Performance evaluation in stochastic process algebra $dtsiPBC$[a]

Igor V. Tarasyuk

A.P. Ershov Institute of Informatics Systems,

Siberian Division of the Russian Academy of Sciences,

6, Acad. Lavrentiev pr., Novosibirsk 630090, Russian Federation

itar@iis.nsk.su

itar.iis.nsk.su

[a]The joint work with Hermenegilda Macià S. and Valentín Valero R., High School of Computer Science Engineering, University of Castilla - La Mancha, Avda. de España s/n, 02071 Albacete, Spain, Hermenegilda.Macia@uclm.es, Valentin.Valero@uclm.es.

Abstract: In [MVF01], a continuous time stochastic extension $sPBC$ of finite Petri box calculus PBC [BDH92] was proposed. In [MVCC03], iteration operator was added to $sPBC$.

Algebra $sPBC$ has an interleaving semantics, but PBC has a step one.

We constructed a discrete time stochastic extension $dtsPBC$ of finite PBC [Tar05] and enriched it with iteration [Tar06].

We present the extension $dtsiPBC$ of $dtsPBC$ with immediate multiactions [TMV10,TMV13]. $dtsiPBC$ is a discrete time analog of $sPBC$ with immediate multiactions.

The step operational semantics is defined in terms of labeled probabilistic transition systems.

The denotational semantics is defined in terms of a subclass of labeled DTSPNs with immediate transitions (LDTSIPNs), called discrete time stochastic and immediate Petri boxes (dtsi-boxes).

The corresponding semi-Markov chain and (reduced) discrete time Markov chain are analyzed to evaluate performance.

We propose step stochastic bisimulation equivalence and explain how to use it for reduction of transition systems and semi-Markov chains.

We demonstrate how to apply this equivalence to compare stationary behaviour and simplify performance analysis.

The case study of performance evaluation is presented: running example of the shared memory system.

Keywords: stochastic Petri net, stochastic process algebra, Petri box calculus, discrete time, immediate multiaction, transition system, operational semantics, immediate transition, dtsi-box, denotational semantics, Markov chain, performance evaluation, stochastic equivalence, reduction, shared memory system.

Contents

Introduction

Previous work

- Continuous time (subsets of $\mathbb{R}_+$): interleaving semantics
  - *Continuous time stochastic Petri nets (CTSPNs)* [Mol82,FN85]:
    exponential transition firing delays,
    Continuous time Markov chain (CTMC).
  - *Generalized stochastic Petri nets (GSPNs)* [MCB84,CMBC93]:
    exponential and zero transition firing delays,
    Semi-Markov chain (SMC).
  - *Extended generalized stochastic Petri nets (EGSPNs)* [HS89,MBBCCC89]:
    hyper-exponential or Erlang or phase and zero transition firing delays.
  - *Deterministic stochastic Petri nets (DSPNs)* [MC87,MCF90]:
    exponential and deterministic transition firing delays,
    Semi-Markov process (SMP), if no two deterministic transitions are enabled in any marking.
  - *Extended deterministic stochastic Petri nets (EDSPNs)* [GL94]:
    non-exponential and deterministic transition firing delays.
  - *Extended stochastic Petri nets (ESPNs)* [DTGN85]:
    arbitrary transition firing delays.

- Discrete time (subsets of $\mathbb{N}$): interleaving and step semantics
  - *Discrete time stochastic Petri nets (DTSPNs)* [Mol85,ZG94]:

    geometric transition firing delays,

    Discrete time Markov chain (DTMC).
  - *Discrete time deterministic and stochastic Petri nets (DTDSPNs)* [ZFH01]:

    geometric and fixed transition firing delays,

    Semi-Markov chain (SMC).
  - *Discrete deterministic and stochastic Petri nets (DDSPNs)* [ZCH97]:

    phase and fixed transition firing delays,

    Semi-Markov chain (SMC).

Stochastic process algebras

- $MTIPP$ [HR94]
- $GSPA$ [BKLL95]
- $PEPA$ [Hil96]
- $S\pi$ [Pri96]
- $EMPA$ [BGo98]
- $GSMPA$ [BBGo98]
- $sACP$ [AHR00]
- TCP^{dst} [MVi08]

More stochastic process calculi

- $TIPP$ [GHR93]
- $WSCCS$ [Tof94]
- $PM-TIPP$ [Ret95]
- $SPADES$ [AKB98]
- $NMSPA$ [LN00]
- $SM-PEPA$ [Brad05]
- $iPEPA$ [HBC13]
- $mCCS$ [DH13]
- $PHASE$ [CR14]

Algebra PBC and its extensions

- *Petri box calculus* PBC [BDH92]
- *Time Petri box calculus* $tPBC$ [Kou00]
- *Timed Petri box calculus* $TPBC$ [MF00]
- *Stochastic Petri box calculus* $sPBC$ [MVF01,MVCC03]
- *Ambient Petri box calculus* $APBC$ [FM03]
- *Arc time Petri box calculus* $atPBC$ [Nia05]
- *Generalized stochastic Petri box calculus* $gsPBC$ [MVCR08]
- *Discrete time stochastic Petri box calculus* $dtsPBC$ [Tar05,Tar06]
- *Discrete time stochastic and immediate Petri box calculus* $dtsiPBC$ [TMV10,TMV13]

SPACLS: Classification of stochastic process algebras

Time	Immediate (multi)actions	Interleaving semantics	Non-interleaving semantics
Continuous	No	$MTIPP$ (CTMC), $PEPA$ (CTMP), $sPBC$ (CTMC)	$GSPA$ (GSMP), $S\pi$, $GSMPA$ (GSMP)
	Yes	$EMPA$ (SMC, CTMC), $gsPBC$ (SMC)	—
Discrete	No	—	$dtsPBC$ (DTMC)
	Yes	TCP^{dst} (DTMRC)	$sACP$, $dtsiPBC$ (SMC, DTMC)

The SPNs-based denotational semantics: orange SPA names.

The underlying stochastic process: in parentheses near the SPA names.

Transition labeling

- CTSPNs [Buc95]
- GSPNs [Buc98]
- DTSPNs [BT00]

Stochastic equivalences

- Probabilistic transition systems (PTSs) [BM89,Chr90,LS91,BHe97,KN98]
- SPAs [HR94,Hil94,BGo98]
- Markov process algebras (MPAs) [Buc94,BKe01]
- CTSPNs [Buc95]
- GSPNs [Buc98]
- Stochastic automata (SAs) [Buc99]
- Stochastic event structures (SESs) [MCW03]

Syntax

The *set of all finite multisets* over X is $\mathbb{N}_{fin}^X$. The *set of all subsets (powerset)* of X is 2^X.

$Act = \{a, b, \ldots\}$ is the set of *elementary actions*.

$\widehat{Act} = \{\hat{a}, \hat{b}, \ldots\}$ is the set of *conjugated actions (conjugates)* s.t. $\hat{a} \neq a$ and $\hat{\hat{a}} = a$.

$\mathcal{A} = Act \cup \widehat{Act}$ is the set of *all actions*. $\mathcal{L} = \mathbb{N}_{fin}^{\mathcal{A}}$ is the set of *all multiactions*.

The *alphabet* of $\alpha \in \mathcal{L}$ is $\mathcal{A}(\alpha) = \{x \in \mathcal{A} \mid \alpha(x) > 0\}$.

A *stochastic multiaction* is a pair (α, ρ) s.t. $\alpha \in \mathcal{L}$ and $\rho \in (0; 1)$ is the *probability* of the multiaction α.

$\mathcal{SL}$ is the set of *all stochastic multiactions*.

An *immediate multiaction* is a pair (α, l) s.t. $\alpha \in \mathcal{L}$ and $l \in \mathbb{N}_{\geq 1}$ is the *weight* of the multiaction α.

$\mathcal{IL}$ is the set of *all immediate multiactions*. $\mathcal{SIL} = \mathcal{SL} \cup \mathcal{IL}$ is the set of *all activities*.

The *alphabet* of $(\alpha, \kappa) \in \mathcal{SIL}$ is $\mathcal{A}(\alpha, \kappa) = \mathcal{A}(\alpha)$, that of $\Upsilon \in \mathbb{N}_{fin}^{\mathcal{SIL}}$ is $\mathcal{A}(\Upsilon) = \cup_{(\alpha,\kappa)\in\Upsilon}\mathcal{A}(\alpha)$.

The *multiaction part* of $\Upsilon \in \mathbb{N}_{fin}^{\mathcal{SIL}}$ is $\mathcal{L}(\Upsilon) = \sum_{(\alpha,\kappa)\in\Upsilon} \alpha$.

The operations: *sequential execution* ;, *choice* $[]$, *parallelism* $\|$, *relabeling* $[f]$, *restriction* rs, *synchronization* sy and *iteration* $[**]$.

Sequential execution and choice have the standard interpretation.

Parallelism does not include synchronization unlike that in standard process algebras.

Relabeling functions $f : \mathcal{A} \to \mathcal{A}$ are bijections preserving conjugates: $\forall x \in \mathcal{A}\ f(\hat{x}) = \widehat{f(x)}$.

Restriction over $a \in Act$: any process behaviour containing a or its conjugate $\hat{a}$ is not allowed.

Let $\alpha, \beta \in \mathcal{L}$ be two multiactions s.t. for $a \in Act$ we have $a \in \alpha$ and $\hat{a} \in \beta$, or $\hat{a} \in \alpha$ and $a \in \beta$.
Synchronization of α and β by a is $\alpha \oplus_a \beta = \gamma$:

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

In the iteration, the initialization subprocess is executed first,

then the body one is performed zero or more times, finally, the termination one is executed.

Static expressions specify the structure of processes.

Definition 1 *Let* $(\alpha,\kappa) \in \mathcal{SIL}$ *and* $a \in Act$. *A* static expression *of* $dtsiPBC$ *is*

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

$StatExpr$ is the set of *all static expressions* of $dtsiPBC$.

Definition 2 *Let* $(\alpha,\kappa) \in \mathcal{SIL}$ *and* $a \in Act$. *A* regular static expression *of* $dtsiPBC$ *is*

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

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

$RegStatExpr$ is the set of *all regular static expressions* of $dtsiPBC$.

Dynamic expressions specify the states of processes.

Dynamic expressions are obtained from static ones annotated with upper or lower bars.

The *underlying static expression* of a dynamic one: removing all upper and lower bars.

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

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

$DynExpr$ is the set of *all dynamic expressions* of $dtsiPBC$.

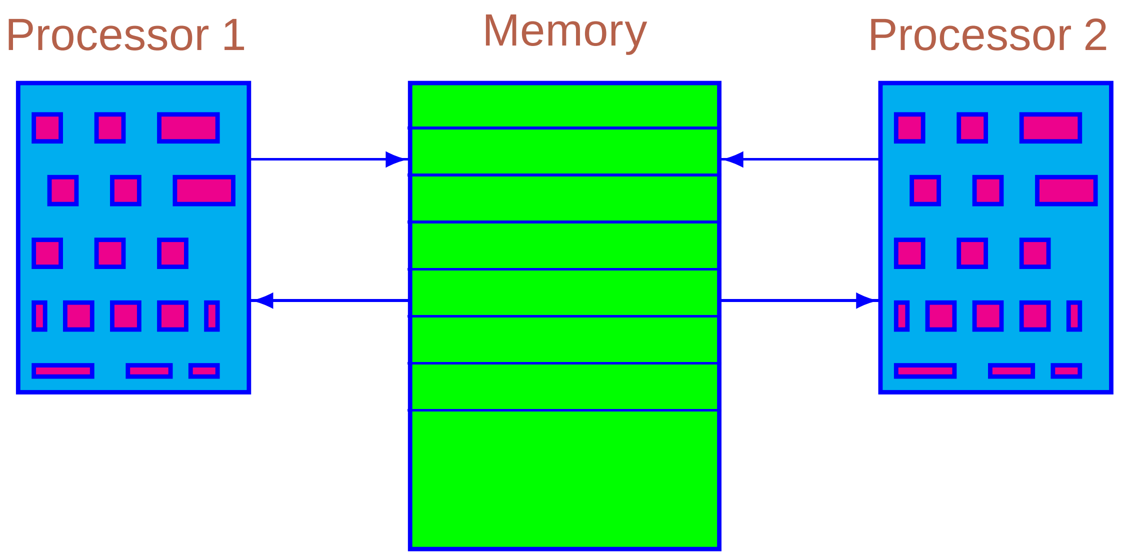
Definition 4 *A dynamic expression is* regular *if its underlying static expression is regular.*

$RegDynExpr$ is the set of *all regular dynamic expressions* of $dtsiPBC$.

We shall consider regular expressions only and omit the word “regular”.

Generalized shared memory system

A model of two processors accessing a common shared memory [MBCDF95]



SHMDIA: The diagram of the shared memory system

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 after which the instantaneous decision is made.

When the decision is made in favour of a processor, it starts an acquisition of the memory, and another processor waits until the former one ends its operations, and the system returns to the state with both active processors and the available memory.

a corresponds to the system activation.

r_i $(1 \leq i \leq 2)$ represent the common memory request of processor i.

d_i correspond to the instantaneous decision on the memory allocation in favour of the processor i.

m_i represent the common memory access of processor i.

The other actions are used for communication purpose only.

$\mathsf{Stop} = (\{c\}, \frac{1}{2})$ rs c is the process that performs empty loops with probability 1 and never terminates.

The static expression of the first processor is

$K_1 = [(\{x_1\}, \rho) * ((\{r_1\}, \rho); (\{d_1, y_1\}, l); (\{m_1, z_1\}, \rho)) * \mathsf{Stop}]$.

The static expression of the second processor is

$K_2 = [(\{x_2\}, \rho) * ((\{r_2\}, \rho); (\{d_2, y_2\}, l); (\{m_2, z_2\}, \rho)) * \mathsf{Stop}]$.

The static expression of the shared memory is

$K_3 = [(\{a, \widehat{x_1}, \widehat{x_2}\}, \rho) * (((\{\widehat{y_1}\}, l); (\{\widehat{z_1}\}, \rho))[]((\{\widehat{y_2}\}, l); (\{\widehat{z_2}\}, \rho))) * \mathsf{Stop}]$.

The static expression of the generalized shared memory system with two processors is

$K = (K_1 \| K_2 \| K_3)$ 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.

Operational semantics

Inaction rules

Inaction rules: instantaneous structural transformations.

Let $E, F, K \in RegStatExpr$ and $a \in Act$.

IRULES1: Inaction rules for overlined and underlined regular static expressions

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

Let $E, F \in RegStatExpr,\ G, H, \widetilde{G}, \widetilde{H} \in RegDynExpr$ and $a \in Act$.

IRULES2: Inaction rules for arbitrary regular dynamic expressions

$$\frac{G \Rightarrow \widetilde{G},\ \circ \in \{;, []\}}{G \circ E \Rightarrow \widetilde{G} \circ E} \quad \frac{G \Rightarrow \widetilde{G},\ \circ \in \{;, []\}}{E \circ G \Rightarrow E \circ \widetilde{G}} \quad \frac{G \Rightarrow \widetilde{G}}{G \| H \Rightarrow \widetilde{G} \| H} \quad \frac{H \Rightarrow \widetilde{H}}{G \| H \Rightarrow G \| \widetilde{H}} \quad \frac{G \Rightarrow \widetilde{G}}{G[f] \Rightarrow \widetilde{G}[f]}$$

$$\frac{G \Rightarrow \widetilde{G},\ \circ \in \{\mathrm{rs}, \mathrm{sy}\}}{G \circ a \Rightarrow \widetilde{G} \circ a} \quad \frac{G \Rightarrow \widetilde{G}}{[G * E * F] \Rightarrow [\widetilde{G} * E * F]} \quad \frac{G \Rightarrow \widetilde{G}}{[E * G * F] \Rightarrow [E * \widetilde{G} * F]} \quad \frac{G \Rightarrow \widetilde{G}}{[E * F * G] \Rightarrow [E * F * \widetilde{G}]}$$

Definition 5 *A regular dynamic expression is operative if no inaction rule can be applied to it.*

$OpRegDynExpr$ is the set of *all operative regular dynamic expressions* of $dtsiPBC$.

We shall consider regular expressions only and omit the word “regular”.

Definition 6 $\approx\ = (\Rightarrow \cup \Leftarrow)^*$ *is the structural equivalence of dynamic expressions in* $dtsiPBC$. *G and G' are structurally equivalent, $G \approx G'$, if they can be reached each from other by applying inaction rules in a forward or backward direction.*

Action and empty loop rules

Action rules with stochastic multiactions: execution of non-empty multisets of stochastic multiactions.

Action rules with immediate multiactions: execution of non-empty multisets of immediate multiactions.

Empty loop rule: execution of the empty multiset of activities at a time step.

Let $(\alpha, \rho), (\beta, \chi) \in \mathcal{SL},\ (\alpha, l), (\beta, m) \in \mathcal{IL}$ and $(\alpha, \kappa) \in \mathcal{SIL}$.

Let $E, F \in RegStatExpr,\ G, H \in OpRegDynExpr,\ \widetilde{G}, \widetilde{H} \in RegDynExpr$ and $a \in Act$.

Let $\Gamma, \Delta \in I\!\!N_{fin}^{\mathcal{SL}} \setminus \{\emptyset\},\ \Gamma' \in I\!\!N_{fin}^{\mathcal{SL}},\ I, J \in I\!\!N_{fin}^{\mathcal{IL}} \setminus \{\emptyset\},\ I' \in I\!\!N_{fin}^{\mathcal{IL}}$ and $\Upsilon \in I\!\!N_{fin}^{\mathcal{SIL}} \setminus \{\emptyset\}$.

The names of the action rules with immediate multiactions have a suffix '**i**'.

ARULES: Action and empty loop rules

El $\dfrac{tang(G)}{G\stackrel{\emptyset}{\rightarrow}G}$

B $\overline{(\alpha,\kappa)}\stackrel{\{(\alpha,\kappa)\}}{\longrightarrow}\underline{(\alpha,\kappa)}$

S $\dfrac{G\stackrel{\Upsilon}{\rightarrow}\widetilde{G}}{G;E\stackrel{\Upsilon}{\rightarrow}\widetilde{G};E\ \ E;G\stackrel{\Upsilon}{\rightarrow}E;\widetilde{G}}$

C $\dfrac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G},\ \neg init(G)\vee(init(G)\wedge tang(\overline{E}))}{G[]E\stackrel{\Gamma}{\rightarrow}\widetilde{G}[]E\ \ E[]G\stackrel{\Gamma}{\rightarrow}E[]\widetilde{G}}$

Ci $\dfrac{G\stackrel{I}{\rightarrow}\widetilde{G}}{G[]E\stackrel{I}{\rightarrow}\widetilde{G}[]E\ \ E[]G\stackrel{I}{\rightarrow}E[]\widetilde{G}}$

P1 $\dfrac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G},\ tang(H)}{G\|H\stackrel{\Gamma}{\rightarrow}\widetilde{G}\|H\ \ H\|G\stackrel{\Gamma}{\rightarrow}H\|\widetilde{G}}$

P1i $\dfrac{G\stackrel{I}{\rightarrow}\widetilde{G}}{G\|H\stackrel{I}{\rightarrow}\widetilde{G}\|H\ \ H\|G\stackrel{I}{\rightarrow}H\|\widetilde{G}}$

P2 $\dfrac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G},\ H\stackrel{\Delta}{\rightarrow}\widetilde{H}}{G\|H\stackrel{\Gamma+\Delta}{\longrightarrow}\widetilde{G}\|\widetilde{H}}$

P2i $\dfrac{G\stackrel{I}{\rightarrow}\widetilde{G},\ H\stackrel{J}{\rightarrow}\widetilde{H}}{G\|H\stackrel{I+J}{\longrightarrow}\widetilde{G}\|\widetilde{H}}$

L $\dfrac{G\stackrel{\Upsilon}{\rightarrow}\widetilde{G}}{G[f]\stackrel{f(\Upsilon)}{\longrightarrow}\widetilde{G}[f]}$

Rs $\dfrac{G\stackrel{\Upsilon}{\rightarrow}\widetilde{G},\ a,\hat{a}\not\in\mathcal{A}(\Upsilon)}{G\ \mathsf{rs}\ a\stackrel{\Upsilon}{\rightarrow}\widetilde{G}\ \mathsf{rs}\ a}$

I1 $\dfrac{G\stackrel{\Upsilon}{\rightarrow}\widetilde{G}}{[G*E*F]\stackrel{\Upsilon}{\rightarrow}[\widetilde{G}*E*F]}$

I2 $\dfrac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G},\ \neg init(G)\vee(init(G)\wedge tang(\overline{F}))}{[E*G*F]\stackrel{\Gamma}{\rightarrow}[E*\widetilde{G}*F]}$

I2i $\dfrac{G\stackrel{I}{\rightarrow}\widetilde{G}}{[E*G*F]\stackrel{I}{\rightarrow}[E*\widetilde{G}*F]}$

I3 $\dfrac{G\stackrel{\Gamma}{\rightarrow}\widetilde{G},\ \neg init(G)\vee(init(G)\wedge tang(\overline{F}))}{[E*F*G]\stackrel{\Gamma}{\rightarrow}[E*F*\widetilde{G}]}$

I3i $\dfrac{G\stackrel{I}{\rightarrow}\widetilde{G}}{[E*F*G]\stackrel{I}{\rightarrow}[E*F*\widetilde{G}]}$

Sy1 $\dfrac{G\stackrel{\Upsilon}{\rightarrow}\widetilde{G}}{G\ \mathsf{sy}\ a\stackrel{\Upsilon}{\rightarrow}\widetilde{G}\ \mathsf{sy}\ a}$

Sy2 $\dfrac{G\ \mathsf{sy}\ a\xrightarrow{\Gamma'+\{(\alpha,\rho)\}+\{(\beta,\chi)\}}\widetilde{G}\ \mathsf{sy}\ a,\ a\in\alpha,\ \hat{a}\in\beta}{G\ \mathsf{sy}\ a\xrightarrow{\Gamma'+\{(\alpha\oplus_a\beta,\rho\cdot\chi)\}}\widetilde{G}\ \mathsf{sy}\ a}$

Sy2i $\dfrac{G\ \mathsf{sy}\ a\xrightarrow{I'+\{(\alpha,l)\}+\{(\beta,m)\}}\widetilde{G}\ \mathsf{sy}\ a,\ a\in\alpha,\ \hat{a}\in\beta}{G\ \mathsf{sy}\ a\xrightarrow{I'+\{(\alpha\oplus_a\beta,l+m)\}}\widetilde{G}\ \mathsf{sy}\ a}$

RULECMP: Comparison of inaction, action and empty loop rules

Rules	State change	Time progress	Activities execution
Inaction rules	$-$	$-$	$-$
Action rules (stochastic multiactions)	$\pm$	$+$	$+$
Action rules (immediate multiactions)	$\pm$	$-$	$+$
Empty loop rule	$-$	$+$	$-$

Transition systems

Definition 7 *The* derivation set $DR(G)$ *of a dynamic expression* G *is the minimal set:*

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

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

The set of *all multisets of activities executable from* s is $Exec(s) = \{\Upsilon \mid \exists H \in s\ \exists\widetilde{H}\ H \stackrel{\Upsilon}{\rightarrow} \widetilde{H}\}$.

The state s is *tangible*, if $Exec(s) \subseteq I\!N_{fin}^{\mathcal{SL}}$. For tangible states we may have $Exec(s) = \{\emptyset\}$.

The state s is *vanishing*, if $Exec(s) \subseteq I\!N_{fin}^{\mathcal{IL}} \setminus \{\emptyset\}$.

The set of *all tangible states from* $DR(G)$ is $DR_T(G)$.

The set of *all vanishing states from* $DR(G)$ is $DR_V(G)$.

Obviously, $DR(G) = DR_T(G) \uplus DR_V(G)$.

Let $\Upsilon \in Exec(s) \setminus \{\emptyset\}$. The *probability of the multiset of stochastic multiactions* or the *weight of the multiset of immediate multiactions* Υ *which is ready for execution in* s:

$$PF(\Upsilon, s) = \begin{cases} \prod_{(\alpha,\rho)\in\Upsilon} \rho \cdot \prod_{\{\{(\beta,\chi)\}\in Exec(s) \mid (\beta,\chi)\not\in\Upsilon\}} (1-\chi), & s \in DR_T(G); \\ \sum_{(\alpha,l)\in\Upsilon} l, & s \in DR_V(G). \end{cases}$$

In the case $\Upsilon = \emptyset$ and $s \in DR_T(G)$ we define

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

Let $\Upsilon \in Exec(s)$. The *probability to execute the multiset of activities* Υ *in* s:

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

If s is tangible, then $PT(\emptyset, s) \in (0; 1]$: the residence time in s is ≥ 1.

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

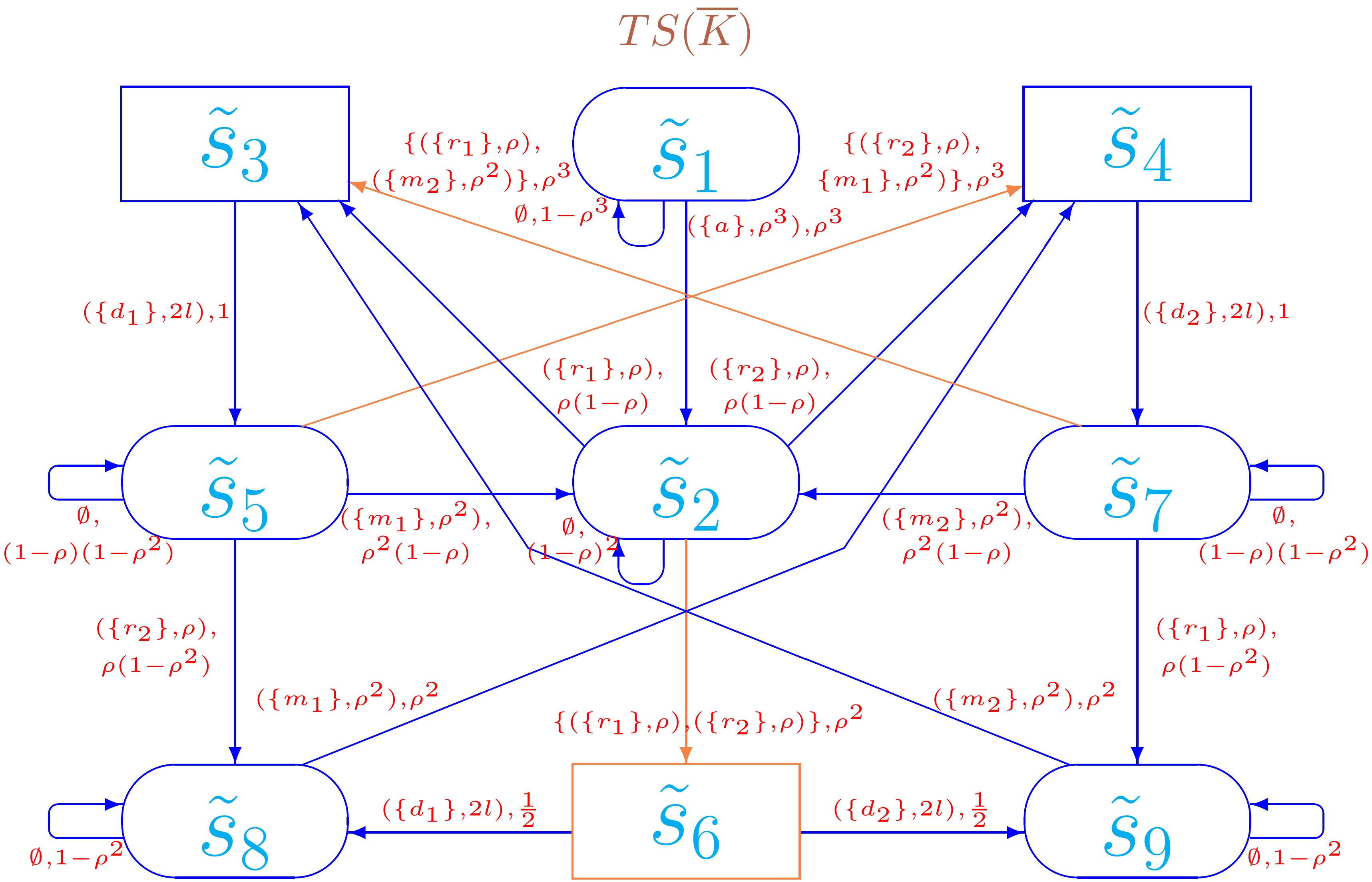
$$PM(s, \tilde{s}) = \sum_{\{\Upsilon \mid \exists H \in s\ \exists \widetilde{H} \in \tilde{s}\ H \xrightarrow{\Upsilon} \widetilde{H}\}} PT(\Upsilon, s).$$

Definition 8 *The* (labeled probabilistic) transition system *of a dynamic expression* G *is* $TS(G) = (S_G, L_G, \mathcal{T}_G, s_G)$, *where*

- *the set of* states *is* $S_G = DR(G)$;
- *the set of* labels *is* $L_G = \mathbb{N}_{fin}^{\mathcal{SIL}} \times (0;1]$;
- *the set of* transitions *is*
  $\mathcal{T}_G = \{(s, (\Upsilon, PT(\Upsilon, s)), \tilde{s}) \mid s, \tilde{s} \in DR(G),\ \exists H \in s\ \exists \widetilde{H} \in \tilde{s}\ H \stackrel{\Upsilon}{\rightarrow} \widetilde{H}\}$;
- *the* initial state *is* $s_G = [G]_\approx$.

A transition $(s, (\Upsilon, \mathcal{P}), \tilde{s}) \in \mathcal{T}_G$ is written as $s \stackrel{\Upsilon}{\rightarrow}_{\mathcal{P}} \tilde{s}$.

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



SHMGTS: The transition system of the generalized shared memory system

(parallel executions of activities and the exclusively reachable states are marked with orange)

Interpretation of the states of the generalized shared memory system

$DR_T(\overline{K}) = \{\tilde{s}_1, \tilde{s}_2, \tilde{s}_5, \tilde{s}_5, \tilde{s}_8, \tilde{s}_9\}$ and $DR_V(\overline{K}) = \{\tilde{s}_3, \tilde{s}_4, \tilde{s}_6\}$.

$\tilde{s}_1$: the initial state,

$\tilde{s}_2$: the system is activated and the memory is not requested,

$\tilde{s}_3$: the memory is requested by the first processor,

$\tilde{s}_4$: the memory is requested by the second processor,

$\tilde{s}_5$: the memory is allocated to the first processor,

$\tilde{s}_6$: the memory is requested by two processors,

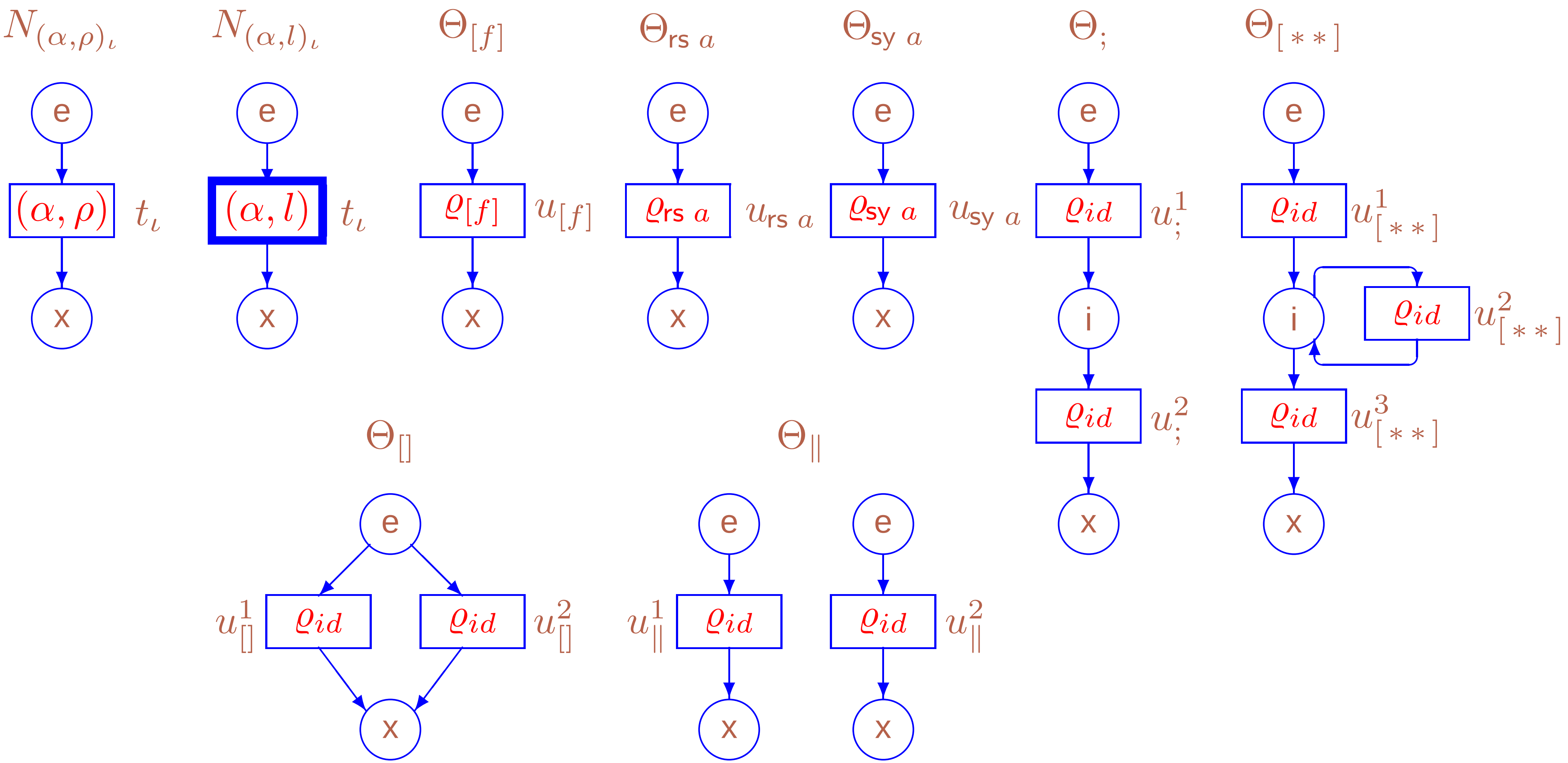
$\tilde{s}_7$: the memory is allocated to the second processor,

$\tilde{s}_8$: the memory is allocated to the first processor and the memory is requested by the second processor,

$\tilde{s}_9$: the memory is allocated to the second processor and the memory is requested by the first processor.

Denotational semantics

Algebra of dtsi-boxes



BOXOPS: The plain and operator dtsi-boxes

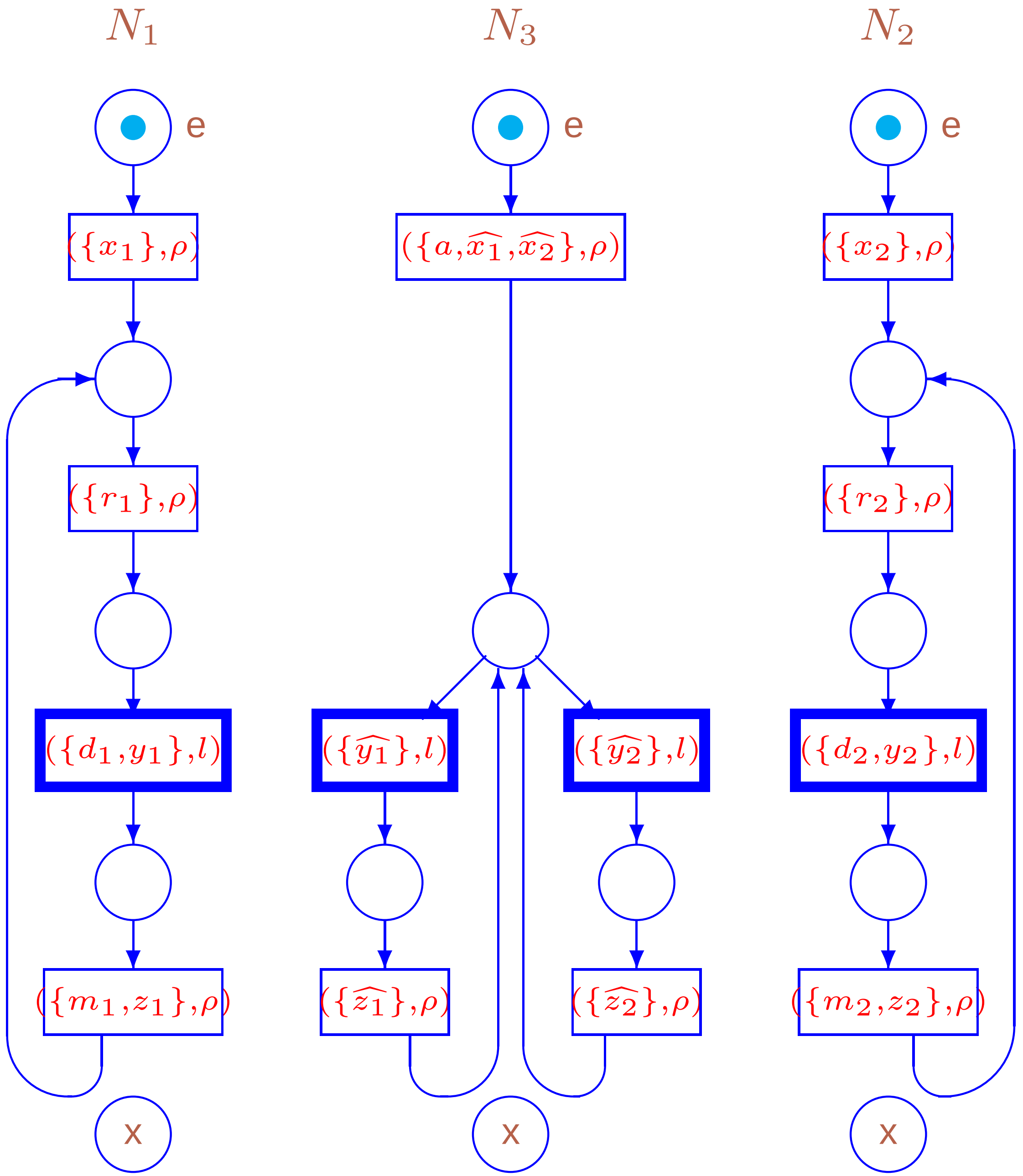
Definition 9 *Let* $(\alpha,\kappa)\in\mathcal{SIL}$, $a\in Act$ *and* $E,F,K\in RegStatExpr$. *The* denotational semantics *of* $dtsiPBC$ *is a mapping* Box_{dtsi} *from* $RegStatExpr$ *into plain dtsi-boxes:*

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

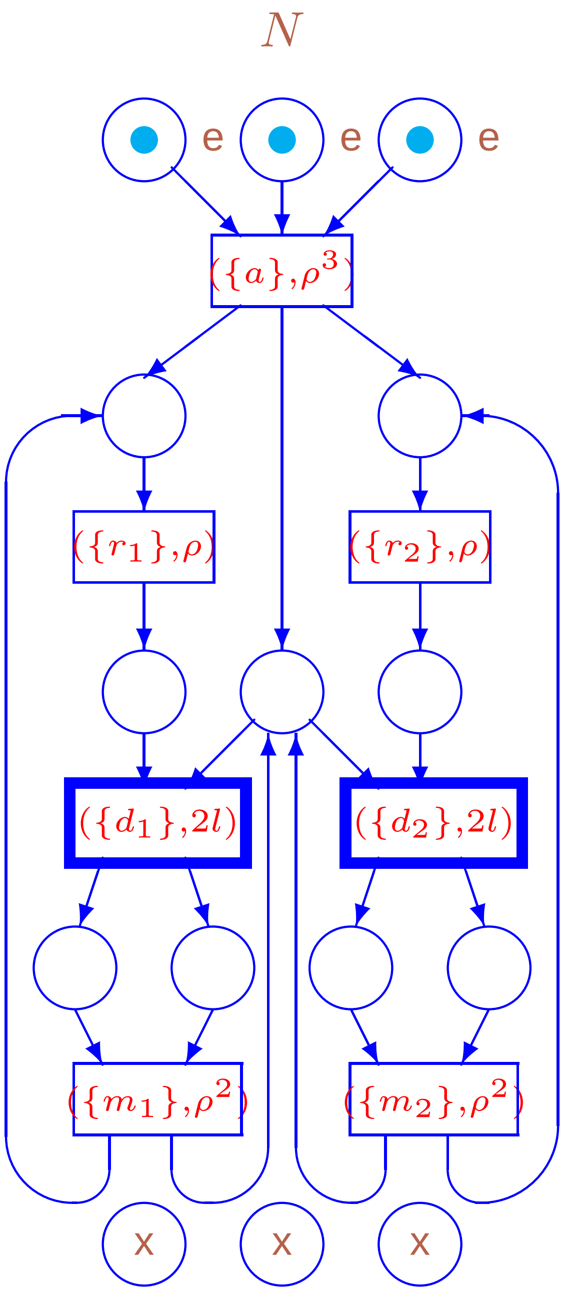
For $E\in RegStatExpr$, let $Box_{dtsi}(\overline{E})=\overline{Box_{dtsi}(E)}$ and $Box_{dtsi}(\underline{E})=\underline{Box_{dtsi}(E)}$.

Theorem 1 *(OPDNSEM) For any static expression* E

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



SHMGPMBOX: The marked dtsi-boxes of the generalized two processors and shared memory



SHMGBOX: The marked dtsi-box of the generalized shared memory system

Performance evaluation

Analysis of the underlying SMC

For a dynamic expression G, a discrete random variable is associated with each state from $DR_T(G)$.

The random variables (residence time in the tangible states) are geometrically distributed:
the probability to stay in the tangible state $s \in DR_T(G)$ for $k-1$ moments
and leave it at the moment $k \geq 1$ is $PM(s,s)^{k-1}(1-PM(s,s))$.

The mean value formula: the *average sojourn time in the tangible state* s is $\frac{1}{1-PM(s,s)}$.

The *average sojourn time in the vanishing state* s is 0.

The *average sojourn time in the state* s is $SJ(s) = \begin{cases} \frac{1}{1-PM(s,s)}, & s \in DR_T(G); \\ 0, & s \in DR_V(G). \end{cases}$

The *average sojourn time vector* SJ of G has the elements $SJ(s),\ s \in DR(G)$.

The *sojourn time variance in the state* s is $VAR(s) = \begin{cases} \frac{PM(s,s)}{(1-PM(s,s))^2}, & s \in DR_T(G); \\ 0, & s \in DR_V(G). \end{cases}$

The *sojourn time variance vector* VAR of G has the elements $VAR(s),\ s \in DR(G)$.

The stochastic process associated with a dynamic expression G: the *underlying semi-Markov chain (SMC)* of G, $SMC(G)$, which is analyzed by extracting the *embedded (absorbing) discrete time Markov chain (EDTMC)* of G, $EDTMC(G)$.

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

Let $s \to s$. The *probability to stay in s due to k $(k \geq 1)$ self-loops* is $PM(s,s)^k$.

Let $s \to \tilde{s}$ and $s \neq \tilde{s}$. The *probability to move from s to $\tilde{s}$ by executing any multiset of activities after possible self-loops* is

$$PM^*(s,\tilde{s}) = \left\{ \begin{array}{ll} PM(s,\tilde{s})\sum_{k=0}^{\infty} PM(s,s)^k = \frac{PM(s,\tilde{s})}{1-PM(s,s)}, & s \to s; \\ PM(s,\tilde{s}), & \text{otherwise}; \end{array} \right\} = SL(s)PM(s,\tilde{s}),$$

where $SL(s) = \begin{cases} \frac{1}{1-PM(s,s)}, & s \to s; \\ 1, & \text{otherwise}; \end{cases}$ is the *self-loops abstraction factor in the state s*.

The *self-loops abstraction vector* SL of G has the elements $SL(s),\ s \in DR(G)$.

Definition 10 *Let G be a dynamic expression. The* embedded (absorbing) discrete time Markov chain (EDTMC) *of G, $EDTMC(G)$, has the state space $DR(G)$, the initial state $[G]_\approx$ and the transitions $s \twoheadrightarrow_{\mathcal{P}} \tilde{s}$, if $s \rightarrow \tilde{s}$ and $s \neq \tilde{s}$, where $\mathcal{P} = PM^*(s, \tilde{s})$.*

The underlying SMC *of G, $SMC(G)$, has the EDTMC $EDTMC(G)$ and the sojourn time in every $s \in DR_T(G)$ is geometrically distributed with the parameter $1 - PM(s, s)$ while the sojourn time in every $s \in DR_V(G)$ is equal to zero.*

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* $EDTMC(G)$:

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

The *transient (k-step, $k \in I\!N$) probability mass function (PMF)* $\psi^*[k] = (\psi^*[k](s_1), \ldots, \psi^*[k](s_n))$ *for* $EDTMC(G)$ is calculated as

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

where $\psi^*[0] = (\psi^*[0](s_1), \ldots, \psi^*[0](s_n))$ is the *initial PMF*:

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

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

The *steady-state PMF* $\psi^* = (\psi^*(s_1), \ldots, \psi^*(s_n))$ *for* $EDTMC(G)$ is a solution of

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

where $\mathbf{I}$ is the identity matrix of order n and $\mathbf{0}$ is a row vector of n values 0, $\mathbf{1}$ is that of n values 1.

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

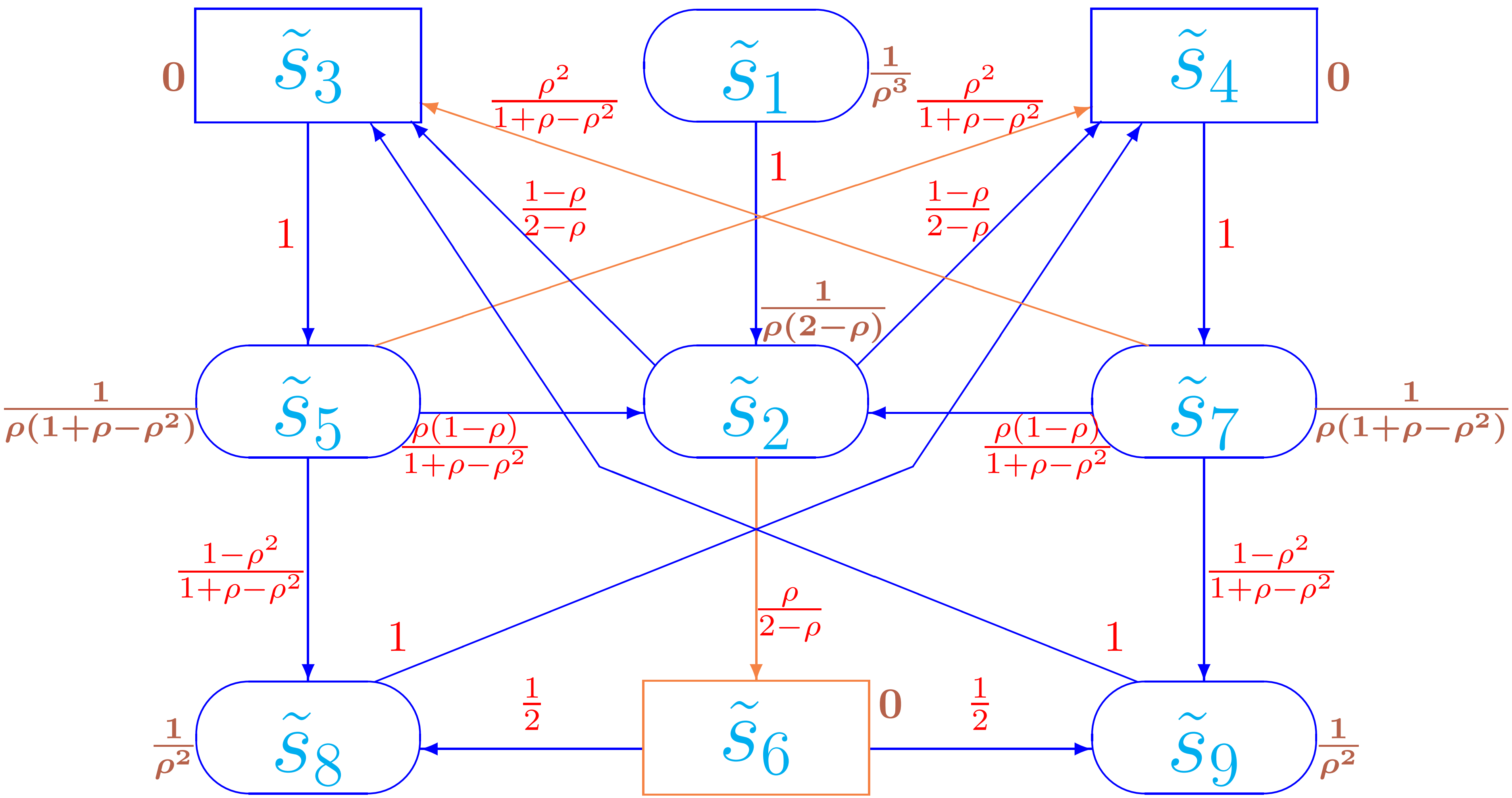
The *steady-state PMF* $\varphi = (\varphi(s_1), \ldots, \varphi(s_n))$ *for* $SMC(G)$:

$$\varphi(s_i) = \begin{cases} \frac{\psi^*(s_i) SJ(s_i)}{\sum_{j=1}^{n} \psi^*(s_j) SJ(s_j)}, & s_i \in DR_T(G); \\ 0, & s_i \in DR_V(G). \end{cases}$$

To calculate φ, we apply abstracting from self-loops to get $\mathbf{P}^*$ and ψ^*, followed by weighting by SJ and normalization.

$EDTMC(G)$ has no self-loops, unlike $SMC(G)$, hence, the behaviour of $EDTMC(G)$ stabilizes quicker than that of $SMC(G)$, since $\mathbf{P}^*$ has only zero elements at the main diagonal.

$SMC(\overline{K})$



SHMGSMC: The underlying SMC of the generalized shared memory system

(parallel executions of activities and the exclusively reachable states are marked with orange)

The average sojourn time vector of $\overline{K}$:

$$\widetilde{SJ} = \left(\frac{1}{\rho^3}, \frac{1}{\rho(2-\rho)}, 0, 0, \frac{1}{\rho(1+\rho-\rho^2)}, 0, \frac{1}{\rho(1+\rho-\rho^2)}, \frac{1}{\rho^2}, \frac{1}{\rho^2}\right).$$

The sojourn time variance vector of $\overline{K}$:

$$\widetilde{VAR} = \left(\frac{1-\rho^3}{\rho^6}, \frac{(1-\rho)^2}{\rho^2(2-\rho)^2}, 0, 0, \frac{(1-\rho)^2(1+\rho)}{\rho^2(1+\rho-\rho^2)^2}, 0, \frac{(1-\rho)^2(1+\rho)}{\rho^2(1+\rho-\rho^2)^2}, \frac{1-\rho^2}{\rho^4}, \frac{1-\rho^2}{\rho^4}\right).$$

The TPM for $EDTMC(\overline{K})$:

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

The steady-state PMF for $EDTMC(\overline{K})$:

$$\tilde{\psi}^* = \frac{1}{2(6+3\rho-9\rho^2+2\rho^3)}(0, 2\rho(2-3\rho-\rho^2), 2+\rho-3\rho^2+\rho^3, 2+\rho-3\rho^2+\rho^3,$$
$$2+\rho-3\rho^2+\rho^3, 2\rho^2(1-\rho), 2+\rho-3\rho^2+\rho^3, 2-\rho-\rho^2, 2-\rho-\rho^2).$$

The steady-state PMF $\tilde{\psi}^*$ weighted by $\widetilde{SJ}$:

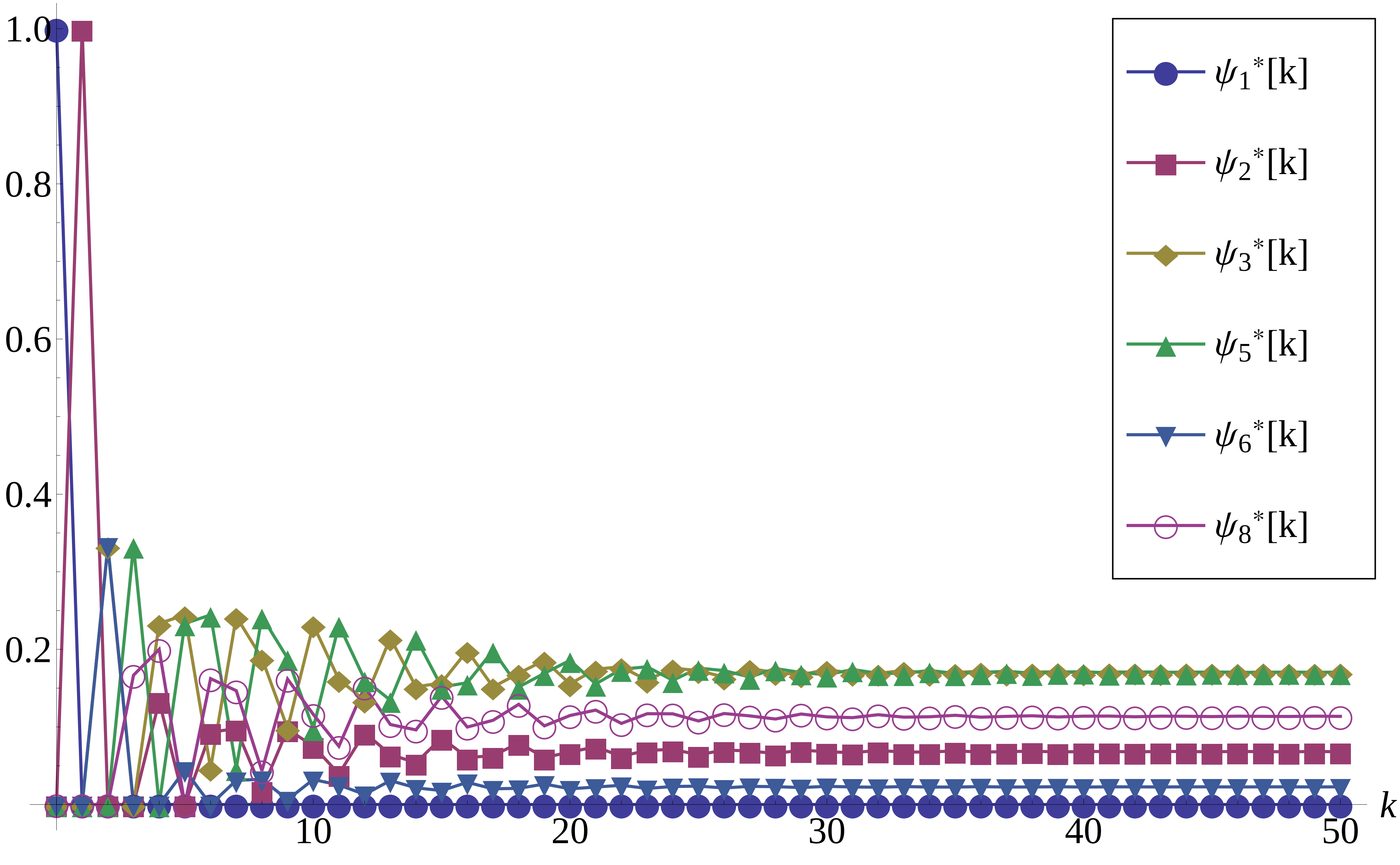
$$\frac{1}{2\rho^2(6+3\rho-9\rho^2+2\rho^3)}(0, 2\rho^2(1-\rho), 0, 0, \rho(2-\rho), 0, \rho(2-\rho), 2-\rho-\rho^2, 2-\rho-\rho^2).$$

We normalize the steady-state weighted PMF dividing it by the sum of its components

$$\tilde{\psi}^*\widetilde{SJ}^T = \frac{2+\rho-\rho^2-\rho^3}{\rho^2(6+3\rho-9\rho^2+2\rho^3)}.$$

The steady-state PMF for $SMC(\overline{K})$:

$$\tilde{\varphi} = \frac{1}{2(2+\rho-\rho^2-\rho^3)}(0, 2\rho^2(1-\rho), 0, 0, \rho(2-\rho), 0, \rho(2-\rho), 2-\rho-\rho^2, 2-\rho-\rho^2).$$



SHMTP: Transient probabilities alteration diagram for the EDTMC of the generalized shared memory system when $\rho = \frac{1}{2}$

Analysis of the reduced DTMC

Definition 11 *Let* G *be a dynamic expression. The* discrete time Markov chain (DTMC) *of* G*,* $DTMC(G)$*, has the state space* $DR(G)$*, the initial state* $[G]_\approx$ *and the transitions* $s \rightarrow_{\mathcal{P}} \tilde{s}$*, where* $\mathcal{P} = PM(s, \tilde{s})$.

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

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

Let G be a dynamic expression and $\mathbf{P}$ be the TPM for $DTMC(G)$.

Reordering the states from $DR(G)$: the first rows and columns of $\mathbf{P}$ correspond to the states from $DR_V(G)$ and the last ones correspond to the states from $DR_T(G)$.

Let $|DR(G)| = n$ and $|DR_T(G)| = m$. The resulting matrix is decomposed as:

$$\mathbf{P} = \begin{pmatrix} \mathbf{C} & \mathbf{D} \\ \mathbf{E} & \mathbf{F} \end{pmatrix}.$$

The elements of the $(n-m)\times(n-m)$ submatrix $\mathbf{C}$: the probabilities to move from vanishing to vanishing states.

The elements of the $(n-m)\times m$ submatrix $\mathbf{D}$: the probabilities to move from vanishing to tangible states.

The elements of the $m\times(n-m)$ submatrix $\mathbf{E}$: the probabilities to move from tangible to vanishing states.

The elements of the $m\times m$ submatrix $\mathbf{F}$: the probabilities to move from tangible to tangible states.

The TPM $\mathbf{P}^\diamond$ for $RDTMC(G)$ is the $m \times m$ matrix:

$$\mathbf{P}^\diamond = \mathbf{F} + \mathbf{EGD},$$

where the elements of the matrix $\mathbf{G}$ are the probabilities to move from vanishing to vanishing states in any number of state transitions, without traversal of the tangible states:

$$\mathbf{G} = \sum_{k=0}^{\infty} \mathbf{C}^k = \begin{cases} \sum_{k=0}^{l} \mathbf{C}^k, & \exists l \in I\!N\ \forall k > l\ \mathbf{C}^k = \mathbf{0}, \quad \text{no loops among vanishing states;} \\ (\mathbf{I} - \mathbf{C})^{-1}, & \lim_{k\to\infty} \mathbf{C}^k = \mathbf{0}, \quad \text{loops among vanishing states;} \end{cases}$$

where $\mathbf{0}$ is the square matrix consisting only of zeros and $\mathbf{I}$ is the identity matrix, both of size $n - m$.

For $1 \leq i, j \leq m$ and $1 \leq k, l \leq n-m$, let
$\mathcal{F}_{ij}$ be the elements of the matrix $\mathbf{F}$, $\mathcal{E}_{ik}$ be those of $\mathbf{E}$, $\mathcal{G}_{kl}$ be those of $\mathbf{G}$ and $\mathcal{D}_{lj}$ be those of $\mathbf{D}$.

The elements $\mathcal{P}^{\diamond}_{ij}$ of the matrix $\mathbf{P}^{\diamond}$ are

$$\mathcal{P}^{\diamond}_{ij} = \mathcal{F}_{ij} + \sum_{k=1}^{n-m} \sum_{l=1}^{n-m} \mathcal{E}_{ik}\mathcal{G}_{kl}\mathcal{D}_{lj} = \mathcal{F}_{ij} + \sum_{k=1}^{n-m} \mathcal{E}_{ik} \sum_{l=1}^{n-m} \mathcal{G}_{kl}\mathcal{D}_{lj} = \mathcal{F}_{ij} + \sum_{l=1}^{n-m} \mathcal{D}_{lj} \sum_{k=1}^{n-m} \mathcal{E}_{ik}\mathcal{G}_{kl},$$

i.e. $\mathcal{P}^{\diamond}_{ij}$ $(1 \leq i, j \leq m)$ is the total probability to move from the tangible state s_i to the tangible state s_j in any number of steps, without traversal of tangible states, but possibly going through vanishing states.

Let $s, \tilde{s} \in DR_T(G)$ such that $s = s_i,\ \tilde{s} = s_j$.

The *probability to move from s to $\tilde{s}$ in any number of steps, without traversal of tangible states* is

$$PM^{\diamond}(s, \tilde{s}) = \mathcal{P}^{\diamond}_{ij}.$$

Definition 12 *Let G be a dynamic expression and $[G]_\approx \in DR_T(G)$.*

The reduced discrete time Markov chain (RDTMC) of G, denoted by $RDTMC(G)$, has the state space $DR_T(G)$, the initial state $[G]_\approx$ and the transitions $s \hookrightarrow_{\mathcal{P}} \tilde{s}$, where $\mathcal{P} = PM^\diamond(s, \tilde{s})$.

Let $DR_T(G) = \{s_1, \ldots, s_m\}$ and $[G]_\approx \in DR_T(G)$. The transient (k-step, $k \in \mathbb{N}$) probability mass function (PMF) $\psi^\diamond[k] = (\psi^\diamond[k](s_1), \ldots, \psi^\diamond[k](s_m))$ for $RDTMC(G)$ is calculated as

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

where $\psi^\diamond[0] = (\psi^\diamond[0](s_1), \ldots, \psi^\diamond[0](s_m))$ is the initial PMF:

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

We have $\psi^\diamond[k+1] = \psi^\diamond[k]\mathbf{P}^\diamond \ (k \in \mathbb{N})$.

The steady-state PMF $\psi^\diamond = (\psi^\diamond(s_1), \ldots, \psi^\diamond(s_m))$ for $RDTMC(G)$ is a solution of:

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

where $\mathbf{I}$ is the identity matrix of size m and $\mathbf{0}$ is a row vector of m values 0, $\mathbf{1}$ is that of m values 1.

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

Proposition 1 (PMFSMCT) *Let G be a dynamic expression, φ be the steady-state PMF for $SMC(G)$ and $\psi^\diamond$ be the steady-state PMF for $RDTMC(G)$. Then $\forall s \in DR(G)$*

$$\varphi(s) = \begin{cases} \psi^\diamond(s), & s \in DR_T(G); \\ 0, & s \in DR_V(G). \end{cases}$$

To calculate φ, we take all the elements of $\psi^\diamond$ as the steady-state probabilities of the tangible states, instead of abstracting from self-loops to get $\mathbf{P}^*$ and ψ^*, followed by weighting by SJ and normalization.

Using $RDTMC(G)$ instead of $EDTMC(G)$ allows one to avoid multistage analysis.

Constructing $\mathbf{P}^\diamond$ requires calculating matrix powers or inverse matrices.

$RDTMC(G)$ has self-loops, unlike $EDTMC(G)$, hence,
the behaviour of $RDTMC(G)$ may stabilize slower than that of $EDTMC(G)$.
$\mathbf{P}^\diamond$ is smaller and denser matrix than $\mathbf{P}^*$, since $\mathbf{P}^\diamond$ has non-zero elements
at the main diagonal and many of them outside it.

The complexity of the analytical calculation of $\psi^\diamond$ w.r.t. ψ^* depends on the model structure:
the number of vanishing states and loops among them.
Usually it is lower, since the matrix size reduction plays an important role.

The elimination of vanishing states.

- The system models with many immediate activities:

  significant simplification of the solution.

- The abstraction level of SMCs:

  decreases their impact to the solution complexity.

- The abstraction level of transition systems:

  allows immediate activities to specify logical structure.

From $TS(\overline{K})$, we can construct $RDTMC(\overline{K})$ and calculate $\tilde{\varphi}$ using it.

$DR_T(\overline{K}) = \{\tilde{s}_1, \tilde{s}_2, \tilde{s}_5, \tilde{s}_7, \tilde{s}_8, \tilde{s}_9\}$ and $DR_V(\overline{K}) = \{\tilde{s}_3, \tilde{s}_4, \tilde{s}_6\}$.

We reorder the elements of $DR(\overline{K})$ by

moving the equivalence classes of vanishing states to the first positions:

$\tilde{s}_3, \tilde{s}_4, \tilde{s}_6, \tilde{s}_1, \tilde{s}_2, \tilde{s}_5, \tilde{s}_7, \tilde{s}_8, \tilde{s}_9$.

The reordered TPM for $DTMC(\overline{K})$:

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

The result of the decomposing $\widetilde{\mathbf{P}}_r$:

$$\widetilde{\mathbf{C}} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \widetilde{\mathbf{D}} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}, \widetilde{\mathbf{E}} = \begin{pmatrix} 0 & 0 & 0 \\ \rho(1-\rho) & \rho(1-\rho) & \rho^2 \\ 0 & \rho^3 & 0 \\ \rho^3 & 0 & 0 \\ 0 & \rho^2 & 0 \\ \rho^2 & 0 & 0 \end{pmatrix},$$

$$\widetilde{\mathbf{F}} = \begin{pmatrix} 1-\rho^3 & \rho^3 & 0 & 0 & 0 & 0 \\ 0 & (1-\rho)^2 & 0 & 0 & 0 & 0 \\ 0 & \rho^2(1-\rho) & (1-\rho)(1-\rho^2) & 0 & \rho(1-\rho^2) & 0 \\ 0 & \rho^2(1-\rho) & 0 & (1-\rho)(1-\rho^2) & 0 & \rho(1-\rho^2) \\ 0 & 0 & 0 & 0 & 1-\rho^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1-\rho^2 \end{pmatrix}.$$

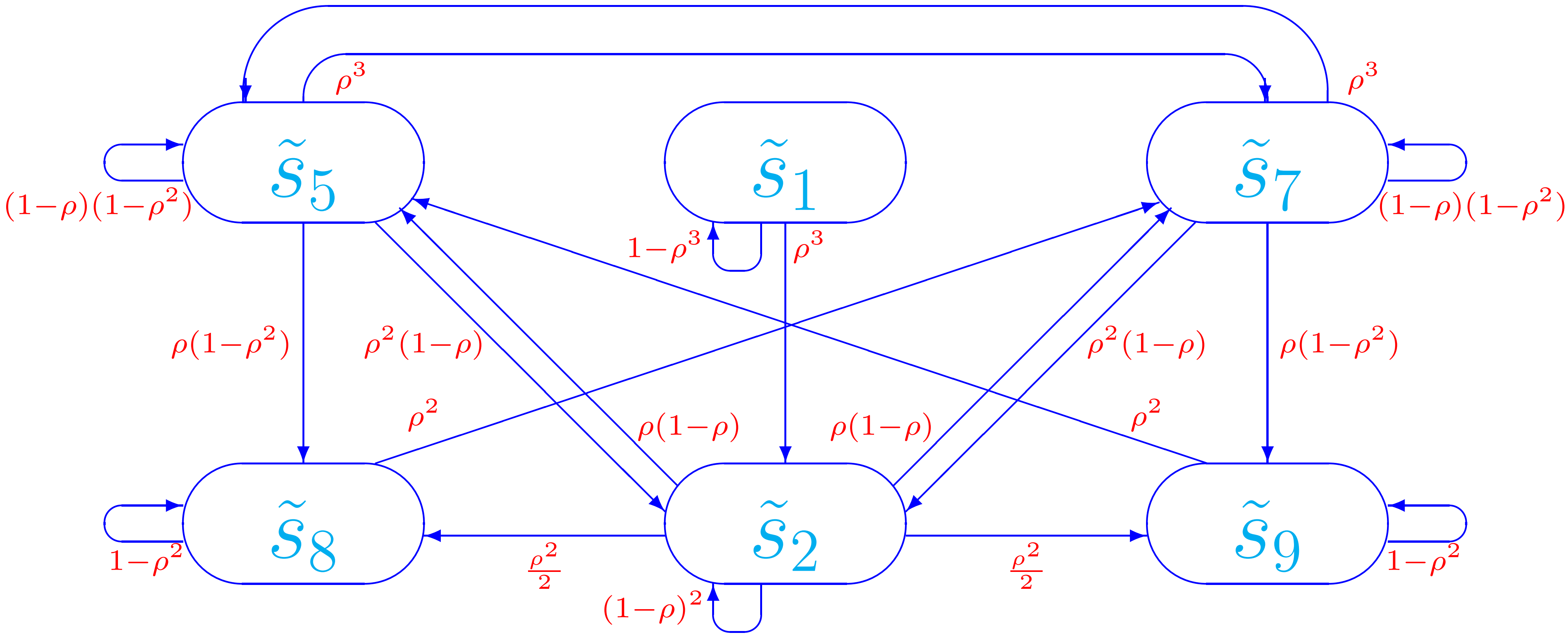
Since $\widetilde{\mathbf{C}}^1 = \mathbf{0}$, we have $\forall k > 0,\ \widetilde{\mathbf{C}}^k = \mathbf{0}$, hence, $l = 0$ and there are no loops among vanishing states. Then

$$\widetilde{\mathbf{G}} = \sum_{k=0}^{l} \widetilde{\mathbf{C}}^k = \widetilde{\mathbf{C}}^0 = \mathbf{I}.$$

The TPM for $RDTMC(\overline{K})$:

$$\widetilde{\mathbf{P}}^{\diamond} = \widetilde{\mathbf{F}} + \widetilde{\mathbf{E}}\widetilde{\mathbf{G}}\widetilde{\mathbf{D}} = \widetilde{\mathbf{F}} + \widetilde{\mathbf{E}}\mathbf{I}\widetilde{\mathbf{D}} = \widetilde{\mathbf{F}} + \widetilde{\mathbf{E}}\widetilde{\mathbf{D}} =$$
$$\begin{pmatrix} 1-\rho^3 & \rho^3 & 0 & 0 & 0 & 0 \\ 0 & (1-\rho)^2 & \rho(1-\rho) & \rho(1-\rho) & \frac{\rho^2}{2} & \frac{\rho^2}{2} \\ 0 & \rho^2(1-\rho) & (1-\rho)(1-\rho^2) & \rho^3 & \rho(1-\rho^2) & 0 \\ 0 & \rho^2(1-\rho) & \rho^3 & (1-\rho)(1-\rho^2) & 0 & \rho(1-\rho^2) \\ 0 & 0 & 0 & \rho^2 & 1-\rho^2 & 0 \\ 0 & 0 & \rho^2 & 0 & 0 & 1-\rho^2 \end{pmatrix}.$$

$RDTMC(\overline{K})$



SHMGRDTMC: The reduced DTMC of the generalized shared memory system

The steady-state PMF for $RDTMC(\overline{K})$:

$$\tilde{\psi}^{\diamond} = \frac{1}{2(2+\rho-\rho^2-\rho^3)}(0, 2\rho^2(1-\rho), \rho(2-\rho), \rho(2-\rho), 2-\rho-\rho^2, 2-\rho-\rho^2).$$

Note that $\tilde{\psi}^{\diamond} = (\tilde{\psi}^{\diamond}(\tilde{s}_1), \tilde{\psi}^{\diamond}(\tilde{s}_2), \tilde{\psi}^{\diamond}(\tilde{s}_5), \tilde{\psi}^{\diamond}(\tilde{s}_7), \tilde{\psi}^{\diamond}(\tilde{s}_8), \tilde{\psi}^{\diamond}(\tilde{s}_9))$.

By Proposition PMFSMCT:

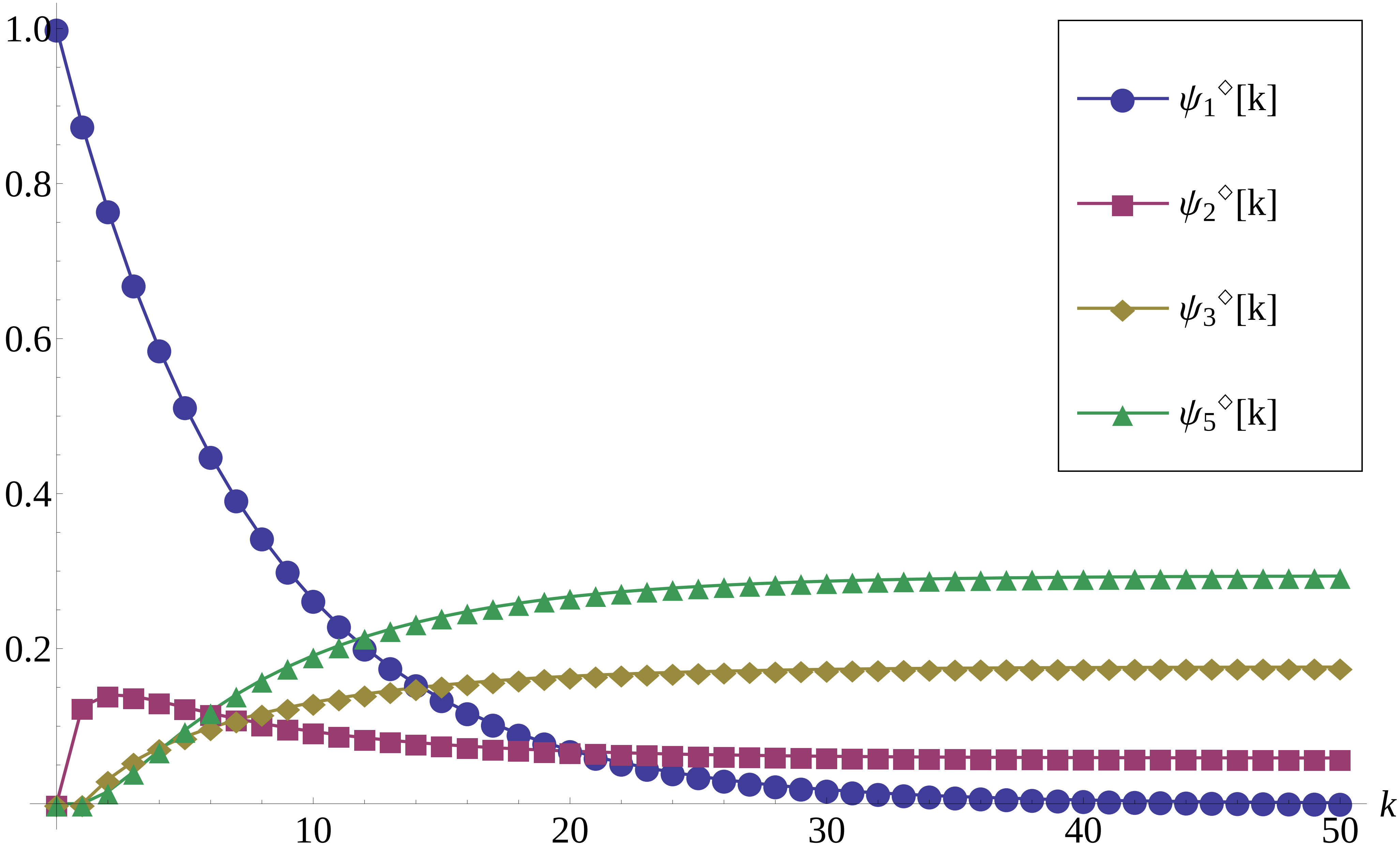
$$\tilde{\varphi}(\tilde{s}_1) = 0, \qquad \tilde{\varphi}(\tilde{s}_2) = \frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3}, \qquad \tilde{\varphi}(\tilde{s}_5) = \frac{\rho(2-\rho)}{2(2+\rho-\rho^2-\rho^3)},$$

$$\tilde{\varphi}(\tilde{s}_7) = \frac{\rho(2-\rho)}{2(2+\rho-\rho^2-\rho^3)}, \qquad \tilde{\varphi}(\tilde{s}_8) = \frac{2-\rho-\rho^2}{2(2+\rho-\rho^2-\rho^3)}, \qquad \tilde{\varphi}(\tilde{s}_9) = \frac{2-\rho-\rho^2}{2(2+\rho-\rho^2-\rho^3)}.$$

The steady-state PMF for $SMC(\overline{K})$:

$$\tilde{\varphi} = \frac{1}{2(2+\rho-\rho^2-\rho^3)}(0, 2\rho^2(1-\rho), 0, 0, \rho(2-\rho), 0, \rho(2-\rho), 2-\rho-\rho^2, 2-\rho-\rho^2).$$

This coincides with the result obtained with the use of $\tilde{\psi}^*$ and $\widetilde{SJ}$.



SHMTRPR: Transient probabilities alteration diagram for the RDTMC of the generalized shared memory system when $\rho = \frac{1}{2}$

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

The following performance indices (measures) are based on the steady-state PMF for $SMC(G)$.

- The *average recurrence (return) time in the state* s (the number of discrete time units or steps required for this) is $\frac{1}{\varphi(s)}$.
- The *fraction of residence time in the state* s is $\varphi(s)$.
- 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 $\sum_{s\in S}\varphi(s)$.
- The *relative fraction of residence time in the set of states* S *w.r.t. that in* $\widetilde{S}$ is $\frac{\sum_{s\in S}\varphi(s)}{\sum_{\tilde{s}\in\widetilde{S}}\varphi(\tilde{s})}$.
- The *rate of leaving the state* s is $\frac{\varphi(s)}{SJ(s)}$.
- The *steady-state probability to perform a step with an activity* (α, κ) is $\sum_{s\in DR(G)}\varphi(s)\sum_{\{\Upsilon\mid(\alpha,\kappa)\in\Upsilon\}}PT(\Upsilon, s)$.
- The *probability of the event determined by a reward function* r *on the states* is $\sum_{s\in DR(G)}\varphi(s)r(s)$, where $\forall s\in DR(G)\ 0\leq r(s)\leq 1$.

Performance indices of the generalized shared memory system

- The average recurrence time in the state $\tilde{s}_2$, where no processor requests the memory, the *average system run-through*, is $\frac{1}{\tilde{\varphi}_2} = \frac{2+\rho-\rho^2-\rho^3}{\rho^2(1-\rho)}$.
- The common memory is available only in the states $\tilde{s}_2, \tilde{s}_3, \tilde{s}_4, \tilde{s}_6$.

  The steady-state probability that the memory is available is
  $\tilde{\varphi}_2 + \tilde{\varphi}_3 + \tilde{\varphi}_4 + \tilde{\varphi}_6 = \frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3} + 0 + 0 + 0 = \frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3}$.
  The steady-state probability that the memory is used (i.e. not available), the *shared memory utilization*, is $1 - \frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3} = \frac{2+\rho-2\rho^2}{2+\rho-\rho^2-\rho^3}$.
- After activation of the system, we leave the state $\tilde{s}_1$ for all, and the common memory is either requested or allocated in every remaining state, with exception of $\tilde{s}_2$.

  The *rate with which the necessity of shared memory emerges* coincides with the rate of leaving $\tilde{s}_2$, calculated as $\frac{\tilde{\varphi}_2}{\widetilde{SJ}_2} = \frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3} \cdot \frac{\rho(2-\rho)}{1} = \frac{\rho^3(1-\rho)(2-\rho)}{2+\rho-\rho^2-\rho^3}$.

- The common memory request of the first processor $(\{r_1\},\rho)$ is only possible from the states $\tilde{s}_2, \tilde{s}_7$.

  The request probability in each of the states is the sum of the execution probabilities for all multisets of activities containing $(\{r_1\},\rho)$.

  The *steady-state probability of the shared memory request from the first processor* is
  $\tilde{\varphi}_2 \sum_{\{\Upsilon \mid (\{r_1\},\rho)\in\Upsilon\}} PT(\Upsilon,\tilde{s}_2) + \tilde{\varphi}_7 \sum_{\{\Upsilon \mid (\{r_1\},\rho)\in\Upsilon\}} PT(\Upsilon,\tilde{s}_7) =$
  $\frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3}(\rho(1-\rho)+\rho^2) + \frac{\rho(2-\rho)}{2(2+\rho-\rho^2-\rho^3)}(\rho(1-\rho^2)+\rho^3) = \frac{\rho^2(2+\rho-2\rho^2)}{2(2+\rho-\rho^2-\rho^3)}$.

Stochastic equivalences

Step stochastic bisimulation equivalence

For $\Upsilon \in I\!N_{fin}^{\mathcal{SIL}}$, we consider $\mathcal{L}(\Upsilon) \in I\!N_{fin}^{\mathcal{L}}$, i.e. (possibly empty) multisets of multiactions.

Let G be a dynamic expression and $\mathcal{H} \subseteq DR(G)$. For $s \in DR(G)$ and $A \in I\!N_{fin}^{\mathcal{L}}$ we write $s \xrightarrow{A}_{\mathcal{P}} \mathcal{H}$, where $\mathcal{P} = PM_A(s, \mathcal{H})$ is the *overall probability to move from s into the set of states $\mathcal{H}$ via steps with the multiaction part A*:

$$PM_A(s, \mathcal{H}) = \sum_{\{\Upsilon \mid \exists \tilde{s} \in \mathcal{H}\ s \xrightarrow{\Upsilon} \tilde{s},\ \mathcal{L}(\Upsilon) = A\}} PT(\Upsilon, s).$$

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

We write $s \rightarrow_{\mathcal{P}} \mathcal{H}$ if $\exists A\ s \xrightarrow{A} \mathcal{H}$, where $\mathcal{P} = PM(s, \mathcal{H})$ is the *overall probability to move from s into the set of states $\mathcal{H}$ via any steps*:

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

Definition 13 *Let G and G' be dynamic expressions. An equivalence relation $\mathcal{R} \subseteq (DR(G) \cup DR(G'))^2$ is a step stochastic bisimulation between G and G', $\mathcal{R} : G \underline{\leftrightarrow}_{ss} G'$, if:*

1. $([G]_\approx, [G']_\approx) \in \mathcal{R}$.
2. $(s_1, s_2) \in \mathcal{R} \Rightarrow \forall \mathcal{H} \in (DR(G) \cup DR(G'))/_\mathcal{R}\ \forall A \in I\!N_{fin}^{\mathcal{L}}$

$$s_1 \xrightarrow{A}_\mathcal{P} \mathcal{H} \Leftrightarrow s_2 \xrightarrow{A}_\mathcal{P} \mathcal{H}.$$

Two dynamic expressions G and G' are step stochastic bisimulation equivalent, $G \underline{\leftrightarrow}_{ss} G'$, if $\exists \mathcal{R} : G \underline{\leftrightarrow}_{ss} G'$.

Proposition 2 *(BISSPL) Let G and G' be dynamic expressions and $\mathcal{R} : G \underline{\leftrightarrow}_{ss} G'$. Then*

$$\mathcal{R} \subseteq (DR_T(G) \cup DR_T(G'))^2 \uplus (DR_V(G) \cup DR_V(G'))^2,$$

where $\uplus$ is disjoint union.

$\mathcal{R}_{ss}(G, G') = \bigcup\{\mathcal{R} \mid \mathcal{R} : G \underline{\leftrightarrow}_{ss} G'\}$ is the *union of all step stochastic bisimulations* between G and G'.

Proposition 3 *(LARBIS) Let G and G' be dynamic expressions and $G \underline{\leftrightarrow}_{ss} G'$. Then $\mathcal{R}_{ss}(G, G')$ is the largest step stochastic bisimulation between G and G'.*

Reduction modulo equivalences

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

For a dynamic expression G and a step stochastic autobisimulation $\mathcal{R} : G\underline{\leftrightarrow}_{ss}G$, let $\mathcal{K} \in DR(G)/_{\mathcal{R}}$ and $s_1, s_2 \in \mathcal{K}$.

We have $\forall \widetilde{\mathcal{K}} \in DR(G)/_{\mathcal{R}}\ \forall A \in I\!N_{fin}^{\mathcal{L}}\ s_1 \stackrel{A}{\rightarrow}_{\mathcal{P}} \widetilde{\mathcal{K}} \ \Leftrightarrow\ s_2 \stackrel{A}{\rightarrow}_{\mathcal{P}} \widetilde{\mathcal{K}}$.

The equality is valid for all $s_1, s_2 \in \mathcal{K}$, hence, we can rewrite it as $\mathcal{K}\stackrel{A}{\rightarrow}_{\mathcal{P}}\widetilde{\mathcal{K}}$, 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 $\mathcal{K}\stackrel{A}{\rightarrow}\widetilde{\mathcal{K}}$ if $\exists \mathcal{P}\ \mathcal{K} \stackrel{A}{\rightarrow}_{\mathcal{P}} \widetilde{\mathcal{K}}$ and $\mathcal{K}\rightarrow\widetilde{\mathcal{K}}$ if $\exists A\ \mathcal{K} \stackrel{A}{\rightarrow} \widetilde{\mathcal{K}}$.

The similar arguments: we write $\mathcal{K}\rightarrow_{\mathcal{P}}\widetilde{\mathcal{K}}$, where $\mathcal{P} = PM(\mathcal{K}, \widetilde{\mathcal{K}}) = PM(s_1, \widetilde{\mathcal{K}}) = PM(s_2, \widetilde{\mathcal{K}})$.

Since $\mathcal{R} \subseteq (DR_T(G))^2 \uplus (DR_V(G))^2$, we have $\forall \mathcal{K} \in DR(G)/_{\mathcal{R}}$, all states from $\mathcal{K}$ are tangible, when $\mathcal{K} \in DR_T(G)/_{\mathcal{R}}$, or all of them are vanishing, when $\mathcal{K} \in DR_V(G)/_{\mathcal{R}}$.

The *average sojourn time in the equivalence class (w.r.t. $\mathcal{R}$) of states $\mathcal{K}$* is

$$SJ_{\mathcal{R}}(\mathcal{K}) = \begin{cases} \frac{1}{1-PM(\mathcal{K},\mathcal{K})}, & \mathcal{K} \in DR_T(G)/_{\mathcal{R}}; \\ 0, & \mathcal{K} \in DR_V(G)/_{\mathcal{R}}. \end{cases}$$

The *average sojourn time vector for the equivalence classes (w.r.t. $\mathcal{R}$) of states* of G, $SJ_{\mathcal{R}}$, has the elements $SJ_{\mathcal{R}}(\mathcal{K}),\ \mathcal{K} \in DR(G)/_{\mathcal{R}}$.

The *sojourn time variance in the equivalence class (w.r.t. $\mathcal{R}$) of states $\mathcal{K}$* is

$$VAR_{\mathcal{R}}(\mathcal{K}) = \begin{cases} \frac{PM(\mathcal{K},\mathcal{K})}{(1-PM(\mathcal{K},\mathcal{K}))^2}, & \mathcal{K} \in DR_T(G)/_{\mathcal{R}}; \\ 0, & \mathcal{K} \in DR_V(G)/_{\mathcal{R}}. \end{cases}$$

The *sojourn time variance vector for the equivalence classes (w.r.t. $\mathcal{R}$) of states* of G, $VAR_{\mathcal{R}}$, has the elements $VAR_{\mathcal{R}}(\mathcal{K}),\ \mathcal{K} \in DR(G)/_{\mathcal{R}}$.

$\mathcal{R}_{ss}(G) = \bigcup\{\mathcal{R} \mid \mathcal{R} : G \underline{\leftrightarrow}_{ss} G\}$ is the *largest step stochastic autobisimulation* on G.

Definition 14 *The* quotient (by $\underline{\leftrightarrow}_{ss}$) (labeled probabilistic) transition system *of a dynamic expression* G *is* $TS_{\underline{\leftrightarrow}_{ss}}(G) = (S_{\underline{\leftrightarrow}_{ss}}, L_{\underline{\leftrightarrow}_{ss}}, \mathcal{T}_{\underline{\leftrightarrow}_{ss}}, s_{\underline{\leftrightarrow}_{ss}})$*, where*

- $S_{\underline{\leftrightarrow}_{ss}} = DR(G)/_{\mathcal{R}_{ss}(G)}$;
- $L_{\underline{\leftrightarrow}_{ss}} \subseteq (I\!N_{fin}^{\mathcal{L}}) \times (0;1]$;
- $\mathcal{T}_{\underline{\leftrightarrow}_{ss}} = \{(\mathcal{K}, (A, PM_A(\mathcal{K}, \widetilde{\mathcal{K}})), \widetilde{\mathcal{K}}) \mid \mathcal{K}, \widetilde{\mathcal{K}} \in DR(G)/_{\mathcal{R}_{ss}(G)},\ \mathcal{K} \xrightarrow{A} \widetilde{\mathcal{K}}\}$;
- $s_{\underline{\leftrightarrow}_{ss}} = [[G]_{\approx}]_{\mathcal{R}_{ss}(G)}$.

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

The abstract generalized shared memory system and its reduction

The static expression of the first processor is

$$L_1 = [(\{x_1\}, \rho) * ((\{r\}, \rho); (\{d, y_1\}, l); (\{m, z_1\}, \rho)) * \mathsf{Stop}].$$

The static expression of the second processor is

$$L_2 = [(\{x_2\}, \rho) * ((\{r\}, \rho); (\{d, y_2\}, l); (\{m, z_2\}, \rho)) * \mathsf{Stop}].$$

The static expression of the shared memory is

$$L_3 = [(\{a, \widehat{x_1}, \widehat{x_2}\}, \rho) * (((\{\widehat{y_1}\}, l); (\{\widehat{z_1}\}, \rho))[]((\{\widehat{y_2}\}, l); (\{\widehat{z_2}\}, \rho))) * \mathsf{Stop}].$$

The static expression of the abstract generalized shared memory system with two processors is

$$L = (L_1 \| L_2 \| L_3) \text{ sy } x_1 \text{ sy } x_2 \text{ sy } y_1 \text{ sy } y_2 \text{ sy } z_1 \text{ sy } z_2 \text{ rs } x_1 \text{ rs } x_2 \text{ rs } y_1 \text{ rs } y_2 \text{ rs } z_1 \text{ rs } z_2.$$

$DR(\overline{L})$ resembles $DR(\overline{K})$, and $TS(\overline{L})$ is similar to $TS(\overline{K})$.

$SMC(\overline{L}) \simeq SMC(\overline{K})$, thus, the average sojourn time vectors of $\overline{L}$ and $\overline{K}$,
the TPMs and the steady-state PMFs for $EDTMC(\overline{L})$ and $EDTMC(\overline{K})$ coincide.

Performance indices of the abstract generalized shared memory system

The first, second and third performance indices are the same

for the generalized system and its abstract modification.

The following performance index: non-identified viewpoint to the processors.

- The common memory request of a processor $(\{r\},\rho)$ is only possible from the states $\tilde{s}_2, \tilde{s}_5, \tilde{s}_7$.

  The request probability in each of the states is the sum of the execution probabilities for all multisets of activities containing $(\{r\},\rho)$.

  The *steady-state probability of the shared memory request from a processor* is

  $\tilde{\varphi}_2 \sum_{\{\Upsilon|(\{r\},\rho)\in\Upsilon\}} PT(\Upsilon,\tilde{s}_2) + \tilde{\varphi}_5 \sum_{\{\Upsilon|(\{r\},\rho)\in\Upsilon\}} PT(\Upsilon,\tilde{s}_5) + \tilde{\varphi}_7 \sum_{\{\Upsilon|(\{r\},\rho)\in\Upsilon\}} PT(\Upsilon,\tilde{s}_7) = \frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3}(\rho(1-\rho)+\rho(1-\rho)+\rho^2) + \frac{\rho(2-\rho)}{2(2+\rho-\rho^2-\rho^3)}(\rho(1-\rho^2)+\rho^3) + \frac{\rho(2-\rho)}{2(2+\rho-\rho^2-\rho^3)}(\rho(1-\rho^2)+\rho^3) = \frac{\rho^2(2-\rho)(1+\rho-\rho^2)}{2+\rho-\rho^2-\rho^3}$.

The quotient of the abstract generalized shared memory system

$DR(\overline{L})/_{\mathcal{R}_{ss}(\overline{L})} = \{\widetilde{\mathcal{K}}_1, \widetilde{\mathcal{K}}_2, \widetilde{\mathcal{K}}_3, \widetilde{\mathcal{K}}_4, \widetilde{\mathcal{K}}_5, \widetilde{\mathcal{K}}_6\}$, where

$\widetilde{\mathcal{K}}_1 = \{\tilde{s}_1\}$ (the initial state),

$\widetilde{\mathcal{K}}_2 = \{\tilde{s}_2\}$ (the system is activated and the memory is not requested),

$\widetilde{\mathcal{K}}_3 = \{\tilde{s}_3, \tilde{s}_4\}$ (the memory is requested by one processor),

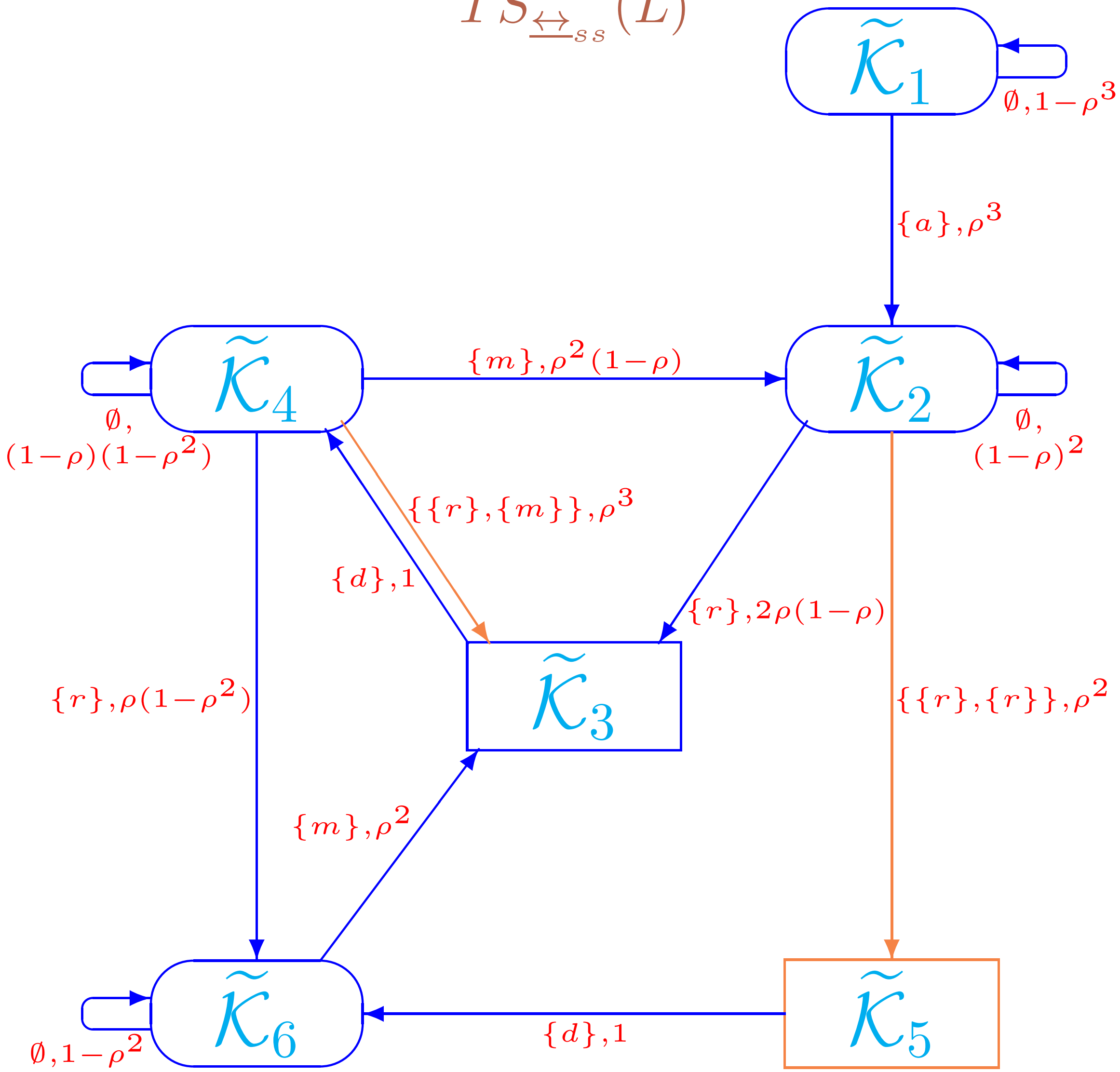
$\widetilde{\mathcal{K}}_4 = \{\tilde{s}_5, \tilde{s}_7\}$ (the memory is allocated to a processor),

$\widetilde{\mathcal{K}}_5 = \{\tilde{s}_6\}$ (the memory is requested by two processors),

$\widetilde{\mathcal{K}}_6 = \{\tilde{s}_8, \tilde{s}_9\}$ (the memory is allocated to a processor and the memory is requested by another processor).

$DR_T(\overline{L})/_{\mathcal{R}_{ss}(\overline{L})} = \{\widetilde{\mathcal{K}}_1, \widetilde{\mathcal{K}}_2, \widetilde{\mathcal{K}}_4, \widetilde{\mathcal{K}}_6\}$ and $DR_V(\overline{L})/_{\mathcal{R}_{ss}(\overline{L})} = \{\widetilde{\mathcal{K}}_3, \widetilde{\mathcal{K}}_5\}$.

$TS_{\underline{\leftrightarrow}_{ss}}(\overline{L})$



SHMGQTS: The quotient transition system of the abstract generalized shared memory system

(parallel executions of activities and the exclusively reachable states are marked with orange)

The *quotient (by $\underline{\leftrightarrow}_{ss}$) average sojourn time vector* of G is $SJ_{\underline{\leftrightarrow}_{ss}} = SJ_{\mathcal{R}_{ss}(G)}$.

The *quotient (by $\underline{\leftrightarrow}_{ss}$) sojourn time variance vector* of G is $VAR_{\underline{\leftrightarrow}_{ss}} = VAR_{\mathcal{R}_{ss}(G)}$.

Let $\mathcal{K} \to \widetilde{\mathcal{K}}$ and $\mathcal{K} \neq \widetilde{\mathcal{K}}$. The *probability to move from $\mathcal{K}$ to $\widetilde{\mathcal{K}}$ by executing any multiset of activities after possible self-loops* is

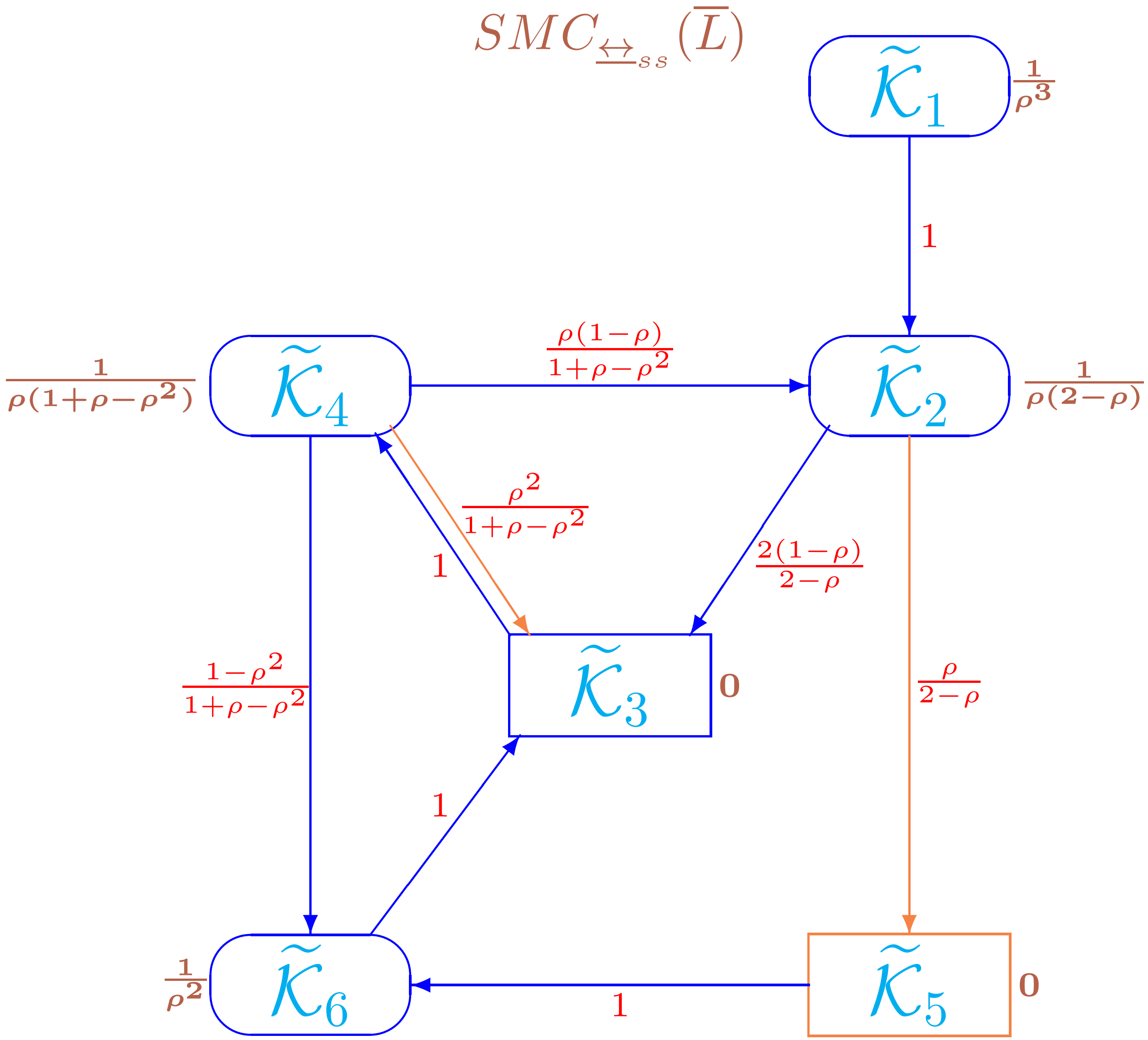
$$PM^*(\mathcal{K}, \widetilde{\mathcal{K}}) = \begin{cases} PM(\mathcal{K}, \widetilde{\mathcal{K}}) \sum_{k=0}^{\infty} PM(\mathcal{K}, \mathcal{K})^k = \frac{PM(\mathcal{K}, \widetilde{\mathcal{K}})}{1 - PM(\mathcal{K}, \mathcal{K})}, & \mathcal{K} \to \mathcal{K}; \\ PM(\mathcal{K}, \widetilde{\mathcal{K}}), & \text{otherwise}. \end{cases}$$

We have $\forall \mathcal{K} \in DR_T(G)/_{\mathcal{R}_{ss}(G)}\ PM^*(\mathcal{K}, \widetilde{\mathcal{K}}) = SJ_{\underline{\leftrightarrow}_{ss}}(\mathcal{K}) PM(\mathcal{K}, \widetilde{\mathcal{K}})$.

Definition 15 *The* quotient (by $\underline{\leftrightarrow}_{ss}$) EDTMC *of a dynamic expression* G, $EDTMC_{\underline{\leftrightarrow}_{ss}}(G)$, *has the state space* $DR(G)/_{\mathcal{R}_{ss}(G)}$*, the initial state* $[[G]_{\approx}]_{\mathcal{R}_{ss}(G)}$ *and the transitions* $\mathcal{K} \twoheadrightarrow_{\mathcal{P}} \widetilde{\mathcal{K}}$*, if* $\mathcal{K} \to \widetilde{\mathcal{K}}$ *and* $\mathcal{K} \neq \widetilde{\mathcal{K}}$*, where* $\mathcal{P} = PM^*(\mathcal{K}, \widetilde{\mathcal{K}})$*.*

The quotient (by $\underline{\leftrightarrow}_{ss}$) underlying SMC *of* G, $SMC_{\underline{\leftrightarrow}_{ss}}(G)$*, has the EDTMC* $EDTMC_{\underline{\leftrightarrow}_{ss}}(G)$ *and the sojourn time in every* $\mathcal{K} \in DR_T(G)/_{\mathcal{R}_{ss}(G)}$ *is geometrically distributed with the parameter* $1 - PM(\mathcal{K}, \mathcal{K})$ *while the sojourn time in every* $\mathcal{K} \in DR_V(G)/_{\mathcal{R}_{ss}(G)}$ *is equal to zero.*

The steady-state PMFs $\psi^*_{\underline{\leftrightarrow}_{ss}}$ for $EDTMC_{\underline{\leftrightarrow}_{ss}}(G)$ and $\varphi_{\underline{\leftrightarrow}_{ss}}$ for $SMC_{\underline{\leftrightarrow}_{ss}}(G)$ are defined like ψ^* for $EDTMC(G)$ and φ for $SMC(G)$.



SHMGQSMC: The quotient underlying SMC of the abstract generalized shared memory system

(parallel executions of activities and the exclusively reachable states are marked with orange)

The quotient average sojourn time vector of $\overline{F}$:

$$\widetilde{SJ}' = \left(\frac{1}{\rho^3}, \frac{1}{\rho(2-\rho)}, 0, \frac{1}{\rho(1+\rho-\rho^2)}, 0, \frac{1}{\rho^2}\right).$$

The quotient sojourn time variance vector of $\overline{F}$:

$$\widetilde{VAR}' = \left(\frac{1-\rho^3}{\rho^6}, \frac{(1-\rho)^2}{\rho^2(2-\rho)^2}, 0, \frac{(1-\rho)^2(1+\rho)}{\rho^2(1+\rho-\rho^2)^2}, 0, \frac{1-\rho^2}{\rho^4}\right).$$

The TPM for $EDTMC_{\underline{\leftrightarrow}_{ss}}(\overline{L})$:

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

The steady-state PMF for $EDTMC_{\underline{\leftrightarrow}_{ss}}(\overline{L})$:

$$\tilde{\psi}'^{*} = \frac{1}{6+3\rho-9\rho^2+2\rho^3}(0, \rho(2-3\rho+\rho^2), 2+\rho-3\rho^2+\rho^3, 2+\rho-3\rho^2+\rho^3, \rho^2(1-\rho), 2-\rho-\rho^2).$$

The steady-state PMF $\tilde{\psi}'^*$ weighted by $\widetilde{SJ}'$:

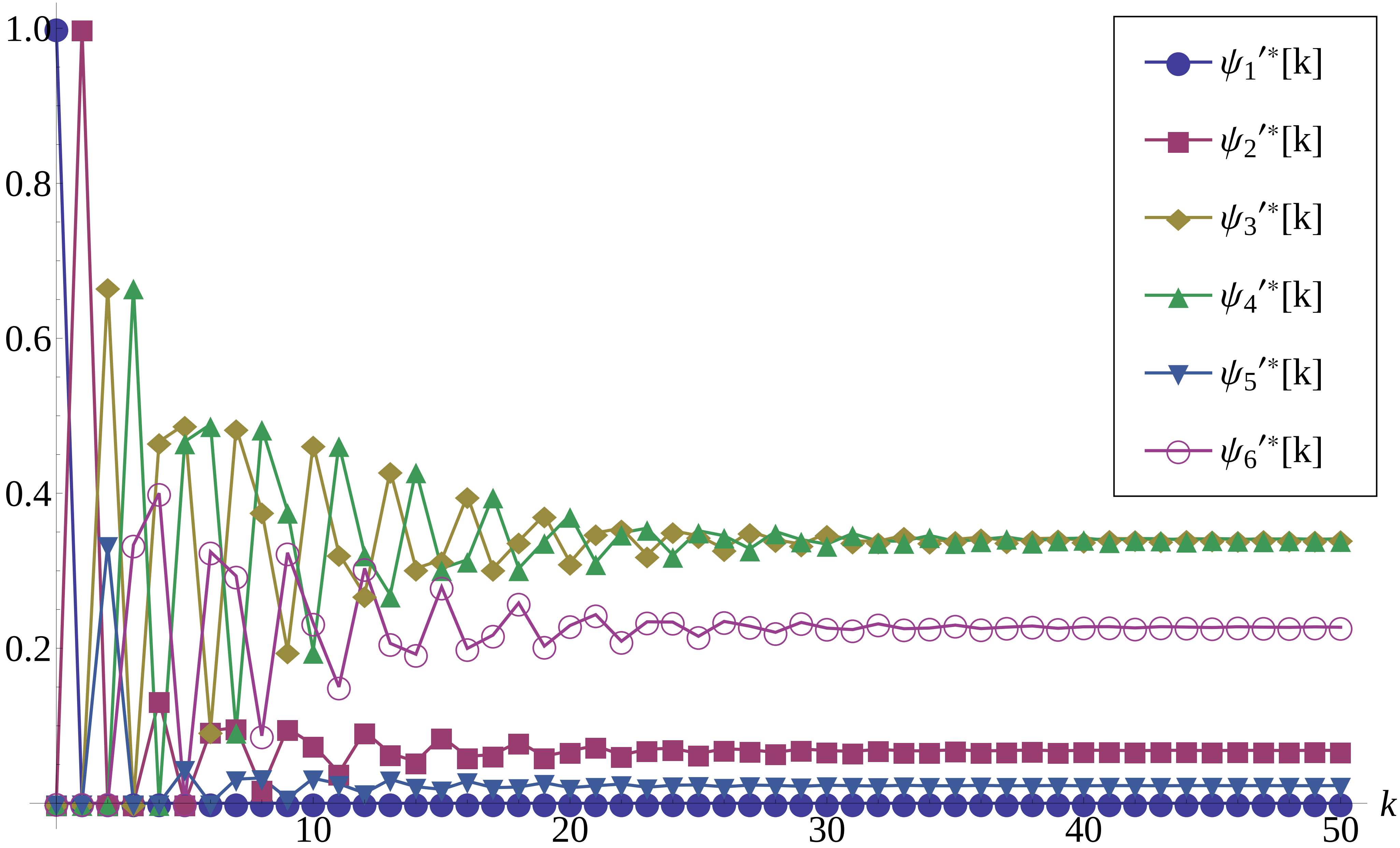
$$\frac{1}{\rho^2(6+3\rho-9\rho^2+2\rho^3)}(0,\rho^2(1-\rho),0,\rho(2-\rho),0,2-\rho-\rho^2).$$

We normalize the steady-state weighted PMF dividing it by the sum of its components

$$\tilde{\psi}'^*\widetilde{SJ}'^T = \frac{2+\rho-\rho^2-\rho^3}{\rho^2(6+3\rho-9\rho^2+2\rho^3)}.$$

The steady-state PMF for $SMC_{\underline{\leftrightarrow}_{ss}}(\overline{L})$:

$$\tilde{\varphi}' = \frac{1}{2+\rho-\rho^2-\rho^3}(0,\rho^2(1-\rho),0,\rho(2-\rho),0,2-\rho-\rho^2).$$



SHMQTP: Transient probabilities alteration diagram for the quotient EDTMC of the abstract generalized shared memory system when $\rho = \frac{1}{2}$

Definition 16 *Let* G *be a dynamic expression. The* quotient (by $\underline{\leftrightarrow}_{ss}$) DTMC *of* G, $DTMC_{\underline{\leftrightarrow}_{ss}}(G)$, *has the state space* $DR(G)/_{\mathcal{R}_{ss}(G)}$, *the initial state* $[[G]_\approx]_{\mathcal{R}_{ss}(G)}$ *and the transitions* $\mathcal{K} \rightarrow_{\mathcal{P}} \widetilde{\mathcal{K}}$, *where* $\mathcal{P} = PM(\mathcal{K}, \widetilde{\mathcal{K}})$.

Definition 17 *The* reduced quotient (by $\underline{\leftrightarrow}_{ss}$) DTMC *of* G, *denoted by* $RDTMC_{\underline{\leftrightarrow}_{ss}}(G)$, *is defined like* $RDTMC(G)$, *but it is constructed from* $DTMC_{\underline{\leftrightarrow}_{ss}}(G)$ *instead of* $DTMC(G)$.

The steady-state PMFs $\psi_{\underline{\leftrightarrow}_{ss}}$ for $DTMC_{\underline{\leftrightarrow}_{ss}}(G)$ and $\psi^\diamond_{\underline{\leftrightarrow}_{ss}}$ for $RDTMC_{\underline{\leftrightarrow}_{ss}}(G)$ are defined like ψ for $DTMC(G)$ and $\psi^\diamond$ for $RDTMC(G)$.

The relationships between the steady-state PMFs $\psi_{\underline{\leftrightarrow}_{ss}}$ and $\psi^*_{\underline{\leftrightarrow}_{ss}}$, $\varphi_{\underline{\leftrightarrow}_{ss}}$ and $\psi_{\underline{\leftrightarrow}_{ss}}$, $\varphi_{\underline{\leftrightarrow}_{ss}}$ and $\psi^\diamond_{\underline{\leftrightarrow}_{ss}}$ are the same as those between their "non-quotient" versions.

From $TS_{\underline{\leftrightarrow}_{ss}}(\overline{L})$, we can construct $RDTMC_{\underline{\leftrightarrow}_{ss}}(\overline{L})$ and calculate $\tilde{\varphi}'$ using it.

$DR_T(\overline{L})/_{\mathcal{R}_{ss}(\overline{L})} = \{\widetilde{\mathcal{K}}_1, \widetilde{\mathcal{K}}_2, \widetilde{\mathcal{K}}_4, \widetilde{\mathcal{K}}_6\}$ and $DR_V(\overline{L})/_{\mathcal{R}_{ss}(\overline{L})} = \{\widetilde{\mathcal{K}}_3, \widetilde{\mathcal{K}}_5\}$.

We reorder the elements of $DR(\overline{L})/_{\mathcal{R}_{ss}(\overline{L})}$ by moving the equivalence classes of vanishing states to the first positions: $\widetilde{\mathcal{K}}_3, \widetilde{\mathcal{K}}_5, \widetilde{\mathcal{K}}_1, \widetilde{\mathcal{K}}_2, \widetilde{\mathcal{K}}_4, \widetilde{\mathcal{K}}_6$.

The reordered TPM for $DTMC_{\underline{\leftrightarrow}_{ss}}(\overline{L})$:

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

The result of the decomposing $\widetilde{\mathbf{P}}'_r$:

$$\widetilde{\mathbf{C}}' = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, \ \widetilde{\mathbf{D}}' = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \ \widetilde{\mathbf{E}}' = \begin{pmatrix} 0 & 0 \\ 2\rho(1-\rho) & \rho^2 \\ \rho^3 & 0 \\ \rho^2 & 0 \end{pmatrix},$$

$$\widetilde{\mathbf{F}}' = \begin{pmatrix} 1-\rho^3 & \rho^3 & 0 & 0 \\ 0 & (1-\rho)^2 & 0 & 0 \\ 0 & \rho^2(1-\rho) & (1-\rho)(1-\rho^2) & \rho(1-\rho^2) \\ 0 & 0 & 0 & 1-\rho^2 \end{pmatrix}.$$

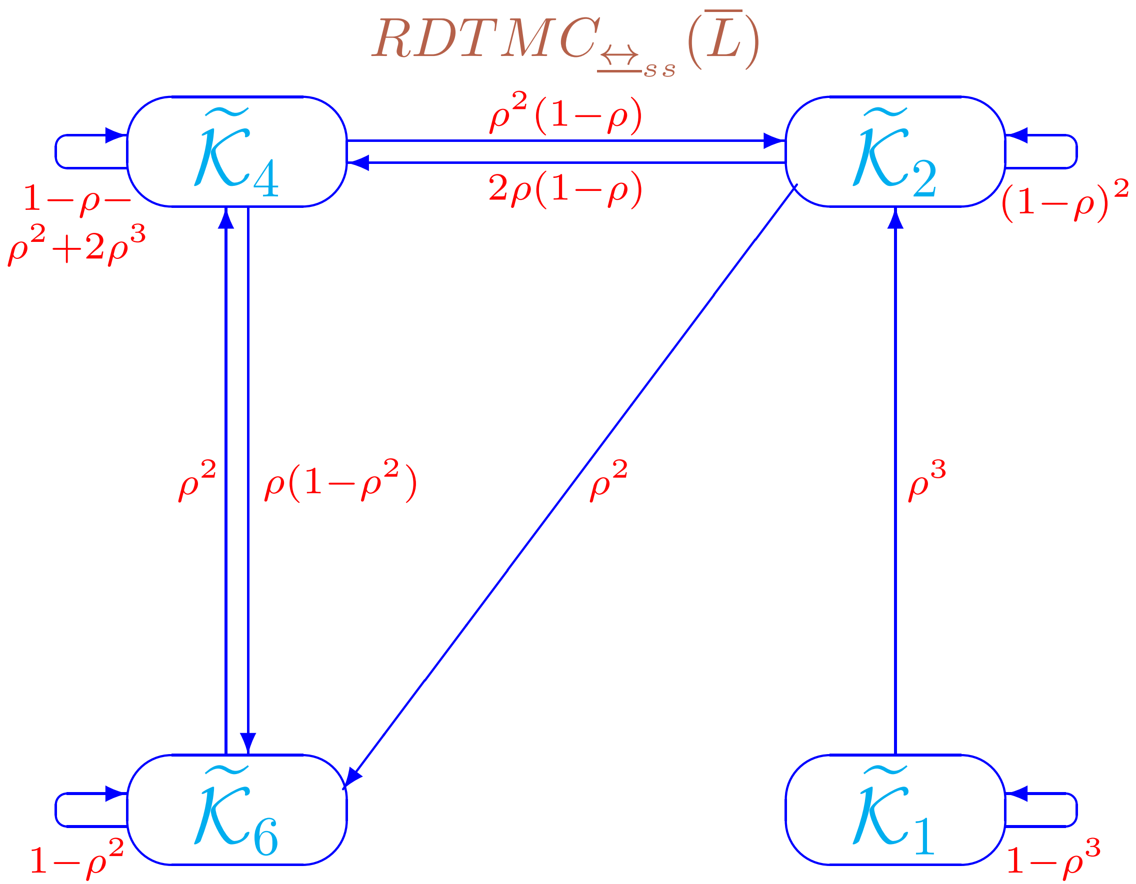
Since $\widetilde{\mathbf{C}}'^1 = \mathbf{0}$, we have $\forall k > 0,\ \widetilde{\mathbf{C}}'^k = \mathbf{0}$, hence, $l = 0$ and there are no loops among vanishing states. Then

$$\widetilde{\mathbf{G}}' = \sum_{k=0}^{l} \widetilde{\mathbf{C}}'^l = \widetilde{\mathbf{C}}'^0 = \mathbf{I}.$$

The TPM for $RDTMC_{\underline{\leftrightarrow}_{ss}}(\overline{L})$:

$$\widetilde{\mathbf{P}}'^{\diamond} = \widetilde{\mathbf{F}}' + \widetilde{\mathbf{E}}'\widetilde{\mathbf{G}}'\widetilde{\mathbf{D}}' = \widetilde{\mathbf{F}}' + \widetilde{\mathbf{E}}'\mathbf{I}\widetilde{\mathbf{D}}' = \widetilde{\mathbf{F}}' + \widetilde{\mathbf{E}}'\widetilde{\mathbf{D}}' =$$

$$\begin{pmatrix} 1-\rho^3 & \rho^3 & 0 & 0 \\ 0 & (1-\rho)^2 & 2\rho(1-\rho) & \rho^2 \\ 0 & \rho^2(1-\rho) & 1-\rho-\rho^2+2\rho^3 & \rho(1-\rho^2) \\ 0 & 0 & \rho^2 & 1-\rho^2 \end{pmatrix}.$$



SHMGQRDTMC: The reduced quotient DTMC of the abstract generalized shared memory system

The steady-state PMF for $RDTMC_{\underline{\leftrightarrow}_{ss}}(\overline{L})$:

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

Note that $\tilde{\psi}'^{\diamond} = (\tilde{\psi}'^{\diamond}(\widetilde{\mathcal{K}}_1), \tilde{\psi}'^{\diamond}(\widetilde{\mathcal{K}}_2), \tilde{\psi}'^{\diamond}(\widetilde{\mathcal{K}}_4), \tilde{\psi}'^{\diamond}(\widetilde{\mathcal{K}}_6))$.

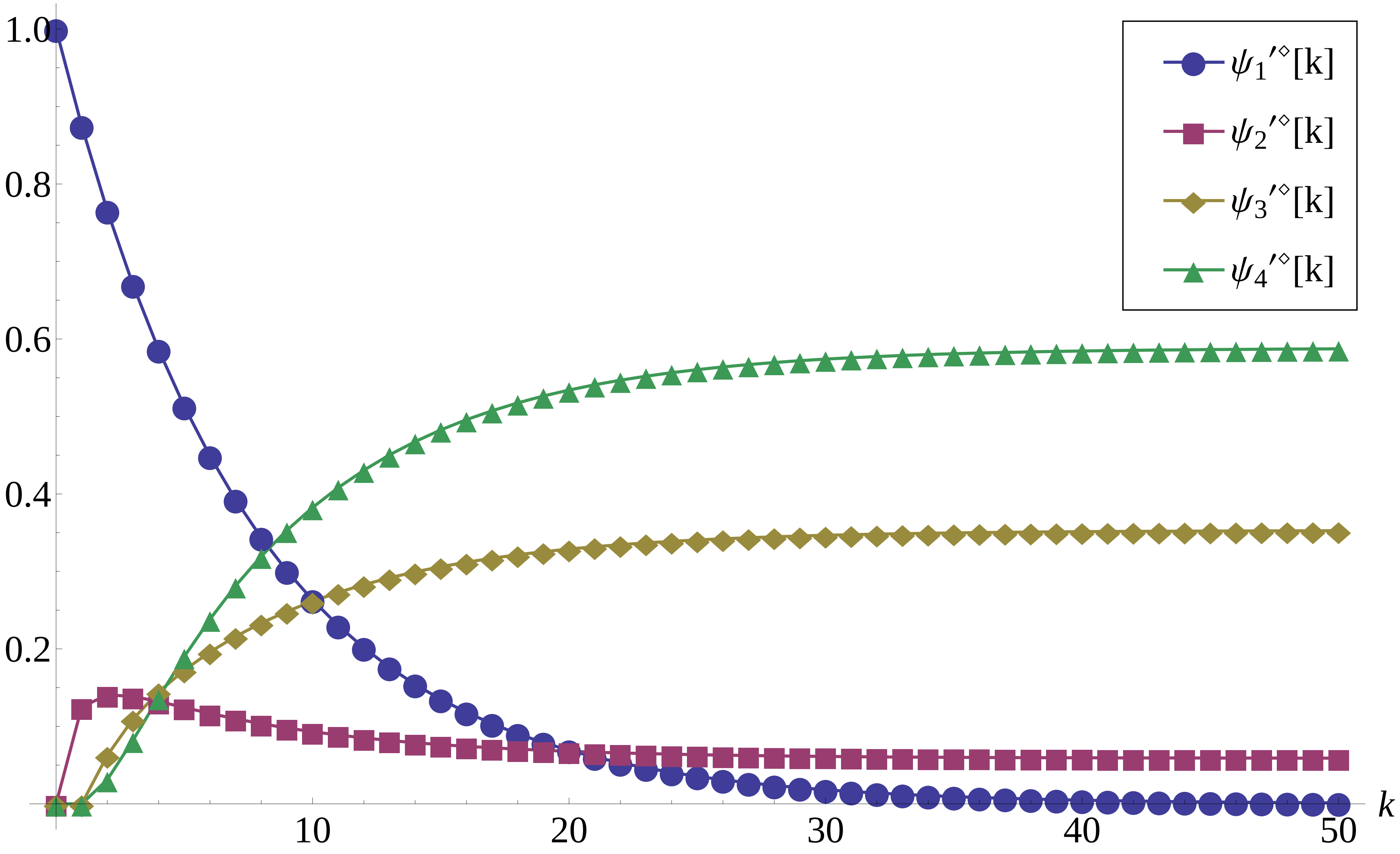
By the "quotient" analogue of Proposition PMFSMCT:

$$\tilde{\varphi}'(\widetilde{\mathcal{K}}_1) = 0, \quad \tilde{\varphi}'(\widetilde{\mathcal{K}}_2) = \frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3}, \quad \tilde{\varphi}'(\widetilde{\mathcal{K}}_3) = 0,$$
$$\tilde{\varphi}'(\widetilde{\mathcal{K}}_4) = \frac{\rho(2-\rho)}{2+\rho-\rho^2-\rho^3}, \quad \tilde{\varphi}'(\widetilde{\mathcal{K}}_5) = 0, \quad \tilde{\varphi}'(\widetilde{\mathcal{K}}_6) = \frac{2-\rho-\rho^2}{2+\rho-\rho^2-\rho^3}.$$

The steady-state PMF for $SMC_{\underline{\leftrightarrow}_{ss}}(\overline{L})$:

$$\tilde{\varphi}' = \frac{1}{2+\rho-\rho^2-\rho^3}(0, \rho^2(1-\rho), 0, \rho(2-\rho), 0, 2-\rho-\rho^2).$$

This coincides with the result obtained with the use of $\tilde{\psi}'^{*}$ and $\widetilde{SJ}'$.



SHMQRTP: Transient probabilities alteration diagram for the reduced quotient DTMC of the abstract generalized shared memory system when $\rho = \frac{1}{2}$

Stationary behaviour

Steady state and equivalences

Proposition 4 *(STPROB) Let* G, G' *be dynamic expressions with* $\mathcal{R} : G \underline{\leftrightarrow}_{ss} G'$ *and* φ *be the steady-state PMF for* $SMC(G)$, φ' *be the steady-state PMF for* $SMC(G')$*. Then*
$\forall \mathcal{H} \in (DR(G) \cup DR(G'))/_{\mathcal{R}}$

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

Let G be a dynamic expression and φ be the steady-state PMF for $SMC(G)$,
$\varphi_{\underline{\leftrightarrow}_{ss}}$ be the steady-state PMF for $SMC_{\underline{\leftrightarrow}_{ss}}(G)$.

By Proposition STPROB: $\forall \mathcal{K} \in DR(G)/_{\mathcal{R}_{ss}(G)}$

$$\varphi_{\underline{\leftrightarrow}_{ss}}(\mathcal{K}) = \sum_{s \in \mathcal{K}} \varphi(s).$$

Definition 18 *A* derived step trace *of a dynamic expression* G *is* $\Sigma = A_1 \cdots A_n \in (\mathbb{N}_{fin}^{\mathcal{L}})^*$, *where* $\exists s \in DR(G)\ s \stackrel{\Upsilon_1}{\to} s_1 \stackrel{\Upsilon_2}{\to} \cdots \stackrel{\Upsilon_n}{\to} s_n,\ \mathcal{L}(\Upsilon_i) = A_i\ (1 \le i \le n)$.

The probability to execute the derived step trace Σ in s:

$$PT(\Sigma, s) = \sum_{\{\Upsilon_1,\ldots,\Upsilon_n \mid s=s_0 \stackrel{\Upsilon_1}{\to} s_1 \stackrel{\Upsilon_2}{\to} \cdots \stackrel{\Upsilon_n}{\to} s_n,\ \mathcal{L}(\Upsilon_i)=A_i\ (1 \le i \le n)\}} \prod_{i=1}^{n} PT(\Upsilon_i, s_{i-1}).$$

Theorem 2 *(STTRAC) Let* G, G' *be dynamic expressions with* $\mathcal{R} : G \underline{\leftrightarrow}_{ss} G'$ *and* φ *be the steady-state PMF for* $SMC(G)$, φ' *be the steady-state PMF for* $SMC(G')$ *and* Σ *be a derived step trace of* G *and* G'. *Then* $\forall \mathcal{H} \in (DR(G) \cup DR(G'))/_{\mathcal{R}}$

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

By Theorem STTRAC: $\forall\mathcal{K} \in DR(G)/_{\mathcal{R}_{ss}(G)}$

$$\varphi_{\underline{\leftrightarrow}_{ss}}(\mathcal{K})PT(\Sigma, \mathcal{K}) = \sum_{s\in\mathcal{K}} \varphi(s)PT(\Sigma, s),$$

where $\forall s \in \mathcal{K}\ PT(\Sigma, \mathcal{K}) = PT(\Sigma, s)$.

Proposition 5 *(SJAVVA) Let* G, G' *be dynamic expressions with* $\mathcal{R} : G\underline{\leftrightarrow}_{ss}G'$. *Then* $\forall\mathcal{H} \in (DR(G) \cup DR(G'))/_{\mathcal{R}}$

$$SJ_{\mathcal{R}\cap(DR(G))^2}(\mathcal{H} \cap DR(G)) = SJ_{\mathcal{R}\cap(DR(G'))^2}(\mathcal{H} \cap DR(G')),$$

$$VAR_{\mathcal{R}\cap(DR(G))^2}(\mathcal{H} \cap DR(G)) = VAR_{\mathcal{R}\cap(DR(G'))^2}(\mathcal{H} \cap DR(G')).$$

Performance indices of the quotient abstract generalized shared memory system

- The average recurrence time in the state $\widetilde{\mathcal{K}}_2$, where no processor requests the memory, the *average system run-through*, is $\frac{1}{\tilde{\varphi}'_2} = \frac{2+\rho-\rho^2-\rho^3}{\rho^2(1-\rho)}$.

- The common memory is available only in the states $\widetilde{\mathcal{K}}_2, \widetilde{\mathcal{K}}_3, \widetilde{\mathcal{K}}_5$.

  The steady-state probability that the memory is available is $\tilde{\varphi}'_2 + \tilde{\varphi}'_3 + \tilde{\varphi}'_5 = \frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3} + 0 + 0 = \frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3}$.

  The steady-state probability that the memory is used (i.e. not available), the *shared memory utilization*, is $1 - \frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3} = \frac{2+\rho-2\rho^2}{2+\rho-\rho^2-\rho^3}$.

- After activation of the system, we leave the state $\widetilde{\mathcal{K}}_1$ for all, and the common memory is either requested or allocated in every remaining state, with exception of $\widetilde{\mathcal{K}}_2$.

  The *rate with which the necessity of shared memory emerges* coincides with the rate of leaving $\widetilde{\mathcal{K}}_2$, calculated as $\frac{\tilde{\varphi}'_2}{\widetilde{SJ}'_2} = \frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3} \cdot \frac{\rho(2-\rho)}{1} = \frac{\rho^3(1-\rho)(2-\rho)}{2+\rho-\rho^2-\rho^3}$.

- The common memory request of a processor $\{r\}$ is only possible from the states $\widetilde{\mathcal{K}}_2, \widetilde{\mathcal{K}}_4$.

  The request probability in each of the states is the sum of the execution probabilities for all multisets of multiactions containing $\{r\}$.

  The *steady-state probability of the shared memory request from a processor* is

  $$\tilde{\varphi}_2' \sum_{\{A,\widetilde{\mathcal{K}} \mid \{r\}\in A,\ \widetilde{\mathcal{K}}_2 \stackrel{A}{\to} \widetilde{\mathcal{K}}\}} PM_A(\widetilde{\mathcal{K}}_2, \widetilde{\mathcal{K}}) + \tilde{\varphi}_4' \sum_{\{A,\widetilde{\mathcal{K}} \mid \{r\}\in A,\ \widetilde{\mathcal{K}}_4 \stackrel{A}{\to} \widetilde{\mathcal{K}}\}} PM_A(\widetilde{\mathcal{K}}_4, \widetilde{\mathcal{K}}) = \frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3}(2\rho(1-\rho)+\rho^2) + \frac{\rho(2-\rho)}{2+\rho-\rho^2-\rho^3}(\rho(1-\rho^2)+\rho^3) = \frac{\rho^2(2-\rho)(1+\rho-\rho^2)}{2+\rho-\rho^2-\rho^3}.$$

The performance indices are the same for the complete and quotient abstract generalized shared memory systems.

The coincidence of the first and second performance indices illustrates Proposition STPROB.

The coincidence of the third performance index illustrates Proposition STPROB and Proposition SJAVVA.

The coincidence of the fourth performance index is by Theorem STTRAC:

one should apply its result to the step traces $\{\{r\}\},\ \{\{r\},\{r\}\},\ \{\{r\},\{m\}\}$ of $\overline{L}$ and itself,

and sum the left and right parts of the three resulting equalities.

Effect of quantitative changes of ρ to performance of the quotient abstract generalized shared memory system in its steady state

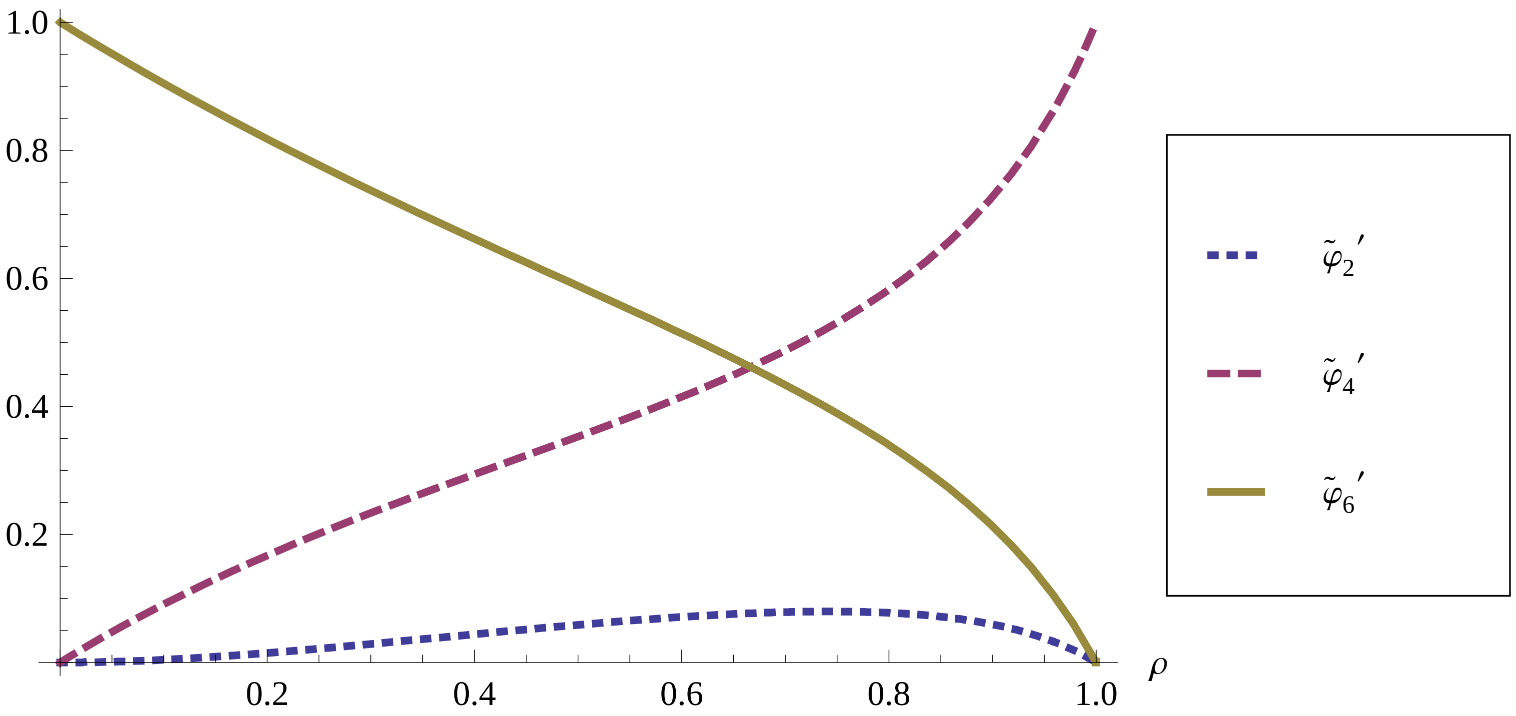
$\rho \in (0;1)$ is the probability of every multiaction of the system.

The closer is ρ to 0, the less is the probability to execute some activities at every discrete time step: the system will most probably *stand idle*.

The closer is ρ to 1, the greater is the probability to execute some activities at every discrete time step: the system will most probably *operate*.

$\tilde{\varphi}'_1 = \tilde{\varphi}'_3 = \tilde{\varphi}'_5 = 0$ are constants, and they do not depend on ρ.

$\tilde{\varphi}'_2 = \frac{\rho^2(1-\rho)}{2+\rho-\rho^2-\rho^3}$, $\tilde{\varphi}'_4 = \frac{\rho(2-\rho)}{2+\rho-\rho^2-\rho^3}$, $\tilde{\varphi}'_6 = \frac{2-\rho-\rho^2}{2+\rho-\rho^2-\rho^3}$ depend on ρ.



SHMGQSSP: Steady-state probabilities $\tilde{\varphi}_2', \tilde{\varphi}_4', \tilde{\varphi}_6'$ as functions of the parameter ρ

$\tilde{\varphi}_2', \tilde{\varphi}_4'$ tend to 0 and $\tilde{\varphi}_6'$ tends to 1 when ρ approaches 0.

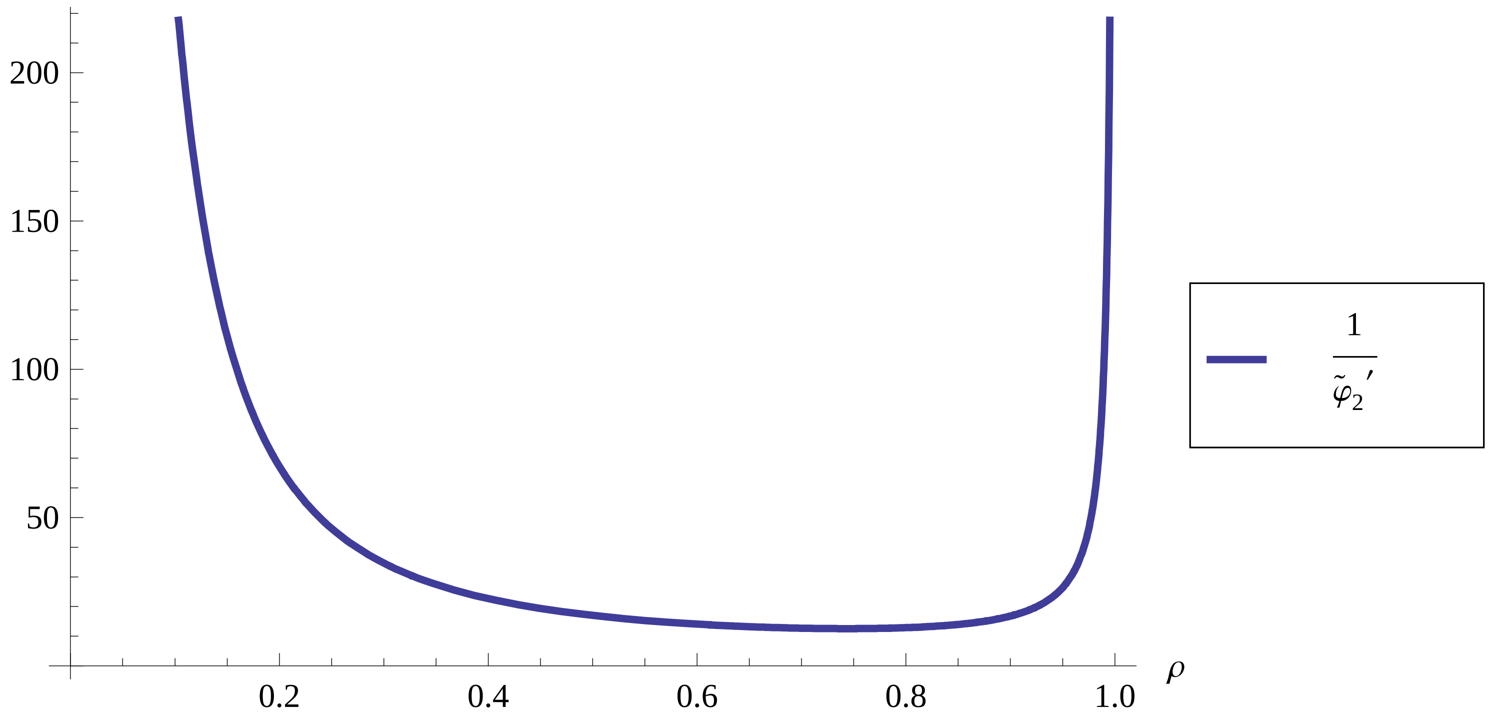
When ρ is closer to 0, the probability that the memory is allocated to a processor and the memory is requested by another processor increases: *more unsatisfied memory requests*.

$\tilde{\varphi}'_2,\ \tilde{\varphi}'_6$ tend to 0 and $\tilde{\varphi}'_4$ tends to 1 when ρ approaches 1.

When ρ is closer to 1, the probability that the memory is allocated to a processor (and not requested by another one) increases: *less unsatisfied memory requests*.

The maximal value 0.0797 of $\tilde{\varphi}'_2$ is reached when $\rho \approx 0.7433$.

In this case, the probability that the system is activated and the memory is not requested is maximal: *maximal shared memory availability* is about 8%.



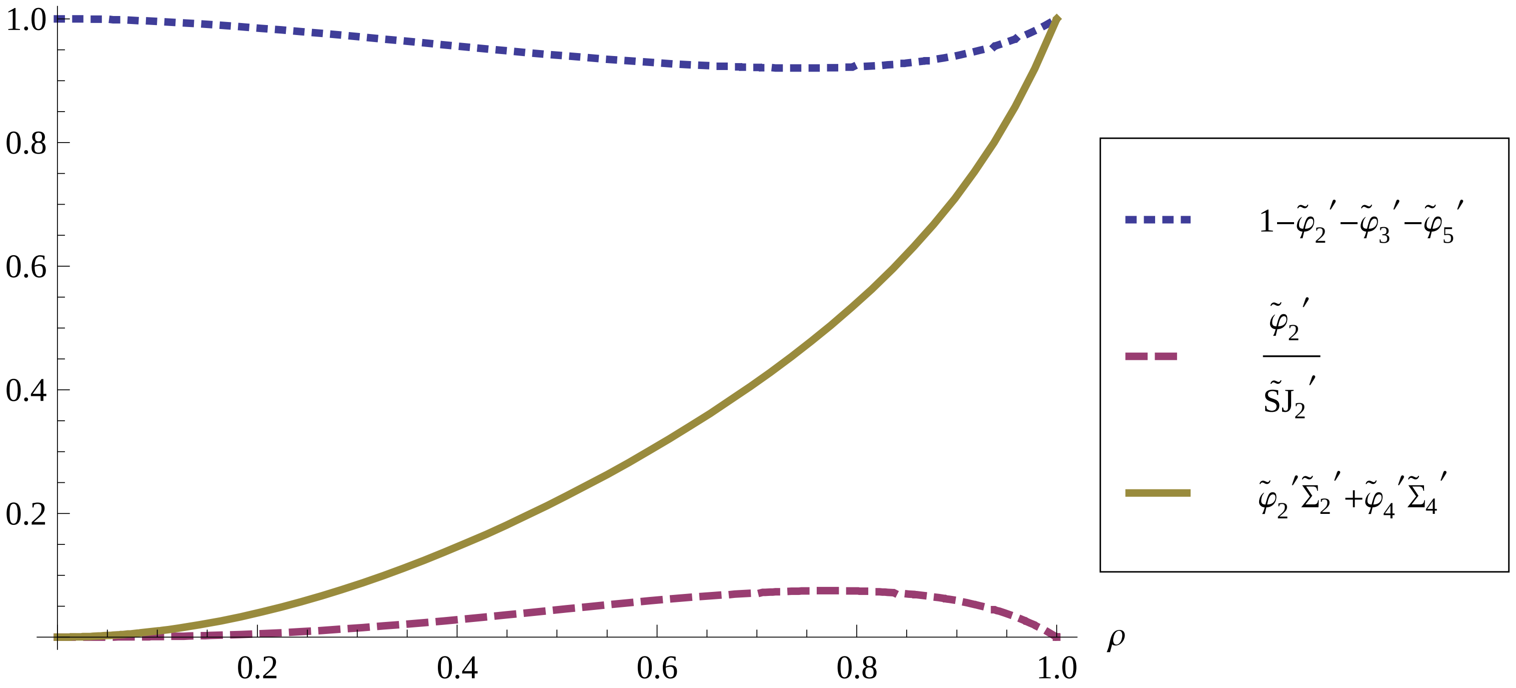
SHMGQART: Average system run-through $\frac{1}{\tilde{\varphi}_2'}$ as a function of the parameter ρ

The average system run-through is $\frac{1}{\tilde{\varphi}_2'}$.

It tends to ∞ when ρ approaches 0 or 1.

The minimal value 12.5516 of $\frac{1}{\tilde{\varphi}_2'}$ is reached when $\rho \approx 0.7433$.

To speed up the system's operation: take the parameter ρ closer to 0.7433.



SHMGQIND: Some performance indices as functions of the parameter ρ

The shared memory utilization is $1-\tilde{\varphi}_2'-\tilde{\varphi}_3'-\tilde{\varphi}_5'$.

It tends to 1 when ρ approaches 0 and when ρ approaches 1.

The minimal value 0.9203 of the utilization is reached when $\rho \approx 0.7433$.

The *minimal shared memory utilization* is about 92%.

To increase the utilization: take the parameter ρ closer to 0 or 1.

The rate with which the necessity of shared memory emerges is $\frac{\tilde{\varphi}'_2}{\widetilde{SJ}'_2}$.

It tends to 0 when ρ approaches 0 and when ρ approaches 1.

The maximal value 0.0751 of the rate is reached when $\rho \approx 0.7743$.

The *maximal rate with which the necessity of shared memory emerges* is about $\frac{1}{13}$.

To decrease the rate: take the parameter ρ closer to 0 or 1.

The steady-state probability of the shared memory request from a processor is $\tilde{\varphi}'_2\widetilde{\Sigma}'_2 + \tilde{\varphi}'_4\widetilde{\Sigma}'_4$,
where $\widetilde{\Sigma}'_i = \sum_{\{A,\widetilde{\mathcal{K}} \mid \{r\} \in A,\ \widetilde{\mathcal{K}}_i \stackrel{A}{\to} \widetilde{\mathcal{K}}\}} PM_A(\widetilde{\mathcal{K}}_i, \widetilde{\mathcal{K}}),\ i \in \{2, 4\}$.

It tends to 0 when ρ approaches 0 and it tends to 1 when ρ approaches 1.

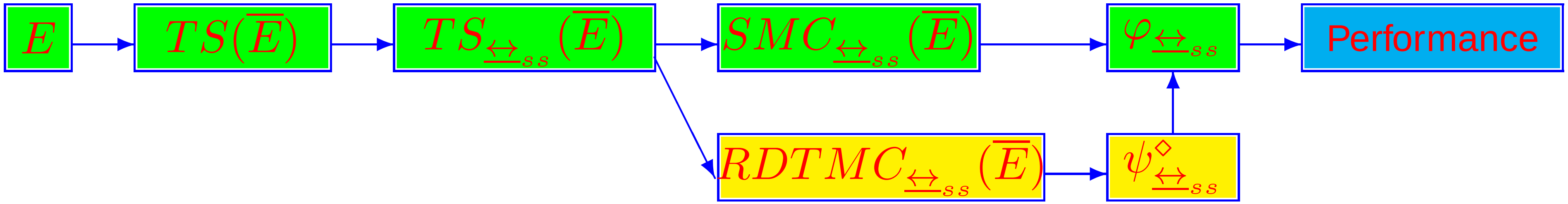
To increase the probability: take the parameter ρ closer to 1.

Simplification of performance analysis

The method of performance analysis simplification.

1. The investigated system is specified by a static expression of $dtsiPBC$.
2. The transition system of the expression is constructed.
3. After treating the transition system for self-similarity,
   a step stochastic autobisimulation equivalence for the expression is determined.
4. The quotient underlying SMC is constructed from the quotient transition system.
5. Stationary probabilities and performance indices are calculated using the SMC.

Simplification of the steps 4 and 5:
constructing the reduced quotient DTMC from the quotient transition system,
calculating the stationary probabilities of the quotient underlying SMC using this DTMC
and obtaining the performance indices.



EQPEVA: Equivalence-based simplification of performance evaluation

The limitation of the method: the expressions with underlying SMCs containing one closed communication class of states, which is ergodic, to ensure uniqueness of the stationary distribution.

If an SMC contains several closed communication classes of states that are all ergodic: several stationary distributions may exist, depending on the initial PMF.

The general steady-state probabilities are then calculated as the sum of the stationary probabilities of all the ergodic classes of states, weighted by the probabilities to enter these classes, starting from the initial state and passing through transient states.

The underlying SMC of each process expression has one initial PMF (that at the time moment 0): the stationary distribution is unique.

It is worth applying the method to the systems with similar subprocesses.

Overview and open questions

The results obtained

- A discrete time stochastic and immediate extension $dtsiPBC$ of finite PBC enriched with iteration.
- The step operational semantics based on labeled probabilistic transition systems.
- The denotational semantics in terms of a subclass of LDTSIPNs.
- The method of performance evaluation based on underlying SMCs.
- Step stochastic bisimulation equivalence of the expressions and dtsi-boxes.
- The transition systems and SMCs reduction modulo the equivalence.
- A comparison of stationary behaviour up to the equivalence.
- Performance analysis simplification with the equivalence.
- The case study: the shared memory system.

Further research

- Constructing a congruence relation: the equivalence that withstands application of the algebraic operations.
- Introducing the deterministically timed multiactions with fixed time delays (including the zero delay).
- Extending the syntax with recursion operator.

References

[AHR00] VAN DER AALST W.M.P., VAN HEE K.M., REIJERS H.A. *Analysis of discrete-time stochastic Petri nets. Statistica Neerlandica* **54(2)**, p. 237–255, 2000, `http://tmitwww.tm.tue.nl/staff/hreijers/H.A. Reijers Bestanden/Statistica.pdf`.

[AKB98] D'ARGENIO P.R., KATOEN J.-P., BRINKSMA E. *A compositional approach to generalised semi-Markov processes. Proceedings of* 4^{th} *International Workshop on Discrete Event Systems - 98 (WODES'98)*, p. 391–397, Cagliary, Italy, IEEE Press, London, UK, 1998, `http://cs.famaf.unc.edu.ar/~dargenio/Publications/papers/wodes98.ps.gz`.

[BGo98] BERNARDO M., GORRIERI R. *A tutorial on EMPA: a theory of concurrent processes with nondeterminism, priorities, probabilities and time. TCS* **202**, p. 1–54, July 1998.

[BDH92] BEST E., DEVILLERS R., HALL J.G. *The box calculus: a new causal algebra with multi-label communication. LNCS* **609**, p. 21–69, 1992.

[Brad05] BRADLEY J.T. *Semi-Markov PEPA: modelling with generally distributed actions. International Journal of Simulation* **6(3–4)**, p. 43–51, February 2005, `http://pubs.doc.ic.ac.uk/semi-markov-pepa/semi-markov-pepa.pdf`.

[BBGo98] BRAVETTI M., BERNARDO M., GORRIERI R. *Towards performance evaluation with general distributions in process algebras.* *LNCS* **1466**, p. 405–422, 1998, http://www.cs.unibo.it/~bravetti/papers/concur98.ps.

[BKLL95] BRINKSMA E., KATOEN J.-P., LANGERAK R., LATELLA D. *A stochastic causality-based process algebra.* *The Computer Journal* **38 (7)**, p. 552–565, 1995, http://eprints.eemcs.utwente.nl/6387/01/552.pdf.

[Buc95] BUCHHOLZ P. *A notion of equivalence for stochastic Petri nets.* *LNCS* **935**, p. 161–180, 1995.

[Buc98] BUCHHOLZ P. *Iterative decomposition and aggregation of labeled GSPNs.* *LNCS* **1420**, p. 226–245, 1998.

[BT00] 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.

[CMBC93] CHIOLA G., MARSAN M.A., BALBO G., CONTE G. *Generalized stochastic Petri nets: a definition at the net level and its implications.* *IEEE Transactions on Software Engineering* **19(2)**, p. 89–107, 1993.

[CR14] CIOBANU G., ROTARU A.S. *PHASE: a stochastic formalism for phase-type distributions. LNCS* **8829**, p. 91–106, 2014.

[DH13] DENG Y., HENNESSY M. *On the semantics of Markov automata. Information and Computation* **222**, p. 139–168, 2013.

[DTGN85] DUGAN J.B., TRIVEDI K.S., GEIST R.M., NICOLA V.F. *Extended stochastic Petri nets: applications and analysis. Proceedings of* 10^{th} *International Symposium on Computer Performance Modelling, Measurement and Evaluation - 84 (Performance'84)*, Paris, France, December 1984, p. 507–519, North-Holland, Amsterdam, The Netherlands, 1985.

[FN85] FLORIN G., NATKIN S. *Les reseaux de Petri stochastiques. Technique et Science Informatique* **4(1)**, 1985.

[FM03] DE FRUTOS E.D., MARROQUÍN A.O. *Ambient Petri nets. Electronic Notes in Theoretical Computer Science* **85(1)**, 27 p., 2003.

[GL94] GERMAN R., LINDEMANN C. *Analysis of stochastic Petri nets by the method of supplementary variables. Performance Evaluation* **20(1–3)**, p. 317–335, 1994.

[GHR93] GÖTZ N., HERZOG U., RETTELBACH M. *Multiprocessor and distributed system design: the integration of functional specification and performance analysis using stochastic process algebras. LNCS* **729**, p. 121–146, 1993.

[HS89] HAAS P.J., SHEDLER G.S. *Stochastic Petri net representation of discrete event simulations. IEEE Transactions on Software Engineering* **15(4)**, p. 381–393, 1987.

[HBC13] HAYDEN R.A., BRADLEY J.T., CLARK A. *Performance specification and evaluation with unified stochastic probes and fluid analysis. IEEE Transactions on Software Engineering* **39(1)**, p. 97–118, IEEE Computer Society Press, January 2013, `http://pubs.doc.ic.ac.uk/fluid-unified-stochastic-probes/fluid-unified-stochastic-probes.pdf`.

[HR94] HERMANNS H., RETTELBACH M. *Syntax, semantics, equivalences and axioms for MTIPP.* In: Herzog U. and Rettelbach M., eds., *Proceedings of the* 2^{nd} *Workshop on Process Algebras and Performance Modelling. Arbeitsberichte des IMMD* **27**, University of Erlangen, 1994.

[Hil96] HILLSTON J. *A compositional approach to performance modelling.* Cambridge University Press, UK, 1996.

[Kou00] KOUTNY M. *A compositional model of time Petri nets. LNCS* **1825**, p. 303–322, 2000.

[LN00] LÓPEZ B.N., NÚÑEZ G.M. *NMSPA: a non-Markovian model for stochastic processes. Proceedings of International Workshop on Distributed System Validation and Verification - 00 (DSVV'00)*, p. 33–40, 2000, `http://dalila.sip.uclm.es/membros/manolo/papers/dsvv2000.ps.gz`.

[MVCC03] MACIÀ S.H., VALERO R.V., CAZORLA L.D., CUARTERO G.F. *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`.

[MVC02] MACIÀ S.H., VALERO R.V., CUARTERO G.F. *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`.

[MVCR08] MACIÀ S.H., VALERO R.V., CUARTERO G.F., RUIZ D.M.C. *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.

[MVF01] MACIÀ S.H., VALERO R.V., DE FRUTOS E.D. *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`.

[MVi08] MARKOVSKI J., DE VINK E.P. *Extending timed process algebra with discrete stochastic time. LNCS* **5140**, p. 268–283, 2008.

[MF00] MARROQUÍN A.O., DE FRUTOS E.D. *TPBC: timed Petri box calculus. Technical Report*, Departamento de Sistemas Infofmáticos y Programación, Universidad Complutense de Madrid, Madrid, Spain, 2000 (in Spanish).

[MBBCCC89] MARSAN M.A., BALBO G., BOBBIO A., CHIOLA G., CONTE G., CUMANI A. *The effect of execution policies on the semantics and analysis of stochastic Petri nets. IEEE Transactions on Software Engineering* **15(7)**, p. 832–846, 1989.

[MBCDF95] 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/`.

[MC87] MARSAN M.A., CHIOLA G. *On Petri nets with deterministic and exponentially distributed firing times. LNCS* **266**, p. 132–145, 1987.

[MCF90] MARSAN M.A., CHIOLA G., FUMAGALLI A. *Improving the efficiency of the analysis of DSPN models. LNCS* **424**, p. 30–50, 1990.

[MCB84] MARSAN M.A., CONTE G., BALBO G. *A class of generalized stochastic Petri nets for performance evaluation of multiprocessor systems. ACM Transactions on Computer Systems* **2(2)**, p. 93–122, 1984.

[Mol82] MOLLOY M. *Performance analysis using stochastic Petri nets. IEEE Transactions on Software Engineering* **31(9)**, p. 913–917, 1982.

[Mol85] MOLLOY M. *Discrete time stochastic Petri nets. IEEE Transactions on Software Engineering* **11(4)**, p. 417–423, 1985.

[Nia05] NIAOURIS A. *An algebra of Petri nets with arc-based time restrictions. LNCS* **3407**, p. 447–462, 2005.

[P81] PETERSON J.L. *Petri net theory and modeling of systems.* Prentice-Hall, 1981.

[Pri96] PRIAMI C. *Stochastic π-calculus with general distributions. Proceedings of 4^{th} International Workshop on Process Algebra and Performance Modelling - 96 (PAPM'96)*, p. 41–57, CLUT Press, Torino, Italy, 1996.

[Ret95] RETTELBACH M. *Probabilistic branching in Markovian process algebras. The Computer Journal* **38(7)**, p. 590–599, 1995.

[Tar05] 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://itar.iis.nsk.su/files/itar/pages/dtspbcib_cov.pdf`.

[Tar06] 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://itar.iis.nsk.su/files/itar/pages/dtsitncc.pdf`.

[TMV10] TARASYUK I.V., MACIÀ S.H., VALERO R.V. *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://itar.iis.nsk.su/files/itar/pages/dtsipbc.pdf`.

[TMV13] TARASYUK I.V., MACIÀ S.H., VALERO R.V. *Discrete time stochastic Petri box calculus with immediate multiactions dtsiPBC. Electronic Notes in Theoretical Computer Science* **296**, p. 229–252, 2013, `http://itar.iis.nsk.su/files/itar/pages/dtsipbcentcs.pdf`.

[Tof94] TOFTS C. *Processes with probabilities, priority and time. Formal Aspects of Computing* **6(5)**, p. 536–564, 1994.

[ZCH97] ZIJAL R., CIARDO G., HOMMEL G. *Discrete deterministic and stochastic Petri nets.* In: K. Irmscher, Ch. Mittaschand and K. Richter, eds., *MMB'97, Aktuelle Probleme der Informatik*: Band **1**. VDE Verlag, 1997.

[ZG94] ZIJAL R., GERMAN R. *A new approach to discrete time stochastic Petri nets.* *LNCS* **199**, p. 198–204, 1994.

[ZFH01] ZIMMERMANN A., FREIHEIT J., HOMMEL G. *Discrete time stochastic Petri nets for modeling and evaluation of real-time systems.* *Proceedings of Workshop on Parallel and Distributed Real Time Systems*, San Francisco, USA, 6 p., 2001, `http://pdv.cs.tu-berlin.de/~azi/texte/WPDRTS01.pdf`.

The slides can be downloaded from Internet:

`http://itar.iis.nsk.su/files/itar/pages/albc15sld.pdf`

Thank you for your attention!