(G)$, is the minimal set such that*

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

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

The set of *all the multisets of activities executable in* s is defined as $Exec(s) = \{\Gamma \mid \exists H \in s\ \exists \widetilde{H}\ H \xrightarrow{\Gamma} \widetilde{H}\}$. 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

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

Thus, $PF(\Gamma, s)$ could be interpreted as a *joint* probability of independent events. Each such an event is interpreted as readiness or not readiness for execution of a particular activity from Γ. The multiplication in the definition is used because it reflects the probability of the events intersection. When only empty multiset of activities can be executed in s, i.e., $Exec(s) = \emptyset$, we have $PF(\emptyset, s) = 1$, since we stay in s in this case. Note that the definition of $PF(\Gamma, s)$ (as well as the definitions of other probability functions which we shall present) is based on the enumeration of activities which is considered implicit.

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

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

Thus, $PT(\Gamma, s)$ is the probability that Γ is ready for execution in s *normalized* by the analagous probability for *any* multiset executable in s. The denominator of the fraction above is a sum since it reflects the probability of the events union.

Note that the sum of outgoing probabilities for the expressions belonging to the derivations of G is equal to 1. More formally, $\forall s \in DR(G)\ \sum_{\Gamma\in Exec(s)} PT(\Gamma, s) = 1$. This obviously follows from the definition of $PT(\Gamma, s)$ and guarantees that $PT(\Gamma, s)$ defines a probability distribution.

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

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

Since $PM(s, \tilde{s})$ is the probability for *any* multiset of activities (including the empty one) to change s to $\tilde{s}$, we use summation in the definition. Note that $\forall s \in DR(G)\ \sum_{\{\tilde{s} \mid \exists H\in s\ \exists\widetilde{H}\in\tilde{s}\ \exists\Gamma\ H\xrightarrow{\Gamma}\widetilde{H}\}} PM(s, \tilde{s}) = \sum_{\{\tilde{s} \mid \exists H\in s\ \exists\widetilde{H}\in\tilde{s}\ \exists\Gamma\ H\xrightarrow{\Gamma}\widetilde{H}\}} \sum_{\{\Gamma \mid \exists H\in s\ \exists\widetilde{H}\in\tilde{s}\ H\xrightarrow{\Gamma}\widetilde{H}\}} PT(\Gamma, s) = \sum_{\Gamma\in Exec(s)} PT(\Gamma, s) = 1$.

Definition 3.4 *Let G be a dynamic expression. The* (labeled probabilistic) 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 $L_G \subseteq I\!N_f^{\mathcal{SL}} \times (0; 1]$;*
- *the set of* transitions *is $\mathcal{T}_G = \{(s, (\Gamma, PT(\Gamma, s)), \tilde{s}) \mid s \in DR(G),\ \exists H \in s\ \exists\widetilde{H} \in \tilde{s}\ H \xrightarrow{\Gamma} \widetilde{H}\}$;*
- *the* initial state *is $s_G = [G]_\approx$.*

The definition of $TS(G)$ is correct, i.e., for every state the sum of the probabilities of all the transitions starting from it is 1. This is guaranteed by the note after the definition of $PT(\Gamma, s)$. Thus, we have defined the *generative* model of probabilistic processes, according to the classification from [37]. The reason is that the sum of the probabilities of the transitions with all possible labels should be equal to 1, not only of those with the same labels (up to enumeration of activities they include) as in the *reactive* models, and we do not have the nested choice as in the *stratified* models.

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 $s \xrightarrow{\Gamma}_{\mathcal{P}} \tilde{s}$. It is interpreted as follows: 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 $s \xrightarrow{\emptyset}_{\mathcal{P}} s$ from a state s to itself in the result of executing the empty multiset. This corresponds to the 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 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 $\exists \mathcal{P}\ s \xrightarrow{\Gamma}_{\mathcal{P}} \tilde{s}$ and $s \to \tilde{s}$ if $\exists \Gamma\ s \xrightarrow{\Gamma} \tilde{s}$. For a one-element multiset of activities $\Gamma = \{(\alpha, \rho)\}$ we write $s \xrightarrow{(\alpha,\rho)}_{\mathcal{P}} \tilde{s}$ and $s \xrightarrow{(\alpha,\rho)} \tilde{s}$.

Isomorphism is a coincidence of systems up to renaming of their components or states.

Definition 3.5 *Let G, G' be dynamic expressions and $TS(G) = (S_G, L_G, \mathcal{T}_G, s_G)$, $TS(G') = (S_{G'}, L_{G'}, \mathcal{T}_{G'}, s_{G'})$ be their transition systems. A mapping $\beta : S_G \to S_{G'}$ is an* isomorphism *between $TS(G)$ and $TS(G')$, denoted by $\beta : TS(G) \simeq TS(G')$, if*

1. *β is a bijection such that $\beta(s_G) = s_{G'}$;*

2. *$\forall s, \tilde{s} \in S_G\ \forall \Gamma\ s \xrightarrow{\Gamma}_{\mathcal{P}} \tilde{s} \ \Leftrightarrow\ \beta(s) \xrightarrow{\Gamma}_{\mathcal{P}} \beta(\tilde{s})$.*

Two transition systems $TS(G)$ and $TS(G')$ are isomorphic*, denoted by $TS(G) \simeq TS(G')$, if $\exists \beta : TS(G) \simeq TS(G')$.*

Transition systems of static expressions can be defined as well. For $E \in \mathit{RegStatExpr}$, let $TS(E) = TS(\overline{E})$.

Definition 3.6 *Two dynamic expressions G and G' are* equivalent with respect to transition systems*, denoted by $G =_{ts} G'$, if $TS(G) \simeq TS(G')$.*

Definition 3.7 *Let G be a dynamic expression. The* underlying discrete time Markov chain (DTMC) *of G, denoted by $DTMC(G)$, has the state space $DR(G)$ and the transitions $s \to_{\mathcal{P}} \tilde{s}$, if $s \to \tilde{s}$ and $\mathcal{P} = PM(s, \tilde{s})$.*

Underlying DTMCs of static expressions can be defined as well. For $E \in \mathit{RegStatExpr}$, let $DTMC(E) = DTMC(\overline{E})$.

For a dynamic expression G, a discrete random variable is associated with every state of $DTMC(G)$. The variable captures a residence time in the state. One can interpret staying in a state in the next discrete time moment as a failure and leaving it as a success of some trial series. It is easy to see that the random variables are geometrically distributed, since the probability to stay in the state $s \in DR(G)$ for $k-1$ time moments and leave it at moment $k \geq 1$ is $PM(s,s)^{k-1}(1 - PM(s,s))$ (the residence time is k in this case). The mean value formula for geometrical distribution allows us to calculate the *average sojourn time in the state s* as

$$SJ(s) = \frac{1}{1 - PM(s,s)}.$$

The *average sojourn time vector* of G, denoted by SJ, is that with the elements $SJ(s),\ s \in DR(G)$.

Example 3.2 *Let $E_1 = (\{a\}, \rho)[](\{a\}, \rho)$, $E_2 = (\{b\}, \chi)$, $E_3 = (\{c\}, \theta)$ and $E = [E_1 * E_2 * E_3]$. The identical activities of the composite static expression are enumerated as follows: $E = [((\{a\}, \rho)_1[](\{a\}, \rho)_2) * (\{b\}, \chi) * (\{c\}, \theta)]$. In Figure 2 the transition system $TS(\overline{E})$ and the underlying DTMC $DTMC(\overline{E})$ are presented. Note that, for simplicity of the graphical representation, states are depicted by expressions belonging to the corresponding equivalence classes, and singleton multisets of activities are written without braces. $DR(\overline{E})$ consists of the equivalence classes $s_1 = [\overline{[E_1 * E_2 * E_3]}]_\approx$, $s_2 = [[E_1 * \overline{E_2} * E_3]]_\approx$, $s_3 = [\underline{[E_1 * E_2 * E_3]}]_\approx$.*

Let us demonstrate how the transition probabilities are calculated. For instance, we have $PF(\{(\{a\}, \rho)_1\}, s_1) = PF(\{(\{a\}, \rho)_2\}, s_1) = \rho(1-\rho)$ and $PF(\emptyset, s_1) = (1-\rho)^2$. Hence, $\sum_{\Delta \in Exec(s_1)} PF(\Delta, s_1) = 2\rho(1-\rho) + (1-\rho)^2 = 1 - \rho^2$. Thus, $PT(\{(\{a\}, \rho)_1\}, s_1) = PT(\{(\{a\}, \rho)_2\}, s_1) = \frac{\rho(1-\rho)}{1-\rho^2} = \frac{\rho(1-\rho)}{(1-\rho)(1+\rho)} = \frac{\rho}{1+\rho}$ and $PT(\emptyset, s_1) = \frac{(1-\rho)^2}{1-\rho^2} = \frac{(1-\rho)^2}{(1-\rho)(1+\rho)} = \frac{1-\rho}{1+\rho}$. The other probabilities are calculated in a similar way.

The average sojourn time vector is

$$SJ = \left(\frac{1+\rho}{2\rho}, \frac{1-\chi\theta}{\theta(1-\chi)}, \infty \right).$$

4 Denotational semantics

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

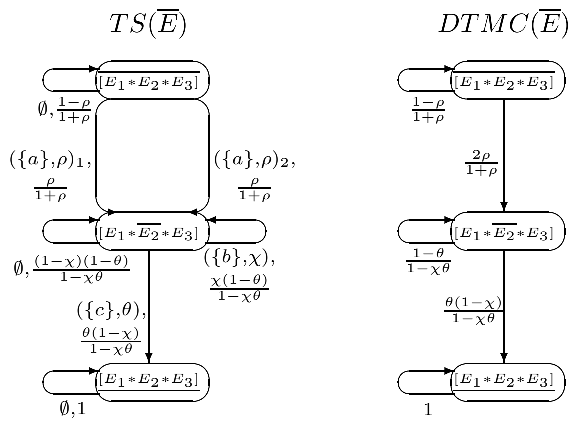

Figure 2: The transition system and the underlying DTMC of $\overline{E}$ for $E = [((\{a\},\rho)_1[](\{a\},\rho)_2)*(\{b\},\chi)*(\{c\},\theta)]$

4.1 Labeled DTSPNs

Let us introduce a class of labeled discrete time stochastic Petri nets. First, we present a formal definition (construction, syntax) of labeled discrete time stochastic Petri nets.

Definition 4.1 *A* labeled DTSPN (LDTSPN) *is a tuple* $N = (P_N, T_N, W_N, \Omega_N, L_N, M_N)$*, where*

- P_N *and* T_N *are finite sets of* places *and* transitions*, respectively, such that* $P_N \cup T_N \neq \emptyset$ *and* $P_N \cap T_N = \emptyset$*;*
- $W_N : (P_N \times T_N) \cup (T_N \times P_N) \to I\!N$ *is a function providing the* weights of arcs *between places and transitions;*
- $\Omega_N : T_N \to (0; 1)$ *is the* transition probability *function associating transitions with probabilities;*
- $L_N : T_N \to \mathcal{L}$ *is the* transition labeling *function assigning multiactions to transitions;*
- $M_N \in I\!N_f^{P_N}$ *is the* initial marking.

A graphical representation of LDTSPNs is like that for standard labeled Petri nets, but with probabilities written near the corresponding transitions. In the case the probabilities are not given in the picture, they are considered to be of no importance in the corresponding examples, such as those used to describe stationary behaviour. The weights of arcs are depicted near them. The names of places and transitions are depicted near them when needed. If the names are omitted but used, it is supposed that the places and transitions are numbered from left to right and from top to down.

Now we define a behaviour (functioning, semantics) of LDTSPNs.

Let N be an LDTSPN and $t \in T_N$, $U \in I\!N_f^{T_N}$. 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 $U^\bullet = \sum_{t\in U} t^\bullet$.

A transition $t \in T_N$ is *enabled* in a marking $M \in I\!N_f^{P_N}$ of LDTSPN N if ${}^\bullet t \subseteq M$. Let $Ena(M)$ be the set of *all transitions (such that each of them is) enabled in a marking* M. A set of transitions $U \subseteq Ena(M)$ is *enabled* in a marking M if ${}^\bullet U \subseteq M$. Firings of transitions are atomic operations, and transitions may fire concurrently in steps. We assume that all transitions participating in a step should differ, hence, only the sets (not multisets) of transitions may fire. Thus, we do not allow self-concurrency, i.e., firing of transitions concurrently to themselves. This restriction is introduced because we would like to avoid technical difficulties while calculating probabilities for multisets of transitions as we shall see after the following formal definitions.

Let M be a marking of an LDTSPN N. A transition $t \in Ena(M)$ fires with probability $\Omega_N(t)$ when no other transitions conflicting with it are enabled.

Let $U \subseteq Ena(M)$, $U \neq \emptyset$ and ${}^\bullet U \subseteq M$. The *probability that the set of transitions* U *is ready for firing in* M is

$$PF(U,M) = \prod_{t\in U} \Omega_N(t) \cdot \prod_{u\in Ena(M)\setminus U} (1 - \Omega_N(u)).$$

In the case $U = \emptyset$ we define

$$PF(\emptyset, M) = \begin{cases} \prod_{u \in Ena(M)}(1 - \Omega_N(u)), & Ena(M) \neq \emptyset; \\ 1, & \text{otherwise.} \end{cases}$$

Thus, $PF(U, M)$ could be interpreted as a *joint* probability of independent events. Each such an event is interpreted as readiness or not readiness for firing of a particular transition from U. The multiplication in the definition is used because it reflects the probability of the events intersection. When no transitions are enabled in M, we have $PF(\emptyset, M) = 1$, since we stay in M in this case.

Let $U \subseteq Ena(M)$, $U \neq \emptyset$ and ${}^\bullet U \subseteq M$. The concurrent firing of the transitions from U changes the marking M to $\widetilde{M} = M - {}^\bullet U + U^\bullet$, denoted by $M \xrightarrow{U}_{\mathcal{P}} \widetilde{M}$, where $\mathcal{P} = PT(U, M)$ is the *probability that the set of transitions U fires in M* defined as

$$PT(U, M) = \frac{PF(U, M)}{\sum_{\{V \mid {}^\bullet V \subseteq M\}} PF(V, M)}.$$

In the case $U = \emptyset$ we have $M = \widetilde{M}$ and

$$PT(\emptyset, M) = \frac{PF(\emptyset, M)}{\sum_{\{V \mid {}^\bullet V \subseteq M\}} PF(V, M)}.$$

Thus, $PT(U, M)$ is the probability that the set U is ready for firing in M *normalized* by the corresponding probability for *any* set enabled in M. The denominator of the fraction above is a sum since it reflects the probability of the events union.

Note that for all markings of an LDTSPN N the sum of outgoing probabilities is equal to 1. More formally, $\forall M \in \mathbb{N}_f^{P_N}\ PT(\emptyset, M) + \sum_{\{U \mid {}^\bullet U \subseteq M\}} PT(U, M) = 1$. This obviously follows from the definition of $PT(U, M)$ and guarantees that it defines a probability distribution.

We write $M \xrightarrow{U} \widetilde{M}$ if $\exists \mathcal{P}\ M \xrightarrow{U}_{\mathcal{P}} \widetilde{M}$ and $M \to \widetilde{M}$ if $\exists U\ M \xrightarrow{U} \widetilde{M}$. For one-element set of transitions $U = \{t\}$ we write $M \xrightarrow{t}_{\mathcal{P}} \widetilde{M}$ and $M \xrightarrow{t} \widetilde{M}$.

Definition 4.2 *Let N be an LDTSPN.*

- *The* reachability set *of N, denoted by $RS(N)$, is the minimal set of markings such that*
  - $M_N \in RS(N)$;
  - *if $M \in RS(N)$ and $M \to \widetilde{M}$ then $\widetilde{M} \in RS(N)$.*
- *The* reachability graph *of N, denoted by $RG(N)$, is a directed labeled graph with the set of nodes $RS(N)$ and an arc labeled with $(U, \mathcal{P})$ between nodes M and $\widetilde{M}$ if $M \xrightarrow{U}_{\mathcal{P}} \widetilde{M}$.*
- *The* underlying discrete time Markov chain (DTMC) *of N, denoted by $DTMC(N)$, has the state space $RS(N)$ and the transitions $M \to_{\mathcal{P}} \widetilde{M}$, if $M \to \widetilde{M}$, where $\mathcal{P} = PM(M, \widetilde{M})$ is the* probability to move from M to $\widetilde{M}$ by firing any set of transitions *defined as*

$$PM(M, \widetilde{M}) = \sum_{\{U \mid M \xrightarrow{U} \widetilde{M}\}} PT(U, M).$$

Since $PM(M, \widetilde{M})$ is the probability for *any* (including the empty one) transition set to change marking M to $\widetilde{M}$, we use summation in the definition. Note that $\forall M \in RS(N)\ \sum_{\{\widetilde{M} \mid M \to \widetilde{M}\}} PM(M, \widetilde{M}) = \sum_{\{\widetilde{M} \mid M \to \widetilde{M}\}} \sum_{\{U \mid M \xrightarrow{U} \widetilde{M}\}} PT(U, M) = \sum_{\{U \mid {}^\bullet U \subseteq M\}} PT(U, M) = 1$.

Let N be an LDTSPN and $M \in RS(N)$. The *average sojourn time in the marking M* is

$$SJ(M) = \frac{1}{1 - PM(M, M)}.$$

The *average sojourn time vector* of N, denoted by SJ, is that with the elements $SJ(M),\ M \in RS(N)$.

Example 4.1 *In Figure 3 an LDTSPN N with two visible transitions t_1 (labeled by $\{a\}$), t_2 (labeled by $\{b\}$) and one invisible transition t_3 (labeled by $\emptyset$) is presented. Transition probabilities of N are denoted by $\rho = \Omega_N(t_1)$, $\chi = \Omega_N(t_2)$, $\theta = \Omega_N(t_3)$. In the figure one can see the reachability graph $RG(N)$ and the underlying DTMC $DTMC(N)$ as well. $RS(N)$ consists of the markings $M_1 = (1,1,0)$, $M_2 = (0,1,1)$, $M_3 = (1,0,1)$, $M_4 = (0,0,2)$.*

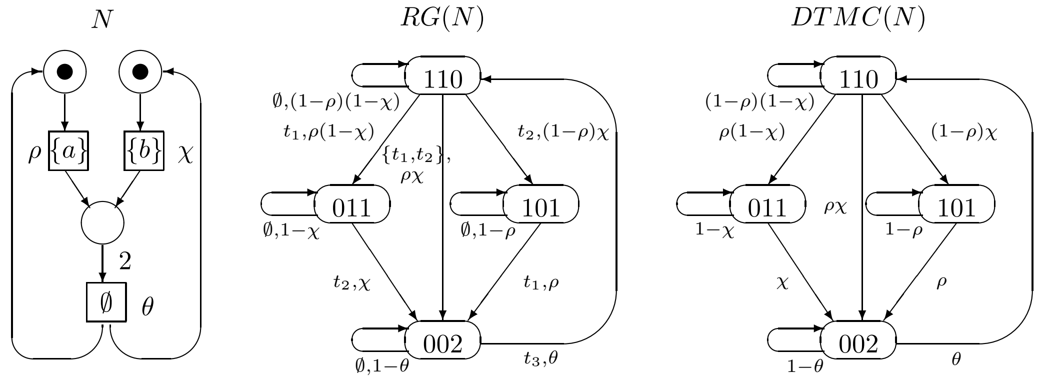

Figure 3: LDTSPN, its reachability graph and the underlying DTMC

The average sojourn time vector is

$$SJ = \left(\frac{1}{\rho+\chi-\rho\chi}, \frac{1}{\chi}, \frac{1}{\rho}, \frac{1}{\theta}\right).$$

The elements $\mathcal{P}_{ij} (1 \leq i, j \leq 4)$ *of (one-step) transition probability matrix (TPM) for* $DTMC(N)$ *are defined as*

$$\mathcal{P}_{ij} = \begin{cases} PM(M_i, M_j), & M_i \to M_j; \\ 0, & \text{otherwise.} \end{cases}$$

Thus, the TPM is

$$\mathbf{P} = \begin{bmatrix} (1-\rho)(1-\chi) & \rho(1-\chi) & \chi(1-\rho) & \rho\chi \\ 0 & 1-\chi & 0 & \chi \\ 0 & 0 & 1-\rho & \rho \\ \theta & 0 & 0 & 1-\theta \end{bmatrix}.$$

The steady-state probability mass function (PMF) $\psi = (\psi_1, \psi_2, \psi_3, \psi_4)$ *for* $DTMC(N)$ *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 dimension four and* $\mathbf{0} = (0,0,0,0)$, $\mathbf{1} = (1,1,1,1)$.
For the case $\rho = \chi = \theta$ *we have*

$$\psi = \left(\frac{1}{5-3\rho}, \frac{1-\rho}{5-3\rho}, \frac{1-\rho}{5-3\rho}, \frac{2-\rho}{5-3\rho}\right).$$

The inverse of the steady-state PMF for $DTMC(N)$ *is its mean recurrence time vector*

$$RC = \left(5-3\rho, \frac{5-3\rho}{1-\rho}, \frac{5-3\rho}{1-\rho}, \frac{5-3\rho}{2-\rho}\right).$$

Each element of RC *is the mean number of steps to return to the corresponding marking. For instance, one can see that the average time to come back to the initial marking* $M_N = M_1$ *in the long-term behaviour belongs in the interval* $(2;5)$, *since* $\rho \in (0;1)$.

4.2 Algebra of dts-boxes

Now we propose discrete time stochastic Petri boxes and associated algebraic operations to define a net representation of $dtsPBC$ expressions.

Definition 4.3 *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, such that* $P_N \cup T_N \neq \emptyset$ *and* $P_N \cap T_N = \emptyset$;

- $W_N : (P_N \times T_N) \cup (T_N \times P_N) \to I\!N$ *is a function providing the* weights of arcs *between places and transitions and vice versa;*

- Λ_N *is the* place and transition labeling *function such that*

  - $\Lambda_N|_{P_N} : P_N \to \{\mathsf{e}, \mathsf{i}, \mathsf{x}\}$ *(it specifies* entry, internal *and* exit *places, respectively);*
  - $\Lambda_N|_{T_N} : T_N \to \{\varrho \mid \varrho \subseteq I\!N_f^{\mathcal{SL}} \times \mathcal{SL}\}$ *(it associates transitions with the* relabeling relations *on activities).*

Moreover, $\forall t \in T_N \ {}^\bullet t \neq \emptyset \neq t^\bullet$*. In addition, for the set of* entry *places of* N *defined as* ${}^\circ N = \{p \in P_N \mid \Lambda_N(p) = \mathsf{e}\}$ *and the set of* exit *places of* N *defined as* $N^\circ = \{p \in P_N \mid \Lambda_N(p) = \mathsf{x}\}$ *the following condition holds:* ${}^\circ N \neq \emptyset \neq N^\circ,\ {}^\bullet({}^\circ N) = \emptyset = (N^\circ)^\bullet$.

A dts-box is *plain* if $\forall t \in T_N \ \Lambda_N(t) \in \mathcal{SL}$, 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 $M_N \in I\!N_f^{P_N}$ is the *initial marking*. We shall use the following notation: $\overline{N} = (N, {}^\circ N)$ and $\underline{N} = (N, N^\circ)$. Note that a marked plain dts-box $(P_N, T_N, W_N, \Lambda_N, M_N)$ could be interpreted as the LDTSPN $(P_N, T_N, W_N, \Omega_N, L_N, M_N)$, where functions Ω_N and L_N are defined as follows: $\forall t \in T_N \ \Omega_N(t) = \Omega(\Lambda_N(t))$ and $L_N(t) = \mathcal{L}(\Lambda_N(t))$. The behaviour of marked dts-boxes follows from the firing rule of LDTSPNs. A plain dts-box N is *n-bounded* ($n \in I\!N$) if $\overline{N}$ is so, i.e., $\forall M \in RS(\overline{N}) \ \forall p \in P_N \ M(p) \leq n$, and it is *safe* if it is 1-bounded. A plain dts-box N is *clean* if $\forall M \in RS(\overline{N}) \ {}^\circ N \subseteq M \ \Rightarrow \ M = {}^\circ N$ and $N^\circ \subseteq M \ \Rightarrow \ M = N^\circ$, if there are tokens in all its entry (exit) places then no other places have tokens.

To define semantic function that associates a plain dts-box with every static expression of $dtsPBC$, we need to propose the *enumeration* function $Enu : T_N \to Num$. It associates the numberings with transitions of plain dts-box N in accordance with those of activities. In the case of synchronization, the function associates with the resulting new transition the concatenation of the parenthesized numberings of the transitions it comes from.

The structure of the plain dts-box corresponding to a static expression is constructed like in PBC, see [17, 18, 8]. I.e., we use simultaneous refinement and relabeling meta-operator (net refinement) in addition to the *operator dts-boxes* corresponding to the algebraic operations of $dtsPBC$ and featuring transformational transition relabelings. Thus, as we shall see in Theorem 4.1, the resulting plain dts-boxes are safe and clean. In the definition of the denotational semantics, we shall apply standard constructions used for PBC. Let Θ denotes *operator box* and u denotes *transition name* from PBC setting.

The relabeling relations $\varrho \subseteq I\!N_f^{\mathcal{SL}} \times \mathcal{SL}$ are defined as follows:

- $\varrho_{id} = \{(\{(\alpha, \rho)\}, (\alpha, \rho)) \mid (\alpha, \rho) \in \mathcal{SL}\}$ 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 $(\alpha, \rho) \in \mathcal{SL}$ itself;

- $\varrho_{[f]} = \{(\{(\alpha, \rho)\}, (f(\alpha), \rho)) \mid (\alpha, \rho) \in \mathcal{SL}\}$;

- $\varrho_{\mathsf{rs}\ a} = \{(\{(\alpha, \rho)\}, (\alpha, \rho)) \mid (\alpha, \rho) \in \mathcal{SL},\ a, \hat{a} \notin \alpha\}$;

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

The plain and operator dts-boxes are presented in Figure 4. Note that the symbol i is usually omitted.

Now we define the enumeration function Enu for every operator of $dtsPBC$. Let $Box_{dts}(E) = (P_E, T_E, W_E, \Lambda_E)$ be the plain dts-box corresponding to a static expression E, and Enu_E be the enumeration function for T_E. We shall use the analogous notation for static expressions F and K.

- $Box_{dts}(E \circ F) = \Theta_\circ(Box_{dts}(E), Box_{dts}(F)),\ \circ \in \{;, [], \|\}$. 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:

$$Enu(t) = Enu_E(t),\ t \in T_E.$$

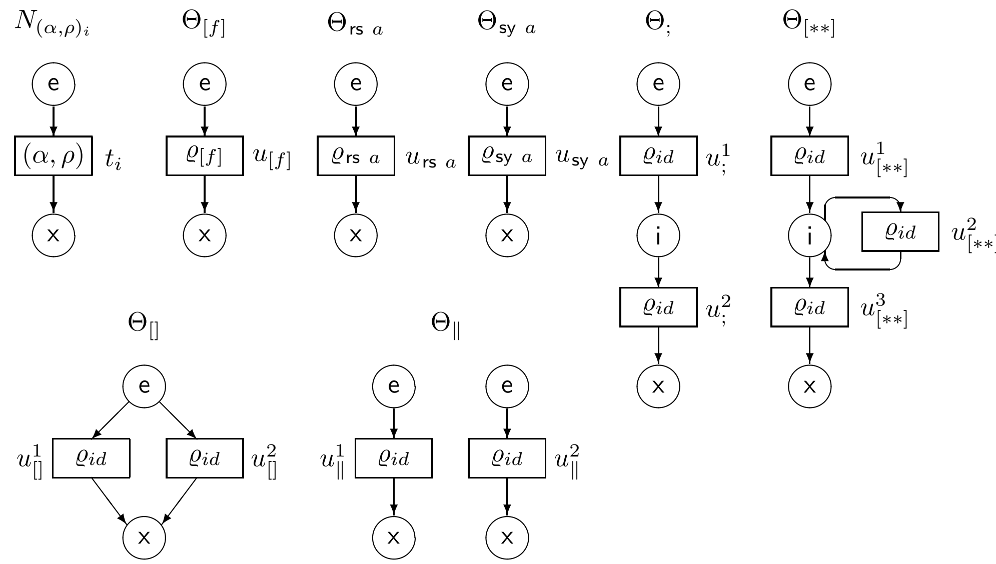

Figure 4: The plain and operator dts-boxes

- $Box_{dts}(E\ \mathsf{rs}\ a) = \Theta_{\mathsf{rs}\ a}(Box_{dts}(E))$. Since we remove all transitions labeled with multiactions containing a or $\hat{a}$, this does not change the numbering of the remaining transitions:

$$Enu(t) = Enu_E(t),\ t \in T_E,\ a, \hat{a} \notin \mathcal{L}(\Lambda_E(t)).$$

- $Box_{dts}(E\ \mathsf{sy}\ a) = \Theta_{\mathsf{sy}\ a}(Box_{dts}(E))$. Note that $\forall v, w \in T_E$ such that $\Lambda_E(v) = (\alpha, \rho),\ \Lambda_E(w) = (\beta, \chi)$ and $a \in \alpha,\ \hat{a} \in \beta$, 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 $Enu(t) = (Enu_E(v))(Enu_E(w))$.

  Thus, the enumeration function is defined as

$$Enu(t) = \begin{cases} Enu_E(t), & t \in T_E; \\ (Enu_E(v))(Enu_E(w)), & t \text{ results from synchronization of } v \text{ and } w. \end{cases}$$

  When we synchronize the same set of transitions in different orders, we obtain several resulting transitions with the same label and probability, but with the different numberings having the same content. In this case, we shall consider only single one from the resulting transitions in the plain dts-box to avoid introducing redundant transitions. For example, if the transitions t and u are generated by synchronizing v and w in different orders, we have $\Lambda(t) = (\alpha \oplus_a \beta, \rho \cdot \chi) = \Lambda(u)$, but $Enu(t) = (Enu_E(v))(Enu_E(w)) \neq (Enu_E(w))(Enu_E(v)) = Enu(u)$ whereas $Cont(Enu(t)) = Cont(Enu(v)) \cup Cont(Enu(w)) = Cont(Enu(u))$. Then only one transition t (or, symmetrically, u) will appear in $Box_{dts}(E\ \mathsf{sy}\ a)$.

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

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

Now we can formally define the denotational semantics as a homomorphism.

Definition 4.4 *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. $Box_{dts}(E \circ a) = \Theta_{\circ a}(Box_{dts}(E)),\ \circ \in \{\mathsf{rs}, \mathsf{sy}\}$;

5. $Box_{dts}([E * F * K]) = \Theta_{[**]}(Box_{dts}(E), Box_{dts}(F), Box_{dts}(K))$.

The dts-boxes of dynamic expressions can be defined as well. For $E \in RegStatExpr$, let $Box_{dts}(\overline{E}) = \overline{Box_{dts}(E)}$ and $Box_{dts}(\underline{E}) = \underline{Box_{dts}(E)}$.

Note that any dynamic expression can be decomposed into overlined or underlined static expressions or those without overlines and underlines. The definition of dts-boxes for arbitrary dynamic expressions should be compositional as well. Hence, we are to apply the net operations to the dts-boxes of these three types of expressions where all and only places containing one token each are the entry or the exit ones or no places contain tokens at all. The operations are applied to the dts-boxes with tokens like to those without them, but preserving the tokens in places.

Theorem 4.1 *For any static expression E, $Box_{dts}(\overline{E})$ is safe and clean.*

Proof. The structure of the net is obtained as in PBC, combining both refinement and relabeling. Consequently, the dts-boxes thus obtained will be safe and clean. □

Let $\simeq$ denote isomorphism between transition systems or between DTMCs and reachability graphs that relates the initial states. Due to the space restrictions, we omit the corresponding definitions as they resemble that of the isomorphism between transition systems. Note that the names of transitions of the dts-box corresponding to a static expression could be identified with the enumerated activities of the latter.

Theorem 4.2 *For any static expression E*

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

Proof. As for the qualitative (functional) behaviour, we have the same isomorphism as in PBC.

The quantitative behaviour is the same by the following reasons. First, the activities of an expression have probability parts coinciding with the probabilities of the transitions belonging to the corresponding dts-box. Second, both in stochastic processes specified by expressions and in dts-boxes, conflicts are resolved via the analogous probability functions used to construct the corresponding transition systems and reachability graphs. □

Proposition 4.1 *For any static expression E*

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

Proof. By Theorem 4.2 and definitions of underlying DTMCs for dynamic expressions and LDTSPNs, since transition probabilities of the associated DTMCs are the sums of those belonging to transition systems or reachability graphs. □

Example 4.2 *Let E be from Example 3.2. In Figure 5 the marked dts-box $N = Box_{dts}(\overline{E})$, its reachability graph $RG(N)$ and the underlying DTMC $DTMC(N)$ are presented. It is easy to see that $TS(\overline{E})$ and $RG(N)$ are isomorphic, as well as $DTMC(\overline{E})$ and $DTMC(N)$.*

Consider the next example that demonstrates synchronization.

Example 4.3 *Let $E_1 = (\{a\}, \rho)$, $E_2 = (\{\hat{a}\}, \chi)$ and $E = (E_1 \| E_2)\ \mathsf{sy}\ a = ((\{a\}, \rho) \| (\{\hat{a}\}, \chi))\ \mathsf{sy}\ a$. In Figure 6 the transition system $TS(\overline{E})$ and the underlying DTMC $DTMC(\overline{E})$ are presented. In Figure 7 the marked dts-box $N = Box_{dts}(\overline{E})$, its reachability graph $RG(N)$ and the underlying DTMC $DTMC(N)$ are depicted. It is easy to see that $TS(\overline{E})$ and $RG(N)$ are isomorphic, as well as $DTMC(\overline{E})$ and $DTMC(N)$.*

The probabilities $\mathcal{P}_{ij}$ $(1 \le i, j \le 4)$ are calculated as follows. Note that the symbol sy inscribes probability of the transition generated by synchronization, and the symbol $\|$ inscribes that of the transition corresponding to the concurrent execution of two activities. To avoid complex notation, we use the normalization factor $\mathcal{N} = \frac{1}{1-\rho^2\chi-\rho\chi^2+\rho^2\chi^2}$.

$\mathcal{P}_{11} = \mathcal{N}(1-\rho)(1-\chi)(1-\rho\chi)$	$\mathcal{P}_{12} = \mathcal{N}\rho(1-\chi)(1-\rho\chi)$	$\mathcal{P}_{13} = \mathcal{N}\chi(1-\rho)(1-\rho\chi)$
$\mathcal{P}_{14}^{\mathsf{sy}} = \mathcal{N}\rho\chi(1-\rho)(1-\chi)$	$\mathcal{P}_{14}^{\|} = \mathcal{N}\rho\chi(1-\rho\chi)$	$\mathcal{P}_{22} = 1-\chi$
$\mathcal{P}_{24} = \chi$	$\mathcal{P}_{33} = 1-\rho$	$\mathcal{P}_{34} = \rho$
$\mathcal{P}_{44} = 1$	$\mathcal{P}_{14} = \mathcal{P}_{14}^{\mathsf{sy}} + \mathcal{P}_{14}^{\|} = \mathcal{N}\rho\chi(2-\rho-\chi)$	

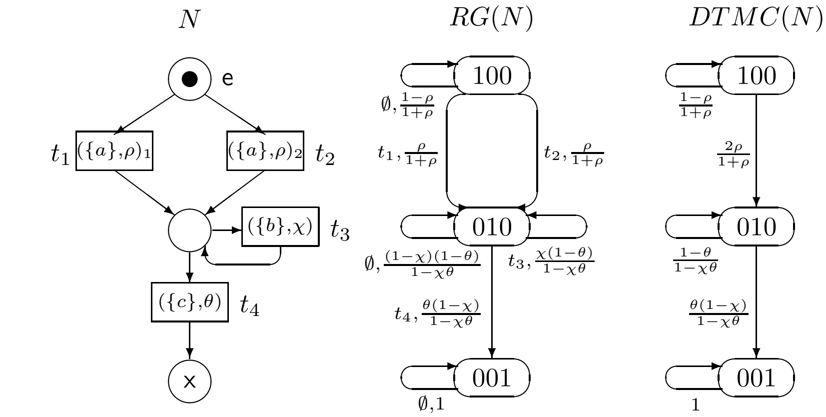

Figure 5: The marked dts-box $N = Box_{dts}(\overline{E})$ for $E = [((\{a\},\rho)_1[](\{a\},\rho)_2) * (\{b\},\chi) * (\{c\},\theta)]$, its reachability graph and the underlying DTMC

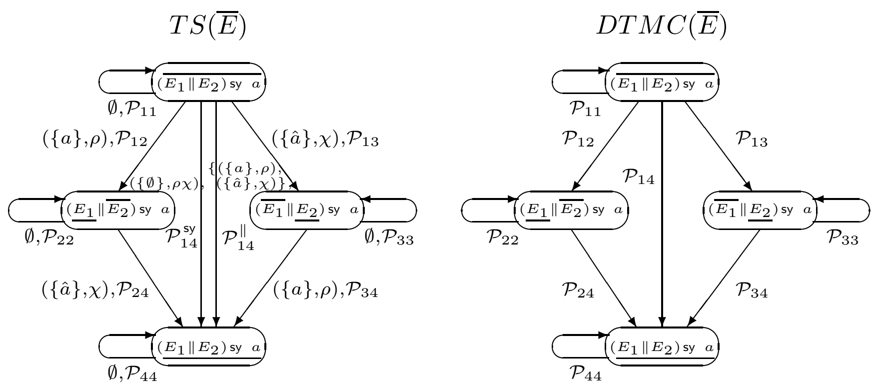

Figure 6: The transition system and the underlying DTMC of $\overline{E}$ for $E = ((\{a\},\rho)\|(\{\hat{a}\},\chi))$ sy a

Consider the case $\rho = \chi = \frac{1}{2}$. Then the transition probabilities will be the following:

$$\mathcal{P}_{11} = \mathcal{P}_{12} = \mathcal{P}_{13} = \mathcal{P}^{\|}_{14} = \frac{3}{13},\ \mathcal{P}^{\mathsf{sy}}_{14} = \frac{1}{13},\ \mathcal{P}_{22} = \mathcal{P}_{24} = \mathcal{P}_{33} = \mathcal{P}_{34} = \frac{1}{2},\ \mathcal{P}_{44} = 1,\ \mathcal{P}_{14} = \frac{4}{13}.$$

The following example demonstrates that without the syntactic restriction on regularity of expressions the corresponding marked dts-boxes may be not safe.

Example 4.4 *Let $E = [((\{a\},\rho) * ((\{b\},\chi)\|(\{c\},\theta)) * (\{d\},\phi)]$. In Figure 8 the marked dts-box $N = Box_{dts}(\overline{E})$ is presented. The initial marking is $M_1 = (1,0,0,0,0,0)$. The marking $M_2 = (0,1,1,1,1,0)$ is obtained from M_1 by firing the transition $(\{a\},\rho)$. Then in the marking $M_3 = (0,1,1,2,0,0)$ obtained from M_2 by firing $(\{b\},\chi)$ there are 2 tokens in the place p_4. Symmetrically, in the marking $M_4 = (0,1,1,0,2,0)$ obtained from M_2 by firing $(\{c\},\theta)$ there are 2 tokens in the place p_5. Thus, allowing concurrency in the second argument of iteration in the expression $\overline{E}$ can lead to non-safeness of the corresponding marked dts-box N, though, it is 2-bounded in the worst case, see [8]. The origin of the problem is that N has as a self-loop with two subnets which can function independently. This explains why do we consider regular expressions only.*

5 Stochastic equivalences

In this section, we propose a number of stochastic equivalences of expressions. The semantic equivalence $=_{ts}$ is too discriminate in many cases, i.e., from our viewpoint, it differentiates too many processes with similar behaviour. Hence, we need weaker equivalence notions to compare behaviour of processes specified by algebraic formulas.

To identify processes with intuitively similar behavior and to be able to apply standard constructions and techniques, we should abstract from infinite internal behaviour. Since *dtsPBC* is a stochastic extension of a

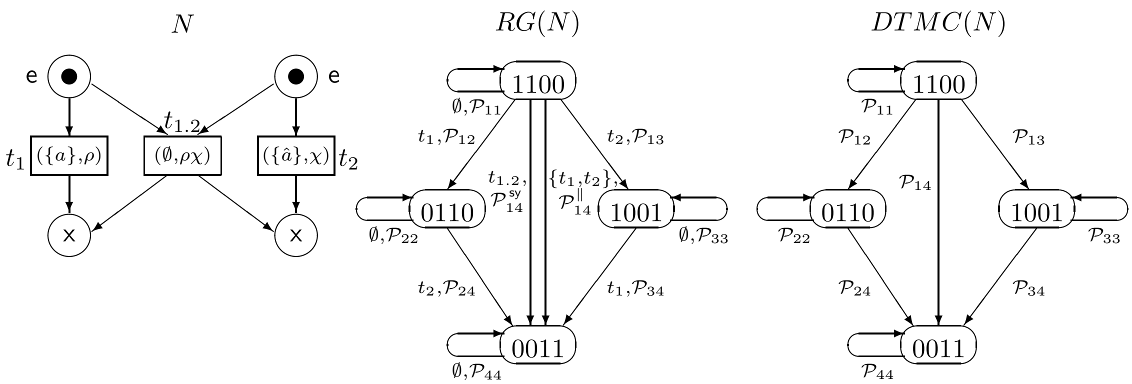

Figure 7: The marked dts-box $N = Box_{dts}(\overline{E})$ for $E = ((\{a\}, \rho)\|(\{\hat{a}\}, \chi))$ sy a, its reachability graph and the underlying DTMC

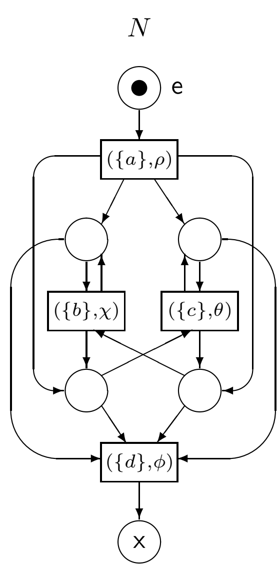

Figure 8: The marked dts-box $N = Box_{dts}(\overline{E})$ for $E = [((\{a\}, \rho) * ((\{b\}, \chi)\|(\{c\}, \theta)) * (\{d\}, \phi)]$

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.

Notice that we do not consider as a silent behaviour the execution of the iteration body built only from activities with the empty multiaction parts, even when the body consists from the single activity $(\emptyset, \rho)$ whose execution does not change the current state of the transition system. The reason is that we skip only the empty steps at the considered abstraction level, but the the iteration body consist at least from one activity.

5.1 Empty loops in transition systems

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* $s \xrightarrow{\emptyset}_{\mathcal{P}} s$ appear when no activities occur at a time step, and this happens with some positive probability. Obviously, the current state remains unchanged in this case.

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

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

where $EL(s) = \frac{1}{1-PT(\emptyset,s)}$ is the *empty loops abstraction factor*. The *empty loops abstraction vector* of G, denoted by EL, is that with the elements $EL(s),\ s \in DR(G)$. The value $k = 0$ in the summation above corresponds to the case when no empty loops occur.

Note that after abstraction from transition probabilities with empty multisets of activities, the remaining transition probabilities are normalized. In order to calculate transition probabilities $PT(\Gamma, s)$, we had to normalize $PF(\Gamma, s)$. Then, to obtain transition probabilities of non-empty steps $PT^*(\Gamma, s)$, we have to normalize $PT(\Gamma, s)$. Thus, we have a two-stage normalization as a result.

Note that $PT^*(\Gamma, s) \leq 1$, 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, since $\forall s \in DR(G)$ such that s is not a terminal state, i.e., there exist transitions from it, we have $\sum_{\Gamma \in Exec(s)\setminus\{\emptyset\}} PT^*(\Gamma, s) = 1$.

Definition 5.1 *The* (labeled probabilistic) transition system without empty loops $TS^*(G)$ *has the state space* $DR(G)$ *and the transitions* $s \overset{\Gamma}{\twoheadrightarrow}_{\mathcal{P}} \tilde{s}$*, if* $s \xrightarrow{\Gamma} \tilde{s},\ \Gamma \neq \emptyset$ *and* $\mathcal{P} = PT^*(\Gamma, s)$.

The definition of $TS^*(G)$ is correct, i.e., for every state excluding the terminal ones the sum of the probabilities of all the transitions starting from it is 1. This is guaranteed by the note after the definition of $PT^*(\Gamma, s)$.

Note that $TS^*(G)$ describes the viewpoint of a person who observes steps only if they include non-empty multisets of activities.

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

We decided to consider empty loops followed only by a non-empty step just for convenience. Alternatively, we could take a non-empty step succeeded by empty loops or a non-empty step preceded and succeeded by empty loops. In all these three cases our sequence begins or/and ends with the loops which do not change states. At the same time, the overall probabilities of the evolutions can differ, since empty loops have positive probabilities. To avoid inconsistency of definitions and too complex description, we consider sequences ending with a non-empty step. It resembles in some sense a construction of branching bisimulation [36].

Transition systems without empty loops of static expressions can be defined as well. For $E \in RegStatExpr$, let $TS^*(E) = TS^*(\overline{E})$.

Definition 5.2 *Two dynamic expressions* G *and* G' *are* equivalent with respect to transition systems without empty loops*, denoted by* $G =_{ts*} G'$*, if* $TS^*(G) \simeq TS^*(G')$.

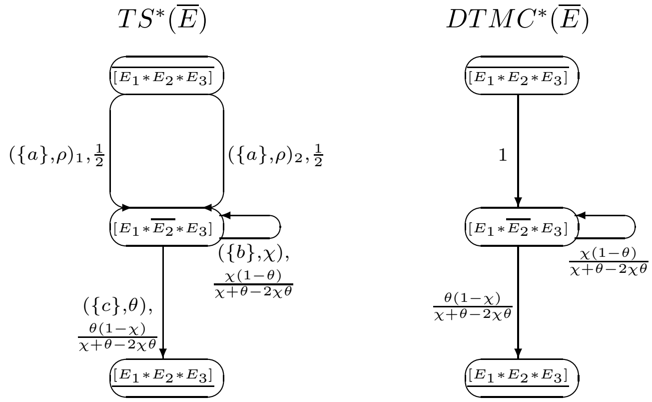

Figure 9: The transition system and the underlying DTMC without empty loops of $\overline{E}$ from Example 3.2

Definition 5.3 *The* underlying DTMC without empty loops $DTMC^*(G)$ *has the state space* $DR(G)$ *and the transitions* $s \twoheadrightarrow_{\mathcal{P}} \tilde{s}$*, if* $s \twoheadrightarrow \tilde{s}$*, 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*

$$PM^*(s,\tilde{s}) = \sum_{\{\Gamma | s \xrightarrow{\Gamma} \tilde{s}\}} PT^*(\Gamma, s) = \begin{cases} EL(s)(PM(s,s) - PT(\emptyset, s)), & s = \tilde{s}; \\ EL(s)PM(s,\tilde{s}), & \text{otherwise}. \end{cases}$$

Note that $\forall s \in DR(G)$ such that s is not a terminal state, i.e., there exist transitions from it, we have $\sum_{\{\tilde{s}|s\twoheadrightarrow\tilde{s}\}} PM^*(s,\tilde{s}) = \sum_{\{\tilde{s}|s\twoheadrightarrow\tilde{s}\}} \sum_{\{\Gamma|s\xrightarrow{\Gamma}\tilde{s}\}} PT^*(\Gamma, s) = \sum_{\Gamma\in Exec(s)\setminus\{\emptyset\}} PT^*(\Gamma, s) = 1$.

Underlying DTMCs without empty loops of static expressions can be defined as well. For $E \in RegStatExpr$, let $DTMC^*(E) = DTMC^*(\overline{E})$.

Example 5.1 *Let E be from Example 3.2. In Figure 9 the transition system $TS^*(\overline{E})$ and the underlying DTMC without empty loops $DTMC^*(\overline{E})$ are presented.*

Let us demonstrate how the transition probabilities of non-empty steps are calculated. For instance, we have $PT(\emptyset, s_1) = \frac{1-\rho}{1+\rho}$ *and* $\frac{1}{1-PT(\emptyset,s_1)} = \frac{1+\rho}{2\rho}$*. Hence, since* $PT(\{(\{a\},\rho)_1\}, s_1) = \frac{\rho}{1+\rho}$*, we have* $PT^*(\{(\{a\},\rho)_1\}, s_1) = \frac{PT(\{(\{a\},\rho)_1\},s_1)}{1-PT(\emptyset,s_1)} = \frac{\rho}{1+\rho} \cdot \frac{1+\rho}{2\rho} = \frac{1}{2}$*. According to the same pattern, we obtain* $PT^*(\{(\{a\},\rho)_2\}, s_1) = \frac{1}{2}$*. The other probabilities are calculated in a similar way.*

5.2 Empty loops in reachability graphs

Let N be an LDTSPN. Reachability graph $RG(N)$ can have loops going from a marking to itself which are labeled by the empty set and have non-zero probability. Such *empty loop* $M \xrightarrow{\emptyset}_{\mathcal{P}} M$ appears when no transitions fire at a time step, and this happens with some positive probability. Obviously, in this case the current marking remains unchanged.

Let N be an LDTSPN and $M \in RS(N)$.

The *probability to stay in M due to k ($k \geq 1$) empty loops* is

$$(PT(\emptyset, M))^k.$$

Let $U \subseteq Ena(M)$, $U \neq \emptyset$ and ${}^\bullet U \subseteq M$. The *probability that the non-empty set of transitions U fires in M after possible empty loops* is

$$PT^*(U,M) = PT(U,M)\sum_{k=0}^{\infty}(PT(\emptyset,M))^k = \frac{PT(U,M)}{1-PT(\emptyset,M)} = EL(M)PT(U,M),$$

where $EL(M) = \frac{1}{1-PT(\emptyset,M)}$ is the *empty loops abstraction factor*. The *empty loops abstraction vector* of N, denoted by EL, is that with the elements $EL(M)$, $M \in RS(N)$. The value $k = 0$ in the summation above corresponds to the case when no empty loops occur.

Note that $PT^*(U,M) \leq 1$, hence, it is really a probability, since $PT(\emptyset,M) + PT(U,M) \leq PT(\emptyset,M) + \sum_{\{V|{}^\bullet V\subseteq M\}} PT(V,M) = 1$. Moreover, $PT^*(U,M)$ defines a probability distribution, since $\forall M \in RS(N)$ such that M is not a terminal marking, i.e., there exist transitions from it, we have $\sum_{\{U\neq\emptyset|{}^\bullet U\subseteq M\}} PT^*(U,M) = 1$.

Definition 5.4 *The* reachability graph without empty loops $RG^*(N)$ *has the set of nodes* $RS(N)$ *and the arcs corresponding to the transitions* $M \overset{U}{\twoheadrightarrow}_{\mathcal{P}} \widetilde{M}$, *if* $M \overset{U}{\to} \widetilde{M}$, $U \neq \emptyset$ *and* $\mathcal{P} = PT^*(U,M)$.

Note that $RG^*(N)$ describes the viewpoint of a person who observes steps only if they include non-empty transition sets.

We write $M \overset{U}{\twoheadrightarrow} \widetilde{M}$ if $\exists\mathcal{P}\ M \overset{U}{\twoheadrightarrow}_{\mathcal{P}} \widetilde{M}$ and $M \twoheadrightarrow \widetilde{M}$ if $\exists U\ M \overset{U}{\twoheadrightarrow} \widetilde{M}$. For a one-element set of transitions $U = \{t\}$ we write $M \overset{t}{\twoheadrightarrow}_{\mathcal{P}} \widetilde{M}$ and $M \overset{t}{\twoheadrightarrow} \widetilde{M}$.

Definition 5.5 *The* underlying DTMC without empty loops $DTMC^*(N)$ *has the state space* $RS(N)$ *and the transitions* $M \twoheadrightarrow_{\mathcal{P}} \widetilde{M}$, *if* $M \twoheadrightarrow \widetilde{M}$, *where* $\mathcal{P} = PM^*(M,\widetilde{M})$ *is the* probability to move from M to $\widetilde{M}$ by firing any non-empty set of transitions after possible empty loops *defined as*

$$PM^*(M,\widetilde{M}) = \sum_{\{U|M\overset{U}{\twoheadrightarrow}\widetilde{M}\}} PT^*(U,M) = \begin{cases} EL(M)(PM(M,M) - PT(\emptyset,M)), & M = \widetilde{M};\\ EL(M)PM(M,\widetilde{M}), & \textit{otherwise}.\end{cases}$$

Note that $\forall M \in RS(N)$ such that M is not a terminal marking, i.e., there exist transitions from it, we have $\sum_{\{\widetilde{M}|M\twoheadrightarrow\widetilde{M}\}} PM^*(M,\widetilde{M}) = \sum_{\{\widetilde{M}|M\twoheadrightarrow\widetilde{M}\}} \sum_{\{U|M\overset{U}{\twoheadrightarrow}\widetilde{M}\}} PT^*(U,M) = \sum_{\{U\neq\emptyset|{}^\bullet U\subseteq M\}} PT^*(U,M) = 1$.

Theorem 5.1 *For any static expression* E

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

Proof. As Theorem 4.2. □

Proposition 5.1 *For any static expression* E

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

Proof. As Proposition 4.1. □

Note that Theorem 5.1 guarantees that the net versions of algebraic equivalences could be easily 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.

Example 5.2 *Let* E *be from Example 3.2 and* N *be from Example 4.2. In Figure 10 the reachability graph* $RG^*(N)$ *and the underlying DTMC without empty loops* $DTMC^*(N)$ *are presented. It is easy to see that* $TS^*(\overline{E})$ *and* $RG^*(N)$ *are isomorphic as well as* $DTMC^*(\overline{E})$ *and* $DTMC^*(N)$.

Consider the next example that demonstrates synchronization.

Example 5.3 *Let* E *and* N *be those from Example 4.3. In Figure 11 the transition system* $TS^*(\overline{E})$ *and the underlying DTMC without empty loops* $DTMC^*(\overline{E})$ *are presented. In Figure 12 the reachability graph* $RG^*(N)$ *and the underlying DTMC without empty loops* $DTMC^*(N)$ *are depicted. It is easy to see that* $TS^*(\overline{E})$ *and* $RG^*(N)$ *are isomorphic as well as* $DTMC^*(\overline{E})$ *and* $DTMC^*(N)$.

The probabilities $\mathcal{P}^*_{ij}$ $(1 \leq i,j \leq 4)$ *are calculated as follows. Note that the symbol* sy *inscribes probability of the transition generated by synchronization, and the symbol* $\|$ *inscribes that of the transition corresponding to the concurrent execution of two activities. To avoid complex notation, we use the normalization factor* $\mathcal{N}^* = \frac{1}{\rho+\chi-2\rho^2\chi-2\rho\chi^2+2\rho^2\chi^2}$. *The probabilities* $\mathcal{P}_{ij}$ $(1 \leq i,j \leq 4)$ *are taken from Example 4.3.*

$$\mathcal{P}^*_{12} = \frac{\mathcal{P}_{12}}{1-\mathcal{P}_{11}} = \mathcal{N}^*\rho(1-\chi)(1-\rho\chi) \qquad \mathcal{P}^*_{13} = \frac{\mathcal{P}_{13}}{1-\mathcal{P}_{11}} = \mathcal{N}^*\chi(1-\rho)(1-\rho\chi)$$
$$\mathcal{P}^{\mathsf{sy}*}_{14} = \frac{\mathcal{P}^{\mathsf{sy}}_{14}}{1-\mathcal{P}_{11}} = \mathcal{N}^*\rho\chi(1-\rho)(1-\chi) \qquad \mathcal{P}^{\|*}_{14} = \frac{\mathcal{P}^{\|}_{14}}{1-\mathcal{P}_{11}} = \mathcal{N}^*\rho\chi(1-\rho\chi)$$
$$\mathcal{P}^*_{24} = \frac{\mathcal{P}_{24}}{1-\mathcal{P}_{22}} = 1 \qquad \mathcal{P}^*_{34} = \frac{\mathcal{P}_{34}}{1-\mathcal{P}_{33}} = 1$$
$$\mathcal{P}^*_{14} = \mathcal{P}^{\mathsf{sy}*}_{14} + \mathcal{P}^{\|*}_{14} = \frac{\mathcal{P}^{\mathsf{sy}}_{14}+\mathcal{P}^{\|}_{14}}{1-\mathcal{P}_{11}} = \mathcal{N}^*\rho\chi(2-\rho-\chi)$$

Consider the case $\rho = \chi = \frac{1}{2}$. *Then the transition probabilities will be the following:*

$$\mathcal{P}^*_{12} = \mathcal{P}^*_{13} = \mathcal{P}^{\|*}_{14} = \frac{3}{10},\ \mathcal{P}^{\mathsf{sy}*}_{14} = \frac{1}{10},\ \mathcal{P}^*_{24} = \mathcal{P}^*_{34} = 1,\ \mathcal{P}^*_{14} = \frac{2}{5}.$$

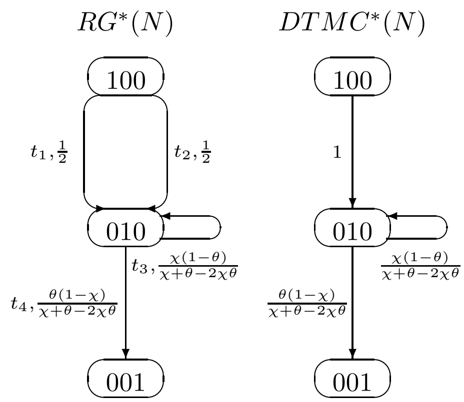

Figure 10: The reachability graph and the underlying DTMC without empty loops of N from Example 4.2

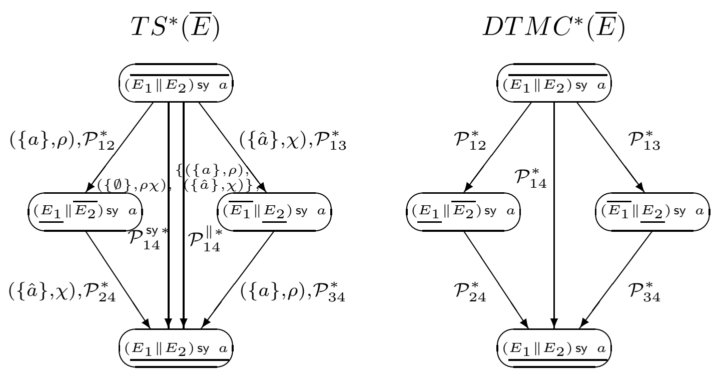

Figure 11: The transition system and the underlying DTMC without empty loops of $\overline{E}$ from Example 4.3

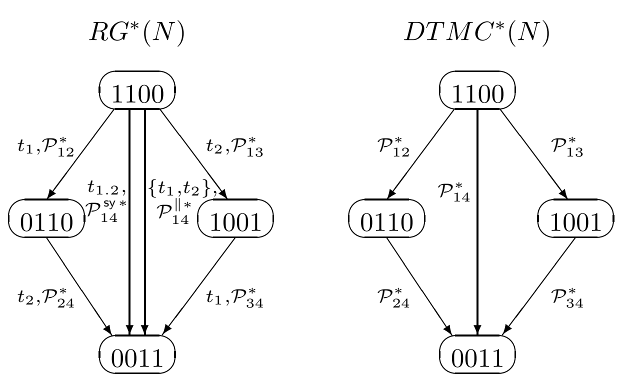

Figure 12: The reachability graph and the underlying DTMC without empty loops of N from Example 4.3

5.3 Stochastic trace equivalences

Trace equivalences are the least discriminating ones. In a trace semantics, the behavior of a system is associated with the set of all possible sequences of activities, i.e., protocols of work or computations. Thus, the points of choice of an external observer between several extensions of a particular computation are not taken into account.

Formal definitions of stochastic trace relations resemble those of trace equivalences for standard Petri nets [77] or process algebras, but additionally we have to take into account the probabilities of sequences of (multisets of) actions like in [28, 89]. First, we have to multiply occurrence probabilities for all (multisets of) activities along every path starting from the initial state of the transition system corresponding to a dynamic expression. The product is the probability of the sequence of multiaction parts of the (multisets of) activities along the path. Second, we should calculate a sum of probabilities for all paths corresponding to the same sequence of multiaction parts.

When concurrency aspects are not relevant, the interleaving behaviour is to be considered. The interleaving semantics abstracts from steps with more than one element. After such an abstracting, one has to normalize the probabilities of the remaining one-element steps. We need to do this since the sum of outgoing probabilities should always be equal to one for each state to form a probability distribution. For this, a special *interleaving transition relation* is proposed. Let G be a dynamic expression, $s, \tilde{s} \in DR(G)$ and $s \stackrel{(\alpha,\rho)}{\twoheadrightarrow} \tilde{s}$. We write $s \stackrel{(\alpha,\rho)}{\twoheadrightarrow}_{\mathcal{P}} \tilde{s}$, 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

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

Note that we have first abstracted from empty loops and then from steps with more than one element. We could perform the abstractions in the reverse order, the result will be the same. The reason is that, at every stage, we abstract from some transitions of a given transition system and then normalize the probabilities of the remaining ones. Hence, the result of each sequence of abstractions coincides with that of the abstraction at once from all the transitions we have abstracted from in this sequence.

For $\Gamma \in I\!N_f^{\mathcal{SL}}$, we define its *multiaction part* by $\mathcal{L}(\Gamma) = \sum_{(\alpha,\rho)\in\Gamma} \alpha$. Note that in the definitions below we shall consider $\mathcal{L}(\Gamma) \in I\!N_f^{\mathcal{L}} \setminus \{\emptyset\}$, i.e., the non-empty multisets of multiactions. These multiactions can be empty, and in this case $\mathcal{L}(\Gamma)$ will contain the elements $\emptyset$, hence, it will not be empty.

Definition 5.6 *An* interleaving stochastic trace *of a dynamic expression G is a pair* $(\sigma, pt^*(\sigma))$, *where* $\sigma = \alpha_1 \cdots \alpha_n \in \mathcal{L}^*$ *and*

$$pt^*(\sigma) = \sum_{\{(\alpha_1,\rho_1),\ldots,(\alpha_n,\rho_n) \mid [G]_\approx = s_0 \stackrel{(\alpha_1,\rho_1)}{\twoheadrightarrow} s_1 \stackrel{(\alpha_2,\rho_2)}{\twoheadrightarrow} \cdots \stackrel{(\alpha_n,\rho_n)}{\twoheadrightarrow} s_n\}} \prod_{i=1}^{n} pt^*((\alpha_i,\rho_i), s_{i-1}).$$

We denote the set of all interleaving stochastic traces *of a dynamic expression G by* $IntStochTraces(G)$. *Two dynamic expressions G and G' are* interleaving stochastic trace equivalent, *denoted by* $G \equiv_{is} G'$, *if*

$$IntStochTraces(G) = IntStochTraces(G').$$

Definition 5.7 *A* step stochastic trace *of a dynamic expression G is a pair* $(\Sigma, PT^*(\Sigma))$, *where* $\Sigma = A_1 \cdots A_n \in (I\!N_f^{\mathcal{L}} \setminus \{\emptyset\})^*$ *and*

$$PT^*(\Sigma) = \sum_{\{\Gamma_1,\ldots,\Gamma_n \mid [G]_\approx = s_0 \stackrel{\Gamma_1}{\twoheadrightarrow} s_1 \stackrel{\Gamma_2}{\twoheadrightarrow} \cdots \stackrel{\Gamma_n}{\twoheadrightarrow} s_n,\ \mathcal{L}(\Gamma_i) = A_i\ (1 \le i \le n)\}} \prod_{i=1}^{n} PT^*(\Gamma_i, s_{i-1}).$$

We denote the set of all step stochastic traces *of a dynamic expression G by* $StepStochTraces(G)$. *Two dynamic expressions G and G' are* step stochastic trace equivalent, *denoted by* $G \equiv_{ss} G'$, *if*

$$StepStochTraces(G) = StepStochTraces(G').$$

Example 5.4 *Let* $E = ((\{a\}, \frac{1}{2}) \| (\{\hat{a}\}, \frac{1}{2}))\ \mathsf{sy}\ a$. *Then* $IntStochTraces(\overline{E}) = \{(\emptyset, \frac{1}{7}),\ (\{a\}, \frac{3}{7}),\ (\{\hat{a}\}, \frac{3}{7}),\ (\{a\}\{\hat{a}\}, \frac{3}{7}),\ (\{\hat{a}\}\{a\}, \frac{3}{7})\}$ *and* $StepStochTraces(\overline{E}) = \{(\{\emptyset\}, \frac{1}{10}),\ (\{\{a\}\}, \frac{3}{10}),\ (\{\{\hat{a}\}\}, \frac{3}{10}),\ (\{\{a\}\}\{\{\hat{a}\}\}, \frac{3}{10}),\ (\{\{\hat{a}\}\}\{\{a\}\}, \frac{3}{10}),\ (\{\{\hat{a}\}, \{a\}\}, \frac{3}{10})\}$.

5.4 Stochastic bisimulation equivalences

Bisimulation equivalences respect the particular points of choice in the behavior of a modeled system. We intend to present a definition of stochastic bisimulation equivalences. The definition is parameterized for the cases of interleaving or step semantics.

To define stochastic bisimulation equivalences, we have to consider a bisimulation as an *equivalence* relation which partitions the states of the *union* of the transition systems $TS^*(G)$ and $TS^*(G')$ of two dynamic expressions G and G' to be compared. For G and G' to be bisimulation equivalent, the initial states of their transition systems, $[G]_\approx$ and $[G']_\approx$, are to be related by a bisimulation having the following transfer property: two states are related if in each of them the same (multisets of) multiactions can occur, and the resulting states *belong to the same equivalence class*. In addition, sums of probabilities for all such occurrences should be the same for both states. Thus, in our definitions, we follow the approach of [50, 51]. Hence, the difference between bisimulation and trace equivalences is that we do not consider *all possible* occurrences of (multisets of) multiactions from the initial states, but only such that lead (stepwise) to the states *belonging to the same equivalence class*. Note that our interleaving stochastic bisimulation equivalence resembles in some sense weak bisimulation one from [14, 15], but we abstract from empty loops only instead of any transitions with the initial and the final states from the same equivalence class (with respect to the mentioned equivalence).

First, we introduce several helpful notations. Let G be a dynamic expression and $\mathcal{H} \subseteq DR(G)$. Then for some $s \in DR(G)$ and $A \in I\!N_f^{\mathcal{L}} \setminus \{\emptyset\}$ we write $s \stackrel{A}{\twoheadrightarrow}_{\mathcal{P}} \mathcal{H}$, 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

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

The summation in the definition above reflects the probability of the events union.

We write $s \stackrel{A}{\twoheadrightarrow} \mathcal{H}$ if $\exists \mathcal{P}\ s \stackrel{A}{\twoheadrightarrow}_{\mathcal{P}} \mathcal{H}$.

We write $s \twoheadrightarrow_{\mathcal{P}} \mathcal{H}$ if $\exists A\ s \stackrel{A}{\twoheadrightarrow} \mathcal{H}$, where $\mathcal{P} = PM^*(s, \mathcal{H})$ is the *overall probability to move from s into the set of states $\mathcal{H}$ via any non-empty steps after possible empty loops* defined as

$$PM^*(s, \mathcal{H}) = \sum_{\{\Gamma \mid \exists \tilde{s} \in \mathcal{H}\ s \stackrel{\Gamma}{\twoheadrightarrow} \tilde{s}\}} PT^*(\Gamma, s).$$

We propose the corresponding *interleaving transition relation* $s \stackrel{\alpha}{\rightharpoondown}_{\mathcal{P}} \mathcal{H}$, 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

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

To introduce stochastic bisimulation equivalence between dynamic expressions G and G', we should consider a "composite" set of states $DR(G) \cup DR(G')$. The reason is that we have to identify the probabilities to come from any two equivalent states into the same "composite" equivalence class on this set. Note that transitions starting from the states of $DR(G)$ (or $DR(G')$) always lead to those from the same set, since $DR(G) \cap DR(G') = \emptyset$, and this allows us to "mix" the sets of states in the definition of stochastic bisimulation.

Definition 5.8 *Let G and G' be dynamic expressions. An* equivalence *relation $\mathcal{R} \subseteq (DR(G) \cup DR(G'))^2$ is a* $\star$-stochastic bisimulation *between G and G', $\star \in$*{interleaving, step}*, denoted by $\mathcal{R} : G \underline{\leftrightarrow}_{\star s} G'$, $\star \in \{i, s\}$, 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 $\hookrightarrow = \rightharpoondown$, if $\star = i$;*
   - *$\forall x \in I\!N_f^{\mathcal{L}} \setminus \{\emptyset\}$ and $\hookrightarrow = \twoheadrightarrow$, if $\star = s$;*

$$s_1 \stackrel{x}{\hookrightarrow}_{\mathcal{P}} \mathcal{H} \ \Leftrightarrow\ s_2 \stackrel{x}{\hookrightarrow}_{\mathcal{P}} \mathcal{H}.$$

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

5.5 Stochastic isomorphism

Stochastic isomorphism is a relation that is weaker than $=_{ts*}$. The main idea of the following definition is to collect the probabilities of all transitions between the same pair of states such that the transition labels have the same multiaction parts. We use summation, since it is the probability of the events union.

Let G be a dynamic expression and $s, \tilde{s} \in DR(G)$ such that $s \xrightarrow{A}\!\!\!\!\rightarrow_{\mathcal{P}} \{\tilde{s}\}$. In this case, we write $s \xrightarrow{A}\!\!\!\!\rightarrow_{\mathcal{P}} \tilde{s}$. Thus, $\mathcal{P}$ is the overall probability to come into the one-element set of states $\{\tilde{s}\}$ starting in s via steps with the multiaction part A. In other words, $\mathcal{P}$ is a sum of all the probabilities of steps with the multiaction part A between the states s and $\tilde{s}$.

Definition 5.9 *Let G, G' be dynamic expressions. A mapping $\beta : DR(G) \to DR(G')$ is a* stochastic isomorphism *between G and G', denoted by $\beta : G =_{sto} G'$, if*

1. *β is a bijection such that $\beta([G]_\approx) = [G']_\approx$;*

2. *$\forall s, \tilde{s} \in DR(G)\ \forall A \in I\!N_f^{\mathcal{L}} \setminus \{\emptyset\}\ s \xrightarrow{A}\!\!\!\!\rightarrow_{\mathcal{P}} \tilde{s} \ \Leftrightarrow\ \beta(s) \xrightarrow{A}\!\!\!\!\rightarrow_{\mathcal{P}} \beta(\tilde{s})$.*

Two dynamic expressions G and G' are stochastically isomorphic*, denoted by $G =_{sto} G'$, if $\exists \beta : G =_{sto} G'$.*

5.6 Interrelations of the stochastic equivalences

Note that all the algebraic equivalences of dynamic expressions we have defined, with the exception of $\approx$, can be transferred to the net level, i.e., to the corresponding marked dts-boxes. It is possible, since by Theorem 5.1 the transition systems without empty loops of the former and the reachability graphs without empty loops of the latter are isomorphic. In the figures with examples of dts-boxes corresponding to the expressions related by the algebraic equivalences, we shall also depict their net analogues (denoted by the same symbols).

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

Proposition 5.2 *Let $\star \in \{i, s\}$. For dynamic expressions G and G' the following holds:*

$$G \underline{\leftrightarrow}_{\star s} G' \ \Rightarrow\ G \equiv_{\star s} G'.$$

Proof. See Appendix A. □

Proposition 5.3 *For dynamic expressions G and G' the following holds:*

$$G =_{ts*} G' \ \Leftrightarrow\ G =_{ts} G'.$$

Proof. ($\Leftarrow$) It is enough to note that the abstraction from empty loops is based on transition probabilities which are the same for isomorphic transition systems.

($\Rightarrow$) Note that $TS(G)$ and $TS^*(G)$ (as well as $TS(G')$ and $TS^*(G')$) differ by presence of empty loops and by values of transition probabilities only. The sets of states, the labeling area, the non-empty multisets of activities which label the transitions and the initial states coincide. We have isomorphism of $TS^*(G)$ and $TS^*(G')$. For a state s of $TS^*(G)$, let s' be the state of $TS^*(G')$ such that these two states are related by the isomorphism of $TS^*(G)$ and $TS^*(G')$. Then $Exec(s) = \{\Gamma \mid \exists \tilde{s}\ s \xrightarrow{\Gamma}\!\!\!\!\rightarrow \tilde{s}\} \cup \{\emptyset\} = \{\Gamma \mid \exists \tilde{s}'\ s' \xrightarrow{\Gamma}\!\!\!\!\rightarrow \tilde{s}'\} \cup \{\emptyset\} = Exec(s')$. Note that in the previous equality we can always find the pairs of states s and s' related by the isomorphism of $TS^*(G)$ and $TS^*(G')$. Further, the definition of $PT(\Gamma, s)$ depends on $Exec(s)$ only rather than on concrete s. Thus, for each state s of $TS(G)$ the probabilities of outgoing transitions will be the same as for the corresponding state s' of $TS(G')$. Hence, $TS(G)$ and $TS(G')$ are isomorphic. □

Note that, though isomorphism of transition systems with and without empty loops appears to be the same relation, the equivalences defined on these two types of transition systems could be different. This is the case when the relations abstract from concrete activities which can happen (more exactly, from their probability parts) and take into account the overall probabilities to execute multiactions only. It is clear that the equivalences defined through transition systems with empty loops imply the relations based on those without empty loops, but the reverse implication is not valid.

For instance, we have defined stochastic isomorphism with the use of transition systems without empty loops. We can define the corresponding relation based on transition systems with empty loops as well. Then the latter equivalence will be strictly stronger than the former. As mentioned above, we decided to abstract from empty loops because of the difficulties with infinite internal behavior. Now we can give another reason for this decision: the equivalences based on transition systems with empty loops are rather cumbersome. The following example explains why.

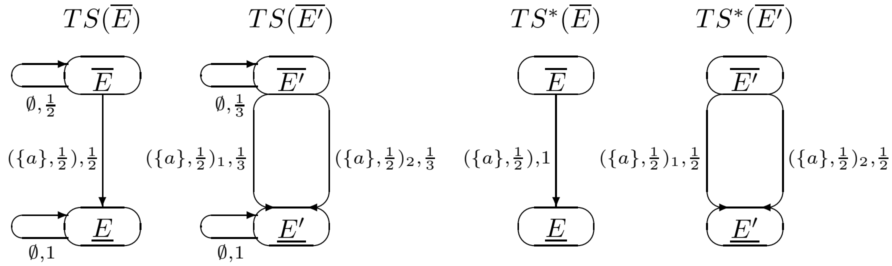

Figure 13: A problem with the stochastic isomorphism based on transition systems with empty loops

$$
\begin{tikzcd}
 & \approx \arrow[d] \\
 & =_{ts} \arrow[d] \\
 & =_{sto} \arrow[d] \\
\underline{\leftrightarrow}_{is} \arrow[d] & \underline{\leftrightarrow}_{ss} \arrow[l] \arrow[d] \\
\equiv_{is} & \equiv_{ss} \arrow[l]
\end{tikzcd}
$$

Figure 14: Interrelations of the stochastic equivalences

Example 5.5 *Let $E = (\{a\}, \frac{1}{2})$ and $E' = (\{a\}, \frac{1}{2})_1[](\{a\}, \frac{1}{2})_2$. Then $\overline{E} =_{sto} \overline{E'}$, but $\overline{E}$ is not equivalent to $\overline{E'}$ according to the stronger version of stochastic isomorphism, since the probability of the only non-empty transition in $TS(\overline{E})$ is $\frac{1}{2}$, whereas the probability of both non-empty transitions in $TS(\overline{E'})$ is $\frac{1}{3}$, and $\frac{1}{2} \neq \frac{1}{3} + \frac{1}{3}$. On the other hand, the probability of the only non-empty transition in $TS^*(\overline{E})$ is 1, the probability of both non-empty transitions in $TS^*(\overline{E'})$ is $\frac{1}{2}$, and $1 = \frac{1}{2} + \frac{1}{2}$. The transition systems with and without empty loops of $\overline{E}$ and $\overline{E}'$ are presented in Figure 13.*

In the continuous time setting of $sPBC$ there are no problems with equivalences like in the example above, but only interleaving relations can be introduced. On the other hand, the concurrency information from expressions has to be preserved in their transition systems to define correctly the congruence relation [59, 60, 63].

In the following, the symbol '_' will denote "nothing", and the equivalences subscribed by it are considered as those without any subscription such as 'is', 'ss', 'sto' or 'ts'.

Theorem 5.2 *Let $\leftrightarrow, \twoheadleftarrow\!\!\!\!\twoheadrightarrow \in \{\equiv, \underline{\leftrightarrow}, =, \approx\}$ and $\star, \star\star \in \{\_, is, ss, sto, ts\}$. For dynamic expressions G and G'*

$$G \leftrightarrow_\star G' \;\Rightarrow\; G \twoheadleftarrow\!\!\!\!\twoheadrightarrow_{\star\star} G'$$

iff there exists a directed path from $\leftrightarrow_\star$ to $\twoheadleftarrow\!\!\!\!\twoheadrightarrow_{\star\star}$ in the graph in Figure 14.

Proof. ($\Leftarrow$) Let us check the validity of implications in the graph in Figure 14.

- The implications $\leftrightarrow_{ss} \to \leftrightarrow_{is},\ \leftrightarrow \in \{\equiv, \underline{\leftrightarrow}\}$ are valid, since single activities are one-element multisets.
- The implications $\underline{\leftrightarrow}_{\star s} \to \equiv_{\star s},\ \star \in \{i, s\}$, are valid by Proposition 5.2.

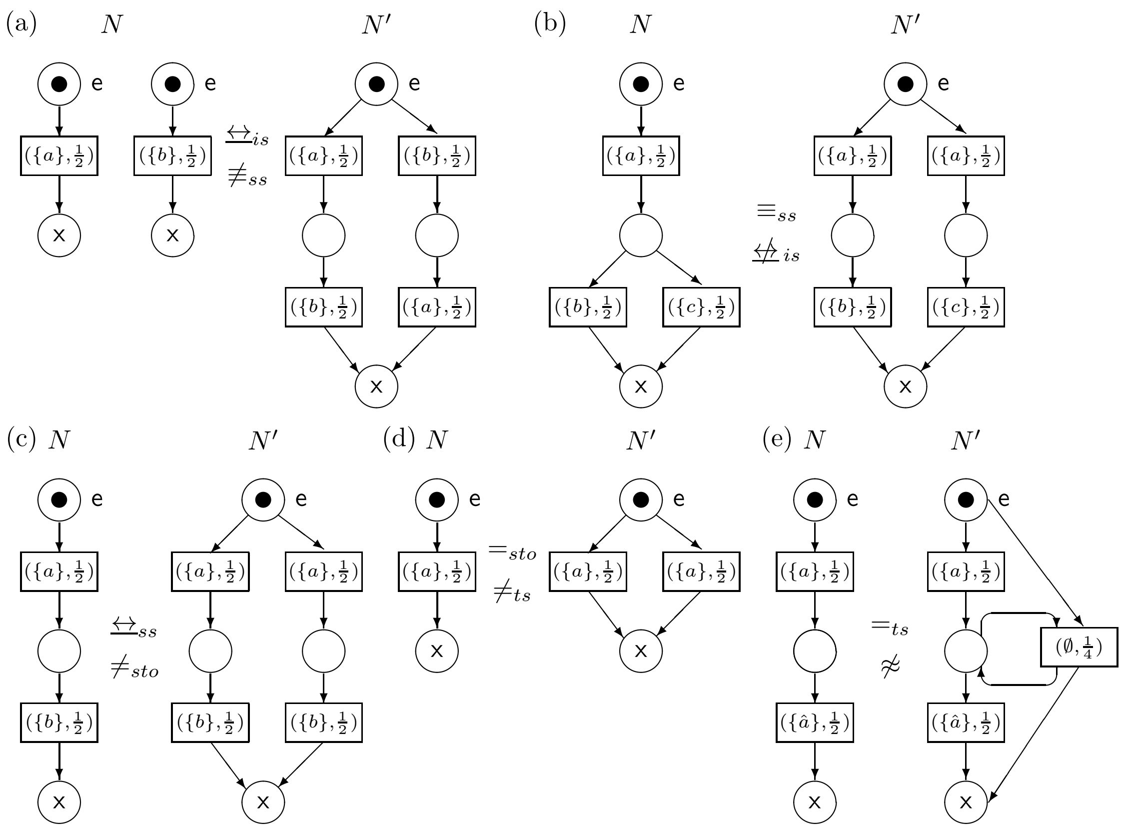

Figure 15: Dts-boxes of the dynamic expressions from equivalence examples of Theorem 5.2

- The implication $=_{sto} \rightarrow \underline{\leftrightarrow}_{ss}$ is proved as follows. Let $\beta : G =_{sto} G'$. Then it is easy to see that $\mathcal{S} : G \underline{\leftrightarrow}_{ss} G'$, where $\mathcal{S} = \{(s, \beta(s)) \mid s \in DR(G)\}$.
- The implication $=_{ts} \rightarrow =_{sto}$ is valid, since stochastic isomorphism is that of transition systems without empty loops up to merging of transitions with labels having identical multiaction parts.
- The implication $\approx \rightarrow =_{ts}$ is valid, since the transition system of a dynamic formula is defined based on its structural equivalence class.

($\Rightarrow$) The absence of additional nontrivial arrows (not resulting from the combination of the existing ones) in the graph in Figure 14 is proved by the following examples.

- Let $E = (\{a\}, \frac{1}{2}) \| (\{b\}, \frac{1}{2})$ and $E' = ((\{a\}, \frac{1}{2}); (\{b\}, \frac{1}{2}))[]((\{b\}, \frac{1}{2}); (\{a\}, \frac{1}{2}))$. Then $\overline{E} \underline{\leftrightarrow}_{is} \overline{E'}$, but $\overline{E} \not\equiv_{ss} \overline{E'}$, since only in $TS^*(\overline{E'})$ multiactions $\{a\}$ and $\{b\}$ cannot be executed concurrently.
- Let $E = (\{a\}, \frac{1}{2}); ((\{b\}, \frac{1}{2})[](\{c\}, \frac{1}{2}))$ and $E' = ((\{a\}, \frac{1}{2}); (\{b\}, \frac{1}{2}))[]((\{a\}, \frac{1}{2}); (\{c\}, \frac{1}{2}))$. Then $\overline{E} \equiv_{ss} \overline{E'}$, but $\overline{E} \underline{\not\leftrightarrow}_{is} \overline{E'}$, since only in $TS^*(\overline{E'})$ a multiaction $\{a\}$ can be executed so that no multiaction $\{b\}$ can occur afterwards.
- Let $E = (\{a\}, \frac{1}{2}); (\{b\}, \frac{1}{2})$ and $E' = (\{a\}, \frac{1}{2}); (\{b\}, \frac{1}{2})[](\{a\}, \frac{1}{2}); (\{b\}, \frac{1}{2})$. Then $\overline{E} \underline{\leftrightarrow}_{ss} \overline{E'}$, but $\overline{E} \neq_{sto} \overline{E'}$, since $TS^*(\overline{E'})$ has more states than $TS^*(\overline{E})$.
- Let $E = (\{a\}, \frac{1}{2})$ and $E' = (\{a\}, \frac{1}{2})_1[](\{a\}, \frac{1}{2})_2$. Then $\overline{E} =_{sto} \overline{E'}$, but $\overline{E} \neq_{ts} \overline{E'}$, since only $TS(\overline{E'})$ has two transitions.
- Let $E = (\{a\}, \frac{1}{2}); (\{\hat{a}\}, \frac{1}{2})$ and $E' = ((\{a\}, \frac{1}{2}); (\{\hat{a}\}, \frac{1}{2}))$ sy a. Then $\overline{E} =_{ts} \overline{E'}$, but $\overline{E} \not\approx \overline{E'}$, since $\overline{E}$ and $\overline{E'}$ cannot be reached from each other by applying inaction rules. □

Example 5.6 *In Figure 15 the marked dts-boxes corresponding to the dynamic expressions from equivalence examples of Theorem 5.2 are presented, i.e.,* $N = Box_{dts}(\overline{E})$ *and* $N' = Box_{dts}(\overline{E'})$ *for each picture (a)–(e).*

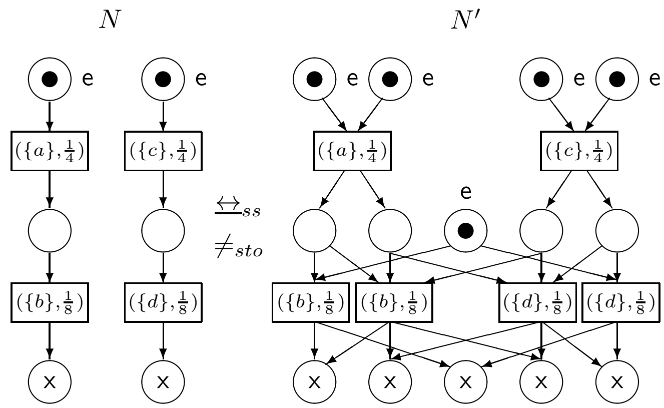

Figure 16: Reduction of a dts-box up to $\underline{\leftrightarrow}_{ss}$

6 Reduction modulo equivalences

The equivalences we proposed can be used to reduce transition systems and DTMCs of expressions (reachability graphs and DTMCs of dts-boxes) as well as the expressions (the dts-boxes) themselves. Under the reductions of graph-based models like transition systems, reachability graphs and DTMCs we understand those with less states (the graph nodes). A reduction of expressions should result to the shorter ones with simpler structure, i.e., to those having less operators and activities. The goal of the reduction is to decrease the number of states in the semantic representation of the modeled system while preserving its important qualitative and quantitative properties. Thus, the reduction allows one to simplify behavioural and performance analysis of systems.

The following example demonstrates how the stochastic equivalences can be used to simplify process expressions. Accordingly, the net analogues of the relations can be used for reduction of dts-boxes.

Example 6.1 *Let* $E = ((\{a\},\frac{1}{2});(\{b\},\frac{1}{2}))\|((\{c\},\frac{1}{2});(\{d\},\frac{1}{2}))$ *and* $E' = (((\{a,x\},\frac{1}{2});((\{b,y_1\},\frac{1}{2})[](\{b,y_2\},\frac{1}{2})))$ $\|((\{a,\hat{x}\},\frac{1}{2});((\{b,\widehat{y_2},y_2'\},\frac{1}{2})[](\{d,v_1\},\frac{1}{2})))\|((\{c,z\},\frac{1}{2});((\{b,\widehat{y_2'}\},\frac{1}{2})[](\{d,\widehat{v_1},v_1'\},\frac{1}{2})))\|((\{c,\hat{z}\},\frac{1}{2});((\{d,\widehat{v_1'}\},\frac{1}{2})[]$ $(\{d,v_2\},\frac{1}{2})))\|((\{b,\widehat{y_1}\},\frac{1}{4})[](\{d,\widehat{v_2}\},\frac{1}{4})))$ $\mathsf{sy}\ x\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ y_2'\ \mathsf{sy}\ z\ \mathsf{sy}\ v_1\ \mathsf{sy}\ v_1'\ \mathsf{sy}\ v_2\ \mathsf{rs}\ x\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ y_2'\ \mathsf{rs}\ z\ \mathsf{rs}\ v_1$ $\mathsf{rs}\ v_1'\ \mathsf{rs}\ v_2$. *Then* $\overline{E}\underline{\leftrightarrow}_{ss}\overline{E'}$, *but* $\overline{E} \neq_{sto} \overline{E'}$, *since* $TS^*(\overline{E'})$ *has more states than* $TS^*(\overline{E})$. *It is clear that the syntax of* E *is much simpler than that of* E', *but both static expressions have the same semantics induced by* $\underline{\leftrightarrow}_{ss}$. *Hence,* E *is a simplification of* E' *with respect to* $\underline{\leftrightarrow}_{ss}$.

In Figure 16 the marked dts-boxes corresponding to the dynamic expressions above are presented, i.e., $N = Box_{dts}(\overline{E})$ *and* $N' = Box_{dts}(\overline{E'})$. *Thus,* N *is a reduction of* N' *up to the net version of* $\underline{\leftrightarrow}_{ss}$.

In the general case, the procedure of expressions reduction cannot be transferred smoothly from a transition systems level. The problem is that the transition system of the reduced expression in some cases can be further reduced in such a way that it will not correspond to any expression anymore. At the net level, the reduced transition system will be isomorphic to the reachability graph of a non-safe net which naturally cannot be a dts-box of any expression.

An *autobisimulation (equivalence)* is a bisimulation (equivalence) between an expression and itself.

For a dynamic expression G and the step stochastic autobisimulation equivalence $G\underline{\leftrightarrow}_{ss}G$ on it let $\mathcal{K} \in DR(G)/_{\underline{\leftrightarrow}_{ss}}$ and $s_1, s_2 \in \mathcal{K}$. We have $\forall \widetilde{\mathcal{K}} \in DR(G)/_{\underline{\leftrightarrow}_{ss}}\ \forall A \in I\!N_f^{\mathcal{L}} \setminus \{\emptyset\}\ s_1 \xrightarrow{A}\!\!\!\!\rightarrow_{\mathcal{P}} \widetilde{\mathcal{K}} \Leftrightarrow s_2 \xrightarrow{A}\!\!\!\!\rightarrow_{\mathcal{P}} \widetilde{\mathcal{K}}$. The previous equality is valid for all $s_1, s_2 \in \mathcal{K}$, hence, we can rewrite it as $\mathcal{K} \xrightarrow{A}\!\!\!\!\rightarrow_{\mathcal{P}} \widetilde{\mathcal{K}}$, where $\mathcal{P} = PM_A^*(\mathcal{K},\widetilde{\mathcal{K}}) = PM_A^*(s_1,\widetilde{\mathcal{K}}) = PM_A^*(s_2,\widetilde{\mathcal{K}})$.

We write $\mathcal{K} \xrightarrow{A}\!\!\!\!\rightarrow \widetilde{\mathcal{K}}$ if $\exists \mathcal{P}\ \mathcal{K} \xrightarrow{A}\!\!\!\!\rightarrow_{\mathcal{P}} \widetilde{\mathcal{K}}$ and $\mathcal{K} \twoheadrightarrow \widetilde{\mathcal{K}}$ if $\exists A\ \mathcal{K} \xrightarrow{A}\!\!\!\!\rightarrow \widetilde{\mathcal{K}}$.

The similar arguments allow us to use the notation $\mathcal{K} \twoheadrightarrow_{\mathcal{P}} \widetilde{\mathcal{K}}$, where $\mathcal{P} = PM^*(\mathcal{K},\widetilde{\mathcal{K}}) = PM^*(s_1,\widetilde{\mathcal{K}}) = PM^*(s_2,\widetilde{\mathcal{K}})$.

Based on the equivalence classes with respect to $\underline{\leftrightarrow}_{ss}$, the quotient transition systems without empty loops and quotient underlying DTMC without empty loops of expressions can be defined.

Definition 6.1 *The* quotient (by $\underline{\leftrightarrow}_{ss}$) (labeled probabilistic) transition system without empty loops *of a dynamic expression* G *is a quadruple* $TS^*_{\underline{\leftrightarrow}_{ss}}(G) = (S_{\underline{\leftrightarrow}_{ss}}, L_{\underline{\leftrightarrow}_{ss}}, \mathcal{T}_{\underline{\leftrightarrow}_{ss}}, s_{\underline{\leftrightarrow}_{ss}})$, *where*

- $S_{\underline{\leftrightarrow}_{ss}} = DR(G)/_{\underline{\leftrightarrow}_{ss}}$;

- $L_{\underline{\leftrightarrow}_{ss}} \subseteq (I\!N_f^{\mathcal{L}} \setminus \{\emptyset\}) \times (0; 1]$;
- $\mathcal{T}_{\underline{\leftrightarrow}_{ss}} = \{(\mathcal{K}, (A, PM_A^*(\mathcal{K}, \widetilde{\mathcal{K}})), \widetilde{\mathcal{K}}) \mid \mathcal{K} \in DR(G)/_{\underline{\leftrightarrow}_{ss}},\ \mathcal{K} \stackrel{A}{\twoheadrightarrow} \widetilde{\mathcal{K}}\}$;
- $s_{\underline{\leftrightarrow}_{ss}} = \{[G]_{\approx}\}$.

The transition $(\mathcal{K}, (A, \mathcal{P}), \widetilde{\mathcal{K}}) \in \mathcal{T}_{\underline{\leftrightarrow}_{ss}}$ will be written as $\mathcal{K} \stackrel{A}{\twoheadrightarrow}_{\mathcal{P}} \widetilde{\mathcal{K}}$.

The quotient (by $\underline{\leftrightarrow}_{ss}$) transition systems without empty loops of static expressions can be defined as well. For $E \in RegStatExpr$, let $TS^*_{\underline{\leftrightarrow}_{ss}}(E) = TS^*_{\underline{\leftrightarrow}_{ss}}(\overline{E})$.

Definition 6.2 *Let G be a dynamic expression. The* quotient (by $\underline{\leftrightarrow}_{ss}$) underlying DTMC without empty loops *of G, denoted by $DTMC^*_{\underline{\leftrightarrow}_{ss}}(G)$, has the state space $DR(G)/_{\underline{\leftrightarrow}_{ss}}$ and the transitions $\mathcal{K} \twoheadrightarrow_{\mathcal{P}} \widetilde{\mathcal{K}}$, where $\mathcal{P} = PM^*(\mathcal{K}, \widetilde{\mathcal{K}})$.*

The quotient (by $\underline{\leftrightarrow}_{ss}$) underlying DTMCs without empty loops of static expressions can be defined as well. For $E \in RegStatExpr$, let $DTMC^*_{\underline{\leftrightarrow}_{ss}}(E) = DTMC^*_{\underline{\leftrightarrow}_{ss}}(\overline{E})$.

The comprehensive reduction examples will be presented in Section 10.

7 Logical characterization

In this section, a logical characterization of stochastic bisimulation equivalences is accomplished via formulas of probabilistic modal logics. The results obtained could be interpreted as an operational characterization of the corresponding logical equivalences. Dynamic expressions are considered as logically equivalent if they satisfy the same formulas.

7.1 Logic $iPML$

The probabilistic modal logic PML has been introduced in [50] on probabilistic transition systems without invisible actions for logical interpretation of the interleaving probabilistic bisimulation equivalence. On the basis of PML, we propose a new interleaving modal logic $iPML$ used for characterization of the interleaving stochastic bisimulation equivalence.

Definition 7.1 *Let $\top$ denote the truth and $\alpha \in \mathcal{L},\ \mathcal{P} \in (0; 1]$. A* formula *of $iPML$ is defined as follows:*

$$\Phi ::= \ \top \mid \neg\Phi \mid \Phi \wedge \Phi \mid \langle\alpha\rangle_{\mathcal{P}}\Phi.$$

We define $\langle\alpha\rangle\Phi = \exists\mathcal{P}\ \langle\alpha\rangle_{\mathcal{P}}\Phi$.

iPML denotes the set of *all formulas of the logic $iPML$.*

Definition 7.2 *Let G be a dynamic expression and $s \in DR(G)$. The* satisfaction relation $\models_G \subseteq DR(G) \times$ **iPML** *is defined as follows:*

1. *$s \models_G \top$ — always;*
2. *$s \models_G \neg\Phi$, if $s \not\models_G \Phi$;*
3. *$s \models_G \Phi \wedge \Psi$, if $s \models_G \Phi$ and $s \models_G \Psi$;*
4. *$s \models_G \langle\alpha\rangle_{\mathcal{P}}\Phi$, if $\exists\mathcal{H} \subseteq DR(G)\ s \xrightarrow{\alpha}_{\mathcal{Q}} \mathcal{H},\ \mathcal{Q} \geq \mathcal{P}$ and $\forall\tilde{s} \in \mathcal{H}\ \tilde{s} \models_G \Phi$.*

Note that $\langle\alpha\rangle_{\mathcal{Q}}\Phi$ implies $\langle\alpha\rangle_{\mathcal{P}}\Phi$, if $\mathcal{Q} \geq \mathcal{P}$.

Definition 7.3 *We write $G \models_G \Phi$, if $[G]_{\approx} \models_G \Phi$. Two dynamic expressions G and G' are* logically equivalent *in $iPML$, denoted by $G =_{iPML} G'$, if $\forall\Phi \in$ **iPML** $G \models_G \Phi\ \Leftrightarrow\ G' \models_{G'} \Phi$.*

Let G be a dynamic expression and $s \in DR(G),\ \alpha \in \mathcal{L}$. The set of states reached from s by execution of multiaction α, the *image set*, is defined as $Image(s, \alpha) = \{\tilde{s} \mid \exists\{(\alpha, \rho)\} \in Exec(s)\ s \stackrel{(\alpha,\rho)}{\twoheadrightarrow} \tilde{s}\}$. A dynamic expression G is an *image-finite* one, if $\forall s \in DR(G)\ \forall\alpha \in \mathcal{L}\ |Image(s, \alpha)| < \infty$.

Theorem 7.1 *For image-finite dynamic expressions G and G'*

$$G \underline{\leftrightarrow}_{is} G'\ \Leftrightarrow\ G =_{iPML} G'.$$

Proof. As the subsequent Theorem 7.2, but with state changes due to execution of single multiactions and the interleaving transition relation. ☐

Hence, in the interleaving semantics, we obtained a logical characterization of the stochastic bisimulation relation or, symmetrically, an operational characterization of the probabilistic modal logic equivalence.

Example 7.1 *Let $E = (\{a\}, \frac{1}{2}); ((\{b\}, \frac{1}{2})[](\{c\}, \frac{1}{2}))$ and $E' = ((\{a\}, \frac{1}{2}); (\{b\}, \frac{1}{2}))[]((\{a\}, \frac{1}{2}); (\{c\}, \frac{1}{2}))$. Then $\overline{E} \neq_{iPML} \overline{E'}$, because for $\Phi = \langle\{a\}\rangle_1\langle\{b\}\rangle_{\frac{1}{2}}\top$ we have $\overline{E} \models_{\overline{E}} \Phi$, but $\overline{E'} \not\models_{\overline{E'}} \Phi$, since in $TS^*(\overline{E'})$ a multiaction $\{a\}$ can be executed so that no multiaction $\{b\}$ can occur afterwards.*

7.2 Logic $sPML$

On the basis of PML, we propose a new step modal logic $sPML$ used for characterization of the step stochastic bisimulation equivalence.

Definition 7.4 *Let $\top$ denote the truth and $A \in I\!N_f^{\mathcal{L}} \setminus \{\emptyset\}$, $\mathcal{P} \in (0; 1]$. A* formula *of $sPML$ is defined as follows:*

$$\Phi ::= \top \mid \neg\Phi \mid \Phi \wedge \Phi \mid \langle A\rangle_{\mathcal{P}}\Phi.$$

We define $\langle A\rangle\Phi = \exists\mathcal{P}\ \langle A\rangle_{\mathcal{P}}\Phi$.

sPML denotes the set of *all formulas of the logic $sPML$.*

Definition 7.5 *Let G be a dynamic expression and $s \in DR(G)$. The* satisfaction relation $\models_G \subseteq DR(G) \times \mathbf{sPML}$ *is defined as follows:*

1. *$s \models_G \top$ — always;*
2. *$s \models_G \neg\Phi$, if $s \not\models_G \Phi$;*
3. *$s \models_G \Phi \wedge \Psi$, if $s \models_G \Phi$ and $s \models_G \Psi$;*
4. *$s \models_G \langle A\rangle_{\mathcal{P}}\Phi$, if $\exists\mathcal{H} \subseteq DR(G)\ s \xrightarrow{A}_{\mathcal{Q}} \mathcal{H}$, $\mathcal{Q} \geq \mathcal{P}$ and $\forall\tilde{s} \in \mathcal{H}\ \tilde{s} \models_G \Phi$.*

Note that $\langle A\rangle_{\mathcal{Q}}\Phi$ implies $\langle A\rangle_{\mathcal{P}}\Phi$, if $\mathcal{Q} \geq \mathcal{P}$.

Definition 7.6 *We write $G \models_G \Phi$, if $[G]_\approx \models_G \Phi$. Two dynamic expressions G and G' are* logically equivalent *in $sPML$, denoted by $G =_{sPML} G'$, if $\forall\Phi \in \mathbf{sPML}\ G \models_G \Phi \Leftrightarrow G' \models_{G'} \Phi$.*

Let G be a dynamic expression and $s \in DR(G)$, $A \in I\!N_f^{\mathcal{L}} \setminus \{\emptyset\}$. The set of states reached from s by execution of a multiset of multiactions A, the *image set*, is defined as $Image(s, A) = \{\tilde{s} \mid \exists\Gamma \in Exec(s)\ \mathcal{L}(\Gamma) = A,\ s \xrightarrow{\Gamma} \tilde{s}\}$. A dynamic expression G is an *image-finite* one, if $\forall s \in DR(G)\ \forall A \in I\!N_f^{Act}\ |Image(s, A)| < \infty$.

Theorem 7.2 *For image-finite dynamic expressions G and G'*

$$G \underline{\leftrightarrow}_{ss} G' \Leftrightarrow G =_{sPML} G'.$$

Proof. ($\Leftarrow$) To simplify the presentation, we propose the *indicator function* Ξ that recovers a dynamic expression by a state belonging to its derivation set. For a dynamic expression G and $s \in DR(G)$ we define $\Xi(s) = G$.

Let us define the equivalence relation $\mathcal{R} = \{(s_1, s_2) \in (DR(G) \cup DR(G'))^2 \mid \forall\Phi \in \mathbf{sPML}\ s_1 \models_{\Xi(s_1)} \Phi \Leftrightarrow s_2 \models_{\Xi(s_2)} \Phi\}$. We have $([G]_\approx, [G']_\approx) \in \mathcal{R}$. Let us prove that $\mathcal{R}$ is a step stochastic bisimulation.

Assume that $[G]_\approx \xrightarrow{A}_{\mathcal{P}} \mathcal{H} \in (DR(G) \cup DR(G'))/_{\mathcal{R}}$. Let $[G']_\approx \xrightarrow{A}_{\mathcal{P}'_1} s'_1, \ldots, [G']_\approx \xrightarrow{A}_{\mathcal{P}'_i} s'_i, [G']_\approx \xrightarrow{A}_{\mathcal{P}'_{i+1}} s'_{i+1}, \ldots, [G']_\approx \xrightarrow{A}_{\mathcal{P}'_n} s'_n$ be changes of the state $[G']_\approx$ in the result of execution of the multiset of multiactions A. Since dynamic expression G' is an image-finite one, the number of such state changes is finite. The state changes are ordered so that $s'_1, \ldots s'_i \in \mathcal{H}$ and $s'_{i+1}, \ldots s'_n \notin \mathcal{H}$.

Then $\exists\Phi_{i+1}, \ldots, \Phi_n \in \mathbf{sPML}$ such that $\forall j\ (i+1 \leq j \leq n)\ \forall s \in \mathcal{H}\ s \models_{\Xi(s)} \Phi_j$, but $s'_j \not\models_{G'} \Phi_j$. We have $[G]_\approx \models_G \langle A\rangle_{\mathcal{P}}(\wedge_{j=i+1}^n \Phi_j)$ and $[G']_\approx \models_{G'} \langle A\rangle_{\left(1-\sum_{j=1}^i \mathcal{P}'_j\right)}\neg(\wedge_{j=i+1}^n \Phi_j)$.

Assume that $\mathcal{P} > \sum_{j=1}^i \mathcal{P}'_j$. Then $[G']_\approx \models_{G'} \langle A\rangle_{(1-\mathcal{P})}\neg(\wedge_{j=i+1}^n \Phi_j)$ and $[G']_\approx \not\models_{G'} \langle A\rangle_{\mathcal{P}}(\wedge_{j=i+1}^n \Phi_j)$ what contradicts to $([G]_\approx, [G']_\approx) \in \mathcal{R}$. Hence, $\mathcal{P} \leq \sum_{j=1}^i \mathcal{P}'_j$. Consequently, $[G']_\approx \xrightarrow{A}_{\mathcal{P}'} \mathcal{H}$, $\mathcal{P} \leq \sum_{j=1}^i \mathcal{P}'_j \leq \mathcal{P}'$. By symmetry of $\underline{\leftrightarrow}_{ss}$, we have $\mathcal{P} \geq \mathcal{P}'$. Thus, $\mathcal{P} = \mathcal{P}'$, and $\mathcal{R}$ is a step stochastic bisimulation.

($\Rightarrow$) It is sufficient to consider only the case $\langle A\rangle_{\mathcal{P}}\Phi$, since all other cases are trivial. Let for dynamic expressions G and G' $G\underline{\leftrightarrow}_{ss}G'$. Then $[G]_\approx\underline{\leftrightarrow}_{ss}[G']_\approx$. Assume that $[G]_\approx \models_G \langle A\rangle_{\mathcal{P}}\Phi$. Then $\exists\mathcal{H} \subseteq DR(G)\cup DR(G')$ such that $[G]_\approx \xrightarrow{A}\!\!\!\!\rightarrow_{\mathcal{Q}} \mathcal{H}$, $\mathcal{Q}\geq\mathcal{P}$ and $\forall s\in\mathcal{H}\ s\models_{\Xi(s)}\Phi$.

Let us define $\widetilde{\mathcal{H}} = \bigcup\{\overline{\mathcal{H}} \in (DR(G)\cup DR(G'))/_{\underline{\leftrightarrow}_{ss}} \mid \overline{\mathcal{H}}\cap\mathcal{H}\neq\emptyset\}$. Then $\forall\tilde{s}\in\widetilde{\mathcal{H}}\ \exists s\in\mathcal{H}\ s\underline{\leftrightarrow}_{ss}\tilde{s}$. Since $\forall s\in\mathcal{H}\ s\models_{\Xi(s)}\Phi$, we have $\forall\tilde{s}\in\widetilde{\mathcal{H}}\ \tilde{s}\models_{\Xi(\tilde{s})}\Phi$ by the induction hypothesis.

Since $\mathcal{H}\subseteq\widetilde{\mathcal{H}}$, we have $[G]_\approx \xrightarrow{A}\!\!\!\!\rightarrow_{\widetilde{\mathcal{Q}}} \widetilde{\mathcal{H}}$, $\widetilde{\mathcal{Q}}\geq\mathcal{Q}$. Since $\widetilde{\mathcal{H}}$ is the union of the equivalence classes with respect to $\underline{\leftrightarrow}_{ss}$, we have $[G]_\approx\underline{\leftrightarrow}_{ss}[G']_\approx$ implies $[G']_\approx \xrightarrow{A}\!\!\!\!\rightarrow_{\widetilde{\mathcal{Q}}} \widetilde{\mathcal{H}}$. Since $\widetilde{\mathcal{Q}}\geq\mathcal{Q}\geq\mathcal{P}$, we have $[G']_\approx\models_{G'}\langle A\rangle_{\mathcal{P}}\Phi$. Therefore, G' satisfies all the formulas which G does. By symmetry of $\underline{\leftrightarrow}_{ss}$, G satisfies all the formulas which G' does. Thus, the sets of formulas satisfiable for G and G' coincide. □

Hence, in the step semantics, we obtained a logical characterization of the stochastic bisimulation relation or, symmetrically, an operational characterization of the probabilistic modal logic equivalence.

Example 7.2 *Let $E = (\{a\},\frac{1}{2})\|(\{b\},\frac{1}{2})$ and $E' = ((\{a\},\frac{1}{2});(\{b\},\frac{1}{2}))[]((\{b\},\frac{1}{2});(\{a\},\frac{1}{2}))$. Then $\overline{E}\underline{\leftrightarrow}_{is}\overline{E'}$ but $\overline{E}\neq_{sPML}\overline{E'}$, because for $\Phi = \langle\{a,b\}\rangle_{\frac{1}{3}}\top$ we have $\overline{E}\models_{\overline{E}}\Phi$, but $\overline{E'}\not\models_{\overline{E'}}\Phi$, since only in $TS^*(\overline{E'})$ multiactions $\{a\}$ and $\{b\}$ cannot be executed concurrently.*

8 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. Note that the iteration operator does not guarantee infiniteness of behaviour, since there can exist a deadlock within the body (the second argument) of iteration when the corresponding subprocess does not reach its final state by some reasons.

Like in the framework of DTMCs, in DTSPNs the most interesting systems for performance analysis are *ergodic* (recurrent non-null, aperiodic and irreducible) ones. For ergodic DTSPNs, the steady-state marking probabilities exist and can be determined. In [58], the following sufficient conditions for ergodicity of DTSPNs are stated: *liveness* (for each transition and any reachable marking there exist a sequence of markings from it leading to the marking enabling that transition), *boundedness* (the number of tokens in every place is not greater than some fixed number for any reachable marking) and *nondeterminism* (the transition probabilites are strictly less than 1). For a dts-box with infinite behaviour these three conditions are partially satisfied: the dts-box is live within the body of each iteration operator it contains, it is safe (1-bounded) and nondeterministic. Hence, the parts of its DTMC corresponding to the execution of the iteration bodies are ergodic. The isomorphism between DTMCs of expressions and the corresponding dts-boxes which is stated by Proposition 5.1 guarantees that DTMCs of expressions with infinite behaviour are ergodic if restricted to the states in which the iteration bodies are executed.

In this section, we shall consider the expressions such that their underlined DTMCs contain one irreducible subset of states to guarantee the existence of the single steady state.

8.1 Theoretical background

Let G be a dynamic expression. The elements $\mathcal{P}^*_{ij}$ ($1\leq i,j\leq n=|DR(G)|$) of (one-step) 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, $k\in I\!N$) probability mass function (PMF) $\psi^*[k] = (\psi^*_1[k],\ldots,\psi^*_n[k])$ for $DTMC^*(G)$ is the solution of the equation system

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

where $\psi^*[0] = (\psi^*_1[0],\ldots,\psi^*_n[0])$ is the initial PMF defined as

$$\psi^*_i[0] = \begin{cases} 1, & s_i = [G]_\approx;\\ 0, & \text{otherwise.}\end{cases}$$

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

The steady-state PMF $\psi^* = (\psi^*_1,\ldots,\psi^*_n)$ 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 dimension n and $\mathbf{0}$ is a vector with n values 0, $\mathbf{1}$ is that with n values 1.

When $DTMC^*(G)$ has the single steady state, we have $\psi^* = \lim_{k\to\infty} \psi^*[k]$.

For $s \in DR(G)$ with $s = s_i$ $(1 \le i \le n)$ we define $\psi^*[k](s) = \psi^*_i[k]$ $(k \in I\!N)$ and $\psi^*(s) = \psi^*_i$.

Let G be a dynamic expression and $s, \tilde{s} \in DR(G)$. The following standard *performance indices (measures)* can be calculated based on the steady-state PMF for $DTMC^*(G)$.

- The *average recurrence (return) time in the state* s (i.e., the number of discrete time units or steps required for this) 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 *with respect to 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 $\sum_{s\in DR(G)} \psi^*(s)r(s)$.

We have intentionally decided to evaluate performance of the modeled systems with the use of the underlying DTMCs without empty loops of the corresponding expressions. This allows us to identify the expressions up to the equivalences defined on the basis of their transition systems without empty loops. Nevertheless, from the theoretical viewpoint, it is interesting to establish a relationship between steady-state PMFs for the underlying DTMCs with and without empty loops. The following theorem proposes the equation that relates the mentioned steady-state PMFs.

First, we introduce some helpful notation. For a vector $v = (v_1, \dots, v_n)$, let $Diag(v)$ be a diagonal matrix of dimension n with the elements $Diag_{ij}(v)$ $(1 \le i, j \le n)$ defined as

$$Diag_{ij}(v) = \begin{cases} v_i, & i = j; \\ 0, & \text{otherwise.} \end{cases}$$

Theorem 8.1 *Let G be a dynamic expression and EL be its empty loops abstraction vector. Then the steady-state PMFs ψ for $DTMC(G)$ and ψ^* for $DTMC^*(G)$ are related as follows: $\forall s \in DR(G)$*

$$\psi(s) = \frac{\psi^*(s)EL(s)}{\sum_{\tilde{s}\in DR(G)} \psi^*(\tilde{s})EL(\tilde{s})}.$$

Proof. Note that the TPM $\mathbf{P}$ and the steady-state PMF ψ for $DTMC(G)$ are defined like the corresponding notions for $DTMC^*(G)$.

Let $PT(\emptyset)$ be a vector with the elements $PT(\emptyset, s),\ s \in DR(G)$. We have $\mathbf{P}^* = Diag(EL)(\mathbf{P} - Diag(PT(\emptyset)))$. Further,

$$\psi^*(\mathbf{P}^* - \mathbf{E}) = \mathbf{0} \text{ and } \psi^*\mathbf{P}^* = \psi^*.$$

After replacement of $\mathbf{P}^*$ by the expression with $\mathbf{P}$ we obtain

$$\psi^* Diag(EL)(\mathbf{P} - Diag(PT(\emptyset))) = \psi^* \text{ and } \psi^* Diag(EL)\mathbf{P} = \psi^*(Diag(EL)Diag(PT(\emptyset)) + \mathbf{E}).$$

Note that $\forall s \in DR(G)$ $EL(s)PT(\emptyset, s) + 1 = \frac{PT(\emptyset,s)}{1-PT(\emptyset,s)} + 1 = \frac{1}{1-PT(\emptyset,s)} = EL(s)$, hence, $Diag(EL)Diag(PT(\emptyset)) + \mathbf{E} = Diag(EL)$. Thus,

$$\psi^* Diag(EL)\mathbf{P} = \psi^* Diag(EL).$$

Then for $v = \psi^* Diag(EL)$ we have

$$v\mathbf{P} = v \text{ and } v(\mathbf{P} - \mathbf{E}) = \mathbf{0}.$$

In order to calculate ψ on the basis of v, we must normalize it by dividing its elements by their sum, since we should have $\psi\mathbf{1}^T = 1$ as a result:

$$\psi = \frac{1}{v\mathbf{1}^T}v = \frac{1}{\psi^* Diag(EL)\mathbf{1}^T}\psi^* Diag(EL).$$

Thus, the elements of ψ are calculated as follows: $\forall s \in DR(G)$

$$\psi(s) = \frac{\psi^*(s)EL(s)}{\sum_{\tilde{s}\in DR(G)} \psi^*(\tilde{s})EL(\tilde{s})}.$$

It is easy to check that ψ is the solution of the equation system

$$\left\{\begin{array}{l} \psi(\mathbf{P} - \mathbf{E}) = \mathbf{0} \\ \psi\mathbf{1}^T = 1 \end{array}\right. ,$$

hence, it is indeed the steady-state PMF for $DTMC(G)$. $\square$

8.2 Steady state and equivalences

The following proposition demonstrates that for two dynamic expressions related by $\underline{\leftrightarrow}_{ss}$ 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 8.1 *Let G, G' be dynamic expressions with $\mathcal{R} : G\underline{\leftrightarrow}_{ss}G'$. 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 B. $\square$

Note that in the proof of Proposition 8.1 a limit construction us used to go from transient to stationary case. Thus, the result of this proposition is valid as well if we replace steady-state probabilities with transient ones in its statement.

We define the expression $\mathsf{Stop} = (\{c\}, \frac{1}{2})$ rs c specifying the special process analogous to the one used in the examples of [59, 60, 63]. The latter is a continuous time stochastic analogue of the stop process proposed in [8]. Stop is a discrete time stochastic analogue of the stop, it is only able to perform empty loops with probability 1 and never terminates. Note that in the specification of Stop one could use an arbitrary action from $\mathcal{A}$ and any probability belonging to the interval $(0; 1)$.

The following example demonstrates that the result of Proposition 8.1 does not hold for $\underline{\leftrightarrow}_{is}$.

Example 8.1 *Let* $E = [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2})\|(\{d\}, \frac{1}{2}))) * \mathsf{Stop}]$ *and*
$E' = [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); (((\{c\}, \frac{1}{2})_1; (\{d\}, \frac{1}{2})_1)[]((\{d\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2))) * \mathsf{Stop}]$. *We have* $\overline{E}\underline{\leftrightarrow}_{is}\overline{E'}$.

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

$s_1 = [[\overline{(\{a\}, \frac{1}{2})} * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2})\|(\{d\}, \frac{1}{2}))) * \mathsf{Stop}]]_\approx$,
$s_2 = [[(\{a\}, \frac{1}{2}) * (\overline{(\{b\}, \frac{1}{2})}; ((\{c\}, \frac{1}{2})\|(\{d\}, \frac{1}{2}))) * \mathsf{Stop}]]_\approx$,
$s_3 = [[(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); (\overline{(\{c\}, \frac{1}{2})}\|\overline{(\{d\}, \frac{1}{2})})) * \mathsf{Stop}]]_\approx$,
$s_4 = [[(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); (\underline{(\{c\}, \frac{1}{2})}\|\overline{(\{d\}, \frac{1}{2})})) * \mathsf{Stop}]]_\approx$,
$s_5 = [[(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); (\overline{(\{c\}, \frac{1}{2})}\|\underline{(\{d\}, \frac{1}{2})})) * \mathsf{Stop}]]_\approx$.

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

$s_1' = [[\overline{(\{a\}, \frac{1}{2})} * ((\{b\}, \frac{1}{2}); (((\{c\}, \frac{1}{2})_1; (\{d\}, \frac{1}{2})_1)[]((\{d\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2))) * \mathsf{Stop}]]_\approx$,
$s_2' = [[(\{a\}, \frac{1}{2}) * (\overline{(\{b\}, \frac{1}{2})}; (((\{c\}, \frac{1}{2})_1; (\{d\}, \frac{1}{2})_1)[]((\{d\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2))) * \mathsf{Stop}]]_\approx$,
$s_3' = [[(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); \overline{(((\{c\}, \frac{1}{2})_1; (\{d\}, \frac{1}{2})_1)[]((\{d\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2))}) * \mathsf{Stop}]]_\approx$,
$s_4' = [[(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); (((\{c\}, \frac{1}{2})_1; \overline{(\{d\}, \frac{1}{2})_1})[]((\{d\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2))) * \mathsf{Stop}]]_\approx$,
$s_5' = [[(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); (((\{c\}, \frac{1}{2})_1; (\{d\}, \frac{1}{2})_1)[]((\{d\}, \frac{1}{2})_2; \overline{(\{c\}, \frac{1}{2})_2}))) * \mathsf{Stop}]]_\approx$.

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

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

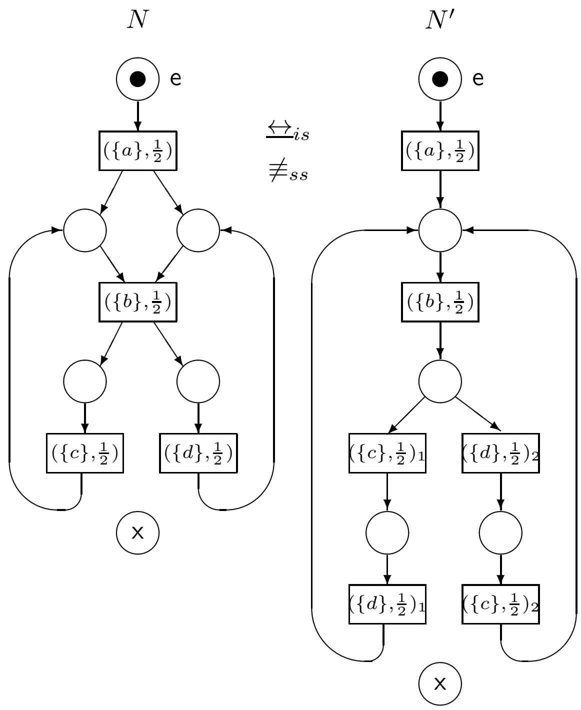

Figure 17: $\underline{\leftrightarrow}_{is}$ does not guarantee a coincidence of steady-state probabilities to come in an equivalence class

Consider the equivalence class $\mathcal{H} = \{s_3, s_3'\}$. We have $\sum_{s\in\mathcal{H}\cap DR(\overline{E})} \psi^*(s) = \psi^*(s_3) = \frac{3}{8}$, whereas $\sum_{s'\in\mathcal{H}\cap DR(\overline{E'})} {\psi'}^*(s') = {\psi'}^*(s_3') = \frac{1}{3}$. Thus, $\underline{\leftrightarrow}_{is}$ does not guarantee a coincidence of steady-state probabilities to come in an equivalence class.

In Figure 17 the marked dts-boxes corresponding to the dynamic expressions above are presented, i.e., $N = Box_{dts}(\overline{E})$ and $N' = Box_{dts}(\overline{E'})$.

The following example demonstrates that the result of Proposition 8.1 does not even hold for the intersection of $\underline{\leftrightarrow}_{is}$ and $\equiv_{ss}$.

Example 8.2 *Let $E = [(\{a\},\frac{1}{2}) * ((\{b\},\frac{1}{2}); ((\{c\},\frac{1}{2}) \| (\{d\},\frac{1}{2}))) * \mathsf{Stop}]$ and $E' = [(\{a\},\frac{1}{2}) * ((\{b\},\frac{1}{2}); (((\{c\},\frac{1}{2})_1 \| (\{d\},\frac{1}{2})_1))[](((\{c\},\frac{1}{2})_2; (\{d\},\frac{1}{2})_2)[]((\{d\},\frac{1}{2})_3; (\{c\},\frac{1}{2})_3))))) * \mathsf{Stop}]$. We have $\overline{E}\underline{\leftrightarrow}_{is}\overline{E'}$ and $\overline{E} \equiv_{ss} \overline{E'}$.*

$DR(\overline{E})$ is given in the Example 8.1. $DR(\overline{E'})$ consists of the equivalence classes

$s_1' = [[\overline{(\{a\},\frac{1}{2})} * ((\{b\},\frac{1}{2}); (((\{c\},\frac{1}{2})_1 \| (\{d\},\frac{1}{2})_1))[](((\{c\},\frac{1}{2})_2; (\{d\},\frac{1}{2})_2)[]((\{d\},\frac{1}{2})_3; (\{c\},\frac{1}{2})_3))))) * \mathsf{Stop}]]_\approx$,

$s_2' = [[(\{a\},\frac{1}{2}) * (\overline{(\{b\},\frac{1}{2})}; (((\{c\},\frac{1}{2})_1 \| (\{d\},\frac{1}{2})_1))[](((\{c\},\frac{1}{2})_2; (\{d\},\frac{1}{2})_2)[]((\{d\},\frac{1}{2})_3; (\{c\},\frac{1}{2})_3))))) * \mathsf{Stop}]]_\approx$,

$s_3' = [[(\{a\},\frac{1}{2}) * ((\{b\},\frac{1}{2}); (\overline{((\{c\},\frac{1}{2})_1 \| (\{d\},\frac{1}{2})_1))[](((\{c\},\frac{1}{2})_2; (\{d\},\frac{1}{2})_2)[]((\{d\},\frac{1}{2})_3; (\{c\},\frac{1}{2})_3))}))) * \mathsf{Stop}]]_\approx$,

$s_4' = [[(\{a\},\frac{1}{2}) * ((\{b\},\frac{1}{2}); (((\underline{(\{c\},\frac{1}{2})_1} \| \overline{(\{d\},\frac{1}{2})_1}))[](((\{c\},\frac{1}{2})_2; (\{d\},\frac{1}{2})_2)[]((\{d\},\frac{1}{2})_3; (\{c\},\frac{1}{2})_3))))) * \mathsf{Stop}]]_\approx$,

$s_5' = [[(\{a\},\frac{1}{2}) * ((\{b\},\frac{1}{2}); (((\overline{(\{c\},\frac{1}{2})_1} \| \underline{(\{d\},\frac{1}{2})_1}))[](((\{c\},\frac{1}{2})_2; (\{d\},\frac{1}{2})_2)[]((\{d\},\frac{1}{2})_3; (\{c\},\frac{1}{2})_3))))) * \mathsf{Stop}]]_\approx$,

$s_6' = [[(\{a\},\frac{1}{2}) * ((\{b\},\frac{1}{2}); (((\{c\},\frac{1}{2})_1 \| (\{d\},\frac{1}{2})_1))[](((\{c\},\frac{1}{2})_2; \overline{(\{d\},\frac{1}{2})_2})[]((\{d\},\frac{1}{2})_3; (\{c\},\frac{1}{2})_3))))) * \mathsf{Stop}]]_\approx$,

$s_7' = [[(\{a\},\frac{1}{2}) * ((\{b\},\frac{1}{2}); (((\{c\},\frac{1}{2})_1 \| (\{d\},\frac{1}{2})_1))[](((\{c\},\frac{1}{2})_2; (\{d\},\frac{1}{2})_2)[]((\{d\},\frac{1}{2})_3; \overline{(\{c\},\frac{1}{2})_3}))))) * \mathsf{Stop}]]_\approx$.

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

$$\psi^* = \left(0, \frac{3}{8}, \frac{3}{8}, \frac{1}{8}, \frac{1}{8}\right),\ {\psi'}^* = \left(0, \frac{13}{38}, \frac{13}{38}, \frac{3}{38}, \frac{3}{38}, \frac{3}{38}, \frac{3}{38}\right).$$

Consider the equivalence class $\mathcal{H} = \{s_3, s_3'\}$. We have $\sum_{s\in\mathcal{H}\cap DR(\overline{E})} \psi^*(s) = \psi^*(s_3) = \frac{3}{8}$, whereas $\sum_{s'\in\mathcal{H}\cap DR(\overline{E'})} {\psi'}^*(s') = {\psi'}^*(s_3') = \frac{13}{38}$. Thus, the intersection of $\underline{\leftrightarrow}_{is}$ and $\equiv_{ss}$ does not guarantee a coincidence of steady-state probabilities to come in an equivalence class.

In Figure 18 the marked dts-boxes corresponding to the dynamic expressions above are presented, i.e., $N = Box_{dts}(\overline{E})$ and $N' = Box_{dts}(\overline{E'})$.

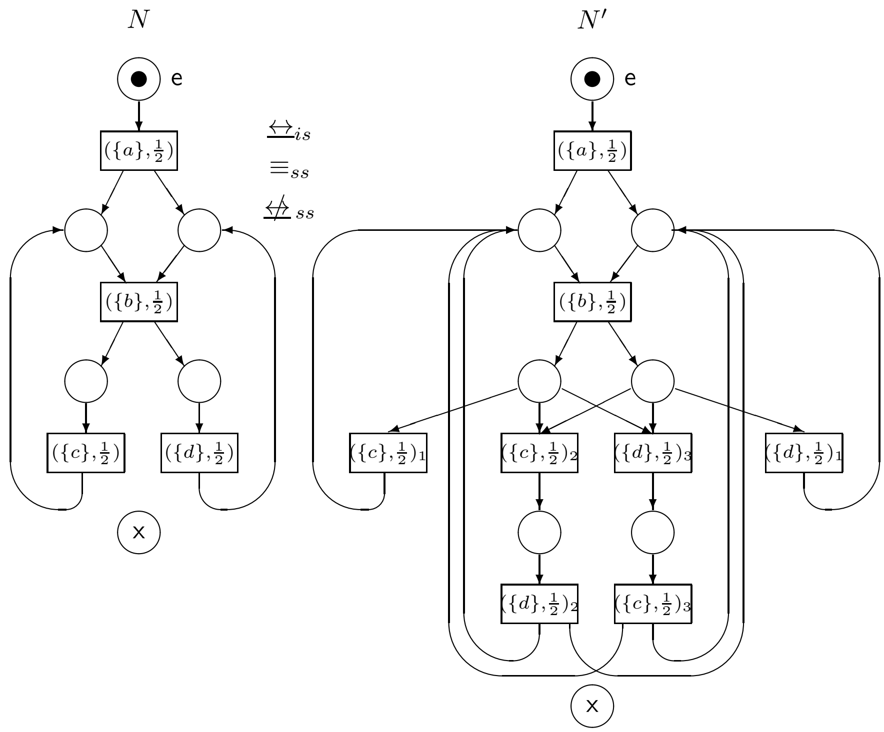

Figure 18: The intersection of $\underline{\leftrightarrow}_{is}$ and $\equiv_{ss}$ does not guarantee a coincidence of steady-state probabilities to come in an equivalence class

By Proposition 8.1, $\underline{\leftrightarrow}_{ss}$ preserves the quantitative properties of stationary behaviour (the level of DTMCs). Now we intend to demonstrate that the qualitative properties of stationary behaviour based of the multiaction labels are preserved as well (the level of transition systems).

Definition 8.1 *A* step trace *of a dynamic expression G is a chain $\Sigma = A_1 \cdots A_n \in (I\!N_f^{\mathcal{L}} \setminus \{\emptyset\})^*$ where $\exists s \in DR(G)\ s \stackrel{\Gamma_1}{\twoheadrightarrow} s_1 \stackrel{\Gamma_2}{\twoheadrightarrow} \cdots \stackrel{\Gamma_n}{\twoheadrightarrow} s_n,\ \mathcal{L}(\Gamma_i) = A_i\ (1 \le i \le n)$. Then the* probability to execute the step trace Σ in s *is*

$$PT^*(\Sigma, s) = \sum_{\{\Gamma_1,\ldots,\Gamma_n \mid s=s_0 \stackrel{\Gamma_1}{\twoheadrightarrow} s_1 \stackrel{\Gamma_2}{\twoheadrightarrow} \cdots \stackrel{\Gamma_n}{\twoheadrightarrow} s_n,\ \mathcal{L}(\Gamma_i)=A_i\ (1\le i\le n)\}} \prod_{i=1}^{n} PT^*(\Gamma_i, s_{i-1}).$$

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

Theorem 8.2 *Let G, G' be dynamic expressions with $\mathcal{R} : G\underline{\leftrightarrow}_{ss}G'$ 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 C. □

Note that in the proof of Theorem 8.2 a limit construction us used to go from transient to stationary case. Thus, the result of this theorem is valid as well if we replace steady-state probabilities with transient ones in its statement.

Example 8.3 *Let* $E = [(\{a\}, \frac{1}{2}) * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2})_1 [] (\{c\}, \frac{1}{2})_2)) * \mathsf{Stop}]$ *and*
$E' = [(\{a\}, \frac{1}{2}) * (((\{b\}, \frac{1}{2})_1; (\{c\}, \frac{1}{2})_1) [] ((\{b\}, \frac{1}{2})_2; (\{c\}, \frac{1}{2})_2)) * \mathsf{Stop}]$. *We have* $\overline{E} =_{sto} \overline{E'}$, *hence,* $\overline{E}\underline{\leftrightarrow}_{ss}\overline{E'}$.

$DR(\overline{E})$ *consists of the equivalence classes*
$s_1 = [[\overline{(\{a\}, \frac{1}{2})} * ((\{b\}, \frac{1}{2}); ((\{c\}, \frac{1}{2})_1 [] (\{c\}, \frac{1}{2})_2)) * \mathsf{Stop}]]_{\approx}$,

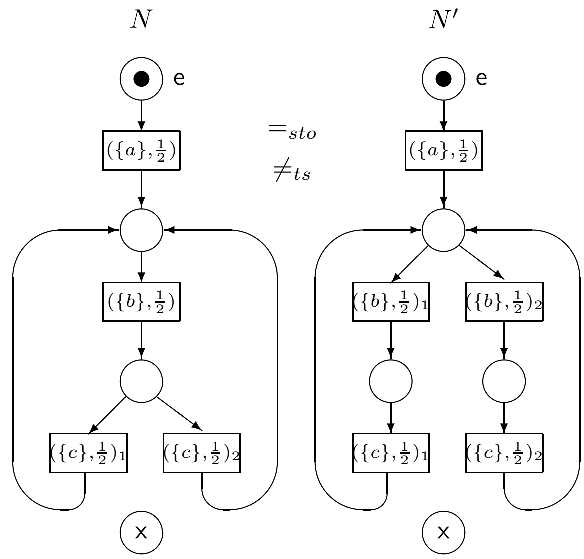

Figure 19: $\underline{\leftrightarrow}_{ss}$ implies a coincidence of the steady-state probabilities to come in an equivalence class and start a trace from it

$s_2 = [[(\{a\},\frac{1}{2}) * (\overline{(\{b\},\frac{1}{2})}; ((\{c\},\frac{1}{2})_1[](\{c\},\frac{1}{2})_2)) * \mathsf{Stop}]]_{\approx}$,
$s_3 = [[(\{a\},\frac{1}{2}) * ((\{b\},\frac{1}{2}); \overline{((\{c\},\frac{1}{2})_1[](\{c\},\frac{1}{2})_2)}) * \mathsf{Stop}]]_{\approx}$.

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

$s'_1 = [[\overline{(\{a\},\frac{1}{2})} * (((\{b\},\frac{1}{2})_1; (\{c\},\frac{1}{2})_1)[]((\{b\},\frac{1}{2})_2; (\{c\},\frac{1}{2})_2)) * \mathsf{Stop}]]_{\approx}$,
$s'_2 = [[(\{a\},\frac{1}{2}) * \overline{(((\{b\},\frac{1}{2})_1; (\{c\},\frac{1}{2})_1)[]((\{b\},\frac{1}{2})_2; (\{c\},\frac{1}{2})_2))} * \mathsf{Stop}]]_{\approx}$,
$s'_3 = [[(\{a\},\frac{1}{2}) * (((\{b\},\frac{1}{2})_1; \overline{(\{c\},\frac{1}{2})_1})[]((\{b\},\frac{1}{2})_2; (\{c\},\frac{1}{2})_2)) * \mathsf{Stop}]]_{\approx}$,
$s'_4 = [[(\{a\},\frac{1}{2}) * (((\{b\},\frac{1}{2})_1; (\{c\},\frac{1}{2})_1)[]((\{b\},\frac{1}{2})_2; \overline{(\{c\},\frac{1}{2})_2})) * \mathsf{Stop}]]_{\approx}$.

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

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

Consider the equivalence class $\mathcal{H} = \{s_3, s'_3, s'_4\}$. One can see that the steady-state probabilities for $\mathcal{H}$ coincide: $\sum_{s\in\mathcal{H}\cap DR(\overline{E})} \psi^*(s) = \psi^*(s_3) = \frac{1}{2} = \frac{1}{4} + \frac{1}{4} = \psi'^*(s'_3) + \psi'^*(s'_4) = \sum_{s'\in\mathcal{H}\cap DR(\overline{E'})} \psi'^*(s')$. 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: $\psi^*(s_3)(PT^*(\{(\{c\},\frac{1}{2})_1\}, s_3) + PT^*(\{(\{c\},\frac{1}{2})_2\}, s_3)) = \frac{1}{2}\left(\frac{1}{2} + \frac{1}{2}\right) = \frac{1}{2} = \frac{1}{4}\cdot 1 + \frac{1}{4}\cdot 1 = \psi'^*(s'_3)PT^*(\{(\{c\},\frac{1}{2})_1\}, s'_3) + \psi'^*(s'_4)PT^*(\{(\{c\},\frac{1}{2})_2\}, s'_4)$.

In Figure 19 the marked dts-boxes corresponding to the dynamic expressions above are presented, i.e., $N = Box_{dts}(\overline{E})$ and $N' = Box_{dts}(\overline{E'})$.

8.3 Preservation of performance 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. Proposition 8.1 and Theorem 8.2 guarantee a coincidence of the mentioned indices for the expressions related by $\underline{\leftrightarrow}_{ss}$. Thus, $\underline{\leftrightarrow}_{ss}$ preserves performance of stochastic systems modeled by expressions of *dtsPBC*. Moreover, Example 8.1 demonstrates that it is the weakest relation we considered that has the performance preservation property.

In addition, obviously, it is easier to evaluate performance with the use of a DTMC with less states, since in this case the dimension of the transition probability matrix will be smaller, and we shall solve systems of less equations to calculate steady-state probabilities. The reasoning above validates the following method of performance analysis simplification.

1. The system under investigation is specified by a static expression of *dtsPBC*.

2. The transition system without empty loops of the expression is constructed.

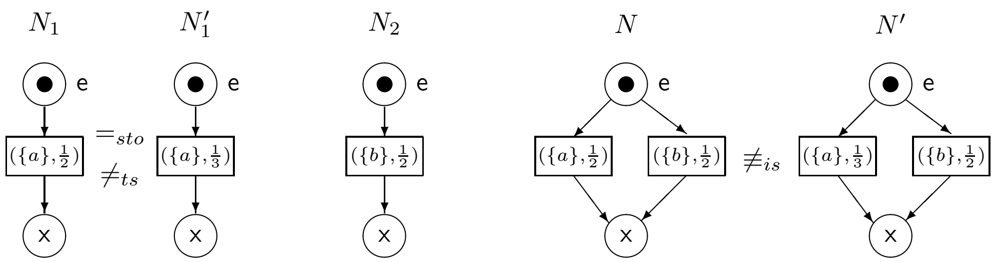

Figure 20: The equivalences between $\equiv_{is}$ and $=_{sto}$ are not congruences

3. After examining this transition system for self-similarity and symmetry, a step stochastic autobisimulation equivalence for the expression is determined.

4. The quotient underlying DTMC without empty loops of the expression is constructed.

5. The steady-state probabilities and performance indices based on this DTMC are calculated.

The limitation of the method above is its applicability only to the expressions such that their corresponding DTMCs contain one irreducible subset of states. I.e., an existence of exactly one stationary state is required. If a DTMC contains several irreducible subsets of states then several steady states can exist which depend on the initial PMF. There is the analytical method to determine the stable states for DTMCs of this kind as well. Note that for every expression the underlying DTMC without empty loops has by definition only one PMF (that at the time moment 0), hence, the stationary state will be only one in this case too. In addition, there is a sense to apply the method only to the systems with similar branches or symmetry in their behaviour.

9 Preservation by algebraic operations

An important question concerning equivalence relations is whether two compound expressions always remain equivalent if they are constructed from pairwise equivalent subexpressions. The equivalence having the mentioned property of preservation by algebraic operations is called a congruence. To be a congruence is a desirable property but not an obligatory one, since many important behavioural equivalences are not congruences. As a rule, a congruence relation is too discriminate, i.e., it differentiates too many formulas. This is the reason why a weaker but more interesting equivalence notion that is not a congruence is preferred in many cases when process behaviour is to be compared.

Definition 9.1 *Let $\leftrightarrow$ be an equivalence of dynamic expressions. Two static expressions E and E' are* equivalent with respect to $\leftrightarrow$*, denoted by $E \leftrightarrow E'$, if $\overline{E} \leftrightarrow \overline{E'}$.*

Let us investigate which algebraic equivalences we proposed are congruences on static expressions.
The following example demonstrates that no equivalence between $\equiv_{is}$ and $=_{sto}$ is a congruence.

Example 9.1 *Let $E = (\{a\}, \frac{1}{2})$, $E' = (\{a\}, \frac{1}{3})$ and $F = (\{b\}, \frac{1}{2})$. We have $\overline{E} =_{sto} \overline{E'}$, since both $TS^*(\overline{E})$ and $TS^*(\overline{E'})$ have the transitions with the multiaction part $\{a\}$ of their labels and probability 1. On the other hand, $\overline{E[]F} \not\equiv_{is} \overline{E'[]F}$, since only in $TS^*(\overline{E'[]F})$ the probabilities of the transitions with the multiaction parts $\{a\}$ and $\{b\}$ of their labels are different ($\frac{1}{3}$ and $\frac{2}{3}$, respectively). Thus, no equivalence between $\equiv_{is}$ and $=_{sto}$ is a congruence.*

In Figure 20 the marked dts-boxes corresponding to the dynamic expressions above are presented, i.e., $N_1 = Box_{dts}(\overline{E})$, $N_1' = Box_{dts}(\overline{E'})$, $N_2 = Box_{dts}(\overline{F})$ and $N = Box_{dts}(\overline{E[]F})$, $N' = Box_{dts}(\overline{E'[]F})$.

The following proposition demonstrates that all the equivalences between $\equiv_{is}$ and $=_{ts}$ are not congruences.

Proposition 9.1 *Let $\star \in \{i, s\}$, $\star\star \in \{sto, ts\}$. The equivalences $\equiv_\star$, $\underline{\leftrightarrow}_\star$, $=_{\star\star}$ are not preserved by algebraic operations.*

Proof. Let $E = (\{a\}, \frac{1}{2})$, $E' = (\{a\}, \frac{1}{2});\mathsf{Stop}$ and $F = (\{b\}, \frac{1}{2})$. We have $\overline{E} =_{ts} \overline{E'}$, since both $TS(\overline{E})$ and $TS(\overline{E'})$ have the transitions with the multiaction part $\{a\}$ of their labels and probability $\frac{1}{2}$. On the other hand, $\overline{E;F} \not\equiv_{is} \overline{E';F}$, since only in $TS^*(\overline{E';F})$ no other transition can fire after the transition with the multiaction part $\{a\}$ of its label. Thus, no equivalence between $\equiv_{is}$ and $=_{ts}$ is a congruence.

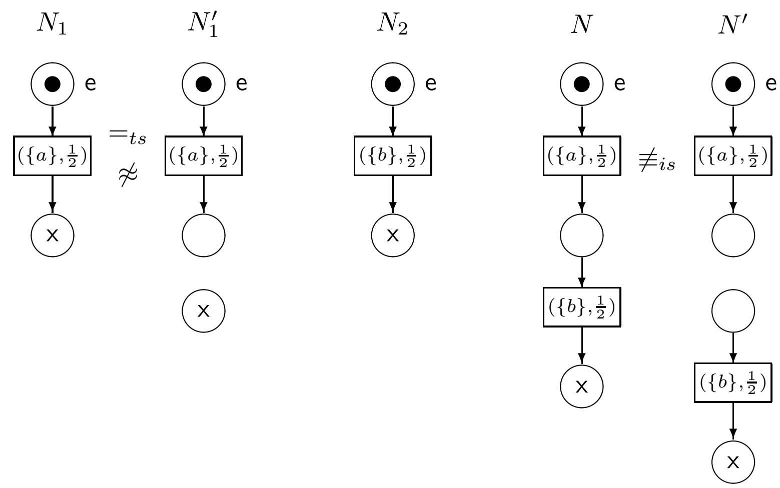

Figure 21: The equivalences between $\equiv_{is}$ and $=_{ts}$ are not congruences

In Figure 21 the marked dts-boxes corresponding to the dynamic expressions above are presented, i.e., $N_1 = Box_{dts}(\overline{E})$, $N_1' = Box_{dts}(\overline{E'})$, $N_2 = Box_{dts}(\overline{F})$ and $N = Box_{dts}(\overline{E[]F})$, $N' = Box_{dts}(\overline{E'[]F})$. $\square$

The following proposition demonstrates that $\approx$ is a congruence.

Proposition 9.2 *The equivalence $\approx$ is preserved by algebraic operations.*

Proof. By definition of $\approx$. $\square$

We suppose that, for an analogue of $=_{ts}$ to be a congruence, we have to equip transition systems of expressions with two extra transitions skip and redo like in [59, 63]. This allows one to avoid difficulties demonstrated in the example from the proof of Proposition 9.1 with unexpected termination due to the Stop process. At the same time, such an enrichment of transition systems does not overcome the problems explained in Example 9.1 with abstraction from empty loops. Hence, the equivalences between $\equiv_{is}$ and $=_{sto}$ defined on the basis of the enriched transition systems will still be non-congruences.

To define the analogue of $=_{ts}$ mentioned above, we shall introduce a notion of sr-transition system. It has the final state and two extra transitions from the initial state to the final one and back. Note that sr-transition systems do not have the loop transitions from the final state to itself. First, we propose the rules for skip and redo. Let $E \in RegStatExpr$.

$$\overline{E} \stackrel{\mathsf{skip}}{\rightarrow} \underline{E} \qquad\qquad \underline{E} \stackrel{\mathsf{redo}}{\rightarrow} \overline{E}$$

Now we can define sr-transition systems of dynamic expressions in the form $\overline{E}$, where E is a static expression. This syntactic restriction is needed to take into account two additional rules above. We assume that skip has probability 0, hence, it will be never executed. On the other hand, redo has probability 1, hence, it will be immediately executed at the next time moment if it is enabled.

Definition 9.2 *Let E be a static expression and $TS(\overline{E}) = (S, L, \mathcal{T}, s)$. The* (labeled probabilistic) sr-transition system *of $\overline{E}$ is a quadruple $TS_{sr}(\overline{E}) = (S_{sr}, L_{sr}, \mathcal{T}_{sr}, s_{sr})$, where*

- $S_{sr} = S \cup \{[\underline{E}]_{\approx}\}$;
- $L_{sr} \subseteq (\mathbb{N}_f^{\mathcal{SL}} \times (0;1]) \cup \{(\mathsf{skip}, 0), (\mathsf{redo}, 1)\}$;
- $\mathcal{T}_{sr} = \mathcal{T} \setminus \{([\underline{E}]_{\approx}, (\emptyset, 1), [\underline{E}]_{\approx})\} \cup \{([\overline{E}]_{\approx}, (\mathsf{skip}, 0), [\underline{E}]_{\approx}), ([\underline{E}]_{\approx}, (\mathsf{redo}, 1), [\overline{E}]_{\approx})\}$;
- $s_{sr} = s$.

We define a new notion of isomorphism for sr-transition systems, since we should take care of their final states.

Definition 9.3 *Let E, E' be static expressions and $TS_{sr}(\overline{E}) = (S_{sr}, L_{sr}, \mathcal{T}_{sr}, s_{sr})$, $TS_{sr}(\overline{E'}) = (S_{sr}', L_{sr}', \mathcal{T}_{sr}', s_{sr}')$ be their sr-transition systems. A mapping $\beta : S_{sr} \to S_{sr}'$ is an* isomorphism *between $TS_{sr}(\overline{E})$ and $TS_{sr}(\overline{E'})$, denoted by $\beta : TS_{sr}(\overline{E}) \simeq TS_{sr}(\overline{E'})$, if*

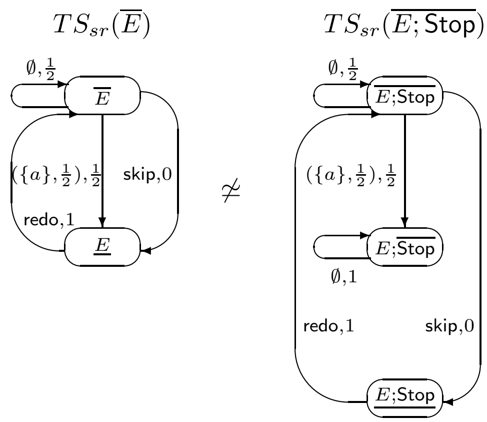

Figure 22: The sr-transition systems of $\overline{E}$ and $\overline{E;\mathsf{Stop}}$ for $E = (\{a\}, \frac{1}{2})$

1. *β is a bijection such that $\beta(s_{sr}) = s'_{sr}$ and $\beta([\underline{E}]_{\approx}) = [\underline{E'}]_{\approx}$;*

2. *$\forall s, \tilde{s} \in S_{sr}\ \forall \Gamma\ s \xrightarrow{\Gamma}_{\mathcal{P}} \tilde{s} \ \Leftrightarrow\ \beta(s) \xrightarrow{\Gamma}_{\mathcal{P}} \beta(\tilde{s})$.*

Two sr-transition systems $TS_{sr}(\overline{E})$ and $TS_{sr}(\overline{E'})$ are isomorphic*, denoted by $TS_{sr}(\overline{E}) \simeq TS_{sr}(\overline{E'})$, if $\exists \beta : TS_{sr}(\overline{E}) \simeq TS_{sr}(\overline{E'})$.*

sr-transition systems of static expressions can be defined as well. For $E \in RegStatExpr$, let $TS_{sr}(E) = TS_{sr}(\overline{E})$.

Example 9.2 *Let $E = (\{a\}, \frac{1}{2})$. In Figure 22 the transition systems $TS_{sr}(\overline{E})$ and $TS_{sr}(\overline{E;\mathsf{Stop}})$ are presented. In the latter sr-transition system (unlike the former one) the final state can be reached by executing the transition* $(\mathsf{skip}, 0)$ *only from the initial state.*

Definition 9.4 *Two dynamic expressions $\overline{E}$ and $\overline{E'}$ are* equivalent with respect to sr-transition systems*, denoted by $\overline{E} =_{tssr} \overline{E'}$, if $TS_{sr}(\overline{E}) \simeq TS_{sr}(\overline{E'})$.*

Note that sr-transition systems without empty loops can be defined, as well as the equivalence $=_{tssr*}$ based on them. At the same time, the coincidence of $=_{tssr}$ and $=_{tssr*}$ can be proved similar to that of $=_{ts}$ and $=_{ts*}$.

Theorem 9.1 *Let $\leftrightarrow, \leftrightarrow\!\!\!\!\leftrightarrow \in \{\equiv, \underline{\leftrightarrow}, =, \approx\}$ and $\star, \star\star \in \{\_, is, ss, sto, ts, tssr\}$. For dynamic expressions G and G'*

$$G \leftrightarrow_{\star} G' \ \Rightarrow\ G \leftrightarrow\!\!\!\!\leftrightarrow_{\star\star} G'$$

iff there exists a directed path from $\leftrightarrow_{\star}$ to $\leftrightarrow\!\!\!\!\leftrightarrow_{\star\star}$ in the graph in Figure 23.

Proof. ($\Leftarrow$) Let us check the validity of implications in the graph in Figure 23.

- The implication $=_{tssr} \rightarrow =_{ts}$ is valid, since sr-transition systems have more states and transitions than usual ones.

- The implication $\approx \rightarrow =_{tssr}$ is valid, since the sr-transition system of a dynamic formula is defined based on its structural equivalence class.

($\Rightarrow$) The absence of additional nontrivial arrows (not resulting from the combination of the existing ones) in the graph in Figure 23 is proved by the following examples.

- Let $E = (\{a\}, \frac{1}{2})$ and $E' = (\{a\}, \frac{1}{2}); \mathsf{Stop}$. We have $\overline{E} =_{ts} \overline{E'}$ as demonstrated in the example from the proof of Proposition 9.1. On the other hand, $\overline{E} \neq_{tssr} \overline{E'}$, since only in $TS_{sr}(\overline{E'})$ after the transition with multiaction part of label $\{a\}$ we do not reach the final state, see Example 9.2.

- Let $E = (\{a\}, \frac{1}{2})$ and $E' = ((\{a\}, \frac{1}{2}); (\{\hat{a}\}, \frac{1}{2}))\ \mathsf{sy}\ a$. Then $\overline{E} =_{tssr} \overline{E'}$, since $\overline{E} =_{ts} \overline{E'}$ as demonstrated in the last example from the proof of Theorem 5.2, and the final states of both $TS_{sr}(\overline{E'})$ and $TS_{sr}(\overline{E'})$ are reachable from the others with "normal" transitions (i.e., not with skip only). On the other hand, $\overline{E} \not\approx \overline{E'}$. $\square$

$$
\begin{tikzcd}
 & \approx \arrow[d] \\
 & =_{tssr} \arrow[d] \\
 & =_{ts} \arrow[d] \\
 & =_{sto} \arrow[d] \\
\underline{\leftrightarrow}_{is} \arrow[d] & \underline{\leftrightarrow}_{ss} \arrow[l] \arrow[d] \\
\equiv_{is} & \equiv_{ss} \arrow[l]
\end{tikzcd}
$$

Figure 23: Interrelations of the stochastic equivalences and the new congruence

The following theorem demonstrates that $=_{tssr}$ is a congruence of static expressions with respect to the operations of $dtsPBC$.

Theorem 9.2 *Let $a \in Act$ and $E, E', F, K \in RegStatExpr$. If $\overline{E} =_{tssr} \overline{E'}$ then*

1. $\overline{E \circ F} =_{tssr} \overline{E' \circ F},\ \overline{F \circ E} =_{tssr} \overline{F \circ E'},\ \circ \in \{;, [], \|\}$;
2. $\overline{E[f]} =_{tssr} \overline{E'[f]}$;
3. $\overline{E \circ a} =_{tssr} \overline{E' \circ a},\ \circ \in \{\mathsf{rs}, \mathsf{sy}\}$;
4. $\overline{[E * F * K]} =_{tssr} \overline{[E' * F * K]},\ \overline{[F * E * K]} =_{tssr} \overline{[F * E' * K]},\ \overline{[F * K * E]} =_{tssr} \overline{[F * K * E']}$.

Proof. First, we have no problems with termination, hence, the composite sr-transition systems built from the isomorphic ones can always execute the same multisets of activities. Second, the probabilities of the corresponding transitions of the composite systems coincide, since the probabilities are calculated from identical values. □

10 Performance evaluation

The standard analysis technique for DTMCs consists in the investigation of their transient and stationary behaviour and the subsequent calculation of some performance indices based on the steady-state probabilities. In this section with a case studies of a number of systems we demonstrate how steady-state distribution can be used for performance evaluation. The examples also illustrate the method of performance analysis simplification described above. The behaviour of all the systems which we consider here includes non-empty transitions only.

10.1 Shared memory system

10.1.1 The standard system

Consider a model of two processors accessing a common shared memory described in [53, 2, 3] in the continuous time setting on GSPNs. We shall analyze this shared memory system in the discrete time stochastic setting of

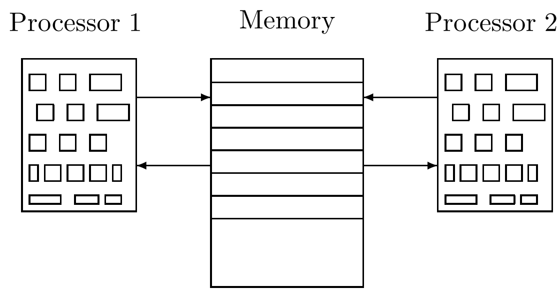

Figure 24: The diagram of the shared memory system

$dtsPBC$ where concurrent execution of activities is possible. The model performs as follows. After activation of the system (turning the computer on), two processors are active, and the common memory is available. Each processor can request an access to the memory. When a processor starts an acquisition of the memory, another processor should wait until the former one ends its memory operations, and the system returns to the state with both active processors and the available common memory. The diagram of the system is depicted in Figure 24.

Let us explain the meaning of actions from syntax of the $dtsPBC$ expressions which will specify the system modules. The action a corresponds to the system activation. The actions r_i $(1 \le i \le 2)$ represent the common memory request of processor i. The actions b_i and e_i correspond to the beginning and the end, respectively, of the common memory access of processor i. The other actions are used for communication purpose only via synchronization, and we abstract from them later using restriction.

The static expression of the first processor is $E_1 = [(\{x_1\}, \frac{1}{2}) * ((\{r_1\}, \frac{1}{2}); (\{b_1, y_1\}, \frac{1}{2}); (\{e_1, z_1\}, \frac{1}{2})) * \mathsf{Stop}]$. The static expression of the second processor is $E_2 = [(\{x_2\}, \frac{1}{2}) * ((\{r_2\}, \frac{1}{2}); (\{b_2, y_2\}, \frac{1}{2}); (\{e_2, z_2\}, \frac{1}{2})) * \mathsf{Stop}]$. The static expression of the shared memory is $E_3 = [(\{a, \widehat{x_1}, \widehat{x_2}\}, \frac{1}{2}) * (((\{\widehat{y_1}\}, \frac{1}{2}); (\{\widehat{z_1}\}, \frac{1}{2}))[]((\{\widehat{y_2}\}, \frac{1}{2}); (\{\widehat{z_2}\}, \frac{1}{2}))) * \mathsf{Stop}]$. The static expression of the shared memory system with two processors is $E = (E_1 \| E_2 \| E_3)$ $\mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2$.

Let us illustrate an effect of synchronization. In the result of synchronization of activities $(\{b_i, y_i\}, \frac{1}{2})$ and $(\{\widehat{y_i}\}, \frac{1}{2})$ we obtain the new activity $(\{b_i\}, \frac{1}{4})$ $(1 \le i \le 2)$. The synchronization of $(\{e_i, z_i\}, \frac{1}{2})$ and $(\{\widehat{z_i}\}, \frac{1}{2})$ produces $(\{e_i\}, \frac{1}{4})$ $(1 \le i \le 2)$. The result of synchronization of $(\{a, \widehat{x_1}, \widehat{x_2}\}, \frac{1}{2})$ with $(\{x_1\}, \frac{1}{2})$ is $(\{a, \widehat{x_2}\}, \frac{1}{4})$, and that of synchronization of $(\{a, \widehat{x_1}, \widehat{x_2}\}, \frac{1}{2})$ with $(\{x_2\}, \frac{1}{2})$ is $(\{a, \widehat{x_1}\}, \frac{1}{4})$. After applying synchronization to $(\{a, \widehat{x_2}\}, \frac{1}{4})$ and $(\{x_2\}, \frac{1}{2})$, as well as to $(\{a, \widehat{x_1}\}, \frac{1}{4})$ and $(\{x_1\}, \frac{1}{2})$ we obtain the same activity $(\{a\}, \frac{1}{8})$.

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

$$s_1 = [([\overline{(\{x_1\}, \tfrac{1}{2})} * ((\{r_1\}, \tfrac{1}{2}); (\{b_1, y_1\}, \tfrac{1}{2}); (\{e_1, z_1\}, \tfrac{1}{2})) * \mathsf{Stop}] \| [\overline{(\{x_2\}, \tfrac{1}{2})} * ((\{r_2\}, \tfrac{1}{2}); (\{b_2, y_2\}, \tfrac{1}{2}); (\{e_2, z_2\}, \tfrac{1}{2})) * \mathsf{Stop}] \| [\overline{(\{a, \widehat{x_1}, \widehat{x_2}\}, \tfrac{1}{2})} * (((\{\widehat{y_1}\}, \tfrac{1}{2}); (\{\widehat{z_1}\}, \tfrac{1}{2}))[]((\{\widehat{y_2}\}, \tfrac{1}{2}); (\{\widehat{z_2}\}, \tfrac{1}{2}))) * \mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2]_{\approx},$$

$$s_2 = [([(\{x_1\}, \tfrac{1}{2}) * (\overline{(\{r_1\}, \tfrac{1}{2})}; (\{b_1, y_1\}, \tfrac{1}{2}); (\{e_1, z_1\}, \tfrac{1}{2})) * \mathsf{Stop}] \| [(\{x_2\}, \tfrac{1}{2}) * (\overline{(\{r_2\}, \tfrac{1}{2})}; (\{b_2, y_2\}, \tfrac{1}{2}); (\{e_2, z_2\}, \tfrac{1}{2})) * \mathsf{Stop}] \| [(\{a, \widehat{x_1}, \widehat{x_2}\}, \tfrac{1}{2}) * (((\{\widehat{y_1}\}, \tfrac{1}{2}); (\{\widehat{z_1}\}, \tfrac{1}{2}))[]((\{\widehat{y_2}\}, \tfrac{1}{2}); (\{\widehat{z_2}\}, \tfrac{1}{2}))) * \mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2]_{\approx},$$

$$s_3 = [([(\{x_1\}, \tfrac{1}{2}) * ((\{r_1\}, \tfrac{1}{2}); \overline{(\{b_1, y_1\}, \tfrac{1}{2})}; (\{e_1, z_1\}, \tfrac{1}{2})) * \mathsf{Stop}] \| [(\{x_2\}, \tfrac{1}{2}) * (\overline{(\{r_2\}, \tfrac{1}{2})}; (\{b_2, y_2\}, \tfrac{1}{2}); (\{e_2, z_2\}, \tfrac{1}{2})) * \mathsf{Stop}] \| [(\{a, \widehat{x_1}, \widehat{x_2}\}, \tfrac{1}{2}) * ((\overline{(\{\widehat{y_1}\}, \tfrac{1}{2})}; (\{\widehat{z_1}\}, \tfrac{1}{2}))[]((\{\widehat{y_2}\}, \tfrac{1}{2}); (\{\widehat{z_2}\}, \tfrac{1}{2}))) * \mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2]_{\approx},$$

$$s_4 = [([(\{x_1\}, \tfrac{1}{2}) * (\overline{(\{r_1\}, \tfrac{1}{2})}; (\{b_1, y_1\}, \tfrac{1}{2}); (\{e_1, z_1\}, \tfrac{1}{2})) * \mathsf{Stop}] \| [(\{x_2\}, \tfrac{1}{2}) * ((\{r_2\}, \tfrac{1}{2}); \overline{(\{b_2, y_2\}, \tfrac{1}{2})}; (\{e_2, z_2\}, \tfrac{1}{2})) * \mathsf{Stop}] \| [(\{a, \widehat{x_1}, \widehat{x_2}\}, \tfrac{1}{2}) * (((\{\widehat{y_1}\}, \tfrac{1}{2}); (\{\widehat{z_1}\}, \tfrac{1}{2}))[](\overline{(\{\widehat{y_2}\}, \tfrac{1}{2})}; (\{\widehat{z_2}\}, \tfrac{1}{2}))) * \mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2]_{\approx},$$

$$s_5 = [([(\{x_1\}, \tfrac{1}{2}) * ((\{r_1\}, \tfrac{1}{2}); (\{b_1, y_1\}, \tfrac{1}{2}); \overline{(\{e_1, z_1\}, \tfrac{1}{2})}) * \mathsf{Stop}] \| [(\{x_2\}, \tfrac{1}{2}) * (\overline{(\{r_2\}, \tfrac{1}{2})}; (\{b_2, y_2\}, \tfrac{1}{2}); (\{e_2, z_2\}, \tfrac{1}{2})) * \mathsf{Stop}] \| [(\{a, \widehat{x_1}, \widehat{x_2}\}, \tfrac{1}{2}) * (((\{\widehat{y_1}\}, \tfrac{1}{2}); \overline{(\{\widehat{z_1}\}, \tfrac{1}{2})})[]((\{\widehat{y_2}\}, \tfrac{1}{2}); (\{\widehat{z_2}\}, \tfrac{1}{2}))) * \mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2]_{\approx},$$

$$s_6 = [([(\{x_1\}, \tfrac{1}{2}) * ((\{r_1\}, \tfrac{1}{2}); \overline{(\{b_1, y_1\}, \tfrac{1}{2})}; (\{e_1, z_1\}, \tfrac{1}{2})) * \mathsf{Stop}] \| [(\{x_2\}, \tfrac{1}{2}) * ((\{r_2\}, \tfrac{1}{2}); \overline{(\{b_2, y_2\}, \tfrac{1}{2})}; (\{e_2, z_2\}, \tfrac{1}{2})) * \mathsf{Stop}] \| [(\{a, \widehat{x_1}, \widehat{x_2}\}, \tfrac{1}{2}) * ((\overline{(\{\widehat{y_1}\}, \tfrac{1}{2})}; (\{\widehat{z_1}\}, \tfrac{1}{2}))[](\overline{(\{\widehat{y_2}\}, \tfrac{1}{2})}; (\{\widehat{z_2}\}, \tfrac{1}{2}))) * \mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2]_{\approx},$$

$$s_7 = [([(\{x_1\}, \tfrac{1}{2}) * (\overline{(\{r_1\}, \tfrac{1}{2})}; (\{b_1, y_1\}, \tfrac{1}{2}); (\{e_1, z_1\}, \tfrac{1}{2})) * \mathsf{Stop}] \| [(\{x_2\}, \tfrac{1}{2}) * ((\{r_2\}, \tfrac{1}{2}); (\{b_2, y_2\}, \tfrac{1}{2}); \overline{(\{e_2, z_2\}, \tfrac{1}{2})}) * \mathsf{Stop}] \| [(\{a, \widehat{x_1}, \widehat{x_2}\}, \tfrac{1}{2}) * (((\{\widehat{y_1}\}, \tfrac{1}{2}); (\{\widehat{z_1}\}, \tfrac{1}{2}))[]((\{\widehat{y_2}\}, \tfrac{1}{2}); \overline{(\{\widehat{z_2}\}, \tfrac{1}{2})})) * \mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{rs}\ x_1$$

$TS^*(\overline{E})$

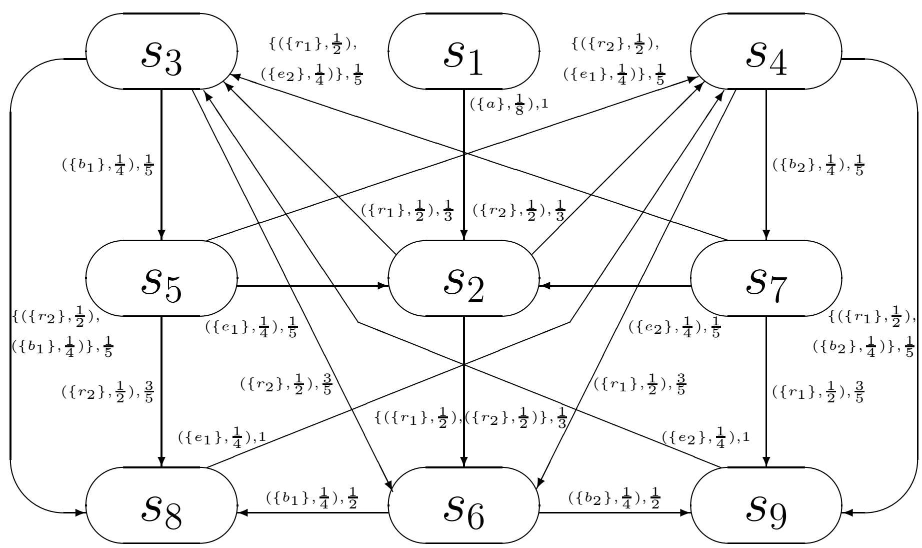

Figure 25: The transition system without empty loops of the shared memory system

rs x_2 rs y_1 rs y_2 rs z_1 rs $z_2]_\approx$,

$s_8 = [([(\{x_1\},\frac{1}{2})*((\{r_1\},\frac{1}{2});(\{b_1,y_1\},\frac{1}{2});\overline{(\{e_1,z_1\},\frac{1}{2}))}*\mathsf{Stop}]\|[(\{x_2\},\frac{1}{2})*((\{r_2\},\frac{1}{2});(\{b_2,y_2\},\frac{1}{2});(\{e_2,z_2\},\frac{1}{2}))*\mathsf{Stop}]\|[(\{a,\widehat{x_1},\widehat{x_2}\},\frac{1}{2})*((((\{\widehat{y_1}\},\frac{1}{2});\overline{(\{\widehat{z_1}\},\frac{1}{2}))}[]((\{\widehat{y_2}\},\frac{1}{2});(\{\widehat{z_2}\},\frac{1}{2})))*\mathsf{Stop}])$ sy x_1 sy x_2 sy y_1 sy y_2 sy z_1 sy z_2 rs x_1 rs x_2 rs y_1 rs y_2 rs z_1 rs $z_2]_\approx$,

$s_9 = [([(\{x_1\},\frac{1}{2})*((\{r_1\},\frac{1}{2});(\{b_1,y_1\},\frac{1}{2});(\{e_1,z_1\},\frac{1}{2}))*\mathsf{Stop}]\|[(\{x_2\},\frac{1}{2})*((\{r_2\},\frac{1}{2});(\{b_2,y_2\},\frac{1}{2});\overline{(\{e_2,z_2\},\frac{1}{2}))}*\mathsf{Stop}]\|[(\{a,\widehat{x_1},\widehat{x_2}\},\frac{1}{2})*((((\{\widehat{y_1}\},\frac{1}{2});(\{\widehat{z_1}\},\frac{1}{2}))[]((\{\widehat{y_2}\},\frac{1}{2});\overline{(\{\widehat{z_2}\},\frac{1}{2})))}*\mathsf{Stop}])$ sy x_1 sy x_2 sy y_1 sy y_2 sy z_1 sy z_2 rs x_1 rs x_2 rs y_1 rs y_2 rs z_1 rs $z_2]_\approx$.

The states are interpreted as follows: s_1 is the initial state, s_2: the system is activated and the memory is not requested, s_3: the memory is requested by the first processor, s_4: the memory is requested by the second processor, s_5: the memory is allocated to the first processor, s_6: the memory is requested by two processors, s_7: the memory is allocated to the second processor, s_8: the memory is allocated to the first processor and the memory is requested by the second processor, s_9: the memory is allocated to the second processor and the memory is requested by the first processor.

In Figure 25 the transition system without empty loops $TS^*(\overline{E})$ is presented. In Figure 26 the underlying DTMC without empty loops $DTMC^*(\overline{E})$ is depicted.

The TPM for $DTMC^*(\overline{E})$ is

$$\mathbf{P}^* = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{5} & \frac{3}{5} & 0 & \frac{1}{5} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{3}{5} & \frac{1}{5} & 0 & \frac{1}{5} \\ 0 & \frac{1}{5} & 0 & \frac{1}{5} & 0 & 0 & 0 & \frac{3}{5} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & \frac{1}{5} & \frac{1}{5} & 0 & 0 & 0 & 0 & 0 & \frac{3}{5} \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

In Table 4 the transient and the steady-state probabilities $\psi_i^*[k]$ $(i \in \{1,2,3,5,6,8\})$ of the shared memory system at the time moments k $(0 \le k \le 10)$ and $k = \infty$ are presented, and in Figure 27 the alteration diagram (evolution in time) for the transient probabilities is depicted. It is sufficient to consider the probabilities for the states $s_1, s_2, s_3, s_5, s_6, s_8$ only, since the corresponding values coincide for s_3, s_4 as well as for s_5, s_7 as well as for s_8, s_9.

The steady-state PMF for $DTMC^*(\overline{E})$ is

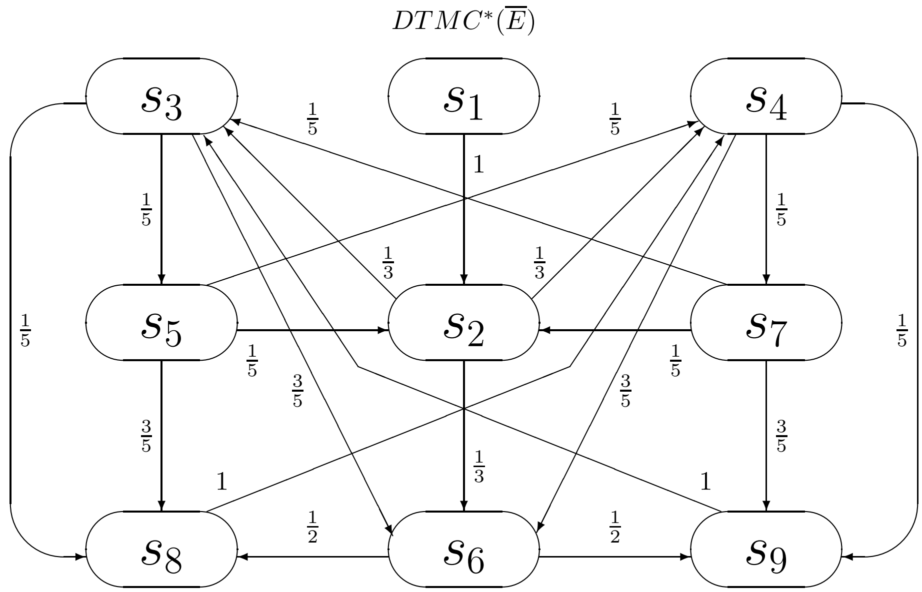

Figure 26: The underlying DTMC without empty loops of the shared memory system

Table 4: Transient and steady-state probabilities of the shared memory system

k	0	1	2	3	4	5	6	7	8	9	10	∞
$\psi_1^*[k]$	1	0	0	0	0	0	0	0	0	0	0	0
$\psi_2^*[k]$	0	1	0	0	0.0267	0	0.0197	0.0199	0.0047	0.0199	0.0160	0.0144
$\psi_3^*[k]$	0	0	0.3333	0	0.2467	0.2489	0.0592	0.2484	0.2000	0.1071	0.2368	0.1794
$\psi_5^*[k]$	0	0	0	0.0667	0	0.0493	0.0498	0.0118	0.0497	0.0400	0.0214	0.0359
$\psi_6^*[k]$	0	0	0.3333	0.4000	0	0.3049	0.2987	0.0776	0.3047	0.2416	0.1351	0.2201
$\psi_8^*[k]$	0	0	0	0.2333	0.2400	0.0493	0.2318	0.1910	0.0956	0.2221	0.1662	0.1675

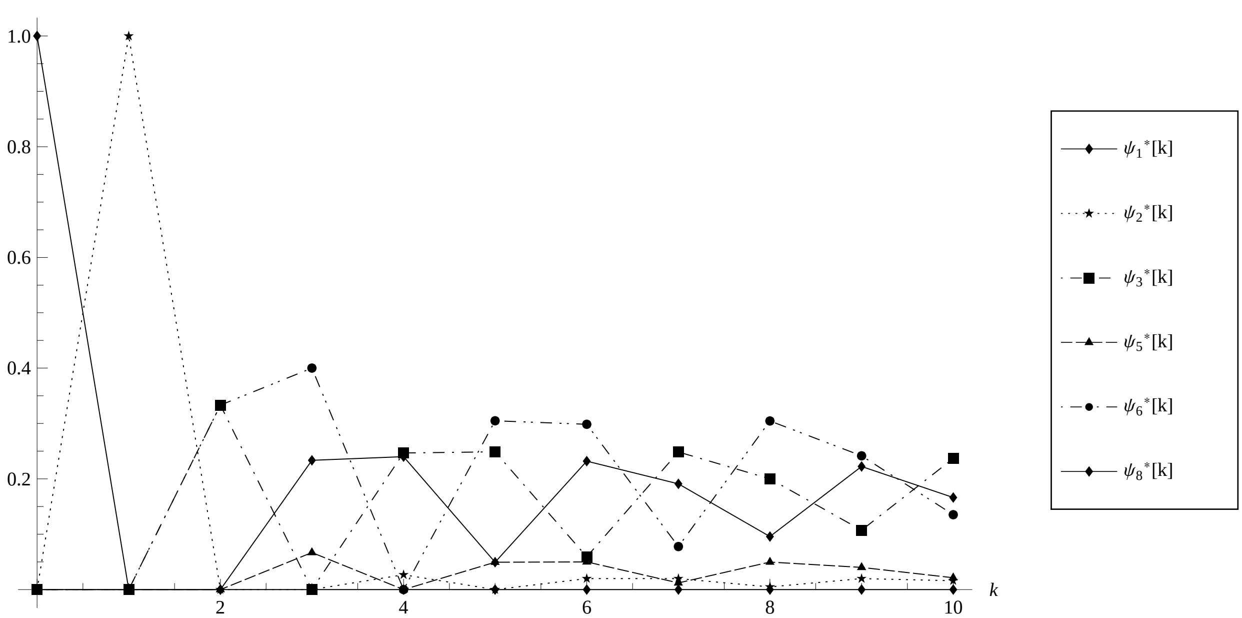

Figure 27: Transient probabilities alteration diagram of the shared memory system

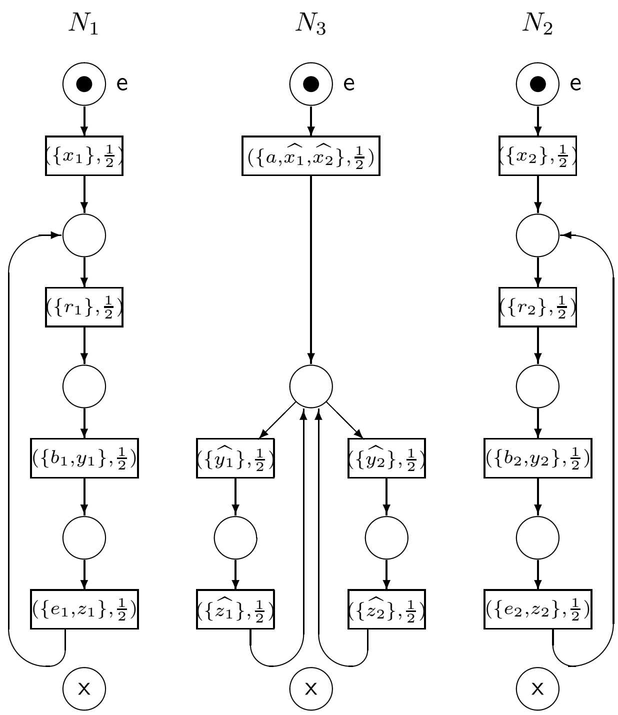

Figure 28: The marked dts-boxes of two processors and shared memory

$$\psi^* = \left(0, \frac{3}{209}, \frac{75}{418}, \frac{75}{418}, \frac{15}{418}, \frac{46}{209}, \frac{15}{418}, \frac{35}{209}, \frac{35}{209}\right).$$

We can now calculate the main performance indices.

- The average recurrence time in the state s_2, where no processor requests the memory, called the *average system run-through*, is $\frac{1}{\psi_2^*} = \frac{209}{3} = 69\frac{2}{3}$.
- The common memory is available only in the states s_2, s_3, s_4, s_6. The steady-state probability that the memory is available is $\psi_2^* + \psi_3^* + \psi_4^* + \psi_6^* = \frac{3}{209} + \frac{75}{418} + \frac{75}{418} + \frac{46}{209} = \frac{124}{209}$. Then the steady-state probability that the memory is used (i.e., not available), called the *shared memory utilization*, is $1 - \frac{124}{209} = \frac{85}{209}$.
- The common memory request of the first processor $(\{r_1\}, \frac{1}{2})$ is only possible from the states s_2, s_4, s_7. In each of the states the request probability is the sum of the execution probabilities for all multisets of activities containing $(\{r_1\}, \frac{1}{2})$. Thus, the *steady-state probability of the shared memory request from the first processor* is $\psi_2^* \sum_{\{\Gamma|(\{r_1\},\frac{1}{2})\in\Gamma\}} PT^*(\Gamma, s_2) + \psi_4^* \sum_{\{\Gamma|(\{r_1\},\frac{1}{2})\in\Gamma\}} PT^*(\Gamma, s_4) +$ $\psi_7^* \sum_{\{\Gamma|(\{r_1\},\frac{1}{2})\in\Gamma\}} PT^*(\Gamma, s_7) = \frac{3}{209}\left(\frac{1}{3} + \frac{1}{3}\right) + \frac{75}{418}\left(\frac{3}{5} + \frac{1}{5}\right) + \frac{15}{418}\left(\frac{3}{5} + \frac{1}{5}\right) = \frac{38}{209}$.

In Figure 28 the marked dts-boxes corresponding to the dynamic expressions of two processors and shared memory are presented, i.e., $N_i = Box_{dts}(\overline{E_i})$ $(1 \leq i \leq 3)$. In Figure 29 the marked dts-box corresponding to the dynamic expression of the shared memory system is depicted, i.e., $N = Box_{dts}(\overline{E})$.

10.1.2 The abstract system and its reduction

Let us consider a modification of the shared memory system with abstraction from identifiers of the processors, i.e., such that the processors are indistinguishable. For example, we can just see that a processor requires memory or the memory is allocated to it but cannot observe which processor is it. We call this system the abstract shared memory one.

The static expression of the first processor is $F_1 = [(\{x_1\}, \frac{1}{2}) * ((\{r\}, \frac{1}{2}); (\{b, y_1\}, \frac{1}{2}); (\{e, z_1\}, \frac{1}{2})) * \mathsf{Stop}]$. The static expression of the second processor is $F_2 = [(\{x_2\}, \frac{1}{2}) * ((\{r\}, \frac{1}{2}); (\{b, y_2\}, \frac{1}{2}); (\{e, z_2\}, \frac{1}{2})) * \mathsf{Stop}]$. The static expression of the shared memory is $F_3 = [(\{a, \widehat{x_1}, \widehat{x_2}\}, \frac{1}{2}) * (((\{\widehat{y_1}\}, \frac{1}{2}); (\{\widehat{z_1}\}, \frac{1}{2}))[]((\{\widehat{y_2}\}, \frac{1}{2}); (\{\widehat{z_2}\}, \frac{1}{2}))) * \mathsf{Stop}]$. The static expression of the abstract shared memory system with two processors is $F = (F_1 \| F_2 \| F_3)$ $\mathsf{sy}\ x_1$ $\mathsf{sy}\ x_2$ $\mathsf{sy}\ y_1$ $\mathsf{sy}\ y_2$ $\mathsf{sy}\ z_1$ $\mathsf{sy}\ z_2$ $\mathsf{rs}\ x_1$ $\mathsf{rs}\ x_2$ $\mathsf{rs}\ y_1$ $\mathsf{rs}\ y_2$ $\mathsf{rs}\ z_1$ $\mathsf{rs}\ z_2$.

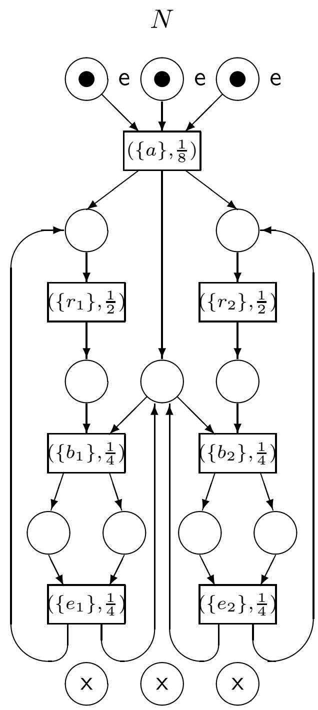

Figure 29: The marked dts-box of the shared memory system

$DR(\overline{F})$ resembles $DR(\overline{E})$, and $TS^*(\overline{F})$ is similar to $TS^*(\overline{E})$. We have $DTMC^*(\overline{F}) = DTMC^*(\overline{E})$. Thus, the TPM and the steady-state PMF for $DTMC^*(\overline{F})$ and $DTMC^*(\overline{E})$ coincide.

The first and second performance indices are the same for the standard and the abstract systems. Let us consider the following performance index based on non-identified viewpoint to the processors.

- The common memory request of a processor $(\{r\},\frac{1}{2})$ is only possible from the states s_2, s_3, s_4, s_5, s_7. In each of the states the request probability is the sum of the execution probabilities for all multisets of activities containing $(\{r_1\},\frac{1}{2})$. Thus, the *steady-state probability of the shared memory request from the first processor* is $\psi_2^* \sum_{\{\Gamma|(\{r\},\frac{1}{2})\in\Gamma\}} PT^*(\Gamma, s_2) + \psi_3^* \sum_{\{\Gamma|(\{r\},\frac{1}{2})\in\Gamma\}} PT^*(\Gamma, s_3) + \psi_4^* \sum_{\{\Gamma|(\{r\},\frac{1}{2})\in\Gamma\}} PT^*(\Gamma, s_4) + \psi_5^* \sum_{\{\Gamma|(\{r\},\frac{1}{2})\in\Gamma\}} PT^*(\Gamma, s_5) + \psi_7^* \sum_{\{\Gamma|(\{r\},\frac{1}{2})\in\Gamma\}} PT^*(\Gamma, s_7) = \frac{3}{209}\left(\frac{1}{3}+\frac{1}{3}+\frac{1}{3}\right) + \frac{75}{418}\left(\frac{3}{5}+\frac{1}{5}\right) + \frac{75}{418}\left(\frac{3}{5}+\frac{1}{5}\right) + \frac{15}{418}\left(\frac{3}{5}+\frac{1}{5}\right) + \frac{15}{418}\left(\frac{3}{5}+\frac{1}{5}\right) = \frac{75}{209}$.

The marked dts-boxes corresponding to the dynamic expressions of the standard and the abstract two processors and shared memory are similar as well as the marked dts-boxes corresponding to the dynamic expression of the standard and the abstract shared memory systems.

Let us consider a reduction of the abstract shared memory system. Note that $TS^*(\overline{F})$ can be reduced by merging the equivalent states s_3, s_4 as well as s_5, s_7 as well as s_8, s_9, thus, it can be transformed into a transition system with six states only. But the resulted reduction of the initial transition system $TS^*(\overline{F})$ will not correspond to some $dtsPBC$ expression anymore.

For the step stochastic autobisimulation equivalence $\overline{F}\underline{\leftrightarrow}_{ss}\overline{F}$ we have $DR(\overline{F})/_{\underline{\leftrightarrow}_{ss}} = \{\mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3, \mathcal{K}_4, \mathcal{K}_5, \mathcal{K}_6\}$, where $\mathcal{K}_1 = \{s_1\}$ (the initial state), $\mathcal{K}_2 = \{s_2\}$ (the system is activated and the memory is not requested), $\mathcal{K}_3 = \{s_3, s_4\}$ (the memory is requested by one processor), $\mathcal{K}_4 = \{s_5, s_7\}$ (the memory is allocated to a processor), $\mathcal{K}_5 = \{s_6\}$ (the memory is requested by two processors), $\mathcal{K}_6 = \{s_8, s_9\}$ (the memory is allocated to a processor and the memory is requested by another processor).

In Figure 30 the quotient transition system without empty loops $TS^*_{\underline{\leftrightarrow}_{ss}}(\overline{F})$ is presented. In Figure 31 the quotient underlying DTMC without empty loops $DTMC^*_{\underline{\leftrightarrow}_{ss}}(\overline{F})$ is depicted.

The TPM for $DTMC^*_{\underline{\leftrightarrow}_{ss}}(\overline{F})$ is

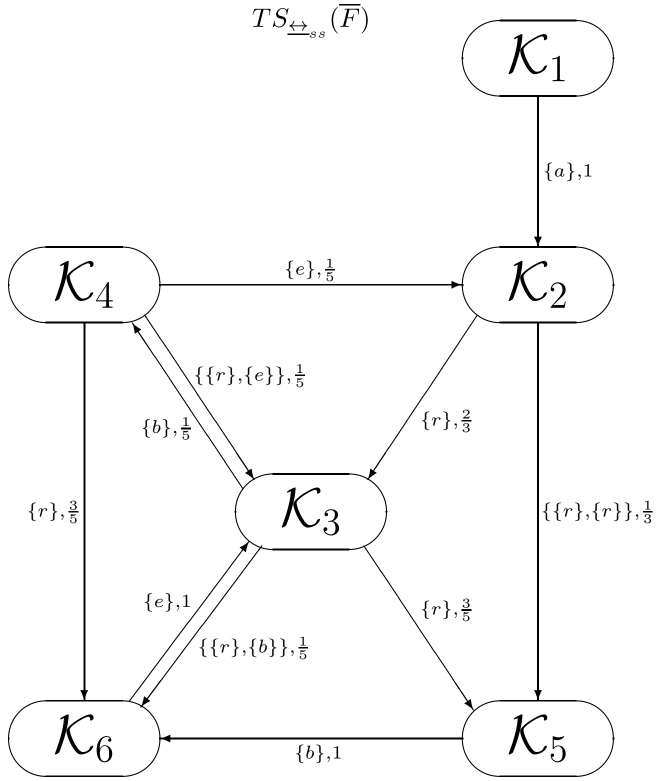

Figure 30: The quotient transition system without empty loops of the abstract shared memory system

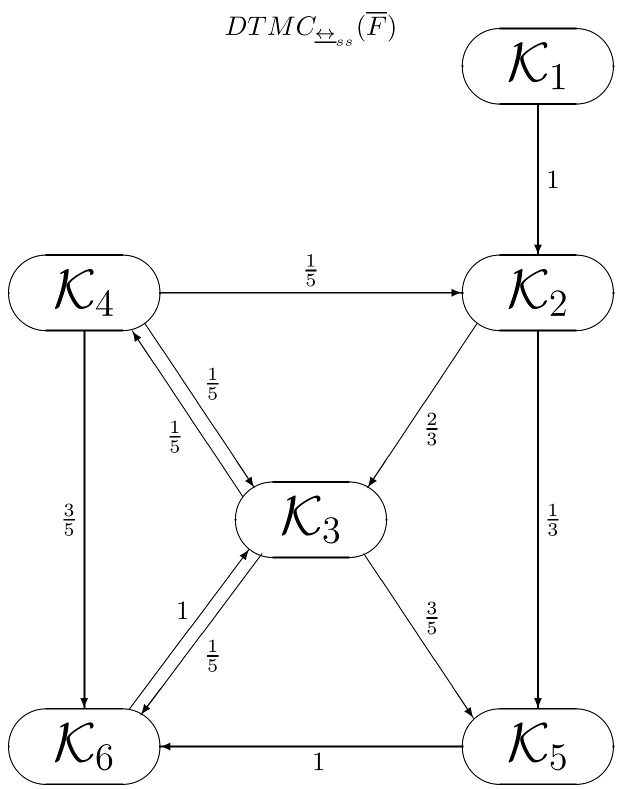

Figure 31: The quotient underlying DTMC without empty loops of the abstract shared memory system

Table 5: Transient and steady-state probabilities of the quotient abstract shared memory system

k	0	1	2	3	4	5	6	7	8	9	10	∞
${\psi'_1}^*[k]$	1	0	0	0	0	0	0	0	0	0	0	0
${\psi'_2}^*[k]$	0	1	0	0	0.0267	0	0.0197	0.0199	0.0047	0.0199	0.0160	0.0144
${\psi'_3}^*[k]$	0	0	0.6667	0	0.4933	0.4978	0.1184	0.4967	0.4001	0.2142	0.4735	0.3589
${\psi'_4}^*[k]$	0	0	0	0.1333	0	0.0987	0.0996	0.0237	0.0993	0.0800	0.0428	0.0718
${\psi'_5}^*[k]$	0	0	0.3333	0.4000	0	0.3049	0.2987	0.0776	0.3047	0.2416	0.1351	0.2201
${\psi'_6}^*[k]$	0	0	0	0.4667	0.4800	0.0987	0.4636	0.3821	0.1912	0.4443	0.3325	0.3349

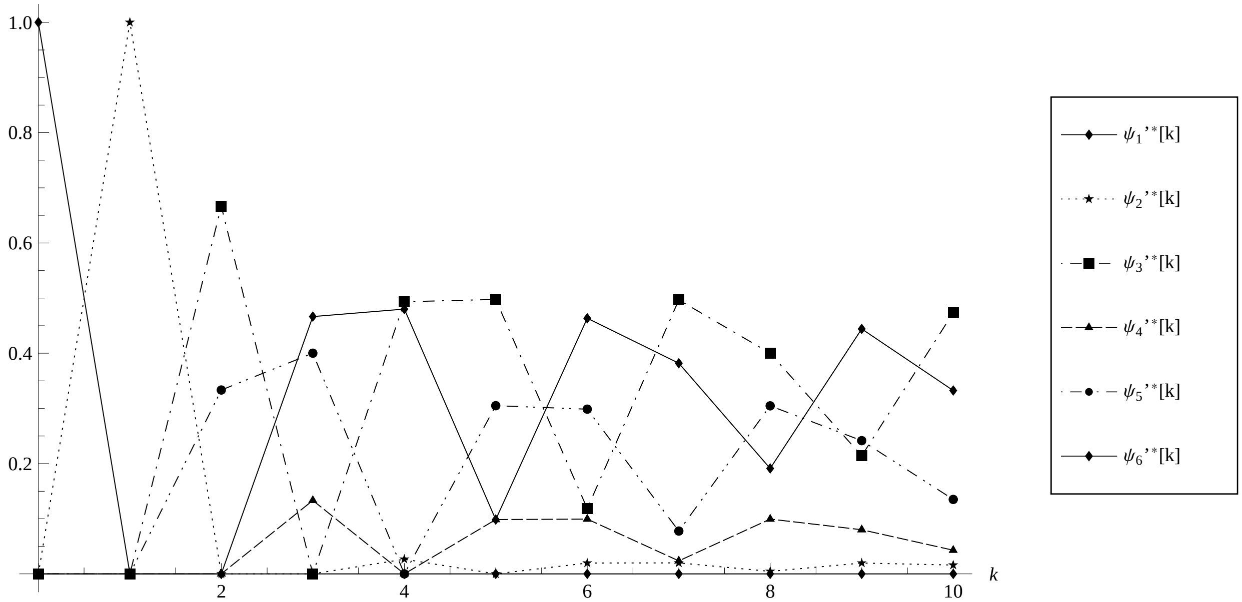

Figure 32: Transient probabilities alteration diagram of the quotient abstract shared memory system

$$\mathbf{P}'^* = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{2}{3} & 0 & \frac{1}{3} & 0 \\ 0 & 0 & 0 & \frac{1}{5} & \frac{3}{5} & \frac{1}{5} \\ 0 & \frac{1}{5} & \frac{1}{5} & 0 & 0 & \frac{3}{5} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

In Table 5 the transient and the steady-state probabilities ${\psi'_i}^*[k]$ $(1 \leq i \leq 6)$ of the quotient abstract shared memory system at the time moments k $(0 \leq k \leq 10)$ and $k = \infty$ are presented, and in Figure 32 the alteration diagram (evolution in time) for the transient probabilities is depicted.

The steady-state PMF for $DTMC^*_{\underline{\leftrightarrow}_{ss}}(\overline{F})$ is

$$\psi'^* = \left(0, \frac{3}{209}, \frac{75}{209}, \frac{15}{418}, \frac{46}{209}, \frac{70}{209}\right).$$

We can now calculate the main performance indices.

- The average recurrence time in the state $\mathcal{K}_2$, where no processor requests the memory, called the *average system run-through*, is $\frac{1}{{\psi'_2}^*} = \frac{209}{3} = 69\frac{2}{3}$.

- The common memory is available only in the states $\mathcal{K}_2, \mathcal{K}_3, \mathcal{K}_5$. The steady-state probability that the memory is available is ${\psi'_2}^* + {\psi'_3}^* + {\psi'_5}^* = \frac{3}{209} + \frac{75}{209} + \frac{46}{209} = \frac{124}{209}$. Then the steady-state probability that the memory is used (i.e., not available), called the *shared memory utilization*, is $1 - \frac{124}{209} = \frac{85}{209}$.

- The common memory request of a processor $\{r\}$ is only possible from the states $\mathcal{K}_2, \mathcal{K}_3, \mathcal{K}_4$. In each of the states the request probability is the sum of the execution probabilities for all multisets of multiactions containing $\{r\}$. Thus, the *steady-state probability of the shared memory request from a processor* is $\psi_2'^* \sum_{\{A,\widetilde{\mathcal{K}} \mid \{r\} \in A,\ \mathcal{K}_2 \stackrel{A}{\twoheadrightarrow} \widetilde{\mathcal{K}}\}} PM_A^*(\mathcal{K}_2, \widetilde{\mathcal{K}}) + \psi_3'^* \sum_{\{A,\widetilde{\mathcal{K}} \mid \{r\} \in A,\ \mathcal{K}_3 \stackrel{A}{\twoheadrightarrow} \widetilde{\mathcal{K}}\}} PM_A^*(\mathcal{K}, \widetilde{\mathcal{K}}) +$
$\psi_4'^* \sum_{\{A,\widetilde{\mathcal{K}} \mid \{r\} \in A,\ \mathcal{K}_4 \stackrel{A}{\twoheadrightarrow} \widetilde{\mathcal{K}}\}} PM_A^*(\mathcal{K}, \widetilde{\mathcal{K}}) = \frac{3}{209}\left(\frac{2}{3} + \frac{1}{3}\right) + \frac{75}{209}\left(\frac{3}{5} + \frac{1}{5}\right) + \frac{15}{209}\left(\frac{3}{5} + \frac{1}{5}\right) = \frac{75}{209}$.

One can see that the performance indices are the same for the complete and the quotient shared memory systems. The coincidence of the first and second performance indices obviously illustrates the result of Proposition 8.1. The coincidence of the third performance index is due to the Theorem 8.2: one should just apply its result to the step traces $\{\{r\}\}$, $\{\{r\},\{r\}\}$, $\{\{r\},\{b\}\}$, $\{\{r\},\{e\}\}$ of the expression $\overline{F}$ and itself, and then sum the left and right parts of the three resulting equalities.

10.1.3 The generalized system

An interesting problem is to find out which influence to performance have the multiaction probabilities from the specification E of the shared memory system. Suppose that all the mentioned multiactions have the same generalized probability ρ. The resulting specification K of the generalized shared memory system is defined as follows.

The static expression of the first processor is $K_1 = [(\{x_1\}, \rho) * ((\{r_1\}, \rho); (\{b_1, y_1\}, \rho); (\{e_1, z_1\}, \rho)) * \mathsf{Stop}]$. The static expression of the second processor is $K_2 = [(\{x_2\}, \rho) * ((\{r_2\}, \rho); (\{b_2, y_2\}, \rho); (\{e_2, z_2\}, \rho)) * \mathsf{Stop}]$. The static expression of the shared memory is $K_3 = [(\{a, \widehat{x_1}, \widehat{x_2}\}, \rho) * (((\{\widehat{y_1}\}, \rho); (\{\widehat{z_1}\}, \rho))[]((\{\widehat{y_2}\}, \rho); (\{\widehat{z_2}\}, \rho))) * \mathsf{Stop}]$. The static expression of the generalized shared memory system with two processors is $K = (K_1 \| K_2 \| K_3)$ $\mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2$.

The TPM for $DTMC^*(\overline{K})$ is

$$\widetilde{\mathbf{P}}^* = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1-\rho}{2-\rho} & \frac{1-\rho}{2-\rho} & 0 & \frac{\rho}{2-\rho} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{\rho(1-\rho)}{1+\rho-\rho^2} & \frac{1-\rho^2}{1+\rho-\rho^2} & 0 & \frac{\rho^2}{1+\rho-\rho^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-\rho^2}{1+\rho-\rho^2} & \frac{\rho(1-\rho)}{1+\rho-\rho^2} & 0 & \frac{\rho^2}{1+\rho-\rho^2} \\ 0 & \frac{\rho(1-\rho)}{1+\rho-\rho^2} & 0 & \frac{\rho^2}{1+\rho-\rho^2} & 0 & 0 & 0 & \frac{1-\rho^2}{1+\rho-\rho^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & \frac{\rho(1-\rho)}{1+\rho-\rho^2} & \frac{\rho^2}{1+\rho-\rho^2} & 0 & 0 & 0 & 0 & 0 & \frac{1-\rho^2}{1+\rho-\rho^2} \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The steady-state PMF for $DTMC^*(\overline{K})$ is

$$\begin{aligned} \tilde{\psi}^* = \Big(0, \ & \frac{\rho^2(-2+5\rho-4\rho^2+\rho^3)}{-6-9\rho+14\rho^2+10\rho^3-14\rho^4+3\rho^5}, \frac{1}{24}\left[4 - 5\rho^2 + 12\rho^3 - 5\rho^4 + \frac{(19\rho^2+36\rho^3-26\rho^4-46\rho^5+15\rho^6)(-2+5\rho-4\rho^2+\rho^3)}{-6-9\rho+14\rho^2+10\rho^3-14\rho^4+3\rho^5}\right], \\ & \frac{1}{24}\left[4 - 5\rho^2 + 12\rho^3 - 5\rho^4 + \frac{(19\rho^2+36\rho^3-26\rho^4-46\rho^5+15\rho^6)(-2+5\rho-4\rho^2+\rho^3)}{-6-9\rho+14\rho^2+10\rho^3-14\rho^4+3\rho^5}\right], \\ & \frac{1}{12}\left[2\rho + \rho^2 - 5\rho^3 + 2\rho^4 + \frac{(-15\rho^2-25\rho^3+13\rho^4+19\rho^5-6\rho^6)(-2+5\rho-4\rho^2+\rho^3)}{-6-9\rho+14\rho^2+10\rho^3-14\rho^4+3\rho^5}\right], \\ & \frac{1}{24}\left[8 - 8\rho + 37\rho^2 - 58\rho^3 + 21\rho^4 + \frac{(-79\rho^2-194\rho^3+46\rho^4+216\rho^5-63\rho^6)(-2+5\rho-4\rho^2+\rho^3)}{-6-9\rho+14\rho^2+10\rho^3-14\rho^4+3\rho^5}\right], \\ & \frac{1}{12}\left[2\rho + \rho^2 - 5\rho^3 + 2\rho^4 + \frac{(-15\rho^2-25\rho^3+13\rho^4+19\rho^5-6\rho^6)(-2+5\rho-4\rho^2+\rho^3)}{-6-9\rho+14\rho^2+10\rho^3-14\rho^4+3\rho^5}\right], \\ & \frac{1}{48}\left[8 - 31\rho^2 + 54\rho^3 - 19\rho^4 + \frac{(77\rho^2+222\rho^3-46\rho^4-200\rho^5+57\rho^6)(-2+5\rho-4\rho^2+\rho^3)}{-6-9\rho+14\rho^2+10\rho^3-14\rho^4+3\rho^5}\right], \\ & \frac{1}{48}\left[8 - 31\rho^2 + 54\rho^3 - 19\rho^4 + \frac{(77\rho^2+222\rho^3-46\rho^4-200\rho^5+57\rho^6)(-2+5\rho-4\rho^2+\rho^3)}{-6-9\rho+14\rho^2+10\rho^3-14\rho^4+3\rho^5}\right]\Big). \end{aligned}$$

One can now proceed with performance evaluation according to the same pattern as in the case $\rho = \frac{1}{2}$ considered earlier.

The abstract generalized shared memory system and its reduction are considered like the corresponding notions for the non-generalized system.

10.2 Dining philosophers system

10.2.1 The standard system

Consider a model of five dining philosophers, for which the Petri net interpretation was proposed in [75]. We shall investigate this dining philosophers system in the discrete time stochastic setting of $dtsPBC$. The philosophers

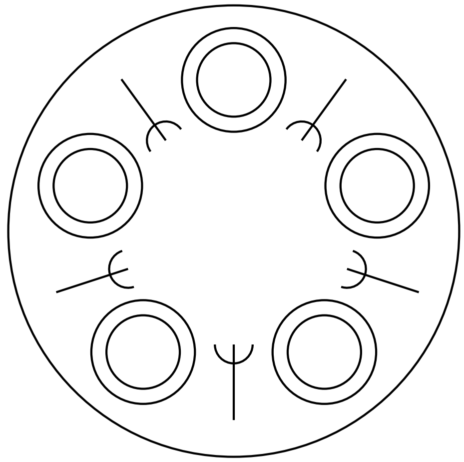

Figure 33: The diagram of the dining philosophers system

occupy a round table, and there is one fork between every neighboring persons, hence, there are five forks on the table. A philosopher needs two forks to eat, namely, his left and right ones. Hence, all five philosophers cannot eat together, since otherwise there will not be enough forks available, but only one of two of them who are not neighbors. The model performs as follows. After activation of the system (coming the philosophers in the dining room), five forks are placed on the table. If the left and right forks are available for a philosopher, he takes them simultaneously and begins eating. At the end of eating, the philosopher places both his forks simultaneously back on the table. The strategy to pick up and release two forks simultaneously prevents the situation when a philosopher takes one fork but is not able to pick up the second one since their neighbor has already done so. In particular, we avoid a deadlock when all the philosophers take their left (right) forks and wait until their right (left) forks will be available. The diagram of the system is depicted in Figure 33.

One can explore what happens if there will be another number of philosophers at the table. The most interesting is to find the maximal sets of philosophers which can dine together, since all other combinations of the dining persons will be the subsets of these maximal sets. For the system with 1 philosopher the only maximal set is $\{1\}$. For the system with 2 philosophers the maximal sets are $\{1\},\ \{2\}$. For the system with 3 philosophers the maximal sets are $\{1\},\ \{2\},\ \{3\}$. For the system with 4 philosophers the maximal sets are $\{1,3\},\ \{2,4\}$. For the system with 5 philosophers the maximal sets are $\{1,3\},\ \{1,4\},\ \{2,4\},\ \{2,5\},\ \{3,5\}$. For the system with 6 philosophers the maximal sets are $\{1,3,5\},\ \{2,4,6\}$. For the system with 7 philosophers the maximal sets are $\{1,3,5\},\ \{1,3,6\},\ \{1,4,6\},\ \{2,4,6\},\ \{2,4,7\},\ \{2,5,7\},\ \{3,5,7\}$. Thus, the system demonstrates a nontrivial behaviour when at least 5 philosophers occupy the table.

Since the neighbors cannot dine together, the maximal number of the dining persons for the system with n philosophers will be $\lfloor\frac{n}{2}\rfloor$, i.e., the maximal natural number that is not greater than $\frac{n}{2}$. Note that if the philosopher i belongs to some maximal set then the philosopher $(i+1)\ \mathsf{mod}\ n$ will belong to the next one. Let us calculate how many such different maximal sets are there. If n is an even number then there will be only 2 maximal sets of $\frac{n}{2}$ dining persons, namely, the philosophers numbered with all odd natural numbers which are not greater than n and those numbered with all even natural numbers which are not greater than n. If n is an odd number then there will be n maximal sets of $\frac{n-1}{2}$ dining persons, since, starting from some maximal set one can "shift" clockwise $n-1$ times by one element modulo n until the next maximal set will coincide with the initial one.

Now we proceed with 5 dining philosophers system. Let us explain the meaning of actions from syntax of the *dtsPBC* expressions which will specify the system modules. The action a corresponds to the system activation. The actions b_i and e_i correspond to the beginning and the end, respectively, of eating of philosopher i $(1\le i\le 5)$. The other actions are used for communication purpose only via synchronization, and we abstract from them later using restriction. Note that the expression of each philosopher includes two alternative subexpressions such that the second one specifies a resource (fork) sharing with the right neighbor.

The static expression of the philosopher i $(1\le i\le 4)$ is $E_i=[(\{x_i\},\frac{1}{2})*(((\{b_i,\widehat{y_i}\},\frac{1}{2});(\{e_i,\widehat{z_i}\},\frac{1}{2}))[\,]((\{y_{i+1}\},\frac{1}{2});(\{z_{i+1}\},\frac{1}{2})))*\mathsf{Stop}]$. The static expression of the philosopher 5 is $E_5=[(\{a,\widehat{x_1},\widehat{x_2},\widehat{x_2},\widehat{x_4}\},\frac{1}{2})*(((\{b_5,\widehat{y_5}\},\frac{1}{2});(\{e_5,\widehat{z_5}\},\frac{1}{2}))[\,]((\{y_1\},\frac{1}{2});(\{z_1\},\frac{1}{2})))*\mathsf{Stop}]$. The static expression of the dining philosophers system is $E=(E_1\|E_2\|E_3\|E_4\|E_5)$ $\mathsf{sy}\ x_1$ $\mathsf{sy}\ x_2$ $\mathsf{sy}\ x_3$ $\mathsf{sy}\ x_4$ $\mathsf{sy}\ y_1$ $\mathsf{sy}\ y_2$ $\mathsf{sy}\ y_3$ $\mathsf{sy}\ y_4$ $\mathsf{sy}\ y_5$ $\mathsf{sy}\ z_1$ $\mathsf{sy}\ z_2$ $\mathsf{sy}\ z_3$ $\mathsf{sy}\ z_4$ $\mathsf{sy}\ z_5$ $\mathsf{rs}\ x_1$ $\mathsf{rs}\ x_2$ $\mathsf{rs}\ x_3$ $\mathsf{rs}\ x_4$ $\mathsf{rs}\ y_1$ $\mathsf{rs}\ y_2$ $\mathsf{rs}\ y_3$ $\mathsf{rs}\ y_4$ $\mathsf{rs}\ y_5$ $\mathsf{rs}\ z_1$ $\mathsf{rs}\ z_2$ $\mathsf{rs}\ z_3$ $\mathsf{rs}\ z_4$ $\mathsf{rs}\ z_5$.

Let us illustrate an effect of synchronization. In the result of synchronization of the activities $(\{b_i, y_i\}, \frac{1}{2})$ and $(\{\widehat{y_i}\}, \frac{1}{2})$ we obtain the new activity $(\{b_i\}, \frac{1}{4})$ $(1 \le i \le 5)$. The synchronization of $(\{e_i, z_i\}, \frac{1}{2})$ and $(\{\widehat{z_i}\}, \frac{1}{2})$ produces $(\{e_i\}, \frac{1}{4})$ $(1 \le i \le 5)$. The result of synchronization of $(\{a, \widehat{x_1}, \widehat{x_2}, \widehat{x_3}, \widehat{x_4}\}, \frac{1}{2})$ and $(\{x_1\}, \frac{1}{2})$ is $(\{a, \widehat{x_2}, \widehat{x_3}, \widehat{x_4}\}, \frac{1}{4})$. The result of synchronization of $(\{a, \widehat{x_2}, \widehat{x_3}, \widehat{x_4}\}, \frac{1}{4})$ and $(\{x_2\}, \frac{1}{2})$ is $(\{a, \widehat{x_3}, \widehat{x_4}\}, \frac{1}{8})$. The result of synchronization of $(\{a, \widehat{x_3}, \widehat{x_4}\}, \frac{1}{8})$ and $(\{x_3\}, \frac{1}{2})$ is $(\{a, \widehat{x_4}\}, \frac{1}{16})$. The result of synchronization of $(\{a, \widehat{x_4}\}, \frac{1}{16})$ and $(\{x_4\}, \frac{1}{2})$ is $(\{a\}, \frac{1}{32})$.

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

$$s_1 = [([\overline{(\{x_1\},\tfrac{1}{2})}*(((\{b_1,\widehat{y_1}\},\tfrac{1}{2});(\{e_1,\widehat{z_1}\},\tfrac{1}{2}))[]((\{y_2\},\tfrac{1}{2});(\{z_2\},\tfrac{1}{2})))*\mathsf{Stop}]|||[\overline{(\{x_2\},\tfrac{1}{2})}*(((\{b_2,\widehat{y_2}\},\tfrac{1}{2});(\{e_2,\widehat{z_2}\},\tfrac{1}{2}))[]((\{y_3\},\tfrac{1}{2});(\{z_3\},\tfrac{1}{2})))*\mathsf{Stop}]|||[\overline{(\{x_3\},\tfrac{1}{2})}*(((\{b_3,\widehat{y_3}\},\tfrac{1}{2});(\{e_3,\widehat{z_3}\},\tfrac{1}{2}))[]((\{y_4\},\tfrac{1}{2});(\{z_4\},\tfrac{1}{2})))*\mathsf{Stop}]|||[\overline{(\{x_4\},\tfrac{1}{2})}*(((\{b_4,\widehat{y_4}\},\tfrac{1}{2});(\{e_4,\widehat{z_4}\},\tfrac{1}{2}))[]((\{y_5\},\tfrac{1}{2});(\{z_5\},\tfrac{1}{2})))*\mathsf{Stop}]|||[\overline{(\{a,\widehat{x_1},\widehat{x_2},\widehat{x_2},\widehat{x_4}\},\tfrac{1}{2})}*(((\{b_5,\widehat{y_5}\},\tfrac{1}{2});(\{e_5,\widehat{z_5}\},\tfrac{1}{2}))[]((\{y_1\},\tfrac{1}{2});(\{z_1\},\tfrac{1}{2})))*\mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ x_3\ \mathsf{sy}\ x_4\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ y_3\ \mathsf{sy}\ y_4\ \mathsf{sy}\ y_5\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{sy}\ z_3\ \mathsf{sy}\ z_4\ \mathsf{sy}\ z_5\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ x_3\ \mathsf{rs}\ x_4\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ y_3\ \mathsf{rs}\ y_4\ \mathsf{rs}\ y_5\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2\ \mathsf{rs}\ z_3\ \mathsf{rs}\ z_4\ \mathsf{rs}\ z_5]_{\approx},$$

$$s_2 = [([(\{x_1\},\tfrac{1}{2})*\overline{(((\{b_1,\widehat{y_1}\},\tfrac{1}{2});(\{e_1,\widehat{z_1}\},\tfrac{1}{2}))[]((\{y_2\},\tfrac{1}{2});(\{z_2\},\tfrac{1}{2})))}*\mathsf{Stop}]|||[(\{x_2\},\tfrac{1}{2})*\overline{(((\{b_2,\widehat{y_2}\},\tfrac{1}{2});(\{e_2,\widehat{z_2}\},\tfrac{1}{2}))[]((\{y_3\},\tfrac{1}{2});(\{z_3\},\tfrac{1}{2})))}*\mathsf{Stop}]|||[(\{x_3\},\tfrac{1}{2})*\overline{(((\{b_3,\widehat{y_3}\},\tfrac{1}{2});(\{e_3,\widehat{z_3}\},\tfrac{1}{2}))[]((\{y_4\},\tfrac{1}{2});(\{z_4\},\tfrac{1}{2})))}*\mathsf{Stop}]|||[(\{x_4\},\tfrac{1}{2})*\overline{(((\{b_4,\widehat{y_4}\},\tfrac{1}{2});(\{e_4,\widehat{z_4}\},\tfrac{1}{2}))[]((\{y_5\},\tfrac{1}{2});(\{z_5\},\tfrac{1}{2})))}*\mathsf{Stop}]|||[(\{a,\widehat{x_1},\widehat{x_2},\widehat{x_2},\widehat{x_4}\},\tfrac{1}{2})*\overline{(((\{b_5,\widehat{y_5}\},\tfrac{1}{2});(\{e_5,\widehat{z_5}\},\tfrac{1}{2}))[]((\{y_1\},\tfrac{1}{2});(\{z_1\},\tfrac{1}{2})))}*\mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ x_3\ \mathsf{sy}\ x_4\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ y_3\ \mathsf{sy}\ y_4\ \mathsf{sy}\ y_5\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{sy}\ z_3\ \mathsf{sy}\ z_4\ \mathsf{sy}\ z_5\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ x_3\ \mathsf{rs}\ x_4\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ y_3\ \mathsf{rs}\ y_4\ \mathsf{rs}\ y_5\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2\ \mathsf{rs}\ z_3\ \mathsf{rs}\ z_4\ \mathsf{rs}\ z_5]_{\approx},$$

$$s_3 = [([(\{x_1\},\tfrac{1}{2})*(((\{b_1,\widehat{y_1}\},\tfrac{1}{2});\overline{(\{e_1,\widehat{z_1}\},\tfrac{1}{2})})[]((\{y_2\},\tfrac{1}{2});(\{z_2\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{x_2\},\tfrac{1}{2})*(((\{b_2,\widehat{y_2}\},\tfrac{1}{2});(\{e_2,\widehat{z_2}\},\tfrac{1}{2}))[]((\overline{\{y_3\},\tfrac{1}{2})};(\{z_3\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{x_3\},\tfrac{1}{2})*\overline{(((\{b_3,\widehat{y_3}\},\tfrac{1}{2});(\{e_3,\widehat{z_3}\},\tfrac{1}{2}))[]((\{y_4\},\tfrac{1}{2});(\{z_4\},\tfrac{1}{2})))}*\mathsf{Stop}]|||[(\{x_4\},\tfrac{1}{2})*(((\overline{\{b_4,\widehat{y_4}\},\tfrac{1}{2})};(\{e_4,\widehat{z_4}\},\tfrac{1}{2}))[]((\{y_5\},\tfrac{1}{2});(\{z_5\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{a,\widehat{x_1},\widehat{x_2},\widehat{x_2},\widehat{x_4}\},\tfrac{1}{2})*(((\{b_5,\widehat{y_5}\},\tfrac{1}{2});(\{e_5,\widehat{z_5}\},\tfrac{1}{2}))[]((\{y_1\},\tfrac{1}{2});\overline{(\{z_1\},\tfrac{1}{2})}))*\mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ x_3\ \mathsf{sy}\ x_4\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ y_3\ \mathsf{sy}\ y_4\ \mathsf{sy}\ y_5\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{sy}\ z_3\ \mathsf{sy}\ z_4\ \mathsf{sy}\ z_5\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ x_3\ \mathsf{rs}\ x_4\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ y_3\ \mathsf{rs}\ y_4\ \mathsf{rs}\ y_5\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2\ \mathsf{rs}\ z_3\ \mathsf{rs}\ z_4\ \mathsf{rs}\ z_5]_{\approx},$$

$$s_4 = [([(\{x_1\},\tfrac{1}{2})*(((\{b_1,\widehat{y_1}\},\tfrac{1}{2});\overline{(\{e_1,\widehat{z_1}\},\tfrac{1}{2})})[]((\{y_2\},\tfrac{1}{2});(\{z_2\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{x_2\},\tfrac{1}{2})*\overline{(((\{b_2,\widehat{y_2}\},\tfrac{1}{2});(\{e_2,\widehat{z_2}\},\tfrac{1}{2}))[]((\{y_3\},\tfrac{1}{2});(\{z_3\},\tfrac{1}{2})))}*\mathsf{Stop}]|||[(\{x_3\},\tfrac{1}{2})*(((\{b_3,\widehat{y_3}\},\tfrac{1}{2});(\{e_3,\widehat{z_3}\},\tfrac{1}{2}))[]((\{y_4\},\tfrac{1}{2});\overline{(\{z_4\},\tfrac{1}{2})}))*\mathsf{Stop}]|||[(\{x_4\},\tfrac{1}{2})*(((\{b_4,\widehat{y_4}\},\tfrac{1}{2});\overline{(\{e_4,\widehat{z_4}\},\tfrac{1}{2})})[]((\{y_5\},\tfrac{1}{2});(\{z_5\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{a,\widehat{x_1},\widehat{x_2},\widehat{x_2},\widehat{x_4}\},\tfrac{1}{2})*(((\{b_5,\widehat{y_5}\},\tfrac{1}{2});(\{e_5,\widehat{z_5}\},\tfrac{1}{2}))[]((\{y_1\},\tfrac{1}{2});\overline{(\{z_1\},\tfrac{1}{2})}))*\mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ x_3\ \mathsf{sy}\ x_4\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ y_3\ \mathsf{sy}\ y_4\ \mathsf{sy}\ y_5\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{sy}\ z_3\ \mathsf{sy}\ z_4\ \mathsf{sy}\ z_5\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ x_3\ \mathsf{rs}\ x_4\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ y_3\ \mathsf{rs}\ y_4\ \mathsf{rs}\ y_5\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2\ \mathsf{rs}\ z_3\ \mathsf{rs}\ z_4\ \mathsf{rs}\ z_5]_{\approx},$$

$$s_5 = [([(\{x_1\},\tfrac{1}{2})*(((\{b_1,\widehat{y_1}\},\tfrac{1}{2});\overline{(\{e_1,\widehat{z_1}\},\tfrac{1}{2})})[]((\{y_2\},\tfrac{1}{2});(\{z_2\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{x_2\},\tfrac{1}{2})*(((\{b_2,\widehat{y_2}\},\tfrac{1}{2});(\{e_2,\widehat{z_2}\},\tfrac{1}{2}))[]((\{y_3\},\tfrac{1}{2});\overline{(\{z_3\},\tfrac{1}{2})}))*\mathsf{Stop}]|||[(\{x_3\},\tfrac{1}{2})*(((\{b_3,\widehat{y_3}\},\tfrac{1}{2});\overline{(\{e_3,\widehat{z_3}\},\tfrac{1}{2})})[]((\{y_4\},\tfrac{1}{2});(\{z_4\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{x_4\},\tfrac{1}{2})*\overline{(((\{b_4,\widehat{y_4}\},\tfrac{1}{2});(\{e_4,\widehat{z_4}\},\tfrac{1}{2}))[]((\{y_5\},\tfrac{1}{2});(\{z_5\},\tfrac{1}{2})))}*\mathsf{Stop}]|||[(\{a,\widehat{x_1},\widehat{x_2},\widehat{x_2},\widehat{x_4}\},\tfrac{1}{2})*(((\{b_5,\widehat{y_5}\},\tfrac{1}{2});(\{e_5,\widehat{z_5}\},\tfrac{1}{2}))[]((\{y_1\},\tfrac{1}{2});\overline{(\{z_1\},\tfrac{1}{2})}))*\mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ x_3\ \mathsf{sy}\ x_4\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ y_3\ \mathsf{sy}\ y_4\ \mathsf{sy}\ y_5\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{sy}\ z_3\ \mathsf{sy}\ z_4\ \mathsf{sy}\ z_5\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ x_3\ \mathsf{rs}\ x_4\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ y_3\ \mathsf{rs}\ y_4\ \mathsf{rs}\ y_5\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2\ \mathsf{rs}\ z_3\ \mathsf{rs}\ z_4\ \mathsf{rs}\ z_5]_{\approx},$$

$$s_6 = [([(\{x_1\},\tfrac{1}{2})*\overline{(((\{b_1,\widehat{y_1}\},\tfrac{1}{2});(\{e_1,\widehat{z_1}\},\tfrac{1}{2}))[]((\{y_2\},\tfrac{1}{2});(\{z_2\},\tfrac{1}{2})))}*\mathsf{Stop}]|||[(\{x_2\},\tfrac{1}{2})*(((\overline{\{b_2,\widehat{y_2}\},\tfrac{1}{2})};(\{e_2,\widehat{z_2}\},\tfrac{1}{2}))[]((\{y_3\},\tfrac{1}{2});(\{z_3\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{x_3\},\tfrac{1}{2})*(((\{b_3,\widehat{y_3}\},\tfrac{1}{2});(\{e_3,\widehat{z_3}\},\tfrac{1}{2}))[]((\{y_4\},\tfrac{1}{2});\overline{(\{z_4\},\tfrac{1}{2})}))*\mathsf{Stop}]|||[(\{x_4\},\tfrac{1}{2})*(((\{b_4,\widehat{y_4}\},\tfrac{1}{2});\overline{(\{e_4,\widehat{z_4}\},\tfrac{1}{2})})[]((\{y_5\},\tfrac{1}{2});(\{z_5\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{a,\widehat{x_1},\widehat{x_2},\widehat{x_2},\widehat{x_4}\},\tfrac{1}{2})*(((\{b_5,\widehat{y_5}\},\tfrac{1}{2});(\{e_5,\widehat{z_5}\},\tfrac{1}{2}))[]((\overline{\{y_1\},\tfrac{1}{2})};(\{z_1\},\tfrac{1}{2})))*\mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ x_3\ \mathsf{sy}\ x_4\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ y_3\ \mathsf{sy}\ y_4\ \mathsf{sy}\ y_5\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{sy}\ z_3\ \mathsf{sy}\ z_4\ \mathsf{sy}\ z_5\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ x_3\ \mathsf{rs}\ x_4\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ y_3\ \mathsf{rs}\ y_4\ \mathsf{rs}\ y_5\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2\ \mathsf{rs}\ z_3\ \mathsf{rs}\ z_4\ \mathsf{rs}\ z_5]_{\approx},$$

$$s_7 = [([(\{x_1\},\tfrac{1}{2})*(((\overline{\{b_1,\widehat{y_1}\},\tfrac{1}{2})};(\{e_1,\widehat{z_1}\},\tfrac{1}{2}))[]((\{y_2\},\tfrac{1}{2});(\{z_2\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{x_2\},\tfrac{1}{2})*(((\{b_2,\widehat{y_2}\},\tfrac{1}{2});(\{e_2,\widehat{z_2}\},\tfrac{1}{2}))[]((\{y_3\},\tfrac{1}{2});\overline{(\{z_3\},\tfrac{1}{2})}))*\mathsf{Stop}]|||[(\{x_3\},\tfrac{1}{2})*(((\{b_3,\widehat{y_3}\},\tfrac{1}{2});\overline{(\{e_3,\widehat{z_3}\},\tfrac{1}{2})})[]((\{y_4\},\tfrac{1}{2});(\{z_4\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{x_4\},\tfrac{1}{2})*(((\{b_4,\widehat{y_4}\},\tfrac{1}{2});(\{e_4,\widehat{z_4}\},\tfrac{1}{2}))[]((\overline{\{y_5\},\tfrac{1}{2})};(\{z_5\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{a,\widehat{x_1},\widehat{x_2},\widehat{x_2},\widehat{x_4}\},\tfrac{1}{2})*\overline{(((\{b_5,\widehat{y_5}\},\tfrac{1}{2});(\{e_5,\widehat{z_5}\},\tfrac{1}{2}))[]((\{y_1\},\tfrac{1}{2});(\{z_1\},\tfrac{1}{2})))}*\mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ x_3\ \mathsf{sy}\ x_4\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ y_3\ \mathsf{sy}\ y_4\ \mathsf{sy}\ y_5\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{sy}\ z_3\ \mathsf{sy}\ z_4\ \mathsf{sy}\ z_5\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ x_3\ \mathsf{rs}\ x_4\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ y_3\ \mathsf{rs}\ y_4\ \mathsf{rs}\ y_5\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2\ \mathsf{rs}\ z_3\ \mathsf{rs}\ z_4\ \mathsf{rs}\ z_5]_{\approx},$$

$$s_8 = [([(\{x_1\},\tfrac{1}{2})*(((\{b_1,\widehat{y_1}\},\tfrac{1}{2});(\{e_1,\widehat{z_1}\},\tfrac{1}{2}))[]((\{y_2\},\tfrac{1}{2});\overline{(\{z_2\},\tfrac{1}{2})}))*\mathsf{Stop}]|||[(\{x_2\},\tfrac{1}{2})*(((\{b_2,\widehat{y_2}\},\tfrac{1}{2});(\{e_2,\widehat{z_2}\},\tfrac{1}{2}))[]((\{y_3\},\tfrac{1}{2});(\{z_3\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{x_3\},\tfrac{1}{2})*(((\{b_3,\widehat{y_3}\},\tfrac{1}{2});(\{e_3,\widehat{z_3}\},\tfrac{1}{2}))[]((\{y_4\},\tfrac{1}{2});\overline{(\{z_4\},\tfrac{1}{2})}))*\mathsf{Stop}]|||[(\{x_4\},\tfrac{1}{2})*(((\{b_4,\widehat{y_4}\},\tfrac{1}{2});\overline{(\{e_4,\widehat{z_4}\},\tfrac{1}{2})})[]((\{y_5\},\tfrac{1}{2});(\{z_5\},\tfrac{1}{2})))*\mathsf{Stop}]|||[(\{a,\widehat{x_1},\widehat{x_2},\widehat{x_2},\widehat{x_4}\},\tfrac{1}{2})*\overline{(((\{b_5,\widehat{y_5}\},\tfrac{1}{2});(\{e_5,\widehat{z_5}\},\tfrac{1}{2}))[]((\{y_1\},\tfrac{1}{2});(\{z_1\},\tfrac{1}{2})))}*\mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ x_3\ \mathsf{sy}\ x_4\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ y_3\ \mathsf{sy}\ y_4\ \mathsf{sy}\ y_5\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{sy}\ z_3\ \mathsf{sy}\ z_4\ \mathsf{sy}\ z_5\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ x_3\ \mathsf{rs}\ x_4\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ y_3\ \mathsf{rs}\ y_4\ \mathsf{rs}\ y_5\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2\ \mathsf{rs}\ z_3\ \mathsf{rs}\ z_4\ \mathsf{rs}\ z_5]_{\approx},$$

$$s_9 = [([(\{x_1\},\tfrac{1}{2})*\overline{(((\{b_1,\widehat{y_1}\},\tfrac{1}{2});(\{e_1,\widehat{z_1}\},\tfrac{1}{2}))[]((\{y_2\},\tfrac{1}{2});(\{z_2\},\tfrac{1}{2})))}*\mathsf{Stop}]|||[(\{x_2\},\tfrac{1}{2})*(((\{b_2,\widehat{y_2}\},\tfrac{1}{2});(\{e_2,\widehat{z_2}\},\tfrac{1}{2}))[]((\{y_3\},\tfrac{1}{2});\overline{(\{z_3\},\tfrac{1}{2})}))*\mathsf{Stop}]|||[(\{x_3\},\tfrac{1}{2})*(((\{b_3,\widehat{y_3}\},\tfrac{1}{2});\overline{(\{e_3,\widehat{z_3}\},\tfrac{1}{2})})[]((\{y_4\},\tfrac{1}{2});(\{z_4\},\tfrac{1}{2})))*\mathsf{Stop}]$$

Table 6: Transient and steady-state probabilities of the dining philosophers system

k	0	1	2	3	4	5	6	7	8	9	10	∞
$\psi_1^*[k]$	1	0	0	0	0	0	0	0	0	0	0	0
$\psi_2^*[k]$	0	1	0	0.2403	0.1541	0.1981	0.1716	0.1884	0.1776	0.1846	0.1800	0.1818
$\psi_3^*[k]$	0	0	0.1500	0.0701	0.1189	0.0878	0.1079	0.0949	0.1033	0.0979	0.1014	0.1000
$\psi_4^*[k]$	0	0	0.0500	0.0818	0.0503	0.0726	0.0578	0.0674	0.0612	0.0652	0.0626	0.0636

$\|[(\{x_4\},\frac{1}{2}) * (((\{b_4,\widehat{y_4}\},\frac{1}{2});(\{e_4,\widehat{z_4}\},\frac{1}{2}))[]((\{y_5\},\frac{1}{2});(\{z_5\},\frac{1}{2}))) * \mathsf{Stop}]\|[(\{a,\widehat{x_1},\widehat{x_2},\widehat{x_2},\widehat{x_4}\},\frac{1}{2}) * (((\{b_5,\widehat{y_5}\},\frac{1}{2});$
$(\{e_5,\widehat{z_5}\},\frac{1}{2}))[]((\{y_1\},\frac{1}{2});(\{z_1\},\frac{1}{2}))) * \mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ x_3\ \mathsf{sy}\ x_4\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ y_3\ \mathsf{sy}\ y_4\ \mathsf{sy}\ y_5\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{sy}\ z_3$
$\mathsf{sy}\ z_4\ \mathsf{sy}\ z_5\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ x_3\ \mathsf{rs}\ x_4\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ y_3\ \mathsf{rs}\ y_4\ \mathsf{rs}\ y_5\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2\ \mathsf{rs}\ z_3\ \mathsf{rs}\ z_4\ \mathsf{rs}\ z_5]_\approx,$

$s_{10} = [([(\{x_1\},\frac{1}{2}) * (((\{b_1,\widehat{y_1}\},\frac{1}{2});(\{e_1,\widehat{z_1}\},\frac{1}{2}))[]((\{y_2\},\frac{1}{2});(\{z_2\},\frac{1}{2}))) * \mathsf{Stop}]\|[(\{x_2\},\frac{1}{2}) * (((\{b_2,\widehat{y_2}\},\frac{1}{2});$
$(\{e_2,\widehat{z_2}\},\frac{1}{2}))[]((\{y_3\},\frac{1}{2});(\{z_3\},\frac{1}{2}))) * \mathsf{Stop}]\|[(\{x_3\},\frac{1}{2}) * (((\{b_3,\widehat{y_3}\},\frac{1}{2});(\{e_3,\widehat{z_3}\},\frac{1}{2}))[]((\{y_4\},\frac{1}{2});(\{z_4\},\frac{1}{2}))) * \mathsf{Stop}]$
$\|[(\{x_4\},\frac{1}{2}) * (((\{b_4,\widehat{y_4}\},\frac{1}{2});(\{e_4,\widehat{z_4}\},\frac{1}{2}))[]((\{y_5\},\frac{1}{2});(\{z_5\},\frac{1}{2}))) * \mathsf{Stop}]\|[(\{a,\widehat{x_1},\widehat{x_2},\widehat{x_2},\widehat{x_4}\},\frac{1}{2}) * (((\{b_5,\widehat{y_5}\},\frac{1}{2});$
$(\{e_5,\widehat{z_5}\},\frac{1}{2}))[]((\{y_1\},\frac{1}{2});(\{z_1\},\frac{1}{2}))) * \mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ x_3\ \mathsf{sy}\ x_4\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ y_3\ \mathsf{sy}\ y_4\ \mathsf{sy}\ y_5\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{sy}\ z_3$
$\mathsf{sy}\ z_4\ \mathsf{sy}\ z_5\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ x_3\ \mathsf{rs}\ x_4\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ y_3\ \mathsf{rs}\ y_4\ \mathsf{rs}\ y_5\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2\ \mathsf{rs}\ z_3\ \mathsf{rs}\ z_4\ \mathsf{rs}\ z_5]_\approx,$

$s_{11} = [([(\{x_1\},\frac{1}{2}) * (((\{b_1,\widehat{y_1}\},\frac{1}{2});(\{e_1,\widehat{z_1}\},\frac{1}{2}))[]((\{y_2\},\frac{1}{2});(\{z_2\},\frac{1}{2}))) * \mathsf{Stop}]\|[(\{x_2\},\frac{1}{2}) * (((\{b_2,\widehat{y_2}\},\frac{1}{2});$
$(\{e_2,\widehat{z_2}\},\frac{1}{2}))[]((\{y_3\},\frac{1}{2});(\{z_3\},\frac{1}{2}))) * \mathsf{Stop}]\|[(\{x_3\},\frac{1}{2}) * (((\{b_3,\widehat{y_3}\},\frac{1}{2});(\{e_3,\widehat{z_3}\},\frac{1}{2}))[]((\{y_4\},\frac{1}{2});(\{z_4\},\frac{1}{2}))) * \mathsf{Stop}]$
$\|[(\{x_4\},\frac{1}{2}) * (((\{b_4,\widehat{y_4}\},\frac{1}{2});(\{e_4,\widehat{z_4}\},\frac{1}{2}))[]((\{y_5\},\frac{1}{2});\overline{(\{z_5\},\frac{1}{2})})) * \mathsf{Stop}]\|[(\{a,\widehat{x_1},\widehat{x_2},\widehat{x_2},\widehat{x_4}\},\frac{1}{2}) * (((\{b_5,\widehat{y_5}\},\frac{1}{2});$
$(\{e_5,\widehat{z_5}\},\frac{1}{2}))[]((\{y_1\},\frac{1}{2});(\{z_1\},\frac{1}{2}))) * \mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ x_3\ \mathsf{sy}\ x_4\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ y_3\ \mathsf{sy}\ y_4\ \mathsf{sy}\ y_5\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{sy}\ z_3$
$\mathsf{sy}\ z_4\ \mathsf{sy}\ z_5\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ x_3\ \mathsf{rs}\ x_4\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ y_3\ \mathsf{rs}\ y_4\ \mathsf{rs}\ y_5\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2\ \mathsf{rs}\ z_3\ \mathsf{rs}\ z_4\ \mathsf{rs}\ z_5]_\approx,$

$s_{12} = [([(\{x_1\},\frac{1}{2}) * (((\{b_1,\widehat{y_1}\},\frac{1}{2});(\{e_1,\widehat{z_1}\},\frac{1}{2}))[]((\{y_2\},\frac{1}{2});\overline{(\{z_2\},\frac{1}{2})})) * \mathsf{Stop}]\|[(\{x_2\},\frac{1}{2}) * (((\{b_2,\widehat{y_2}\},\frac{1}{2});$
$(\{e_2,\widehat{z_2}\},\frac{1}{2}))[]((\{y_3\},\frac{1}{2});(\{z_3\},\frac{1}{2}))) * \mathsf{Stop}]\|[(\{x_3\},\frac{1}{2}) * (((\{b_3,\widehat{y_3}\},\frac{1}{2});(\{e_3,\widehat{z_3}\},\frac{1}{2}))[]((\{y_4\},\frac{1}{2});(\{z_4\},\frac{1}{2}))) * \mathsf{Stop}]$
$\|[(\{x_4\},\frac{1}{2}) * (((\{b_4,\widehat{y_4}\},\frac{1}{2});(\{e_4,\widehat{z_4}\},\frac{1}{2}))[]((\{y_5\},\frac{1}{2});\overline{(\{z_5\},\frac{1}{2})})) * \mathsf{Stop}]\|[(\{a,\widehat{x_1},\widehat{x_2},\widehat{x_2},\widehat{x_4}\},\frac{1}{2}) * (((\{b_5,\widehat{y_5}\},\frac{1}{2});$
$(\{e_5,\widehat{z_5}\},\frac{1}{2}))[]((\{y_1\},\frac{1}{2});(\{z_1\},\frac{1}{2}))) * \mathsf{Stop}])\ \mathsf{sy}\ x_1\ \mathsf{sy}\ x_2\ \mathsf{sy}\ x_3\ \mathsf{sy}\ x_4\ \mathsf{sy}\ y_1\ \mathsf{sy}\ y_2\ \mathsf{sy}\ y_3\ \mathsf{sy}\ y_4\ \mathsf{sy}\ y_5\ \mathsf{sy}\ z_1\ \mathsf{sy}\ z_2\ \mathsf{sy}\ z_3$
$\mathsf{sy}\ z_4\ \mathsf{sy}\ z_5\ \mathsf{rs}\ x_1\ \mathsf{rs}\ x_2\ \mathsf{rs}\ x_3\ \mathsf{rs}\ x_4\ \mathsf{rs}\ y_1\ \mathsf{rs}\ y_2\ \mathsf{rs}\ y_3\ \mathsf{rs}\ y_4\ \mathsf{rs}\ y_5\ \mathsf{rs}\ z_1\ \mathsf{rs}\ z_2\ \mathsf{rs}\ z_3\ \mathsf{rs}\ z_4\ \mathsf{rs}\ z_5]_\approx.$

The states are interpreted as follows: s_1 is the initial state, s_2: the system is activated and no philosophers dine, s_3: philosopher 1 dines, s_4: philosophers 1 and 4 dine, s_5: philosophers 1 and 3 dine, s_6: philosopher 4 dines, s_7: philosopher 3 dines, s_8: philosophers 2 and 4 dine, s_9: philosophers 3 and 5 dine, s_{10}: philosopher 2 dines, s_{11}: philosopher 5 dines, s_{12}: philosophers 2 and 5 dine.

In Figure 34 the transition system without empty loops $TS^*(\overline{E})$ is presented. In Figure 35 the underlying DTMC without empty loops $DTMC^*(\overline{E})$ is depicted.

The TPM for $DTMC^*(\overline{E})$ is

$$\mathbf{P}^* = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{3}{20} & \frac{1}{20} & \frac{1}{20} & \frac{3}{20} & \frac{3}{20} & \frac{1}{20} & \frac{1}{20} & \frac{3}{20} & \frac{3}{20} & \frac{1}{20} \\ 0 & \frac{3}{11} & 0 & \frac{3}{11} & \frac{3}{11} & \frac{1}{11} & \frac{1}{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{7} & \frac{3}{7} & 0 & 0 & \frac{3}{7} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{7} & \frac{3}{7} & 0 & 0 & 0 & \frac{3}{7} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{3}{11} & \frac{1}{11} & \frac{3}{11} & 0 & 0 & 0 & \frac{3}{11} & 0 & \frac{1}{11} & 0 & 0 \\ 0 & \frac{3}{11} & \frac{1}{11} & 0 & \frac{3}{11} & 0 & 0 & 0 & \frac{3}{11} & 0 & \frac{1}{11} & 0 \\ 0 & \frac{1}{7} & 0 & 0 & 0 & \frac{3}{7} & 0 & 0 & 0 & \frac{3}{7} & 0 & 0 \\ 0 & \frac{1}{7} & 0 & 0 & 0 & 0 & \frac{3}{7} & 0 & 0 & 0 & \frac{3}{7} & 0 \\ 0 & \frac{3}{11} & 0 & 0 & 0 & \frac{1}{11} & 0 & \frac{3}{11} & 0 & 0 & \frac{1}{11} & \frac{3}{11} \\ 0 & \frac{3}{11} & 0 & 0 & 0 & 0 & \frac{1}{11} & 0 & \frac{3}{11} & \frac{1}{11} & 0 & \frac{3}{11} \\ 0 & \frac{1}{7} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{3}{7} & \frac{3}{7} & 0 \end{bmatrix}.$$

In Table 6 the transient and the steady-state probabilities $\psi_i^*[k]$ $(1 \le i \le 4)$ of the dining philosophers system at the time moments k $(0 \le k \le 10)$ and $k = \infty$ are presented, and in Figure 36 the alteration diagram (evolution in time) for the transient probabilities is depicted. It is sufficient to consider the probabilities for the states $s_1, \dots, s_4$ only, since the corresponding values coincide for $s_3, s_6, s_7, s_{10}, s_{11}$ as well as for $s_4, s_5, s_8, s_9, s_{12}$.

The steady-state PMF for $DTMC^*(\overline{E})$ is

$TS^*(\overline{E})$

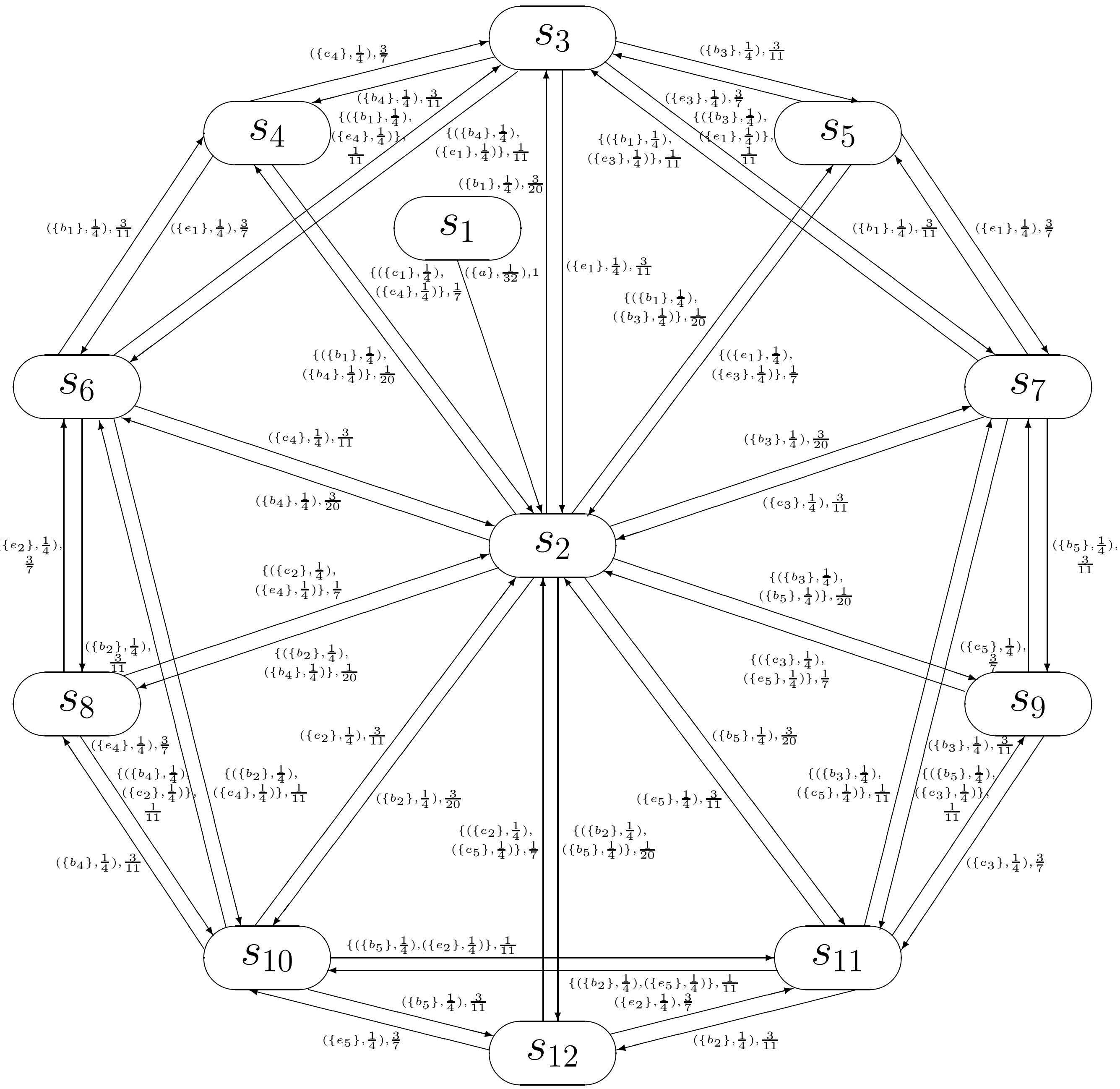

Figure 34: The transition system without empty loops of the dining philosophers system

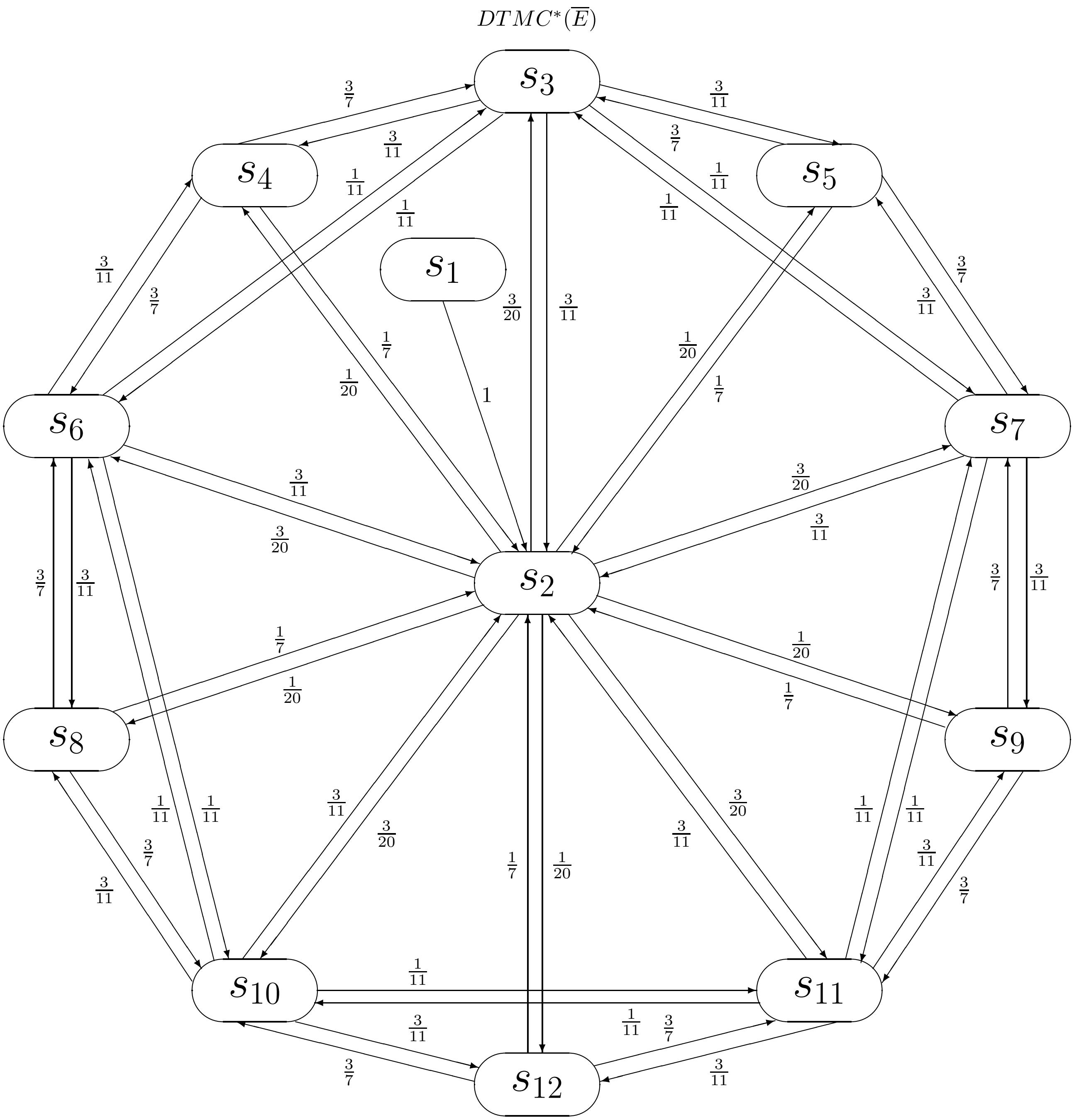

Figure 35: The underlying DTMC without empty loops of the dining philosophers system

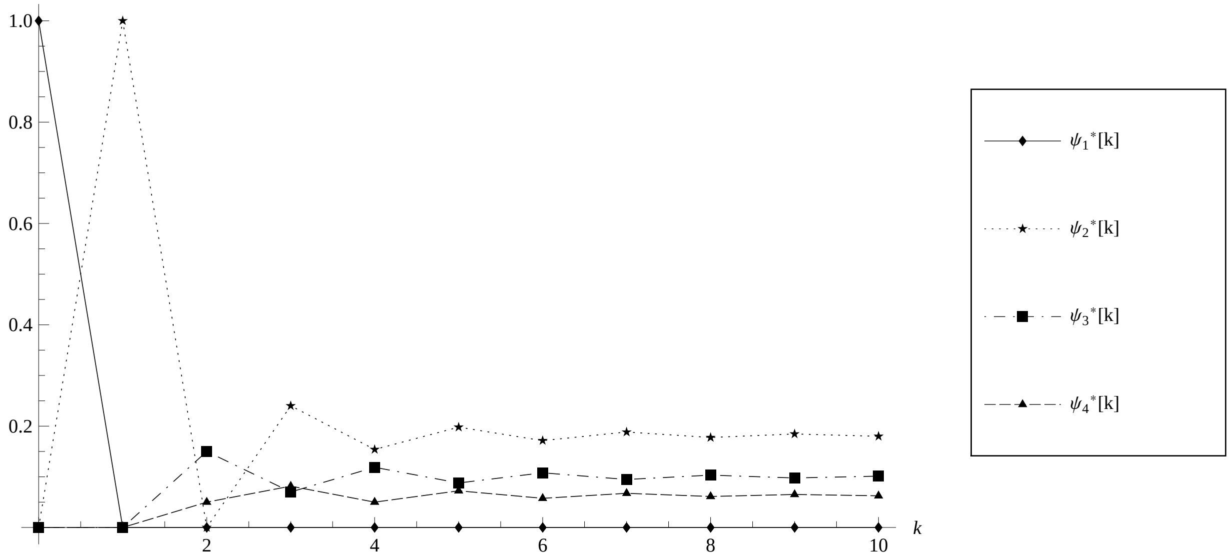

Figure 36: Transient probabilities alteration diagram of the dining philosophers system

$$\psi^* = \left(0, \frac{2}{11}, \frac{1}{10}, \frac{7}{110}, \frac{7}{110}, \frac{1}{10}, \frac{1}{10}, \frac{7}{110}, \frac{7}{110}, \frac{1}{10}, \frac{1}{10}, \frac{7}{110}\right).$$

We can now calculate the main performance indices.

- The average recurrence time in the state s_2, where all the forks are available, called the *average system run-through*, is $\frac{1}{\psi_2^*} = \frac{11}{2} = 5\frac{1}{2}$.
- Nobody eats in the state s_2. Then, the *fraction of time when no philosophers dine* is $\psi_2^* = \frac{2}{11}$.

  Only one philosopher eats in the states $s_3, s_6, s_7, s_{10}, s_{11}$. Then, the *fraction of time when only one philosopher dines* is $\psi_3^* + \psi_6^* + \psi_7^* + \psi_{10}^* + \psi_{11}^* = \frac{1}{10} + \frac{1}{10} + \frac{1}{10} + \frac{1}{10} + \frac{1}{10} = \frac{1}{2}$.

  Two philosophers eat together in the states $s_4, s_5, s_8, s_9, s_{12}$. Then, the *fraction of time when two philosophers dine* is $\psi_4^* + \psi_5^* + \psi_8^* + \psi_9^* + \psi_{12}^* = \frac{7}{110} + \frac{7}{110} + \frac{7}{110} + \frac{7}{110} + \frac{7}{110} = \frac{7}{22}$.

  The *relative fraction of time when two philosophers dine with respect to when only one philosopher dines* is $\frac{7}{22} \cdot \frac{2}{1} = \frac{7}{11}$.
- The beginning of eating of first philosopher $(\{b_1\}, \frac{1}{4})$ is only possible from the states s_2, s_6, s_7. In each of the states the beginning of eating probability is the sum of the execution probabilities for all multisets of activities containing $(\{b_1\}, \frac{1}{4})$. Thus, the *steady-state probability of the beginning of eating of first philosopher* is $\psi_2^* \sum_{\{\Gamma|(\{b_1\},\frac{1}{4})\in\Gamma\}} PT^*(\Gamma, s_2) + \psi_6^* \sum_{\{\Gamma|(\{b_1\},\frac{1}{4})\in\Gamma\}} PT^*(\Gamma, s_6) + \psi_7^* \sum_{\{\Gamma|(\{b_1\},\frac{1}{4})\in\Gamma\}} PT^*(\Gamma, s_7) = \frac{2}{11}\left(\frac{3}{20} + \frac{1}{20} + \frac{1}{20}\right) + \frac{1}{10}\left(\frac{3}{11} + \frac{1}{11}\right) + \frac{1}{10}\left(\frac{3}{11} + \frac{1}{11}\right) = \frac{13}{110}$.

In Figure 37 the marked dts-boxes corresponding to the dynamic expressions of the dining philosophers are presented, i.e., $N_i = Box_{dts}(\overline{E_i})$ $(1 \le i \le 5)$. In Figure 38 the marked dts-box corresponding to the dynamic expression of the dining philosophers system is depicted, i.e., $N = Box_{dts}(\overline{E})$.

10.2.2 The abstract system and its reductions

Let us consider a modification of the dining philosophers system with abstraction from personalities, i.e., such that all the philosophers are indistinguishable. For example, we can just see that one or two philosophers dine but cannot observe who they are. We call this system the abstract dining philosophers one.

The static expression of the philosopher i $(1 \le i \le 4)$ is $F_i = [(\{x_i\}, \frac{1}{2}) * (((\{b, \widehat{y_i}\}, \frac{1}{2}); (\{e, \widehat{z_i}\}, \frac{1}{2}))[] ((\{y_{i+1}\}, \frac{1}{2}); (\{z_{i+1}\}, \frac{1}{2}))) * \mathsf{Stop}]$. The static expression of the philosopher 5 is $F_5 = [(\{a, \widehat{x_1}, \widehat{x_2}, \widehat{x_2}, \widehat{x_4}\}, \frac{1}{2}) * (((\{b, \widehat{y_5}\}, \frac{1}{2}); (\{e, \widehat{z_5}\}, \frac{1}{2}))[]((\{y_1\}, \frac{1}{2}); (\{z_1\}, \frac{1}{2}))) * \mathsf{Stop}]$. The static expression of the abstract dining philosophers system is $F = (F_1 \| F_2 \| F_3 \| F_4 \| F_5)$ sy x_1 sy x_2 sy x_3 sy x_4 sy y_1 sy y_2 sy y_3 sy y_4 sy y_5 sy z_1 sy z_2 sy z_3 sy z_4 sy z_5 rs x_1 rs x_2 rs x_3 rs x_4 rs y_1 rs y_2 rs y_3 rs y_4 rs y_5 rs z_1 rs z_2 rs z_3 rs z_4 rs z_5.

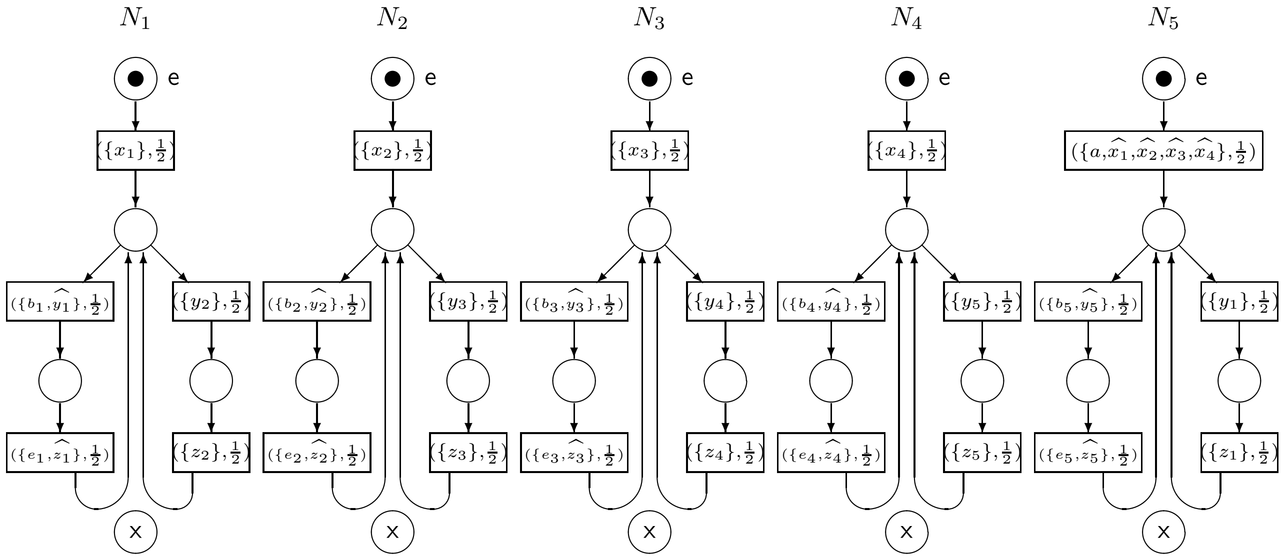

Figure 37: The marked dts-boxes of the dining philosophers

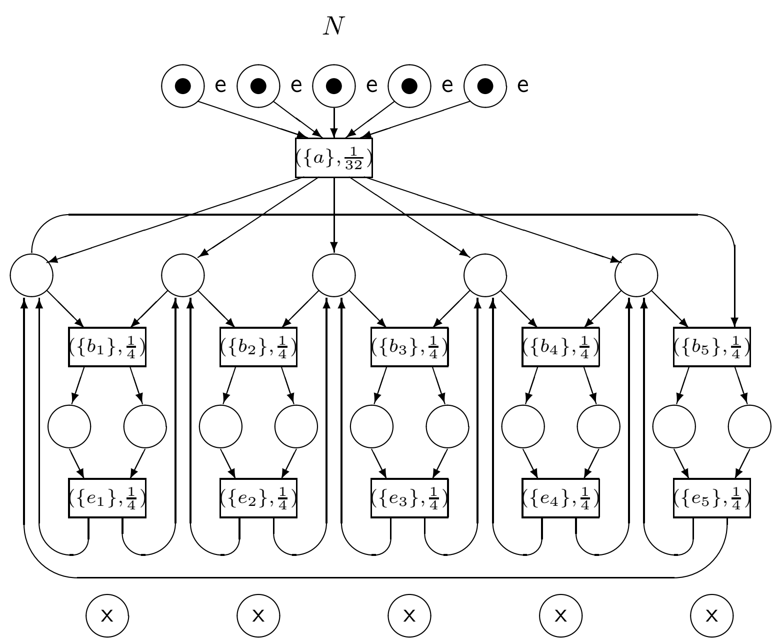

Figure 38: The marked dts-box of the dining philosophers system

Table 7: Transient and steady-state probabilities of the reduced abstract dining philosophers system

k	0	1	2	3	4	5	6	7	8	9	10	∞
${\psi'_1}^*[k]$	1	0	0	0	0	0	0	0	0	0	0	0
${\psi'_2}^*[k]$	0	1	0	0.2403	0.1541	0.1981	0.1716	0.1884	0.1776	0.1846	0.1800	0.1818
${\psi'_3}^*[k]$	0	0	0.3750	0.1753	0.2973	0.2195	0.2697	0.2372	0.2583	0.2446	0.2535	0.2500
${\psi'_5}^*[k]$	0	0	0.2500	0.4091	0.2513	0.3628	0.2890	0.3371	0.3059	0.3261	0.3130	0.3182

$DR(\overline{F})$ resembles $DR(\overline{E})$, and $TS^*(\overline{F})$ is similar to $TS^*(\overline{E})$. We have $DTMC^*(\overline{F}) = DTMC^*(\overline{E})$. Thus, the TPM and the steady-state PMF for $DTMC^*(\overline{F})$ and $DTMC^*(\overline{E})$ coincide.

The first performance index and the second group of the indices are the same for the standard and the abstract systems. Let us consider the following performance index based on non-personalized viewpoint to the philosophers.

- The beginning of eating of a philosopher $(\{b\}, \frac{1}{4})$ is only possible from the states $s_2, s_3, s_6, s_7, s_{10}, s_{11}$. In each of the states the beginning of eating probability is the sum of the execution probabilities for all multisets of activities containing $(\{b\}, \frac{1}{4})$. Thus, the *steady-state probability of the beginning of eating of a philosopher* is
$\psi_2^* \sum_{\{\Gamma|(\{b\},\frac{1}{4})\in\Gamma\}} PT^*(\Gamma, s_2) + \psi_3^* \sum_{\{\Gamma|(\{b\},\frac{1}{4})\in\Gamma\}} PT^*(\Gamma, s_3) + \psi_6^* \sum_{\{\Gamma|(\{b\},\frac{1}{4})\in\Gamma\}} PT^*(\Gamma, s_6) +$
$\psi_7^* \sum_{\{\Gamma|(\{b\},\frac{1}{4})\in\Gamma\}} PT^*(\Gamma, s_7) + \psi_{10}^* \sum_{\{\Gamma|(\{b\},\frac{1}{4})\in\Gamma\}} PT^*(\Gamma, s_{10}) + \psi_{11}^* \sum_{\{\Gamma|(\{b\},\frac{1}{4})\in\Gamma\}} PT^*(\Gamma, s_{11}) =$
$\frac{2}{11}\left(\frac{3}{20} + \frac{1}{20} + \frac{3}{20} + \frac{1}{20} + \frac{3}{20} + \frac{1}{20} + \frac{3}{20} + \frac{1}{20} + \frac{3}{20} + \frac{1}{20}\right) + \frac{1}{4}\left(\frac{3}{11} + \frac{1}{11} + \frac{3}{11} + \frac{1}{11}\right) + \frac{1}{4}\left(\frac{3}{11} + \frac{1}{11} + \frac{3}{11} + \frac{1}{11}\right) +$
$\frac{1}{4}\left(\frac{3}{11} + \frac{1}{11} + \frac{3}{11} + \frac{1}{11}\right) + \frac{1}{4}\left(\frac{3}{11} + \frac{1}{11} + \frac{3}{11} + \frac{1}{11}\right) + \frac{1}{4}\left(\frac{3}{11} + \frac{1}{11} + \frac{3}{11} + \frac{1}{11}\right) = \frac{6}{11}$.

The marked dts-boxes corresponding to the dynamic expressions of the standard and the abstract dining philosophers are similar as well as the marked dts-boxes corresponding to the dynamic expression of the standard and the abstract dining philosophers systems.

Let us consider a reduction of the abstract dining philosophers system. The static expression of the philosopher 1 is $F'_1 = [(\{x\}, \frac{1}{2}) * ((\{b\}, \frac{2}{5}); (\{e\}, \frac{1}{4})) * \mathsf{Stop}]$. The static expression of the philosopher 2 is $F'_2 = [(\{a, \hat{x}\}, \frac{1}{16}) * ((\{b\}, \frac{2}{5}); (\{e\}, \frac{1}{4})) * \mathsf{Stop}]$. The static expression of the reduced abstract dining philosophers system is $F' = (F'_1 \| F'_2)\ \mathsf{sy}\ x\ \mathsf{rs}\ x$.

$DR(\overline{F'})$ consists of the equivalence classes
$s'_1 = [([\overline{(\{x\}, \frac{1}{2})} * ((\{b\}, \frac{2}{5})_1; (\{e\}, \frac{1}{4})_1) * \mathsf{Stop}] \| [\overline{(\{a, \hat{x}\}, \frac{1}{16})} * ((\{b\}, \frac{2}{5})_2; (\{e\}, \frac{1}{4})_2) * \mathsf{Stop}])\ \mathsf{sy}\ x\ \mathsf{rs}\ x]_\approx$,
$s'_2 = [([(\{x\}, \frac{1}{2}) * (\overline{(\{b\}, \frac{2}{5})_1}; (\{e\}, \frac{1}{4})_1) * \mathsf{Stop}] \| [(\{a, \hat{x}\}, \frac{1}{16}) * (\overline{(\{b\}, \frac{2}{5})_2}; (\{e\}, \frac{1}{4})_2) * \mathsf{Stop}])\ \mathsf{sy}\ x\ \mathsf{rs}\ x]_\approx$,
$s'_3 = [([(\{x\}, \frac{1}{2}) * ((\{b\}, \frac{2}{5})_1; \overline{(\{e\}, \frac{1}{4})_1}) * \mathsf{Stop}] \| [(\{a, \hat{x}\}, \frac{1}{16}) * (\overline{(\{b\}, \frac{2}{5})_2}; (\{e\}, \frac{1}{4})_2) * \mathsf{Stop}])\ \mathsf{sy}\ x\ \mathsf{rs}\ x]_\approx$,
$s'_4 = [([(\{x\}, \frac{1}{2}) * (\overline{(\{b\}, \frac{2}{5})_1}; (\{e\}, \frac{1}{4})_1) * \mathsf{Stop}] \| [(\{a, \hat{x}\}, \frac{1}{16}) * ((\{b\}, \frac{2}{5})_2; \overline{(\{e\}, \frac{1}{4})_2}) * \mathsf{Stop}])\ \mathsf{sy}\ x\ \mathsf{rs}\ x]_\approx$,
$s'_5 = [([(\{x\}, \frac{1}{2}) * ((\{b\}, \frac{2}{5})_1; \overline{(\{e\}, \frac{1}{4})_1}) * \mathsf{Stop}] \| [(\{a, \hat{x}\}, \frac{1}{16}) * ((\{b\}, \frac{2}{5})_2; \overline{(\{e\}, \frac{1}{4})_2}) * \mathsf{Stop}])\ \mathsf{sy}\ x\ \mathsf{rs}\ x]_\approx$.

The states are interpreted as follows: s'_1 is the initial state, s'_2: the system is activated and no philosophers dine, s'_3, s'_4: one philosopher dines, s'_5: two philosophers dine.

We have $\overline{F} \underline{\leftrightarrow}_{ss} \overline{F'}$ with $(DR(\overline{F}) \cup DR(\overline{F'}))/_{\underline{\leftrightarrow}_{ss}} = \{\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3, \mathcal{H}_4\}$, where $\mathcal{H}_1 = \{s_1, s'_1\}$ (the initial state), $\mathcal{H}_2 = \{s_2, s'_2\}$ (the system is activated and no philosophers dine), $\mathcal{H}_3 = \{s_3, s_6, s_7, s_{10}, s_{11}, s'_3, s'_4\}$ (one philosopher dines), $\mathcal{H}_4 = \{s_4, s_5, s_8, s_9, s_{12}, s'_5\}$ (two philosophers dine). One can see that F' is a reduction of F with respect to $\underline{\leftrightarrow}_{ss}$.

In Figure 39 the transition system without empty loops $TS^*(\overline{F'})$ is presented. In Figure 40 the underlying DTMC without empty loops $DTMC^*(\overline{F'})$ is depicted.

The TPM for $DTMC^*(\overline{F'})$ is

$$\mathbf{P}'^* = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{3}{8} & \frac{3}{8} & \frac{1}{4} \\ 0 & \frac{3}{11} & 0 & \frac{2}{11} & \frac{6}{11} \\ 0 & \frac{3}{11} & \frac{2}{11} & 0 & \frac{6}{11} \\ 0 & \frac{1}{7} & \frac{3}{7} & \frac{3}{7} & 0 \end{bmatrix}.$$

In Table 7 the transient and the steady-state probabilities ${\psi'_i}^*[k]$ $(i \in \{1, 2, 3, 5\})$ of the reduced abstract dining philosophers system at the time moments k $(0 \le k \le 10)$ and $k = \infty$ are presented, and in Figure 41 the alteration diagram (evolution in time) for the transient probabilities is depicted. It is sufficient to consider the probabilities for the states s'_1, s'_2, s'_3, s'_5 only, since the corresponding values coincide for s'_3, s'_4.

The steady-state PMF for $DTMC^*(\overline{F'})$ is

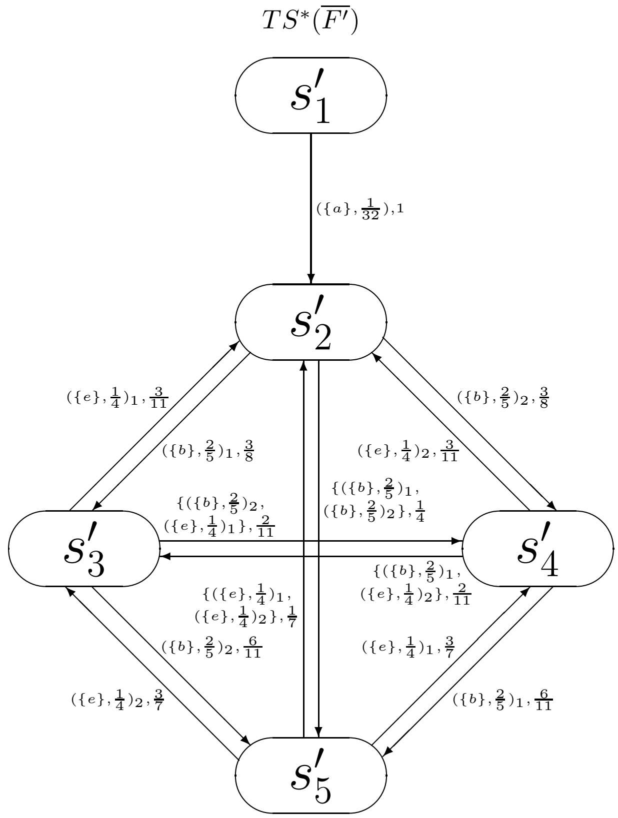

Figure 39: The transition system without empty loops of the reduced abstract dining philosophers system

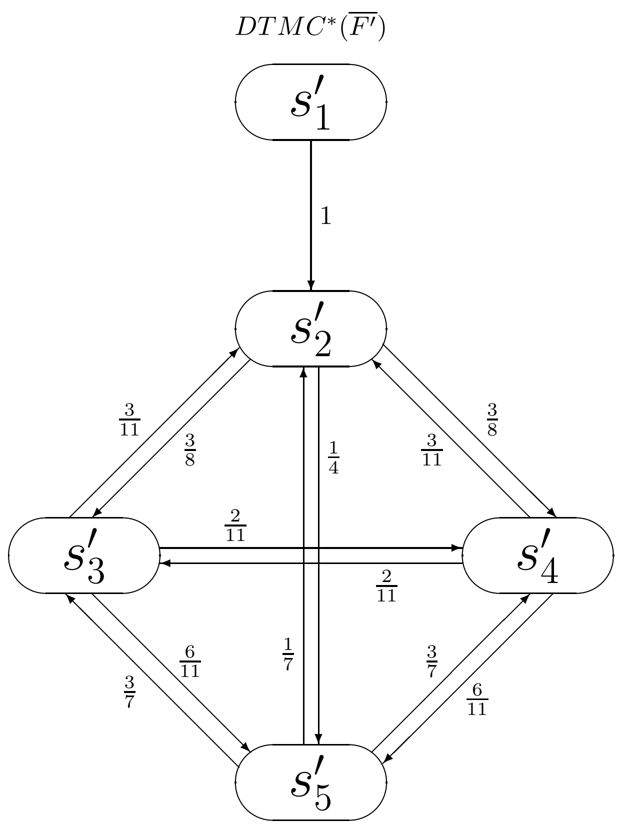

Figure 40: The underlying DTMC without empty loops of the reduced abstract dining philosophers system

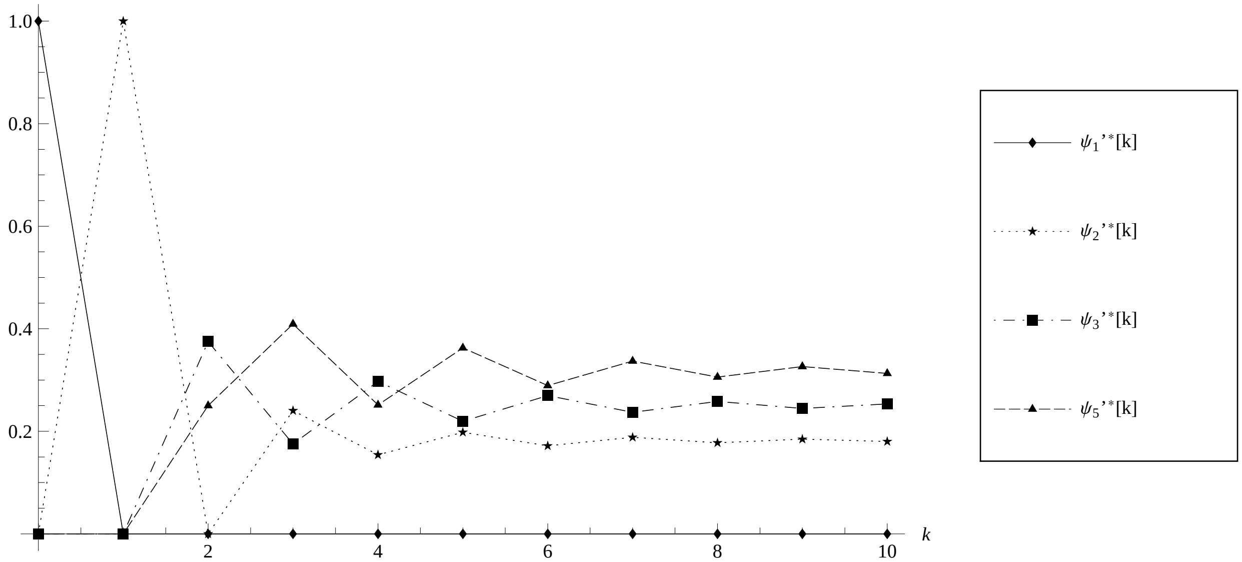

Figure 41: Transient probabilities alteration diagram of the reduced abstract dining philosophers system

$$\psi'^* = \left(0, \frac{2}{11}, \frac{1}{4}, \frac{1}{4}, \frac{7}{22}\right).$$

We can now calculate the main performance indices.

- The average recurrence time in the state s_2', where all the forks are available, called the *average system run-through*, is $\frac{1}{\psi_2'^*} = \frac{11}{2} = 5\frac{1}{2}$.
- Nobody eats in the state s_2'. Then, the *fraction of time when no philosophers dine* is $\psi_2'^* = \frac{2}{11}$.

  Only one philosopher eats in the states s_3', s_4'. Then, the *fraction of time when only one philosopher dines* is $\psi_3'^* + \psi_4'^* = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$.

  Two philosophers eat together in the state s_5'. Then, the *fraction of time when two philosophers dine* is $\psi_5'^* = \frac{7}{22}$.

  The *relative fraction of time when two philosophers dine with respect to when only one philosopher dines* is $\frac{7}{22} \cdot \frac{2}{1} = \frac{7}{11}$.
- The beginning of eating of a philosopher $(\{b\}, \frac{2}{5})$ is only possible from the states s_2', s_3', s_4'. In each of the states the beginning of eating probability is the sum of the execution probabilities for all multisets of activities containing $(\{b\}, \frac{2}{5})$. Thus, the *steady-state probability of the beginning of eating of a philosopher* is $\psi_2'^* \sum_{\{\Gamma \mid (\{b\}, \frac{2}{5}) \in \Gamma\}} PT^*(\Gamma, s_2') + \psi_3'^* \sum_{\{\Gamma \mid (\{b\}, \frac{2}{5}) \in \Gamma\}} PT^*(\Gamma, s_3') + \psi_4'^* \sum_{\{\Gamma \mid (\{b\}, \frac{2}{5}) \in \Gamma\}} PT^*(\Gamma, s_4') = \frac{2}{11}\left(\frac{3}{8} + \frac{3}{8} + \frac{1}{4}\right) + \frac{1}{4}\left(\frac{6}{11} + \frac{2}{11}\right) + \frac{1}{4}\left(\frac{6}{11} + \frac{2}{11}\right) = \frac{6}{11}$.

One can see that the performance indices are the same for the complete and the reduced abstract dining philosophers systems. The coincidence of the first performance index as well as the second group of indices obviously illustrates the result of Proposition 8.1. The coincidence of the third performance index is due to the Theorem 8.2: one should just apply its result to the step traces $\{\{b\}\}$, $\{\{b\},\{b\}\}$, $\{\{b\},\{e\}\}$ of the expressions $\overline{F}$ and $\overline{F'}$, and then sum the left and right parts of the three resulting equalities.

In Figure 42 the marked dts-boxes corresponding to the dynamic expressions of the reduced abstract dining philosophers are presented, i.e., $N_i' = Box_{dts}(\overline{F_i'})$ $(1 \le i \le 2)$. In Figure 43 the marked dts-box corresponding to the dynamic expression of the reduced abstract dining philosophers system is depicted, i.e., $N' = Box_{dts}(\overline{F'})$.

Note that $TS^*(\overline{F'})$ can be reduced further by merging the equivalent states s_3' and s_4', thus, it can be transformed into a transition system with four states only. But the resulted reduction of the initial transition system $TS^*(\overline{F})$ will not correspond to some $dtsPBC$ expression anymore.

For the step stochastic autobisimulation equivalence $\overline{F} \underline{\leftrightarrow}_{ss} \overline{F}$ we have $DR(\overline{F})/_{\underline{\leftrightarrow}_{ss}} = \{\mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3, \mathcal{K}_4\}$, where $\mathcal{K}_1 = \{s_1\}$ (the initial state), $\mathcal{K}_2 = \{s_2\}$ (the system is activated and no philosophers dine), $\mathcal{K}_3 = \{s_3, s_6, s_7, s_{10}, s_{11}\}$ (one philosopher dines), $\mathcal{K}_4 = \{s_4, s_5, s_8, s_9, s_{12}\}$ (two philosophers dine).

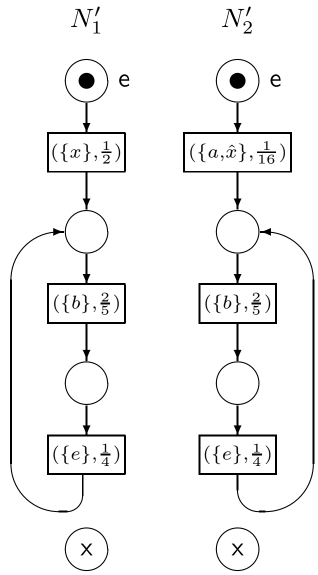

Figure 42: The marked dts-boxes of the reduced abstract dining philosophers

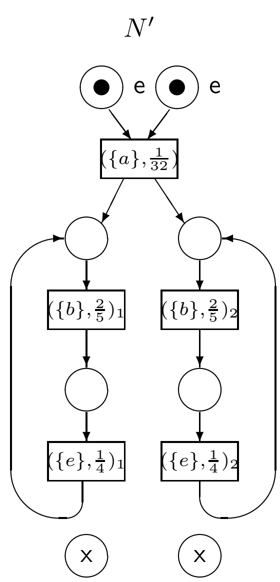

Figure 43: The marked dts-box of the reduced abstract dining philosophers system

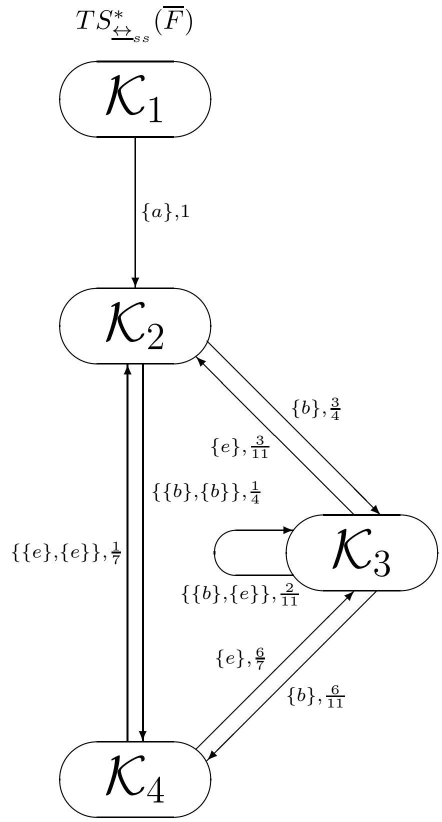

Figure 44: The quotient transition system without empty loops of the abstract dining philosophers system

Table 8: Transient and steady-state probabilities of the quotient abstract dining philosophers system

k	0	1	2	3	4	5	6	7	8	9	10	∞
$\psi_1''^*[k]$	1	0	0	0	0	0	0	0	0	0	0	0
$\psi_2''^*[k]$	0	1	0	0.2403	0.1541	0.1981	0.1716	0.1884	0.1776	0.1846	0.1800	0.1818
$\psi_3''^*[k]$	0	0	0.7500	0.3506	0.5946	0.4391	0.5394	0.4745	0.5165	0.4893	0.5069	0.5000
$\psi_4''^*[k]$	0	0	0.2500	0.4091	0.2513	0.3628	0.2890	0.3371	0.3059	0.3261	0.3130	0.3182

In Figure 44 the quotient transition system without empty loops $TS^*_{\underline{\leftrightarrow}_{ss}}(\overline{F})$ is presented. In Figure 45 the quotient underlying DTMC without empty loops $DTMC^*_{\underline{\leftrightarrow}_{ss}}(\overline{F})$ is depicted.

The TPM for $DTMC^*_{\underline{\leftrightarrow}_{ss}}(\overline{F})$ is

$$\mathbf{P}''^* = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{3}{4} & \frac{1}{4} \\ 0 & \frac{3}{11} & \frac{2}{11} & \frac{6}{11} \\ 0 & \frac{1}{7} & \frac{6}{7} & 0 \end{bmatrix}.$$

In Table 8 the transient and the steady-state probabilities $\psi_i''^*[k]$ $(1 \le i \le 4)$ of the quotient abstract dining philosophers system at the time moments k $(0 \le k \le 10)$ and $k = \infty$ are presented, and in Figure 46 the alteration diagram (evolution in time) for the transient probabilities is depicted.

The steady-state PMF for $DTMC^*_{\underline{\leftrightarrow}_{ss}}(\overline{F})$ is

$$\psi''^* = \left(0, \frac{2}{11}, \frac{1}{2}, \frac{7}{22}\right).$$

We can now calculate the main performance indices.

- The average recurrence time in the state $\mathcal{K}_2$, where all the forks are available, called the *average system run-through*, is $\frac{1}{\psi_2''^*} = \frac{11}{2} = 5\frac{1}{2}$.

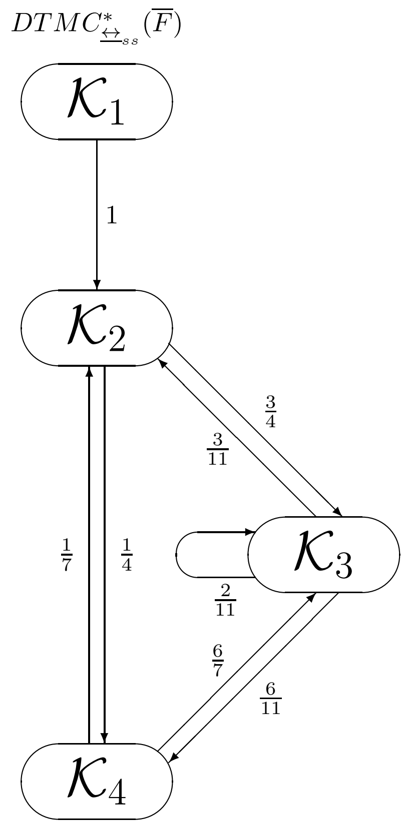

Figure 45: The quotient underlying DTMC without empty loops of the abstract dining philosophers system

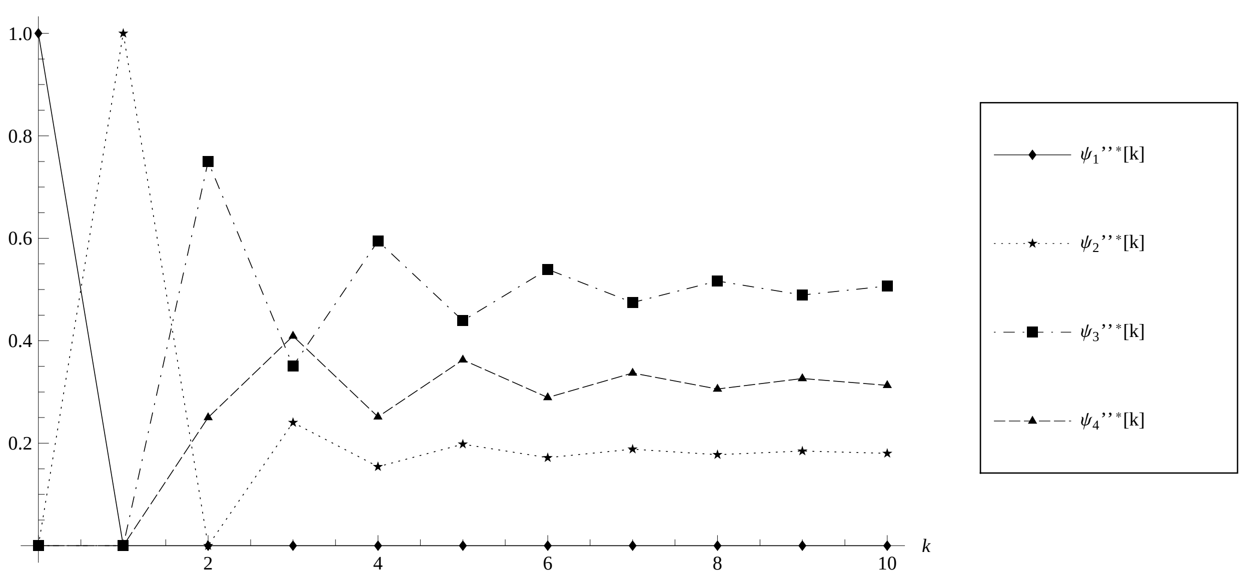

Figure 46: Transient probabilities alteration diagram of the quotient abstract dining philosophers system

- Nobody eats in the state $\mathcal{K}_2$. Then, the *fraction of time when no philosophers dine* is $\psi_2''^{*} = \frac{2}{11}$.

  Only one philosopher eats in the state $\mathcal{K}_3$. Then, the *fraction of time when only one philosopher dines* is $\psi_3''^{*} = \frac{1}{2}$.

  Two philosophers eat together in the state $\mathcal{K}_4$. Then, the *fraction of time when two philosophers dine* is $\psi_4''^{*} = \frac{7}{22}$.

  The *relative fraction of time when two philosophers dine with respect to when only one philosopher dines* is $\frac{7}{22} \cdot \frac{2}{1} = \frac{7}{11}$.

- The beginning of eating of a philosopher $\{b\}$ is only possible from the states $\mathcal{K}_2, \mathcal{K}_3$. In each of the states the beginning of eating probability is the sum of the execution probabilities for all multisets of multiactions containing $\{b\}$. Thus, the *steady-state probability of the beginning of eating of a philosopher* is $\psi_2''^{*} \sum_{\{A,\widetilde{\mathcal{K}} \mid \{b\} \in A,\ \mathcal{K}_2 \xrightarrow{A} \widetilde{\mathcal{K}}\}} PM_A^*(\mathcal{K}_2, \widetilde{\mathcal{K}}) + \psi_3''^{*} \sum_{\{A,\widetilde{\mathcal{K}} \mid \{b\} \in A,\ \mathcal{K}_3 \xrightarrow{A} \widetilde{\mathcal{K}}\}} PM_A^*(\mathcal{K}_3, \widetilde{\mathcal{K}}) = \frac{2}{11}\left(\frac{3}{4} + \frac{1}{4}\right) + \frac{1}{2}\left(\frac{6}{11} + \frac{2}{11}\right) = \frac{6}{11}$.

One can see that the performance indices are the same for the complete and the quotient abstract dining philosophers systems. The explanation of this fact is just the same as that presented earlier for the complete and the reduced abstract dining philosophers systems.

10.2.3 The generalized system

An interesting problem is to find out which influence to performance have the multiaction probabilities from the specification E of the dining philosophers system. Suppose that all the mentioned multiactions have the same probability ρ. The resulting specification K of the generalized dining philosophers system is defined as follows.

The static expression of the philosopher i $(1 \le i \le 4)$ is $K_i = [(\{x_i\}, \rho) * (((\{b_i, \widehat{y_i}\}, \rho); (\{e_i, \widehat{z_i}\}, \rho))[] ((\{y_{i+1}\}, \rho); (\{z_{i+1}\}, \rho))) * \mathsf{Stop}]$. The static expression of the philosopher 5 is $K_5 = [(\{a, \widehat{x_1}, \widehat{x_2}, \widehat{x_2}, \widehat{x_4}\}, \rho) * (((\{b_5, \widehat{y_5}\}, \rho); (\{e_5, \widehat{z_5}\}, \rho))[]((\{y_1\}, \rho); (\{z_1\}, \rho))) * \mathsf{Stop}]$. The static expression of the generalized dining philosophers system is $K = (K_1 \| K_2 \| K_3 \| K_4 \| K_5)$ $\mathsf{sy}\ x_1$ $\mathsf{sy}\ x_2$ $\mathsf{sy}\ x_3$ $\mathsf{sy}\ x_4$ $\mathsf{sy}\ y_1$ $\mathsf{sy}\ y_2$ $\mathsf{sy}\ y_3$ $\mathsf{sy}\ y_4$ $\mathsf{sy}\ y_5$ $\mathsf{sy}\ z_1$ $\mathsf{sy}\ z_2$ $\mathsf{sy}\ z_3$ $\mathsf{sy}\ z_4$ $\mathsf{sy}\ z_5$ $\mathsf{rs}\ x_1$ $\mathsf{rs}\ x_2$ $\mathsf{rs}\ x_3$ $\mathsf{rs}\ x_4$ $\mathsf{rs}\ y_1$ $\mathsf{rs}\ y_2$ $\mathsf{rs}\ y_3$ $\mathsf{rs}\ y_4$ $\mathsf{rs}\ y_5$ $\mathsf{rs}\ z_1$ $\mathsf{rs}\ z_2$ $\mathsf{rs}\ z_3$ $\mathsf{rs}\ z_4$ $\mathsf{rs}\ z_5$.

$DR(\overline{K})$ consists of the 12 states which are interpreted as follows: $\tilde{s}_1$ is the initial state, $\tilde{s}_2$: the system is activated and no philosophers dine, $\tilde{s}_3$: philosopher 1 dines, $\tilde{s}_4$: philosophers 1 and 4 dine, $\tilde{s}_5$: philosophers 1 and 3 dine, $\tilde{s}_6$: philosopher 4 dines, $\tilde{s}_7$: philosopher 3 dines, $\tilde{s}_8$: philosophers 2 and 4 dine, $\tilde{s}_9$: philosophers 3 and 5 dine, $\tilde{s}_{10}$: philosopher 2 dines, $\tilde{s}_{11}$: philosopher 5 dines, $\tilde{s}_{12}$: philosophers 2 and 5 dine.

The TPM for $DTMC^*(\overline{K})$ is

$$\widetilde{\mathbf{P}}^* = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1-\rho^2}{5} & \frac{\rho^2}{5} & \frac{\rho^2}{5} & \frac{1-\rho^2}{5} & \frac{1-\rho^2}{5} & \frac{\rho^2}{5} & \frac{\rho^2}{5} & \frac{1-\rho^2}{5} & \frac{1-\rho^2}{5} & \frac{\rho^2}{5} \\
0 & \frac{1-\rho^2}{3-\rho^2} & 0 & \frac{1-\rho^2}{3-\rho^2} & \frac{1-\rho^2}{3-\rho^2} & \frac{\rho^2}{3-\rho^2} & \frac{\rho^2}{3-\rho^2} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{\rho^2}{2-\rho^2} & \frac{1-\rho^2}{2-\rho^2} & 0 & 0 & \frac{1-\rho^2}{2-\rho^2} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{\rho^2}{2-\rho^2} & \frac{1-\rho^2}{2-\rho^2} & 0 & 0 & 0 & \frac{1-\rho^2}{2-\rho^2} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1-\rho^2}{3-\rho^2} & \frac{\rho^2}{3-\rho^2} & \frac{1-\rho^2}{3-\rho^2} & 0 & 0 & 0 & \frac{1-\rho^2}{3-\rho^2} & 0 & \frac{\rho^2}{3-\rho^2} & 0 & 0 \\
0 & \frac{1-\rho^2}{3-\rho^2} & \frac{\rho^2}{3-\rho^2} & 0 & \frac{1-\rho^2}{3-\rho^2} & 0 & 0 & 0 & \frac{1-\rho^2}{3-\rho^2} & 0 & \frac{\rho^2}{3-\rho^2} & 0 \\
0 & \frac{\rho^2}{2-\rho^2} & 0 & 0 & 0 & \frac{1-\rho^2}{2-\rho^2} & 0 & 0 & 0 & \frac{1-\rho^2}{2-\rho^2} & 0 & 0 \\
0 & \frac{\rho^2}{2-\rho^2} & 0 & 0 & 0 & 0 & \frac{1-\rho^2}{2-\rho^2} & 0 & 0 & 0 & \frac{1-\rho^2}{2-\rho^2} & 0 \\
0 & \frac{1-\rho^2}{3-\rho^2} & 0 & 0 & 0 & \frac{\rho^2}{3-\rho^2} & 0 & \frac{1-\rho^2}{3-\rho^2} & 0 & 0 & \frac{\rho^2}{3-\rho^2} & \frac{1-\rho^2}{3-\rho^2} \\
0 & \frac{1-\rho^2}{3-\rho^2} & 0 & 0 & 0 & 0 & \frac{\rho^2}{3-\rho^2} & 0 & \frac{1-\rho^2}{3-\rho^2} & \frac{\rho^2}{3-\rho^2} & 0 & \frac{1-\rho^2}{3-\rho^2} \\
0 & \frac{\rho^2}{2-\rho^2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1-\rho^2}{2-\rho^2} & \frac{1-\rho^2}{2-\rho^2} & 0
\end{bmatrix}.$$

The steady-state PMF for $DTMC^*(\overline{K})$ is

$$\tilde{\psi}^* = \left(0, \frac{1}{2(3-\rho^2)}, \frac{1}{10}, \frac{2-\rho^2}{10(3-\rho^2)}, \frac{2-\rho^2}{10(3-\rho^2)}, \frac{1}{10}, \frac{1}{10}, \frac{2-\rho^2}{10(3-\rho^2)}, \frac{2-\rho^2}{10(3-\rho^2)}, \frac{1}{10}, \frac{1}{10}, \frac{2-\rho^2}{10(3-\rho^2)}\right).$$

We can now calculate the main performance indices.

- The average recurrence time in the state $\tilde{s}_2$, where all the forks are available, called the *average system run-through*, is $\frac{1}{\tilde{\psi}_2^*} = 2(3-\rho^2)$.

- Nobody eats in the state $\tilde{s}_2$. Then, the *fraction of time when no philosophers dine* is $\tilde{\psi}_2^* = \frac{1}{2(3-\rho^2)}$.

  Only one philosopher eats in the states $\tilde{s}_3, \tilde{s}_6, \tilde{s}_7, \tilde{s}_{10}, \tilde{s}_{11}$. Then, the *fraction of time when only one philosopher dines* is $\tilde{\psi}_3^* + \tilde{\psi}_6^* + \tilde{\psi}_7^* + \tilde{\psi}_{10}^* + \tilde{\psi}_{11}^* = \frac{1}{10} + \frac{1}{10} + \frac{1}{10} + \frac{1}{10} + \frac{1}{10} = \frac{1}{2}$.

  Two philosophers eat together in the states $\tilde{s}_4, \tilde{s}_5, \tilde{s}_8, \tilde{s}_9, \tilde{s}_{12}$. Then, the *fraction of time when two philosophers dine* is $\tilde{\psi}_4^* + \tilde{\psi}_5^* + \tilde{\psi}_8^* + \tilde{\psi}_9^* + \tilde{\psi}_{12}^* = \frac{2-\rho^2}{10(3-\rho^2)} + \frac{2-\rho^2}{10(3-\rho^2)} + \frac{2-\rho^2}{10(3-\rho^2)} + \frac{2-\rho^2}{10(3-\rho^2)} + \frac{2-\rho^2}{10(3-\rho^2)} = \frac{2-\rho^2}{2(3-\rho^2)}$.

  The *relative fraction of time when two philosophers dine with respect to when only one philosopher dines* is $\frac{2-\rho^2}{2(3-\rho^2)} \cdot \frac{2}{1} = \frac{2-\rho^2}{3-\rho^2}$.

- The beginning of eating of first philosopher $(\{b_1\}, \rho^2)$ is only possible from the states $\tilde{s}_2, \tilde{s}_6, \tilde{s}_7$. In each of the states the beginning of eating probability is the sum of the execution probabilities for all multisets of activities containing $(\{b_1\}, \rho^2)$. Thus, the *steady-state probability of the beginning of eating of first philosopher* is $\tilde{\psi}_2^* \sum_{\{\Gamma|(\{b_1\},\rho^2)\in\Gamma\}} PT^*(\Gamma, \tilde{s}_2) + \tilde{\psi}_6^* \sum_{\{\Gamma|(\{b_1\},\rho^2)\in\Gamma\}} PT^*(\Gamma, \tilde{s}_6) + \tilde{\psi}_7^* \sum_{\{\Gamma|(\{b_1\},\rho^2)\in\Gamma\}} PT^*(\Gamma, \tilde{s}_7) = \frac{1}{2(3-\rho^2)}\left(\frac{1-\rho^2}{5} + \frac{\rho^2}{5} + \frac{\rho^2}{5}\right) + \frac{1}{10}\left(\frac{1-\rho^2}{3-\rho^2} + \frac{\rho^2}{3-\rho^2}\right) + \frac{1}{10}\left(\frac{1-\rho^2}{3-\rho^2} + \frac{\rho^2}{3-\rho^2}\right) = \frac{3+\rho^2}{10(3-\rho^2)}$.

Let us consider a modification of the generalized dining philosophers system with abstraction from personalities. We call this system the abstract generalized dining philosophers one.

The static expression of the philosopher i $(1 \le i \le 4)$ is $L_i = [(\{x_i\}, \rho) * (((\{b, \widehat{y_i}\}, \rho); (\{e, \widehat{z_i}\}, \rho))[]((\{y_{i+1}\}, \rho); (\{z_{i+1}\}, \rho))) * \mathsf{Stop}]$. The static expression of the philosopher 5 is $L_5 = [(\{a, \widehat{x_1}, \widehat{x_2}, \widehat{x_2}, \widehat{x_4}\}, \rho) * (((\{b, \widehat{y_5}\}, \rho); (\{e, \widehat{z_5}\}, \rho))[]((\{y_1\}, \rho); (\{z_1\}, \rho))) * \mathsf{Stop}]$. The static expression of the abstract generalized dining philosophers system is $L = (L_1 \| L_2 \| L_3 \| L_4 \| L_5)$ sy x_1 sy x_2 sy x_3 sy x_4 sy y_1 sy y_2 sy y_3 sy y_4 sy y_5 sy z_1 sy z_2 sy z_3 sy z_4 sy z_5 rs x_1 rs x_2 rs x_3 rs x_4 rs y_1 rs y_2 rs y_3 rs y_4 rs y_5 rs z_1 rs z_2 rs z_3 rs z_4 rs z_5.

$DR(\overline{L})$ resembles $DR(\overline{K})$, and $TS^*(\overline{L})$ is similar to $TS^*(\overline{K})$. We have $DTMC^*(\overline{L}) = DTMC^*(\overline{K})$. Thus, the TPM and the steady-state PMF for $DTMC^*(\overline{L})$ and $DTMC^*(\overline{K})$ coincide.

The first performance index and the second group of the indices are the same for the standard and the abstract generalized systems. Let us consider the following performance index based on non-personalized viewpoint to the philosophers.

- The beginning of eating of a philosopher $(\{b\}, \rho^2)$ is only possible from the states $\tilde{s}_2, \tilde{s}_3, \tilde{s}_6, \tilde{s}_7, \tilde{s}_{10}, \tilde{s}_{11}$. In each of the states the beginning of eating probability is the sum of the execution probabilities for all multisets of activities containing $(\{b\}, \rho^2)$. Thus, the *steady-state probability of the beginning of eating of a philosopher* is
$\tilde{\psi}_2^* \sum_{\{\Gamma|(\{b\},\rho^2)\in\Gamma\}} PT^*(\Gamma, \tilde{s}_2) + \tilde{\psi}_3^* \sum_{\{\Gamma|(\{b\},\rho^2)\in\Gamma\}} PT^*(\Gamma, \tilde{s}_3) + \tilde{\psi}_6^* \sum_{\{\Gamma|(\{b\},\rho^2)\in\Gamma\}} PT^*(\Gamma, \tilde{s}_6) + \tilde{\psi}_7^* \sum_{\{\Gamma|(\{b\},\rho^2)\in\Gamma\}} PT^*(\Gamma, \tilde{s}_7) + \tilde{\psi}_{10}^* \sum_{\{\Gamma|(\{b\},\rho^2)\in\Gamma\}} PT^*(\Gamma, \tilde{s}_{10}) + \tilde{\psi}_{11}^* \sum_{\{\Gamma|(\{b\},\rho^2)\in\Gamma\}} PT^*(\Gamma, \tilde{s}_{11}) = \frac{1}{2(3-\rho^2)}\left(\frac{1-\rho^2}{5} + \frac{\rho^2}{5} + \frac{1-\rho^2}{5} + \frac{\rho^2}{5} + \frac{1-\rho^2}{5} + \frac{\rho^2}{5} + \frac{1-\rho^2}{5} + \frac{\rho^2}{5} + \frac{1-\rho^2}{5} + \frac{\rho^2}{5}\right) + \frac{1}{10}\left(\frac{1-\rho^2}{3-\rho^2} + \frac{\rho^2}{3-\rho^2} + \frac{1-\rho^2}{3-\rho^2} + \frac{\rho^2}{3-\rho^2}\right) + \frac{1}{10}\left(\frac{1-\rho^2}{3-\rho^2} + \frac{\rho^2}{3-\rho^2} + \frac{1-\rho^2}{3-\rho^2} + \frac{\rho^2}{3-\rho^2}\right) + \frac{1}{10}\left(\frac{1-\rho^2}{3-\rho^2} + \frac{\rho^2}{3-\rho^2} + \frac{1-\rho^2}{3-\rho^2} + \frac{\rho^2}{3-\rho^2}\right) + \frac{1}{10}\left(\frac{1-\rho^2}{3-\rho^2} + \frac{\rho^2}{3-\rho^2} + \frac{1-\rho^2}{3-\rho^2} + \frac{\rho^2}{3-\rho^2}\right) + \frac{1}{10}\left(\frac{1-\rho^2}{3-\rho^2} + \frac{\rho^2}{3-\rho^2} + \frac{1-\rho^2}{3-\rho^2} + \frac{\rho^2}{3-\rho^2}\right) = \frac{3}{2(3-\rho^2)}$.

Let us consider a reduction of the abstract generalized dining philosophers system. The static expression of the philosopher 1 is $L_1' = [(\{x\}, \rho) * ((\{b\}, \frac{2\rho^2}{3-\rho^2}); (\{e\}, \rho^2)) * \mathsf{Stop}]$. The static expression of the philosopher 2 is $L_2' = [(\{a, \hat{x}\}, \rho^4) * ((\{b\}, \frac{2\rho^2}{3-\rho^2}); (\{e\}, \rho^2)) * \mathsf{Stop}]$. The static expression of the reduced abstract generalized dining philosophers system is $L' = (L_1' \| L_2')$ sy x rs x.

$DR(\overline{L'})$ consists of the 5 states which are interpreted as follows: $\tilde{s}_1'$ is the initial state, $\tilde{s}_2'$: the system is activated and no philosophers dine, $\tilde{s}_3', \tilde{s}_4'$: one philosopher dines, $\tilde{s}_5'$: two philosophers dine.

We have $\overline{L} \underline{\leftrightarrow}_{ss} \overline{L'}$ with $(DR(\overline{L}) \cup DR(\overline{L'}))/_{\underline{\leftrightarrow}_{ss}} = \{\widetilde{\mathcal{H}}_1, \widetilde{\mathcal{H}}_2, \widetilde{\mathcal{H}}_3, \widetilde{\mathcal{H}}_4\}$, where $\widetilde{\mathcal{H}}_1 = \{\tilde{s}_1, \tilde{s}_1'\}$ (the initial state), $\widetilde{\mathcal{H}}_2 = \{\tilde{s}_2, \tilde{s}_2'\}$ (the system is activated and no philosophers dine), $\widetilde{\mathcal{H}}_3 = \{\tilde{s}_3, \tilde{s}_6, \tilde{s}_7, \tilde{s}_{10}, \tilde{s}_{11}, \tilde{s}_3', \tilde{s}_4'\}$ (one philosopher dines), $\widetilde{\mathcal{H}}_4 = \{\tilde{s}_4, \tilde{s}_5, \tilde{s}_8, \tilde{s}_9, \tilde{s}_{12}, \tilde{s}_5'\}$ (two philosophers dine). One can see that L' is a reduction of L with respect to $\underline{\leftrightarrow}_{ss}$.

The TPM for $DTMC^*(\overline{L'})$ is

$$\widetilde{\mathbf{P}}'^* = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{1-\rho^2}{2} & \frac{1-\rho^2}{2} & \rho^2 \\ 0 & \frac{1-\rho^2}{3-\rho^2} & 0 & \frac{2\rho^2}{3-\rho^2} & \frac{2(1-\rho^2)}{3-\rho^2} \\ 0 & \frac{1-\rho^2}{3-\rho^2} & \frac{2\rho^2}{3-\rho^2} & 0 & \frac{2(1-\rho^2)}{3-\rho^2} \\ 0 & \frac{\rho^2}{2-\rho^2} & \frac{1-\rho^2}{2-\rho^2} & \frac{1-\rho^2}{2-\rho^2} & 0 \end{bmatrix}.$$

The steady-state PMF for $DTMC^*(\overline{L'})$ is

$$\tilde{\psi}'^* = \left(0, \frac{1}{2(3-\rho^2)}, \frac{1}{4}, \frac{1}{4}, \frac{2-\rho^2}{2(3-\rho^2)}\right).$$

We can now calculate the main performance indices.

- The average recurrence time in the state $\tilde{s}'_2$, where all the forks are available, called the *average system run-through*, is $\frac{1}{\tilde{\psi}'^*_2} = 2(3-\rho^2)$.

- Nobody eats in the state $\tilde{s}'_2$. Then, the *fraction of time when no philosophers dine* is $\tilde{\psi}'^*_2 = \frac{1}{2(3-\rho^2)}$.

  Only one philosopher eats in the states $\tilde{s}'_3, \tilde{s}'_4$. Then, the *fraction of time when only one philosopher dines* is $\tilde{\psi}'^*_3 + \tilde{\psi}'^*_4 = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$.

  Two philosophers eat together in the state $\tilde{s}'_5$. Then, the *fraction of time when two philosophers dine* is $\tilde{\psi}'^*_5 = \frac{2-\rho^2}{2(3-\rho^2)}$.

  The *relative fraction of time when two philosophers dine with respect to when only one philosopher dines* is $\frac{2-\rho^2}{2(3-\rho^2)} \cdot \frac{2}{1} = \frac{2-\rho^2}{3-\rho^2}$.

- The beginning of eating of a philosopher $(\{b\}, \frac{2\rho^2}{3-\rho^2})$ is only possible from the states $\tilde{s}'_2, \tilde{s}'_3, \tilde{s}'_4$. In each of the states the beginning of eating probability is the sum of the execution probabilities for all multisets of activities containing $(\{b\}, \frac{2\rho^2}{3-\rho^2})$. Thus, the *steady-state probability of the beginning of eating of a philosopher* is $\tilde{\psi}'^*_2 \sum_{\{\Gamma|(\{b\},\frac{2\rho^2}{3-\rho^2})\in\Gamma\}} PT^*(\Gamma, \tilde{s}'_2) + \tilde{\psi}'^*_3 \sum_{\{\Gamma|(\{b\},\frac{2\rho^2}{3-\rho^2})\in\Gamma\}} PT^*(\Gamma, \tilde{s}'_3) + \tilde{\psi}'^*_4 \sum_{\{\Gamma|(\{b\},\frac{2\rho^2}{3-\rho^2})\in\Gamma\}} PT^*(\Gamma, \tilde{s}'_4) = \frac{1}{2(3-\rho^2)}\left(\frac{1-\rho^2}{2} + \frac{1-\rho^2}{2} + \rho^2\right) + \frac{1}{4}\left(\frac{2(1-\rho^2)}{3-\rho^2} + \frac{2\rho^2}{3-\rho^2}\right) + \frac{1}{4}\left(\frac{2(1-\rho^2)}{3-\rho^2} + \frac{2\rho^2}{3-\rho^2}\right) = \frac{3}{2(3-\rho^2)}$.

One can see that the performance indices are the same for the complete and the reduced abstract generalized dining philosophers systems. The coincidence of the first performance index as well as the second group of indices obviously illustrates the result of Proposition 8.1. The coincidence of the third performance index is due to the Theorem 8.2: one should just apply its result to the step traces $\{\{b\}\}$, $\{\{b\},\{b\}\}$, $\{\{b\},\{e\}\}$ of the expressions $\overline{L}$ and $\overline{L'}$, and then sum the left and right parts of the three resulting equalities.

Note that $TS^*(\overline{L'})$ can be reduced further by merging the equivalent states $\tilde{s}'_3$ and $\tilde{s}'_4$, thus, it can be transformed into a transition system with four states only. But the resulted reduction of the initial transition system $TS^*(\overline{L})$ will not correspond to some $dtsPBC$ expression anymore.

For the step stochastic autobisimulation equivalence $\overline{L}\underline{\leftrightarrow}_{ss}\overline{L}$ we have $DR(\overline{L})/_{\underline{\leftrightarrow}_{ss}} = \{\widetilde{\mathcal{K}}_1, \widetilde{\mathcal{K}}_2, \widetilde{\mathcal{K}}_3, \widetilde{\mathcal{K}}_4\}$, where $\widetilde{\mathcal{K}}_1 = \{\tilde{s}_1\}$ (the initial state), $\widetilde{\mathcal{K}}_2 = \{\tilde{s}_2\}$ (the system is activated and no philosophers dine), $\widetilde{\mathcal{K}}_3 = \{\tilde{s}_3, \tilde{s}_6, \tilde{s}_7, \tilde{s}_{10}, \tilde{s}_{11}\}$ (one philosopher dines), $\widetilde{\mathcal{K}}_4 = \{\tilde{s}_4, \tilde{s}_5, \tilde{s}_8, \tilde{s}_9, \tilde{s}_{12}\}$ (two philosophers dine).

The TPM for $DTMC^*_{\underline{\leftrightarrow}_{ss}}(\overline{L})$ is

$$\widetilde{\mathbf{P}}''^* = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1-\rho^2 & \rho^2 \\ 0 & \frac{1-\rho^2}{3-\rho^2} & \frac{2\rho^2}{3-\rho^2} & \frac{2(1-\rho^2)}{3-\rho^2} \\ 0 & \frac{\rho^2}{2-\rho^2} & \frac{2(1-\rho^2)}{2-\rho^2} & 0 \end{bmatrix}.$$

The steady-state PMF for $DTMC^*_{\underline{\leftrightarrow}_{ss}}(\overline{L})$ is

$$\tilde{\psi}''^* = \left(0, \frac{1}{2(3-\rho^2)}, \frac{1}{2}, \frac{2-\rho^2}{2(3-\rho^2)}\right).$$

We can now calculate the main performance indices.

- The average recurrence time in the state $\widetilde{\mathcal{K}}_2$, where all the forks are available, called the *average system run-through*, is $\frac{1}{\tilde{\psi}''^*_2} = 2(3-\rho^2)$.

- Nobody eats in the state $\widetilde{\mathcal{K}}_2$. Then, the *fraction of time when no philosophers dine* is $\tilde{\psi}''^*_2 = \frac{1}{2(3-\rho^2)}$.

  Only one philosopher eats in the state $\widetilde{\mathcal{K}}_3$. Then, the *fraction of time when only one philosopher dines* is $\tilde{\psi}''^*_3 = \frac{1}{2}$.

  Two philosophers eat together in the state $\widetilde{\mathcal{K}}_4$. Then, the *fraction of time when two philosophers dine* is $\tilde{\psi}''^*_4 = \frac{2-\rho^2}{2(3-\rho^2)}$.

The *relative fraction of time when two philosophers dine with respect to when only one philosopher dines* is $\frac{2-\rho^2}{2(3-\rho^2)} \cdot \frac{2}{1} = \frac{2-\rho^2}{3-\rho^2}$.

- The beginning of eating of a philosopher $\{b\}$ is only possible from the states $\widetilde{\mathcal{K}}_2, \widetilde{\mathcal{K}}_3$. In each of the states the beginning of eating probability is the sum of the execution probabilities for all multisets of multiactions containing $\{b\}$. Thus, the *steady-state probability of the beginning of eating of a philosopher* is $\tilde{\psi}_2^{\prime\prime *} \sum_{\{A,\widetilde{\mathcal{K}}|\{b\}\in A,\ \widetilde{\mathcal{K}}_2 \xrightarrow{A} \widetilde{\mathcal{K}}\}} PM_A^*(\widetilde{\mathcal{K}}_2, \widetilde{\mathcal{K}}) + \tilde{\psi}_3^{\prime\prime *} \sum_{\{A,\widetilde{\mathcal{K}}|\{b\}\in A,\ \widetilde{\mathcal{K}}_3 \xrightarrow{A} \widetilde{\mathcal{K}}\}} PM_A^*(\widetilde{\mathcal{K}}_3, \widetilde{\mathcal{K}}) = \frac{1}{2(3-\rho^2)}((1-\rho^2)+\rho^2) + \frac{1}{2}\left(\frac{2(1-\rho^2)}{3-\rho^2} + \frac{2\rho^2}{3-\rho^2}\right) = \frac{3}{2(3-\rho^2)}$.

One can see that the performance indices are the same for the complete and the quotient abstract generalized dining philosophers systems. The explanation of this fact is just the same as that presented earlier for the complete and the reduced abstract generalized dining philosophers systems.

11 Conclusion

In this paper, we have considered a discrete time stochastic extension *dtsPBC* of a finite part of *PBC* enriched with iteration. The calculus has the concurrent step operational semantics based on transition systems and the denotational semantics in terms of a subclass of LDTSPNs. Within the context of *dtsPBC* with iteration, we have defined a number of stochastic algebraic equivalences which have natural net analogues on LDTSPNs. The equivalences abstract from empty loops in transition systems corresponding to dynamic expressions. The diagram of interrelations for the algebraic equivalences has been constructed. We have explained how one can reduce transition systems and DTMCs as well as expressions and dts-boxes modulo the stochastic equivalences. We have presented a logical characterization of stochastic bisimulation equivalences. An application of the equivalences to comparison of stationary behaviour has been demonstrated, and we have found which equivalences from those we proposed guarantee an identity of stationary behaviour. We have proved that the weakest one from the relations having the property is the step stochastic bisimulation equivalence. A congruence relation has been proposed. Case studies of performance evaluation in the framework of the calculus have been presented. An advantage of our framework is twofold. First, one can specify in it concurrent composition and synchronization of (multi)actions, whereas this is not possible in classical Markov chains. Second, algebraic formulas represent processes in a more compact way than Petri nets and allow one to apply syntactic transformations and comparisons.

Future work consists in abstracting from the silent activities in the definitions of the equivalences, i.e., from the activities with empty multiaction part. The abstraction from empty loops and that from silent activities could be done in one step as well. The main point here is that we should collect probabilities during such abstractions from an internal activity. As a result, we shall have the algebraic analogues of the net stochastic equivalences from [21, 22]. Moreover, we plan to extend *dtsPBC* with recursion to enhance the specification power of the calculus.

Acknowledgements I would like to thank Eike Best for many advices and encouraging discussions during my research work at Computer Science Department, Carl von Ossietzky University of Oldenburg, Germany.

References

[1] Andova S. *Process algebra with probabilistic choice. Lecture Notes in Computer Science* **1601**, p. 111–129, 1999.

[2] Balbo G. *Introduction to stochastic Petri nets. Lecture Notes in Computer Science* **2090**, p. 84–155, 2001.

[3] Balbo G. *Introduction to generalized stochastic Petri nets. Lecture Notes in Computer Science* **4486**, p. 83–131, 2007.

[4] Best E., Devillers R., Esparza J. *General refinement and recursion operations in the box calculus. Lecture Notes in Computer Science* **665**, p. 130–140, 1993.

[5] Best E., Devillers R., Hall J.G. *The box calculus: a new causal algebra with multi-label communication. Lecture Notes in Computer Science* **609**, p. 21–69, 1992.

[6] Best E., Devillers R., Koutny M. *Petri nets, process algebras and concurrent programming languages. Lecture Notes in Computer Science* **1492**, p. 1–84, 1998, `http://parsys.informatik.uni-oldenburg.de/~best/publications/apnf.ps.gz`.

[7] Best E., Devillers R., Koutny M. *The box algebra: a model of nets and process expressions. Lecture Notes in Computer Science* **1639**, p. 344–363, 1999.

[8] Best E., Devillers R., Koutny M. *Petri net algebra. EATCS Monographs on Theoretical Computer Science*, 378 p., Springer Verlag, 2001.

[9] Best E., Devillers R., Koutny M. *A unified model for nets and process algebras.* In: *Handbook of Process Algebra*, Chapter **14**, p. 873–944, (Bergstra J.A., Ponse A., Smolka S.A., eds.), Elsevier Science B.V., Amsterdam, The Netherlands, 2001, `http://parsys.informatik.uni-oldenburg.de/~best/publications/handbook.ps.gz`.

[10] Best E., Devillers R., Koutny M. *Recursion and Petri nets. Acta Informatica* **37(11–12)**, p. 781–829, 2001, `http://parsys.informatik.uni-oldenburg.de/~best/publications/bdk-recursion.ps.gz`.

[11] Best E., Devillers R., Koutny M. *The box algebra = Petri nets + process expressions. Information and Computation* **178**, p. 44–100, 2002.

[12] Best E. *COSY and its relationship to CSP. Lecture Notes in Computer Science* **255**, p. 416–440, 1986.

[13] Bernardo M., Gorrieri R. *A tutorial on EMPA: a theory of concurrent processes with nondeterminism, priorities, probabilities and time. Theoretical Computer Science* **202**, p. 1–54, July 1998, `http://www.sti.uniurb.it/bernardo/documents/tcs202.pdf`.

[14] Baier C., Hermanns H. *Weak bisimulation for fully probabilistic processes. Lecture Notes in Computer Science* **1254**, p. 119–130, 1997, `http://web.informatik.uni-bonn.de/l/baier/papers/BH97.ps`.

[15] Baier C., Hermanns H., Katoen J.-P., Wolf V. *Comparative branching-time semantics for Markov chains. Lecture Notes in Computer Science* **2761**, p. 492–507, 2003, `http://www.inf.tu-dresden.de/content/institutes/thi/algi/publikationen/texte/11_00.pdf`.

[16] Buchholz P., Kemper P. *Quantifying the dynamic behavior of process algebras. Lecture Notes in Computer Science* **2165**, p. 184–199, 2001.

[17] Best E., Koutny M. *A refined view of the box algebra. Lecture Notes in Computer Science* **935**, p. 1–20, 1995.

[18] Best E., Koutny M. *Solving recursive net equations. Lecture Notes in Computer Science* **944**, p. 605–623, 1995.

[19] Best E., Linde-Göers H.G. *Compositional process semantics of Petri boxes. Lecture Notes in Computer Science* **802**, p. 250–270, 1993.

[20] Bloom B., Meyer A. *A remark on bisimulation between probabilistic processes. Lecture Notes in Computer Science* **363**, p. 26–40, 1989.

[21] Buchholz P., Tarasyuk I.V. *A class of stochastic Petri nets with step semantics and related equivalence notions. Technische Berichte* **TUD-FI00-12**, 18 p., Fakultät Informatik, Technische Universität Dresden, Germany, November 2000, `ftp://ftp.inf.tu-dresden.de/pub/berichte/tud00-12.ps.gz`.

[22] Buchholz P., Tarasyuk I.V. *Net and algebraic approaches to probablistic modeling. Joint Novosibirsk Computing Center and Institute of Informatics Systems Bulletin, Series Computer Science* **15**, p. 31–64, Novosibirsk, 2001, `http://www.iis.nsk.su/persons/itar/spnpancc.pdf`.

[23] Buchholz P., Tarasyuk I.V. *Equivalences for stochastic Petri nets and stochastic process algebras. Vestnik, Quartal Journal of Novosibirsk State University, Series: Mathematics, Mechanics and Informatics* **6(1)**, p. 14–42, Novosibirsk State University, Novosibirsk, 2006 (in Russian), `http://www.iis.nsk.su/persons/itar/vestnik06.pdf`.

[24] Buchholz P. *Markovian process algebra: composition and equivalence.* In: Herzog U. and Rettelbach M., editors, *Proceedings of 2^{nd} Workshop on Process Algebras and Performance Modelling, Arbeitsberichte des IMMD* **27**, p. 11–30, University of Erlangen, Germany, 1994.

[25] Buchholz P. *Equivalence relations for stochastic automata networks.* In: W.J. Stewart, ed., *Computation with Markov Chains*, p. 197–216, Kluwer Academic Publishers, USA, 1995.

[26] Buchholz P. *A notion of equivalence for stochastic Petri nets.* Lecture Notes in Computer Science **935**, p. 161–180, 1995.

[27] Buchholz P. *Iterative decomposition and aggregation of labeled GSPNs.* Lecture Notes in Computer Science **1420**, p. 226–245, 1998.

[28] Christoff I. *Testing equivalence and fully abstract models of probabilistic processes.* Lecture Notes in Computer Science **458**, p. 126–140, 1990.

[29] Devillers R. *Construction of S-invariants and S-components for refined Petri boxes.* Lecture Notes in Computer Science **691**, p. 242–261, 1993.

[30] Devillers R. *S-invariant analysis of general recursive Petri boxes.* Acta Informatica **32(4)**, p. 313–345, 1995.

[31] Devillers R. *Petri boxes and finite processes.* Lecture Notes in Computer Science **1119**, p. 465–480, 1996.

[32] Donatelli S., Ribaudo M., Hillston J. *A comparison of perfomance evaluation process algebra and generalized stochastic Petri nets.* Proceedings of 6^{th} International Workshop on Petri Nets and Performance Models, Durham, USA, IEEE Computer Society Press, p. 158–168, 1995.

[33] Esparza J., Bruns G. *Trapping mutual exclusion in the box calculus.* Theoretical Computer Science **153**, p. 95–128, 1996.

[34] de Frutos D.E., Marroquín O.A. *Ambient Petri nets.* Electronic Notes in Theoretical Computer Science **85(1)**, 27 p., 2003, `http://www.elsevier.nl/gej-ng/31/29/23/138/47/27/85.1.005.ps`, `http://www.elsevier.nl/locate/entcs/volume85.html`.

[35] Florin G., Natkin S. *Les reseaux de Petri stochastiques.* Technique et Science Informatique **4(1)**, 1985.

[36] van Glabbeek R.J. *The linear time – branching time spectrum II: the semantics of sequential systems with silent moves. Extended abstract.* Lecture Notes in Computer Science **715**, p. 66–81, 1993.

[37] van Glabbeek R.J., Smolka S.A., Steffen B. *Reactive, generative, and stratified models of probabilistic processes.* Information and Computation **121(1)**, p. 59–80, 1995, `http://boole.stanford.edu/pub/prob.ps.gz`.

[38] Hansson H. *Time and probability in formal design of distributed systems.* In: *Real-Time Safety Critical Systems*, Volume **1**, Elsevier, The Netherlands, 1994.

[39] Hillston J. *A compositional approach to performance modelling.* 158 p., Cambridge University Press, Great Britain, 1996, `http://www.dcs.ed.ac.uk/pepa/book.pdf`.

[40] Hesketh M., Koutny M. *An axiomatization of duplication equivalence in the Petri box calculus.* Lecture Notes in Computer Science **1420**, p. 165–184, 1998.

[41] Hoare C.A.R. *Communicating sequential processes.* Prentice-Hall, London, Great Britain, 1985.

[42] Hermanns H., Rettelbach M. *Syntax, semantics, equivalences and axioms for MTIPP.* Proceedings of 2^{nd} Workshop on Process Algebras and Performance Modelling, Regensberg / Erlangen (Herzog U., Rettelbach M., eds.), *Arbeitsberichte des IMMD* **27**, p. 71–88, University of Erlangen, Germany, 1994, `http://ftp.informatik.uni-erlangen.de/local/inf7/papers/Hermanns/syntax_semantics_equivalences_axioms_for_MTIPP.ps.gz`.

[43] Jonsson B., Yi W., Larsen K.G. *Probabilistic extensions of process algebras.* In: *Handbook of Process Algebra*, Chapter **11**, p. 685–710, (Bergstra J.A., Ponse A., Smolka S.A., eds.), Elsevier Science B.V., Amsterdam, The Netherlands, 2001.

[44] Koutny M., Best E. *Operational and denotational semantics for the box algebra.* Theoretical Computer Science **211(1–2)**, p. 1–83, 1999, `http://parsys.informatik.uni-oldenburg.de/~best/publications/tcs.ps.gz`.

[45] Kotov V.E., Cherkasova L.A. *On structural properties of generalized processes.* Lecture Notes in Computer Science **188**, p. 288–306, 1985.

[46] Koutny M., Esparza J., Best E. *Operational semantics for the Petri box calculus.* Lecture Notes in Computer Science **836**, p. 210–225, 1994.

[47] Kotov V.E. *An algebra for parallelism based on Petri nets. Lecture Notes in Computer Science* **64**, p. 39–55, 1978.

[48] Koutny M. *Partial order semantics of box expressions. Lecture Notes in Computer Science* **815**, p. 318–337, 1994.

[49] Koutny M. *A compositional model of time Petri nets. Lecture Notes in Computer Science* **1825**, p. 303–322, 2000.

[50] Larsen K.G., Skou A. *Bisimulation through probabilistic testing. Information and Computation* **94(1)**, p. 1–28, 1991.

[51] Larsen K.G., Skou A. *Compositional verification of probabilistic processes. Lecture Notes in Computer Science* **630**, p. 456–471, 1992.

[52] Macià H.S. *sPBC: Una extensión Markoviana del Petri box calculus. Ph.D. thesis*, 249 p., Departamento de Informática, Universidad de Castilla-La Mancha, Albacete, Spain, December 2003 (in Spanish), `http://www.info-ab.uclm.es/retics/publications/2003/sPBCthesis03.pdf`.

[53] Marsan M.A., Balbo G., Conte G., Donatelli S., Franceschinis G. *Modelling with generalized stochastic Petri nets. Wiley Series in Parallel Computing*, John Wiley and Sons, 316 p., 1995, `http://www.di.unito.it/~greatspn/GSPN-Wiley/`.

[54] Marroquín O.A., de Frutos D.E. *TPBC: timed Petri box calculus. Technical Report*, Departamento de Sistemas Infofmáticos y Programación, Universidad Complutense de Madrid, Madrid, Spain, 2000 (in Spanish).

[55] Marroquín O.A., de Frutos D.E. *Extending the Petri box calculus with time. Lecture Notes in Computer Science* **2075**, p. 303–322, 2001.

[56] Milner R.A.J. *Communication and concurrency.* Prentice-Hall, 260 p., Upper Saddle River, NJ, USA, 1989.

[57] Molloy M. *Performance analysis using stochastic Petri nets. IEEE Transactions on Software Engineering* **31(9)**, p. 913–917, 1982.

[58] Molloy M. *Discrete time stochastic Petri nets. IEEE Transactions on Software Engineering* **11(4)**, p. 417–423, 1985.

[59] Macià H.S., Valero V.R., Cuartero F.G. *A congruence relation in finite sPBC. Technical Report* **DIAB-02-01-31**, 34 p., Department of Computer Science, University of Castilla-La Mancha, Albacete, Spain, October 2002, `http://www.info-ab.uclm.es/retics/publications/2002/tr020131.ps`.

[60] Macià H.S., Valero V.R., Cuartero F.G. *Defining equivalence relations in sPBC. Proceedings of* 1^{st} *International Conference on the Principles of Software Engineering - 04 (PriSE'04)*, p. 195–205, Buenos Aires, Argentina, November 2004, `http://www.info-ab.uclm.es/retics/publications/2004/prise04.pdf`.

[61] Macià H.S., Valero V.R., Cazorla D.L., Cuartero F.G. *Introducing the iteration in sPBC. Technical Report* **DIAB-03-01-37**, 20 p., Department of Computer Science, University of Castilla-La Mancha, Albacete, Spain, September 2003, `http://www.info-ab.uclm.es/descargas/tecnicalreports/DIAB-03-01-37/diab030137.zip`.

[62] Macià H.S., Valero V.R., Cazorla D.L., Cuartero F.G. *Introducing the iteration in sPBC. Proceedings of the* 24^{th} *International Conference on Formal Techniques for Networked and Distributed Systems - 04 (FORTE'04)*, Madrid, Spain, *Lecture Notes in Computer Science* **3235**, p. 292–308, October 2004, `http://www.info-ab.uclm.es/retics/publications/2004/forte04.pdf`.

[63] Macià H.S., Valero V.R., Cuartero F.G., de Frutos D.E. *A congruence relation for sPBC. Formal Methods in System Design* **32(2)**, p. 85–128, Springer, The Netherlands, April 2008.

[64] Macià H.S., Valero V.R., Cuartero F.G., Pelayo F.L. *A new proposal for the synchronization in sPBC. Technical Report* **DIAB-02-01-26**, 15 p., Department of Computer Science, University of Castilla-La Mancha, Albacete, Spain, June 2002, `http://www.info-ab.uclm.es/sec-ab/Tecrep/newproposalsysPBC.ps`.

[65] Macià H.S., Valero V.R., Cuartero F.G., Pelayo F.L. *Improving the synchronization in stochastic Petri box calculus. Actas de las II Jornadas sobre Programacion y Lenguajes - 02 (PROLE'02)*, El Escorial, Spain, November 2002.

[66] Macià H.S., Valero V.R., Cuartero F.G., Pelayo F.L. *A new synchronization in finite stochastic Petri box calculus. Proceedings of 3^{rd} International IEEE Conference on Application of Concurrency to System Design - 03 (ACSD'03)*, p. 216–225, Guimarães, Portugal, IEEE Computer Society Press, June 2003, `http://www.info-ab.uclm.es/retics/publications/2003/acsd03.pdf`.

[67] Macià H.S., Valero V.R., Cuartero F.G., Ruiz M.C.D. *sPBC: a Markovian extension of Petri box calculus with immediate multiactions. Fundamenta Informaticae* **87(3–4)**, p. 367–406, IOS Press, Amsterdam, The Netherlands, 2008.

[68] Macià H.S., Valero V.R., de Frutos D.E. *sPBC: a Markovian extension of finite Petri box calculus. Proceedings of 9^{th} IEEE International Workshop on Petri Nets and Performance Models - 01 (PNPM'01)*, p. 207–216, Aachen, Germany, IEEE Computer Society Press, September 2001, `http://www.info-ab.uclm.es/retics/publications/2001/pnpm01.ps`.

[69] Macià H.S., Valero V.R., de Frutos D.E. *sPBC: a Markovian extension of finite PBC. Actas de IX Jornadas de Concurrencia - 01 (JC'01)*, p. 243–256, Sitges, Spain, June 2001, `http://www.info-ab.uclm.es/retics/publications/2001/mvfjc01.ps`.

[70] Macià H.S., Valero V.R., de Frutos D.E., Cuartero F.G. *Extending PBC with Markovian multiactions. Proceedings of XXVII Conferencia Latinoamericana de Informática - 01 (CLEI'01)* (Montilva, J.A., Besembel, I., eds.), 12 p., Mérida, Venezuela, Universidad de los Andes, September 2001, `http://www.info-ab.uclm.es/retics/publications/2001/clei01.ps`.

[71] Núñez M.G., de Frutos D.E., Llana L.D. *Acceptance trees for probabilistic processes. Lecture Notes in Computer Science* **962**, p. 249–263, 1995.

[72] Niaouris A. *An algebra of Petri nets with arc-based time restrictions. Lecture Notes in Computer Science* **3407**, p. 447–462, 2005.

[73] Niaouris A., Koutny M. *An algebra timed-arc Petri nets. Technical Report Series* **CS-TR-895**, 60 p., School of Computer Science, University of Newcastle upon Tyne, Great Britain, March 2005, `http://www.cs.ncl.ac.uk/publications/trs/papers/895.pdf`.

[74] Núñez M.G. *An axiomatization of probabilistic testing. Lecture Notes in Computer Science* **1601**, p. 130–150, 1999, `http://dalila.sip.ucm.es/miembros/manolo/papers/arts99.ps.gz`.

[75] Peterson J.L. *Petri net theory and modeling of systems.* Prentice-Hall, 1981.

[76] Ribaudo M. *Stochastic Petri net semantics for stochastic process algebra. Proceedings of 6^{th} International Workshop on Petri Nets and Performance Models*, p. 148–157, Durham, NC, USA, IEEE Computer Society Press, 1995.

[77] Tarasyuk I.V. *Equivalence notions applied to designing concurrent systems with the use of Petri nets. Programming and Computer Software* **24(4)**, p. 162–175, Nauka, Moscow, 1998, `http://www.maik.rssi.ru/journals/procom.htm`.

[78] Tarasyuk I.V. *Logical characterization of probabilistic τ-bisimulation equivalences. Joint Novosibirsk Computing Center and Institute of Informatics Systems Bulletin, Series Computer Science* **20**, p. 97–111, Novosibirsk, 2004, `http://www.iis.nsk.su/persons/itar/prlogncc.pdf`.

[79] Tarasyuk I.V. *Stochastic Petri nets: a formalism for modeling and performance analysis of computing processes. System Informatics* **9**, p. 135–194, SB RAS Publisher, Novosibirsk, 2004 (in Russian), `http://www.iis.nsk.su/persons/itar/spnsi.pdf`.

[80] Tarasyuk I.V. *Discrete time stochastic Petri box calculus. Berichte aus dem Department für Informatik* **3/05**, 25 p., Carl von Ossietzky Universität Oldenburg, Germany, November 2005, `http://www.iis.nsk.su/persons/itar/dtspbcib_cov.pdf`.

[81] Tarasyuk I.V. *Iteration in discrete time stochastic Petri box calculus. Bulletin of the Novosibirsk Computing Center, Series Computer Science, IIS Special Issue* **24**, p. 129–148, NCC Publisher, Novosibirsk, 2006, `http://www.iis.nsk.su/persons/itar/dtsitncc.pdf`.

[82] Tarasyuk I.V. *Stochastic Petri box calculus with discrete time. Fundamenta Informaticae* **76(1–2)**, p. 189–218, IOS Press, Amsterdam, The Netherlands, February 2007, http://www.iis.nsk.su/persons/itar/dtspbcfi.pdf.

[83] Tarasyuk I.V. *Equivalences for behavioural analysis of concurrent and distributed computing systems.* 321 p., Academic Publisher "Geo", Novosibirsk, 2007 (ISBN 978-5-9747-0098-9, in Russian).

[84] Tarasyuk I.V. *Investigating equivalence relations in dtsPBC. Berichte aus dem Department für Informatik* **5/08**, 57 p., Carl von Ossietzky Universität Oldenburg, Germany, October 2008, http://www.iis.nsk.su/persons/itar/dtspbcit_cov.pdf.

[85] Tarasyuk I.V. *A notion of congruence for dtsPBC. Bulletin of the Novosibirsk Computing Center, Series Computer Science, IIS Special Issue* **28**, p. 121–141, NCC Publisher, Novosibirsk, 2008, http://www.iis.nsk.su/persons/itar/dtcgncc.pdf.

[86] Tarasyuk I.V. *Performance evaluation in dtsPBC. Proceedings of* 18^{th} *Workshop on Concurrency, Specification and Programming - 09 (CS&P'09)*, Kraków-Przegorzały, Poland, September 28–30, 2009, L. Czaja, M. Szczuka, editors, p. 602–615, Warsaw University, 2009, http://www.iis.nsk.su/persons/itar/dtsshmcsp.pdf.

[87] Tarasyuk I.V. *Modeling and performance analysis of concurrent processes in the algebra dtsPBC. Vestnik, Quartal Journal of Novosibirsk State University, Series: Mathematics, Mechanics, Informatics* **9(4)**, p. 90–117, Novosibirsk State University, Novosibirsk, 2009 (in Russian), http://www.iis.nsk.su/persons/itar/vestnik09.pdf.

[88] Tarasyuk I.V., Macià H.S., Valero V.R. *Discrete time stochastic Petri box calculus with immediate multiactions. Technical Report* **DIAB-10-03-1**, 25 p., Department of Computer Systems, High School of Computer Science Engineering, University of Castilla-La Mancha, Albacete, Spain, March 2010, http://www.dsi.uclm.es/descargas/thecnicalreports/DIAB-10-03-1/dtsipbc.pdf.

[89] Wolf V., Baier C., Majster-Cederbaum M. *Trace machines for observing continuous-time Markov chains. Proceedings of the* 3^{rd} *International Workshop on Quantitative Aspects of Programming Languages (QAPL'05)*, Edinburgh, The Great Britain, 2005, *Electronic Notes in Theoretical Computer Science* **153(2)**, p. 259–277, 2005.

A Proof of Proposition 5.2

It is enough to prove the statement of the proposition for $\star = s$, since $\star = i$ is a particular case of the previous one with one-element multisets of multiactions and interleaving transition relation.

Let $\mathcal{R} : N \underline{\leftrightarrow}_{ss} N'$, $\mathcal{H} \in (DR(G) \cup DR(G'))/_{\mathcal{R}}$ and $s, \bar{s} \in \mathcal{H}$. We have $\forall \widetilde{\mathcal{H}} \in (DR(G) \cup DR(G'))/_{\mathcal{R}}\ \forall A \in \mathbb{N}_f^{\mathcal{L}} \setminus \{\emptyset\}\ s \stackrel{A}{\twoheadrightarrow}_{\mathcal{P}} \widetilde{\mathcal{H}} \ \Leftrightarrow\ \bar{s} \stackrel{A}{\twoheadrightarrow}_{\mathcal{P}} \widetilde{\mathcal{H}}$. The previous equality is valid for all $s, \bar{s} \in \mathcal{H}$, hence, we can rewrite it as $\mathcal{H} \stackrel{A}{\twoheadrightarrow}_{\mathcal{P}} \widetilde{\mathcal{H}}$ and denote $PM_A^*(\mathcal{H}, \widetilde{\mathcal{H}}) = PM_A^*(s, \widetilde{\mathcal{H}}) = PM_A^*(\bar{s}, \widetilde{\mathcal{H}})$. Note that transitions from the states of $DR(G)$ always lead to those from the same set, hence, $\forall s \in DR(G)\ PM_A^*(s, \widetilde{\mathcal{H}}) = PM_A^*(s, \widetilde{\mathcal{H}} \cap DR(G))$. The same is true for $DR(G')$.

Let $(A_1 \cdots A_n, \mathcal{Q}) \in StepStochTraces(G)$. Taking into account the notes above and $\mathcal{R} : G \underline{\leftrightarrow}_{ss} G'$, we have $\forall \mathcal{H}_1, \ldots, \forall \mathcal{H}_n \in (DR(G) \cup DR(G'))/_{\mathcal{R}}\ [G]_{\approx} \stackrel{A_1}{\twoheadrightarrow}_{\mathcal{P}_1} \mathcal{H}_1 \stackrel{A_2}{\twoheadrightarrow}_{\mathcal{P}_2} \cdots \stackrel{A_n}{\twoheadrightarrow}_{\mathcal{P}_n} \mathcal{H}_n \ \Leftrightarrow\ [G']_{\approx} \stackrel{A_1}{\twoheadrightarrow}_{\mathcal{P}_1} \mathcal{H}_1 \stackrel{A_2}{\twoheadrightarrow}_{\mathcal{P}_2} \cdots \stackrel{A_n}{\twoheadrightarrow}_{\mathcal{P}_n} \mathcal{H}_n$.

Now we intend to prove that the sum of probabilities of all the paths starting in $[G]_{\approx}$ and going through the states from $\mathcal{H}_1, \ldots, \mathcal{H}_n$ is equal to the product of $\mathcal{P}_1, \ldots, \mathcal{P}_n$, which is essentially the probability of the "composite" path starting in $\mathcal{H}_0 = [[G]_{\approx}]_{\mathcal{R}}$ and going through the equivalence classes $\mathcal{H}_1, \ldots, \mathcal{H}_n$ in $TS^*(G)$:

$$\sum_{\{\Gamma_1, \ldots, \Gamma_n \mid [G]_{\approx} \stackrel{\Gamma_1}{\twoheadrightarrow} \cdots \stackrel{\Gamma_n}{\twoheadrightarrow} s_n,\ \mathcal{L}(\Gamma_i) = A_i,\ s_i \in \mathcal{H}_i\ (1 \le i \le n)\}} \prod_{i=1}^{n} PT^*(\Gamma_i, s_{i-1}) = \prod_{i=1}^{n} PM_{A_i}^*(\mathcal{H}_{i-1}, \mathcal{H}_i).$$

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

- $n = 1$

  $\sum_{\{\Gamma_1 \mid [G]_{\approx} \stackrel{\Gamma_1}{\twoheadrightarrow} s_1,\ \mathcal{L}(\Gamma_1) = A_1,\ s_1 \in \mathcal{H}_1\}} PT^*(\Gamma_1, [G]_{\approx}) = PM_{A_1}^*([G]_{\approx}, \mathcal{H}_1) = PM_{A_1}^*(\mathcal{H}_0, \mathcal{H}_1).$

- $n \to n+1$

  $\sum_{\{\Gamma_1,\ldots,\Gamma_n,\Gamma_{n+1}|[G]_\approx \overset{\Gamma_1}{\twoheadrightarrow}\cdots\overset{\Gamma_n}{\twoheadrightarrow} s_n \overset{\Gamma_{n+1}}{\twoheadrightarrow} s_{n+1},\ \mathcal{L}(\Gamma_i)=A_i,\ s_i\in\mathcal{H}_i\ (1\le i\le n+1)\}} \prod_{i=1}^{n+1} PT^*(\Gamma_i, s_{i-1}) =$
  $\sum_{\{\Gamma_{n+1}|s_n \overset{\Gamma_{n+1}}{\twoheadrightarrow} s_{n+1},\ \mathcal{L}(\Gamma_{n+1})=A_{n+1},\ s_n\in\mathcal{H}_n,\ s_{n+1}\in\mathcal{H}_{n+1}\}} \sum_{\{\Gamma_1,\ldots,\Gamma_n|[G]_\approx \overset{\Gamma_1}{\twoheadrightarrow}\cdots\overset{\Gamma_n}{\twoheadrightarrow} s_n,\ \mathcal{L}(\Gamma_i)=A_i,\ s_i\in\mathcal{H}_i\ (1\le i\le n)\}}$
  $\prod_{i=1}^{n} PT^*(\Gamma_i, s_{i-1}) PT^*(\Gamma_{n+1}, s_n) =$
  $\sum_{\{\Gamma_1,\ldots,\Gamma_n|[G]_\approx \overset{\Gamma_1}{\twoheadrightarrow}\cdots\overset{\Gamma_n}{\twoheadrightarrow} s_n,\ \mathcal{L}(\Gamma_i)=A_i,\ s_i\in\mathcal{H}_i\ (1\le i\le n)\}}$
  $\left[\prod_{i=1}^{n} PT^*(\Gamma_i, s_{i-1}) \sum_{\{\Gamma_{n+1}|s_n \overset{\Gamma_{n+1}}{\twoheadrightarrow} s_{n+1},\ \mathcal{L}(\Gamma_{n+1})=A_{n+1},\ s_n\in\mathcal{H}_n,\ s_{n+1}\in\mathcal{H}_{n+1}\}} PT^*(\Gamma_{n+1}, s_n)\right] =$
  $\sum_{\{\Gamma_1,\ldots,\Gamma_n|[G]_\approx \overset{\Gamma_1}{\twoheadrightarrow}\cdots\overset{\Gamma_n}{\twoheadrightarrow} s_n,\ \mathcal{L}(\Gamma_i)=A_i,\ s_i\in\mathcal{H}_i\ (1\le i\le n)\}} \prod_{i=1}^{n} PT^*(\Gamma_i, s_{i-1}) PM^*_{A_{n+1}}(s_n, \mathcal{H}_{n+1}) =$
  $\sum_{\{\Gamma_1,\ldots,\Gamma_n|[G]_\approx \overset{\Gamma_1}{\twoheadrightarrow}\cdots\overset{\Gamma_n}{\twoheadrightarrow} s_n,\ \mathcal{L}(\Gamma_i)=A_i,\ s_i\in\mathcal{H}_i\ (1\le i\le n)\}} \prod_{i=1}^{n} PT^*(\Gamma_i, s_{i-1}) PM^*_{A_{n+1}}(\mathcal{H}_n, \mathcal{H}_{n+1}) =$
  $PM^*_{A_{n+1}}(\mathcal{H}_n, \mathcal{H}_{n+1}) \sum_{\{\Gamma_1,\ldots,\Gamma_n|[G]_\approx \overset{\Gamma_1}{\twoheadrightarrow}\cdots\overset{\Gamma_n}{\twoheadrightarrow} s_n,\ \mathcal{L}(\Gamma_i)=A_i,\ s_i\in\mathcal{H}_i\ (1\le i\le n)\}} \prod_{i=1}^{n} PT^*(\Gamma_i, s_{i-1}) =$
  $PM^*_{A_{n+1}}(\mathcal{H}_n, \mathcal{H}_{n+1}) \prod_{i=1}^{n} PM^*_{A_i}(\mathcal{H}_{i-1}, \mathcal{H}_i) = \prod_{i=1}^{n+1} PM^*_{A_i}(\mathcal{H}_{i-1}, \mathcal{H}_i).$

Note that the equality we have just proved can also be applied to G'.

Now we only need to see that the summation over *all multisets of activities* is the same as the summation over *all equivalence classes*: $\mathcal{Q} = \sum_{\{\Gamma_1,\ldots,\Gamma_n|[G]_\approx \overset{\Gamma_1}{\twoheadrightarrow}\cdots\overset{\Gamma_n}{\twoheadrightarrow} s_n,\ \mathcal{L}(\Gamma_i)=A_i,\ (1\le i\le n)\}} \prod_{i=1}^{n} PT^*(\Gamma_i, s_{i-1}) =$
$\sum_{\mathcal{H}_1,\ldots,\mathcal{H}_n} \sum_{\{\Gamma_1,\ldots,\Gamma_n|[G]_\approx \overset{\Gamma_1}{\twoheadrightarrow}\cdots\overset{\Gamma_n}{\twoheadrightarrow} s_n,\ \mathcal{L}(\Gamma_i)=A_i,\ s_i\in\mathcal{H}_i\ (1\le i\le n)\}} \prod_{i=1}^{n} PT^*(\Gamma_i, s_{i-1}) =$
$\sum_{\mathcal{H}_1,\ldots,\mathcal{H}_n} \prod_{i=1}^{n} PM^*_{A_i}(\mathcal{H}_{i-1}, \mathcal{H}_i) =$
$\sum_{\mathcal{H}_1,\ldots,\mathcal{H}_n} \sum_{\{\Gamma'_1,\ldots,\Gamma'_n|[G']_\approx \overset{\Gamma'_1}{\twoheadrightarrow}\cdots\overset{\Gamma'_n}{\twoheadrightarrow} s'_n,\ \mathcal{L}(\Gamma'_i)=A_i,\ s'_i\in\mathcal{H}_i\ (1\le i\le n)\}} \prod_{i=1}^{n} PT^*(\Gamma'_i, s'_{i-1}) =$
$\sum_{\{\Gamma'_1,\ldots,\Gamma'_n|[G']_\approx \overset{\Gamma'_1}{\twoheadrightarrow}\cdots\overset{\Gamma'_n}{\twoheadrightarrow} s'_n,\ \mathcal{L}(\Gamma'_i)=A_i,\ (1\le i\le n)\}} \prod_{i=1}^{n} PT^*(\Gamma'_i, s'_{i-1}).$

Hence, $(A_1 \cdots A_n, \mathcal{Q}) \in StepStochTraces(G')$, and we have $StepStochTraces(G) \subseteq StepStochTraces(G')$. The reverse inclusion is proved by symmetry. □

B Proof of Proposition 8.1

The proof is an extension of results from [26] to the process algebra framework and discrete time case.

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

- $k = 0$

  Note that 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 $\sum_{s\in\mathcal{H}\cap DR(G)} \psi^*[0](s) = 0 = \sum_{s'\in\mathcal{H}\cap DR(G')} \psi'^*[0](s')$.

- $k \to k+1$

  Let $\mathcal{H} \in (DR(G) \cup DR(G'))/_{\mathcal{R}}$ and $s_1, s_2 \in \mathcal{H}$. We have $\forall \widetilde{\mathcal{H}} \in (DR(G) \cup DR(G'))/_{\mathcal{R}}\ \forall A \in \mathbb{N}_f^{\mathcal{L}} \setminus \{\emptyset\}$ $s_1 \overset{A}{\twoheadrightarrow}_{\mathcal{P}} \widetilde{\mathcal{H}} \Leftrightarrow s_2 \overset{A}{\twoheadrightarrow}_{\mathcal{P}} \widetilde{\mathcal{H}}$. Therefore, $PM^*(s_1, \widetilde{\mathcal{H}}) = \sum_{\{\Gamma|\exists \tilde{s}_1\in\widetilde{\mathcal{H}}\ s_1 \overset{\Gamma}{\twoheadrightarrow} \tilde{s}_1\}} PT^*(\Gamma, s_1) =$
  $\sum_{A\in\mathbb{N}_f^{\mathcal{L}}\setminus\{\emptyset\}} \sum_{\{\Gamma|\exists \tilde{s}_1\in\widetilde{\mathcal{H}}\ s_1 \overset{\Gamma}{\twoheadrightarrow} \tilde{s}_1,\ \mathcal{L}(\Gamma)=A\}} PT^*(\Gamma, s_1) = \sum_{A\in\mathbb{N}_f^{\mathcal{L}}\setminus\{\emptyset\}} PM^*_A(s_1, \widetilde{\mathcal{H}}) =$
  $\sum_{A\in\mathbb{N}_f^{\mathcal{L}}\setminus\{\emptyset\}} PM^*_A(s_2, \widetilde{\mathcal{H}}) = \sum_{A\in\mathbb{N}_f^{\mathcal{L}}\setminus\{\emptyset\}} \sum_{\{\Gamma|\exists \tilde{s}_2\in\widetilde{\mathcal{H}}\ s_2 \overset{\Gamma}{\twoheadrightarrow} \tilde{s}_2,\ \mathcal{L}(\Gamma)=A\}} PT^*(\Gamma, s_2) =$
  $\sum_{\{\Gamma|\exists \tilde{s}_2\in\widetilde{\mathcal{H}}\ s_2 \overset{\Gamma}{\twoheadrightarrow} \tilde{s}_2\}} PT^*(\Gamma, s_2) = PM^*(s_2, \widetilde{\mathcal{H}})$. Since we have the previous equality for all $s_1, s_2 \in \mathcal{H}$, we can denote $PM^*(\mathcal{H}, \widetilde{\mathcal{H}}) = PM^*(s_1, \widetilde{\mathcal{H}}) = PM^*(s_2, \widetilde{\mathcal{H}})$. Note that transitions from the states of $DR(G)$ always lead to those from the same set, hence, $\forall s \in DR(G)\ PM^*(s, \widetilde{\mathcal{H}}) = PM^*(s, \widetilde{\mathcal{H}} \cap DR(G))$. 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,
  $\sum_{\tilde{s}\in\widetilde{\mathcal{H}}\cap DR(G)} \psi^*[k+1](\tilde{s}) = \sum_{\tilde{s}\in\widetilde{\mathcal{H}}\cap DR(G)} \sum_{s\in DR(G)} \psi^*[k](s) PM^*(s, \tilde{s}) =$
  $\sum_{s\in DR(G)} \sum_{\tilde{s}\in\widetilde{\mathcal{H}}\cap DR(G)} \psi^*[k](s) PM^*(s, \tilde{s}) = \sum_{s\in DR(G)} \psi^*[k](s) \sum_{\tilde{s}\in\widetilde{\mathcal{H}}\cap DR(G)} PM^*(s, \tilde{s}) =$
  $\sum_{\mathcal{H}} \sum_{s\in\mathcal{H}\cap DR(G)} \psi^*[k](s) \sum_{\tilde{s}\in\widetilde{\mathcal{H}}\cap DR(G)} PM^*(s, \tilde{s}) =$
  $\sum_{\mathcal{H}} \sum_{s\in\mathcal{H}\cap DR(G)} \psi^*[k](s) \sum_{\tilde{s}\in\widetilde{\mathcal{H}}\cap DR(G)} \sum_{\{\Gamma|s \overset{\Gamma}{\twoheadrightarrow} \tilde{s}\}} PT^*(\Gamma, s) =$

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

C Proof of Theorem 8.2

The main idea of the proof is similar to that from [21, 22] but in the algebraic setting.

Let $\mathcal{H}\in(DR(G)\cup DR(G'))/_{\mathcal{R}}$ and $s,\bar{s}\in\mathcal{H}$. We have $\forall\widetilde{\mathcal{H}}\in(DR(G)\cup DR(G'))/_{\mathcal{R}}\ \forall A\in\mathbb{N}_f^{\mathcal{L}}\setminus\{\emptyset\}\ s\xrightarrow{A}_{\mathcal{P}}\widetilde{\mathcal{H}}\ \Leftrightarrow\ \bar{s}\xrightarrow{A}_{\mathcal{P}}\widetilde{\mathcal{H}}$. The previous equality is valid for all $s,\bar{s}\in\mathcal{H}$, hence, we can rewrite it as $\mathcal{H}\xrightarrow{A}_{\mathcal{P}}\widetilde{\mathcal{H}}$ and denote $PM_A^*(\mathcal{H},\widetilde{\mathcal{H}})=PM_A^*(s,\widetilde{\mathcal{H}})=PM_A^*(\bar{s},\widetilde{\mathcal{H}})$. Note that transitions from the states of $DR(G)$ always lead to those from the same set, hence, $\forall s\in DR(G)\ PM_A^*(s,\widetilde{\mathcal{H}})=PM_A^*(s,\widetilde{\mathcal{H}}\cap DR(G))$. The same is true for $DR(G')$.

Let $\Sigma=A_1\cdots A_n$ be a step trace of G and G'. We have $\exists\mathcal{H}_0,\dots,\exists\mathcal{H}_n\in(DR(G)\cup DR(G'))/_{\mathcal{R}}\ \mathcal{H}_0\xrightarrow{A_1}_{\mathcal{P}_1}\mathcal{H}_1\xrightarrow{A_2}_{\mathcal{P}_2}\cdots\xrightarrow{A_n}_{\mathcal{P}_n}\mathcal{H}_n$. Now we intend to 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 $\mathcal{P}_1,\dots,\mathcal{P}_n$:

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

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

- $n=1$

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

- $n\to n+1$

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

Let $s_0,\bar{s}_0\in\mathcal{H}_0$. We have
$PM^*(A_1\cdots A_n,s_0)=\sum_{\{\Gamma_1,\dots,\Gamma_n\mid s_0\xrightarrow{\Gamma_1}\cdots\xrightarrow{\Gamma_n}s_n,\ \mathcal{L}(\Gamma_i)=A_i,\ (1\le i\le n)\}}\prod_{i=1}^{n}PT^*(\Gamma_i,s_{i-1})=$
$\sum_{\mathcal{H}_1,\dots,\mathcal{H}_n}\sum_{\{\Gamma_1,\dots,\Gamma_n\mid s_0\xrightarrow{\Gamma_1}\cdots\xrightarrow{\Gamma_n}s_n,\ \mathcal{L}(\Gamma_i)=A_i,\ s_i\in\mathcal{H}_i\ (1\le i\le n)\}}\prod_{i=1}^{n}PT^*(\Gamma_i,s_{i-1})=$
$\sum_{\mathcal{H}_1,\dots,\mathcal{H}_n}\prod_{i=1}^{n}PM_{A_i}^*(\mathcal{H}_{i-1},\mathcal{H}_i)=$
$\sum_{\mathcal{H}_1,\dots,\mathcal{H}_n}\sum_{\{\overline{\Gamma}_1,\dots,\overline{\Gamma}_n\mid \bar{s}_0\xrightarrow{\overline{\Gamma}_1}\cdots\xrightarrow{\overline{\Gamma}_n}\bar{s}_n,\ \mathcal{L}(\overline{\Gamma}_i)=A_i,\ \bar{s}_i\in\mathcal{H}_i\ (1\le i\le n)\}}\prod_{i=1}^{n}PT^*(\overline{\Gamma}_i,\bar{s}_{i-1})=$
$\sum_{\{\overline{\Gamma}_1,\dots,\overline{\Gamma}_n\mid \bar{s}_0\xrightarrow{\overline{\Gamma}_1}\cdots\xrightarrow{\overline{\Gamma}_n}\bar{s}_n,\ \mathcal{L}(\overline{\Gamma}_i)=A_i,\ (1\le i\le n)\}}\prod_{i=1}^{n}PT^*(\overline{\Gamma}_i,\bar{s}_{i-1})=PM^*(A_1\cdots A_n,\bar{s}_0).$
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 8.1, $\sum_{s\in\mathcal{H}\cap DR(G)}\psi^*(s)=\sum_{s'\in\mathcal{H}\cap DR(G')}{\psi'}^*(s')$. Now we can complete the proof: $\sum_{s\in\mathcal{H}\cap DR(G)}\psi^*(s)PT^*(\Sigma,s)=\sum_{s\in\mathcal{H}\cap DR(G)}\psi^*(s)PT^*(\Sigma,\mathcal{H})=PT^*(\Sigma,\mathcal{H})\sum_{s\in\mathcal{H}\cap DR(G)}\psi^*(s)=$ $PT^*(\Sigma,\mathcal{H})\sum_{s'\in\mathcal{H}\cap DR(G')}{\psi'}^*(s')=\sum_{s'\in\mathcal{H}\cap DR(G')}{\psi'}^*(s')PT^*(\Sigma,\mathcal{H})=$ $\sum_{s'\in\mathcal{H}\cap DR(G')}{\psi'}^*(s')PT^*(\Sigma,s')$. □