Rational Automata Networks
- A Non-Markovian Modeling Approach -

Peter Buchholz
Informatik IV, TU Dortmund, D-44221 Dortmund, Germany, {peter.buchholz@tu-dortmund.de}
Miklós Telek
Department of Telecommunications, Technical University of Budapest, H-1521 Budapest, Hungary, {telek@hit.hme.hu}

A new class of non-Markovian models is introduced that results from the combination of stochastic automata networks and a very general class of stochastic processes namely rational arrival processes which are derived from matrix exponential distributions. It is shown that the modeling formalism allows a compact representation of complex models with large state spaces. The resulting stochastic process is non-Markovian but it can be analyzed with numerical techniques like a Markov chain and the results at the level of the automata are stochastic distributions that can be used to compute standard performance and dependability results. The model class includes stochastic automata networks with phase type distributed and correlated event times and includes also models that have a finite state space but cannot be represented by finite Markov chains. The paper introduces the model class, shows how the descriptor matrix can be represented in compact form, presents some example models and outlines methods to analyze the new models.

Key words: Applied Probability, Matrix Exponential Distributions, Rational Arrival Processes, Numerical Analysis, Automata Networks, Structured Analysis;
History: submitted March 2011, revised July 2011.

---

1. Introduction

Markov models have been used in performance and dependability modeling for a long time (e.g., Stewart, 1994). Their major advantage is the possibility of analyzing the resulting Markov chain using numerical, simulative and a large number of approximate techniques. However, the major disadvantage of Markov models is the problem of state space explosion which implies that even harmless looking models result in state spaces of enormous size. Even on contemporary hardware many realistic models cannot be analyzed numerically. Consequently, the handling of large state spaces is a major research topic in Markov modeling.

The most successful approach in handling large state spaces is based on a structured description of models as a set of communicating components. This approach which has been published in different variants during the last decades (e.g., in Buchholz et al., 2000; Buchholz, 1992; Donatelli, 1994; Hermanns, 2002; Hillston, 1995; Plateau, 1985) allows one to represent the generator matrix of the Markov chain in a compact form that avoids the exponential growth and can be exploited in efficient numerical analysis techniques. The reason for state space explosion is the combinatorial growth of the state space in terms of the size of the state spaces of several components. This means that a large number of components or components with larger state spaces result in huge Markov chains. So called structured numerical analysis techniques represent the transition matrix in compact form such that the storage and computation of the solution vector becomes the major bottleneck. One reason for large component state spaces is the modeling of non-exponential distributions by phase type (PH) distributions (Neuts, 1981) with a large number of phases. PH distributions are often implicitly used in structured Markovian models as for example in (Buchholz et al., 2000; Buchholz, 1992; Hermanns, 2002) but the integration of PH distribution in compositional models is still an open topic (e.g., Sbeity et al., 2008). Often the reason for a large number of phases are distributions with a small coefficient of variation or density functions with a large number of local maxima and minima. A sufficiently accurate modeling of those distributions with PH distributions requires many phases. The use of general distributions instead of PH distributions destroys the Markov property such that the models are not or only with high effort solvable by numerical techniques.

A generalization of PH distributions are Matrix Exponential (ME) distributions as described in (Lipsky, 2008) which are based on a poorly algebraic description using matrices and vectors. Although ME distributions have been known for some time they have rarely been used in performance and dependability modeling. The reasons for this limited use are missing stochastic interpretation, missing methods to fit distribution parameters and missing analysis algorithms. However, recent results indicate that ME distributions or their extension Rational Arrival Processes (RAPs) developed in (Asmussen and Bladt, 1999) are an interesting extension of PH distributions and Markovian Arrival Processes (MAPs) (Neuts, 1979) since they are more general and models including ME distributions and RAPs can be analyzed like Markov chains with efficient numerical techniques as recently shown in (Bean and Nielsen, 2010; Buchholz and Telek, 2010).

However, up to now ME distributions and RAPs are not used in structured descriptions

of models and it has not been proved whether the compositionality of Markov models can be extended to models including more general distributions. In this paper we propose a new class of non-Markovian models based on stochastic automata networks (SANs) with firing time distributions that are defined by ME distributions or RAPs. Since the model class does not have a probabilistic interpretation, we name the corresponding components as Rational Automata (RA), their networks as Rational Automata Networks (RANs) and the underlying processes as Rational Processes (RPs). It will be shown that RPs can be analyzed with similar techniques as Markov chains and that the descriptor of a RP resulting from a RAN can be represented in a compact form similar to the descriptor of a SAN using Kronecker products and sums. Although we restrict the model class to networks of automata it can be easily extended to other approaches like stochastic Petri nets or stochastic process algebras that can also be used to specify RPs in a structured form.

The paper is structured as follows. In the next section we define ME distributions and RAPs. Furthermore, we extend the class of RAPs to RAPs with multiple events denoted as MRAPs and show that ME distributions, RAPs and MRAPs are closed under composition realized by a restricted form of synchronized products. Section 3 introduces rational automata as an extension of stochastic automata where event durations are described by ME distributions and MRAPs. Then, in Section 4, networks of rational automata are built and it is shown that the descriptor of a network of rational automata can be represented by a sum of Kronecker products like the descriptor of a SAN. It is also outlined how numerical solution techniques can exploit this representation. The paper ends with the conclusions. Proofs of the theorems that are presented in the paper can be found in the appendix.

2. Matrix Exponential Distributions and Related Processes

In this section we describe the class of distributions and processes that build the base of our modeling approach.

2.1. ME Distributions

A ME distribution is defined as follows in (Lipsky, 2008).

Definition 1 *Let* $\mathbf{G}_0$ *be a non-singular* $n \times n$ *real valued matrix where the real parts of all eigenvalues are negative,* $\mathbb{1}$ *is a column vector of ones and* π *is an* n*-dimensional row vector*

with $\pi\mathbb{1} = 1$. *The pair* $(\pi, \mathbf{G}_0)$ *defines a Matrix Exponential (ME) distribution of size* n *if and only if*

$$F_{(\pi,\mathbf{G}_0)}(t) = 1 - \pi e^{\mathbf{G}_0 t}\mathbb{1}$$

defines a valid distribution function.

Observe that $F_{(\pi,\mathbf{G}_0)}(0) = 0$ follows from $\pi\mathbb{1} = 1$. Several results on ME distributions can be found in the literature (e.g., (Bladt and Neuts, 2003; Fackrell, 2005; Lipsky, 2008)). Obviously, a PH distribution is a special case of a ME distribution where all off-diagonal elements of matrix $\mathbf{G}_0$ are non-negative and all elements of vector π are non-negative. However, it can be shown that the class of ME distributions of size $n > 2$ is strictly larger than the class of PH distributions of the same size (see Horváth et al., 2009) and the class of ME distributions of finite size contains some distributions where the density becomes 0 in $(0, \infty)$ which is not possible with a PH distribution of finite size.

Example 1 *The following vector matrix pair from (Fackrell, 2003) defines for* $h \leq 2.30033340858$ *a ME distribution and can be represented as a PH distribution of size 3 for* $h \leq 0.552748375$.

$$\pi = (0.2, 0.3, 0.5)\,,\ \mathbf{G}_0 = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -3 & h \\ 0 & -h & -3 \end{pmatrix}.$$

At $h = 0.552748375$ *the canonical PH representation is (see Horváth and Telek, 2009)*

$$\tilde{\pi} = (0.988469, 0.0115306, 0)\,,\ \tilde{\mathbf{G}}_0 = \begin{pmatrix} -2.91865 & 0 & 0.176486 \\ 2.91865 & -2.91865 & 0 \\ 0 & 1.16269 & -1.16269 \end{pmatrix}.$$

The following matrix vector pair from (Éltető et al., 2006) describes also a ME distribution.

$$\pi = (2.63479, -1.22850, -0.406283)\,,$$
$$\mathbf{G}_0 = \begin{pmatrix} -2.25709 & 0 & 0 \\ 0 & -2.25709 & -2.338187 \\ 0 & 2.338187 & -2.25709 \end{pmatrix}.$$

This distribution has a squared coefficient of variation of 0.2009. *Observe that a PH distribution with* 5 *phases is necessary to reach such a low coefficient of variation.*

2.2. RAPs and MRAPs

To model correlations, ME distributions can be extended to rational arrival processes (RAPs) (Asmussen and Bladt, 1999) like PH distributions have been extended to MAPs. We define an extension of RAPs, namely MRAPs which consider multiple event types like MMAPs (He and Neuts, 1998) which extend MAPs by introducing multiple event types. It should be mentioned that MRAPs are structurally identical to BRAPs which have been defined in (Bean and Nielsen, 2010) to model batch arrivals with inter-arrival times defined by a RAP.

Definition 2 *A* marked rational arrival process *(MRAP) of size n with K event types is defined by a set of $K+1$ matrices of size $n \times n$ $(\mathbf{G}_0, \ldots, \mathbf{G}_K)$ such that*

1. $(\mathbf{G}_0 + \sum_{k=1}^{K} \mathbf{G}_k)\mathbb{1} = \mathbf{0}$,
2. *all eigenvalues of $\mathbf{G}_0$ have a negative real part which implies that the matrix is non-singular (Lipsky, 2008),*
3. $\mathbf{P} = -\mathbf{G}_0^{-1}\left(\sum_{k=1}^{K} \mathbf{G}_k\right)$ *has a unique eigenvalue 1 such that the solution $\pi\mathbf{P} = \pi$, $\pi\mathbb{1} = 1$ is unique and*
4. $f_{(\mathbf{G}_0,\ldots,\mathbf{G}_K)}(t_1, k_1, \ldots, t_j, k_j) = \pi e^{\mathbf{G}_0 t_1}\mathbf{G}_{k_1} e^{\mathbf{G}_0 t_2}\mathbf{G}_{k_2}\ldots e^{\mathbf{G}_0 t_j}\mathbf{G}_{k_j}\mathbb{1}$ *is a valid joint density for all $t_i \geq 0$ and $k_i \in \{1, \ldots, K\}$ $(i = 1, \ldots, j)$. That is $f_{(\mathbf{G}_0,\ldots,\mathbf{G}_K)}(t_1, k_1, \ldots, t_j, k_j) \geq 0$ and*
   $\sum_{k_1}\ldots\sum_{k_j}\int_{t_1}\ldots\int_{t_j} f_{(\mathbf{G}_0,\ldots,\mathbf{G}_K)}(t_1, k_1, \ldots, t_j, k_j)\, dt_j \ldots dt_1 = 1$.

$f_{(\mathbf{G}_0,\ldots,\mathbf{G}_K)}(t_1, k_1, \ldots, t_j, k_j)$ is the probability density of the event that the first j arrivals happen with inter-arrival times $t_1, \ldots, t_j$ and these arrivals are of types $k_1, \ldots, k_j$. If $\mathbf{G}_k \geq \mathbf{0}$ for $k = 1, \ldots, K$ and $\mathbf{G}_0$ has non-negative off-diagonal elements, then the process is an MMAP and $f_{(\mathbf{G}_0,\ldots,\mathbf{G}_K)}(t_1, k_1, \ldots, t_j, k_j)$ is necessarily a probability distribution. For $K = 1$ we obtain the definition of a RAP. Each ME$(\pi, \mathbf{G}_0)$ distribution can be expanded into a ME renewal process, which is a RAP with $K = 1$, $(\mathbf{G}_0, \mathbf{G}_1 = -\mathbf{G}_0\mathbb{1}\pi)$. To represent the transient behavior of MRAP$(\mathbf{G}_0, \ldots, \mathbf{G}_K)$ its description is extended by an initial vector ϕ. The initial vector ϕ is valid for a MRAP$(\mathbf{G}_0, \ldots, \mathbf{G}_K)$, if $\phi e^{\mathbf{G}_0 t_1}\mathbf{G}_{k_1} e^{\mathbf{G}_0 t_2}\mathbf{G}_{k_2}\ldots e^{\mathbf{G}_0 t_j}\mathbf{G}_{k_j}\mathbb{1}$ is a valid joint density for all $t_i \geq 0$ and $k_i \in \{1, \ldots, K\}$ $(i = 1, \ldots, j)$. In the rest of the paper, sometimes we add the initial vector ϕ to the MRAP definition such that $(\phi, \mathbf{G}_0, \ldots, \mathbf{G}_K)$ is denoted as MRAP with initial vector ϕ. If $\phi = \pi$ which is the initial vector given by the left

eigenvector of matrix $\mathbf{P}$, we have the stationary behavior of the MRAP and do not indicate the initial vector.

If $\phi \neq \pi$, then the MRAP is considered with another initial vector which differs from the stationary vector after an arbitrary event. E.g., if one wants to analyze the process after an event of class k, then $\phi = \pi(-\mathbf{G}_0)^{-1}\mathbf{G}_k/(\pi(-\mathbf{G}_0)^{-1}\mathbf{G}_k\mathbb{1})$.

Definition 3 *For a MRAP $(\phi, \mathbf{G}_0, \ldots, \mathbf{G}_K)$ we define the set of available initial vectors $\mathcal{V}_{(\phi,\mathbf{G}_0,\ldots,\mathbf{G}_K)}$ as follows. Let*

$$\mathcal{V}^{(0)}_{(\phi,\mathbf{G}_0,\ldots,\mathbf{G}_K)} = \{\phi\},$$

$$\mathcal{V}^{(i+1)}_{(\phi,\mathbf{G}_0,\ldots,\mathbf{G}_K)} = \left\{\nu \middle| \nu = \eta e^{\mathbf{G}_0 t}\mathbf{G}_k \quad \text{for } i > 0, t \geq 0, k \in \{1,\ldots K\}, \eta \in \mathcal{V}^{(i)}_{(\phi,\mathbf{G}_0,\ldots,\mathbf{G}_K)}\right\}$$

and

$$\mathcal{V}_{(\phi,\mathbf{G}_0,\ldots,\mathbf{G}_K)} = \bigcup_{i=0}^{\infty} \mathcal{V}^{(i)}_{(\phi,\mathbf{G}_0,\ldots,\mathbf{G}_K)}.$$

Corollary 1 *$(\phi, \mathbf{G}_0, \ldots, \mathbf{G}_K)$ defines a MRAP if and only if $\nu\mathbb{1} \geq 0$ for all vectors $\nu \in \mathcal{V}_{(\phi,\mathbf{G}_0,\ldots,\mathbf{G}_K)}$.*

Definition 3 avoids the normalization of vectors. It is implicitly assumed that if $\nu \in \mathcal{V}_{(\phi,\mathbf{G}_0,\ldots,\mathbf{G}_K)}$, then also $c\nu \in \mathcal{V}_{(\phi,\mathbf{G}_0,\ldots,\mathbf{G}_K)}$ for all $c > 0$. It is because a multiplication with c does not affect the nonnegativity of $\nu\mathbb{1}$.

Example 2 *The following matrixes define a MRAP*

$$\mathbf{G}_0 = \begin{pmatrix} -1.89167 & -0.6 & -0.05 \\ 0.198611 & -3.1 & -0.0916667 \\ 2.18472 & -1.1 & -4.00833 \end{pmatrix}, \quad \mathbf{G}_1 = \begin{pmatrix} 1.09167 & -0.145833 & 0.554167 \\ 2.00139 & -1.18403 & 1.09931 \\ 2.01528 & -1.02431 & 0.0923611 \end{pmatrix},$$

$$\mathbf{G}_2 = \begin{pmatrix} 0 & 0.545833 & 0.495833 \\ 0.166667 & 1.00069 & -0.0909722 \\ 1.83333 & 0.00763889 & -0.000694444 \end{pmatrix}.$$

The stationary initial vector, satisfying $\pi = -\pi\mathbf{G}_0^{-1}\left(\sum_{k=1}^{K}\mathbf{G}_k\right)$, $\pi\mathbb{1} = 1$, is $\pi = (0.9867, -0.1933, 0.2067)$. A valid transient initial vector is $\phi = (1.96, -1.08, 0.12)$. These vectors and matrixes define a valid process because $f_{(\pi,\mathbf{G}_0,\ldots,\mathbf{G}_K)}(t_1, k_1, \ldots, t_j, k_j)$ and $f_{(\phi,\mathbf{G}_0,\ldots,\mathbf{G}_K)}(t_1, k_1, \ldots, t_j, k_j)$ are non-negative for all valid set of parameters. An invalid transient initial vector is $\hat{\phi} = (-1, 3, -1)$, because with this initial vector $f_{(\hat{\phi},\mathbf{G}_0,\mathbf{G}_1,\mathbf{G}_2)}(t_1 = 1, k_1 = 2) = -0.0237813$.

If we remove the events from a MRAP, we obtain a Rational Process (RP) defined as follows.

Definition 4 *Let* $(\mathbf{G}_0, \ldots, \mathbf{G}_K)$ *be a MRAP, then* $\mathbf{G} = \sum_{k=0}^{K} \mathbf{G}_k$ *is the transition matrix of a rational process (RP).*

Due to the conditions for MRAPs, matrix $\mathbf{G}$ of a RP has a unique eigenvalue 0 such that the solution $\psi\mathbf{G} = \mathbf{0}$ with $\psi\mathbb{1} = 1$ exists. ψ is the stationary vector of the RP which can be computed from the solution of a linear set of equations. Similarly, the transient vector is given by $\nu_t = \nu_0 e^{\mathbf{G}t}$ where ν_0 is the initial vector which has to be a valid initial vector for the MRAP from which the RP has been generated.

Theorem 1 *If* $\mathbf{G}$ *is the matrix of a RP and* ν_0 *is an initial vector with* $\nu_0\mathbb{1} = 1$*, then* $\nu_t\mathbb{1} = 1$ *for all* $t \geq 0$*.*

As it is seen in Example 2 neither ψ nor ν_t need to be probability vectors with non-negative elements. They may contain negative values but the elements should sum up to 1.

2.3. Composition of MRAPs

It is known that the class of PH distributions and MAPs are closed under several composition operations. We now show that some composition results hold for MRAPs too.

We begin with the thinning of MRAPs. Thinning is a way to generate an (thinned) arrival process from an original arrival process. It works such that an arrival of the original process is probabilistically decided whether it is accepted or not. The set of accepted arrivals form the thinned arrival process. The following theorem shows that the class of MRAPs is closed under thinning and presents the matrix representation of the thinned process.

Theorem 2 *Let* $(\mathbf{G}_0, \ldots, \mathbf{G}_K)$ *be a MRAP,* $p_k \in [0, 1]$ *(*$k = 1, \ldots, K$*) and for at least one* $k \in \{1, \ldots, K\}$ $p_k > 0$*, then* $(\mathbf{H}_0, \ldots, \mathbf{H}_K) = (\mathbf{G}_0 + \sum_{k=1}^{K}(1 - p_k)\mathbf{G}_k, p_1\mathbf{G}_1, \ldots, p_K\mathbf{G}_K)$ *is the thinned process which is also a MRAP.*

In the above theorem $p_k = 0$ for all $k \in \{1, \ldots, K\}$ has been excluded since in this case we obtain a RP rather than a RAP. If some of the values p_k are zero, then the corresponding classes are no longer visible in the thinned process.

The second operation is the synchronized composition of MRAPs. To define this composition let $\hat{\mathbf{G}}_k = diag(\mathbf{G}_k\mathbb{1})$ where $diag(\mathbf{a})$ for a vector $\mathbf{a}$ is a diagonal matrix with $\mathbf{a}(i)$ in position (i,i) and for a MRAP $(\pi, \mathbf{G}_0, \ldots, \mathbf{G}_K)$ let $\mathbf{G'}_0 = \mathbf{G}_0 + \sum_{k=1}^{K} \hat{\mathbf{G}}_k$.

For synchronized composition we consider MRAPs that are defined over the same set of events $\mathcal{K} = \{1, \ldots, K\}$ and synchronization is performed over an event set $\mathcal{C} \subseteq \mathcal{K}$. Now assume that two MRAPs $(\mathbf{G}_0, \mathbf{G}_l(l \in \mathcal{G}))$ and $(\mathbf{H}_0, \mathbf{H}_m(m \in \mathcal{H}))$ with $\mathcal{K} = \mathcal{G} \cup \mathcal{H}$ should be composed. Then first the event sets are made identical by adding so called pseudo events. Thus, we define $\mathbf{G}_k = \mathbf{I}$ for $k \in \mathcal{C} \setminus \mathcal{G}$, $\mathbf{G}_k = \mathbf{0}$ for $k \in \mathcal{K} \setminus (\mathcal{C} \cup \mathcal{G})$, $\mathbf{H}_k = \mathbf{I}$ for $k \in \mathcal{C} \setminus \mathcal{H}$, and $\mathbf{H}_k = \mathbf{0}$ for $k \in \mathcal{K} \setminus (\mathcal{C} \cup \mathcal{H})$. The resulting MRAPs are $(\mathbf{G}_0, \ldots, \mathbf{G}_K)$ and $(\mathbf{H}_0, \ldots, \mathbf{H}_K)$. Substituting $\mathbf{I}$ and $\mathbf{0}$ matrices in the composition operations defined in the following theorem shows that these extensions do not modify the behavior of the composed process.

Theorem 3 *Let $(\mathbf{G}_0, \ldots, \mathbf{G}_K)$ and $(\mathbf{H}_0, \ldots, \mathbf{H}_K)$ be two MRAPs of size n and m, respectively. The synchronized composition via a set of classes $\mathcal{C} \subseteq \mathcal{K} = \{1, \ldots, K\}$ is denoted as $(\mathbf{G}_0, \ldots, \mathbf{G}_K) \| _{\mathcal{C}} (\mathbf{H}_0, \ldots, \mathbf{H}_K)$ and is defined as $(\mathbf{F}_0, \ldots, \mathbf{F}_K)$ of size nm with*

$$\begin{array}{ll} \mathbf{F}_0 & = \mathbf{G'}_0 \oplus \mathbf{H'}_0 - \sum\limits_{k \notin \mathcal{C}} \hat{\mathbf{G}}_k \oplus \hat{\mathbf{H}}_k - \sum\limits_{k \in \mathcal{C}} \hat{\mathbf{G}}_k \otimes \hat{\mathbf{H}}_k, \\ \mathbf{F}_k & = \mathbf{G}_k \oplus \mathbf{H}_k, \ \textit{for } k \notin \mathcal{C}, \\ \mathbf{F}_k & = \mathbf{G}_k \otimes \mathbf{H}_k, \ \textit{for } k \in \mathcal{C}, \end{array} \tag{1}$$

where $\otimes$ is the Kronecker product and $\oplus$ the Kronecker sum (Stewart, 1994). $(\mathbf{F}_0, \ldots, \mathbf{F}_K)$ is a MRAP if the solution of $\psi\mathbf{F} = \mathbf{0}$, $\psi\mathbb{1} = 1$ is unique, where $\mathbf{F} = \mathbf{F}_0 + \sum_{k=1}^{K} \mathbf{F}_k$ and one of the following conditions hold:

1. *$(\mathbf{G}_0, \ldots, \mathbf{G}_K)$ and $(\mathbf{H}_0, \ldots, \mathbf{H}_K)$ are both MMAPs, or*
2. *$\mathcal{C} = \emptyset$, or*
3. *$(\mathbf{G}_0, \ldots, \mathbf{G}_K)$ and $(\mathbf{H}_0, \ldots, \mathbf{H}_K)$ are both MRAPs and there exists $\mathcal{F} \subseteq \mathcal{C}$, such that*
   - *for $k \in \mathcal{F}$ $\mathbf{H}_k\mathbb{1} = d_k\mathbb{1}$, and*
   - *for $k \in \mathcal{C} \setminus \mathcal{F}$ $\mathbf{G}_k\mathbb{1} = c_k\mathbb{1}$*

   furthermore
   - *$(\tilde{\mathbf{G}}_0, \mathbf{G}_k$ (for $k \in \mathcal{K} \setminus \mathcal{F}$), $d_k\mathbf{G}_k$ (for $k \in \mathcal{F}$)) and*
   - *$(\tilde{\mathbf{H}}_0, \mathbf{H}_k$ (for $k \in \mathcal{K} \setminus (\mathcal{C} \setminus \mathcal{F})$), $c_k\mathbf{H}_k$ (for $k \in \mathcal{C} \setminus \mathcal{F}$))*

are MRAPs too, where $\tilde{\mathbf{G}}_0 = \mathbf{G'}_0 - \sum_{k\in\mathcal{K}\setminus\mathcal{F}} \hat{\mathbf{G}}_k - \sum_{k\in\mathcal{F}} d_k\hat{\mathbf{G}}_k$ *and* $\tilde{\mathbf{H}}_0 = \mathbf{H'}_0 - \sum_{k\in\mathcal{K}\setminus(\mathcal{C}\setminus\mathcal{F})} \hat{\mathbf{H}}_k - \sum_{k\in\mathcal{C}\setminus\mathcal{F}} c_k\hat{\mathbf{H}}_k$.

Naturally, the roles of $(\mathbf{G}_0,\ldots,\mathbf{G}_K)$ *and* $(\mathbf{H}_0,\ldots,\mathbf{H}_K)$ *may be interchanged.*

The theorem is central for the composition of MRAPs and we briefly explain the different cases. Case 2 is the simplest one since it describes independent MRAPs which run in parallel. It is intuitively clear that the resulting process is again an MRAP. The first case considers the composition of two MMAPs where the result is again an MMAP. However, even if the resulting process is an MMAP, the general composition has some strange properties. It is for example sensitive to similarity transformation including speed up (change of the time scale). Let $(\mathbf{G}_0^*,\ldots,\mathbf{G}_K^*)$ be a MRAP similar to $(\mathbf{G}_0,\ldots,\mathbf{G}_K)$ (i.e., there exists a non-singular $\mathbf{B}$ such that $\mathbf{G}_k^* = \mathbf{B}^{-1}\mathbf{G}_k\mathbf{B}$) then $(\mathbf{G}_0,\ldots,\mathbf{G}_K)\|_{\mathcal{C}}(\mathbf{H}_0,\ldots,\mathbf{H}_K)$ is not similar to $(\mathbf{G}_0^*,\ldots,\mathbf{G}_K^*)\|_{\mathcal{C}}(\mathbf{H}_0,\ldots,\mathbf{H}_K)$ in general. Furthermore, let $(\mathbf{G}_0^*,\ldots,\mathbf{G}_K^*)$ and $(\mathbf{H}_0^*,\ldots,\mathbf{H}_K^*)$ be two MRAPs c times faster than $(\mathbf{G}_0,\ldots,\mathbf{G}_K)$ and $(\mathbf{H}_0,\ldots,\mathbf{H}_K)$ (i.e., $\mathbf{G}_k^* = c\mathbf{G}_k$ and $\mathbf{H}_k^* = c\mathbf{H}_k$) then $(\mathbf{G}_0,\ldots,\mathbf{G}_K)\|_{\mathcal{C}}(\mathbf{H}_0,\ldots,\mathbf{H}_K)$ and $(\mathbf{G}_0^*,\ldots,\mathbf{G}_K^*)\|_{\mathcal{C}}(\mathbf{H}_0^*,\ldots,\mathbf{H}_K^*)$ are two different processes, in general. This behavior indicates that the composition of two processes where the timing of an event may depend on the state of both processes has a behavior that is not well defined since the composed process behaves differently at different time scales which destroys the basic assumption of time scale invariance in modeling. If the composed processes are MRAPs, then the general composition may even result in something that is not a stochastic process. In the third case, the composition is restricted by defining for each k one MRAP that determines the timing. The other MRAP only reacts which means that it changes its state when the event occurs. Of course, different events can be triggered by different MRAPs. This composition behaves well since it is invariant under similarity transformations and assures that the resulting process is an MRAP. Constants $c_k, d_k \neq 1.0$ can be used to slow down speed up events independently of the state of the other MRAP.

The following theorem introduces a specific case of a MRAP that is reset by or superposed with a Poisson process.

Theorem 4 *If* $(\phi, \mathbf{G}_0, \mathbf{G}_1, \ldots, \mathbf{G}_K)$ *is a MRAP, then*
$(\phi, \mathbf{G}_0 - \lambda\mathbf{I}, \mathbf{G}_1, \ldots, \mathbf{G}_K, \lambda\mathbb{1}\phi)$ *and* $(\phi, \mathbf{G}_0 - \lambda\mathbf{I}, \mathbf{G}_1, \ldots, \mathbf{G}_K, \lambda\mathbf{I})$ *are MRAPs for all* $\lambda \geq 0$.

If the solution $\psi\mathbf{F} = \mathbf{0}$, $\psi\mathbb{1} = 1$ is non-unique and one of the three conditions holds, then $\mathbf{F}$ is still the matrix of a rational process but does not fulfill the unique stationary condition

of MRAPs. For the process $(\eta, \mathbf{F}_0, \ldots, \mathbf{F}_K)$ this can imply that $\mathcal{V}_{(\eta,\mathbf{F}_0,\ldots,\mathbf{F}_K)}$ does not span $\mathbb{R}^n$. This property is the counterpart of reducibility of Markov models.

3. Rational Automata

The class of rational automata which is defined in this section is based on the class of stochastic automata as they have been proposed by (Plateau, 1985; Plateau and Fourneau, 1991). The original version of stochastic automata has exponentially distributed event times and contains state dependent rates. Only recently stochastic automata have been formally extended by integrating PH distributions in (Sbeity et al., 2008). Implicitly PH distributions have already been used in composed stochastic automata, stochastic Petri nets or stochastic process algebras before for example in (Buchholz et al., 2000; Dayar et al., 1997; El-Rayes et al., 1999). Here we use similar ideas. However, instead of using PH distributions, we use MRAPs to model durations of events. The approach follows in some sense the generalization of (Sbeity et al., 2008) but our approach differs in several details apart from the use of MRAPs instead of PH distributions.

3.1. The Skeleton Automaton

We define the skeleton automaton as a finite automaton and denote the states of the skeleton automaton as locations. Denote by $\mathcal{L}$ the finite set of locations for an automaton. The dynamic behavior of an automaton is defined by transitions which are either local or belong to a class $\mathcal{E}_s$ of global events that are later used for synchronization. Define $\mathcal{E} = \mathcal{E}_s \cup \{\varepsilon\}$ where ε denotes local transitions. Let $L = |\mathcal{L}|$ be the number of locations, then the behavior of the skeleton automaton can be described by a set of $L \times L$ matrices. Matrix $\mathbb{P}_e$ $(e \in \mathcal{E})$ describes the transitions due to event e. If event e is disabled in location x, then row x is zero in $\mathbb{P}_e$. If e is enabled in x, then row x defines a probability distribution, i.e., $\mathbb{P}_e(x, y) \geq 0$ and $\sum_{y \in \mathcal{L}} \mathbb{P}_e(x, y) = 1$. Events may also start and end in the same location but we assume that $\mathbb{P}_\epsilon(x, x) = 0$ to simplify the presentation of the descriptor matrix as introduced in the following section. We furthermore assume that $\mathbb{P}_e$ equals the identity matrix if it is not explicitly defined which means that additional events without any effect are added. We define for a location x: $Ena(x) \subseteq \mathcal{E}$ as the set of events e such that $\mathbb{P}_e(x, y) > 0$ for some y.

Furthermore, we define the initial location of the automaton as $x_{ini} \in \mathcal{L}$. This definition can be easily extended to a initial distribution by defining a probability distribution over the

set of locations.

Examples

Different example automata are shown in Figures 1-3. We will briefly describe them and continue to introduce the complete models in the following sections.

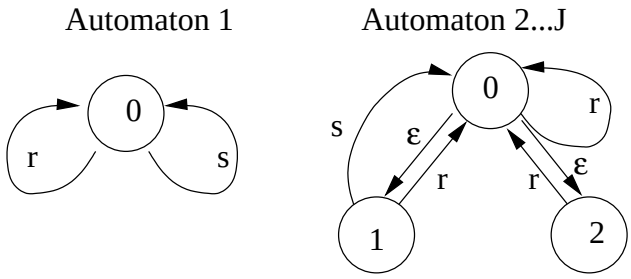

Figure 1: Automata of the first example RAN

Example 3 *The example (see Fig. 1) consists of two types of automata,* Automaton 1 *has only a single location that enables the transitions r and s. It will act like a control process for the other automata. The remaining automata belong to the second type with three locations. In location* 0 *the automaton does some local processing which ends with a local transition and results in a successful computation (location* 1*) or an unsuccessful computation (location* 2*). From location* 2 *the automaton has to be reset with an r transition. After a successful computation the automaton can also be reset via an r transition or it can start a new service via an s transition. Transitions labeled with r and s occur only synchronously in all automata. The automata are described by the following matrices.*

$$\mathbb{P}_s^{(1)} = \mathbb{P}_r^{(1)} = (1)\,,\ \mathbb{P}_\epsilon^{(j)} = \begin{pmatrix} 0 & 1-p & p \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix},\ \mathbb{P}_r^{(j)} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix},\ \mathbb{P}_s^{(j)} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

where $j = 2, \ldots, J$ and p is the probability of an unsuccessful computation.

Example 4 *The example (see Fig. 2) describes an automaton generating arrivals (*Automaton 1*) followed by two queues (*Automata 2 and 3*) with capacity N_i ($i = 1, 2$), respectively. The queues have specific failure event that is enabled whenever the queue is non-empty. If the failure occurs in one of the queues both queues become empty and the input process is reset. All automata are synchronized via the failure events f_1, f_2, event a describes an arrival to the first queue and event s a service at the first queue corresponding to an arrival at the second*

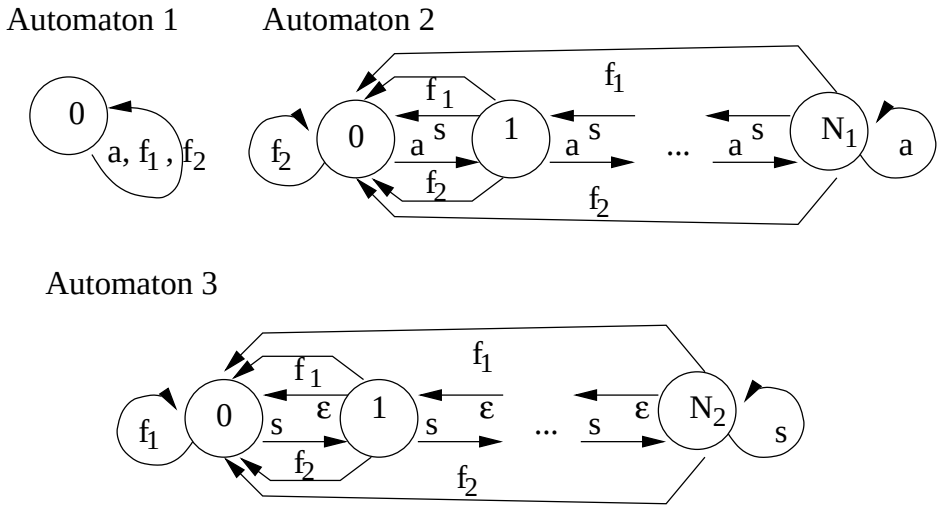

Figure 2: Automata of the second example RAN

queue. We skip the presentation of the matrices $\mathbb{P}_e^{(j)}$ *since they follow from the graphical representation of the automata.*

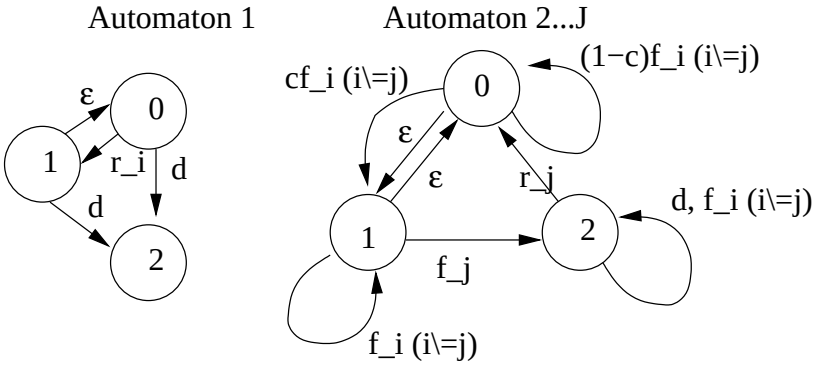

Figure 3: Automata of the third example RAN

Example 5 *The example consists again of two types of automata. The first is a repair unit which performs a repair operation of automaton* i *(*$i = 2, \ldots, J$*) via event* r_i*. Afterwards it requires some spare time in location* 1 *before the next repair can be started. If an event of type* d *occurs, the repair component goes down. The second type of automata describes components subject to failures and recovery. In location* 0 *a component is working properly. Via an internal transition, the component might get into recovery location* 1*. From the recovery location, the component may recover by an internal transition or a second failure may occur bringing the component down to location* 2*. Failures in component* j *(*$j = 2, \ldots, J$*) are described by event* f_j*. If a failure occurs in a component, then other working components*

are affected with probability $(1-c)$*. If a component is affected it performs a transition from location* 0 *to* 1*. A component in failure location* 2 *goes back to the working location* 0 *via event* r_j*. Event* d *is also enabled in location* 2 *but does not modify the location if it occurs. The same holds for the events* f_i *(*$i \in \{2,\ldots,J\} \setminus \{j\}$*) in the locations* 1 *and* 2 *of automaton* j*. Automaton 1 and* j *(*$j = 2,\ldots,J$*) are synchronized by transitions labeled with* d*,* r_j *and automaton* j *(*$j = 2,\ldots,J$*) and* i *(*$i \in \{2,\ldots,J\} \setminus \{j\}$*) are synchronized by transitions labeled with* f_i*.*

3.2. Adding Timing and Weight Information

The usual timing in stochastic automata (Plateau, 1985) is defined by exponential distributions such that the resulting automaton can be interpreted as a CTMC with labeled transitions. We now extend this interpretation by using MRAPs for the timing. Let $\mathcal{D}$ be a set of MRAPs and denote by $(\pi^{(d)}, \mathbf{G}_0^{(d)}, \ldots, \mathbf{G}_{K_d}^{(d)})$ the dth MRAP in the set with $n^{(d)}$ phases. Denote by $Class(d) = \{1,\ldots,K_d\}$ the set of all classes of MRAP d. This definition naturally applies to a set of MRAPs, $Class(\mathcal{D}) = \bigcup_{d\in\mathcal{D}} Class(d)$. Observe that the class of MRAPs contains RAPs (with $K_d = 1$), MMAPs (with $\mathbf{G}_0^{(d)}(x,y) \geq 0$ for $x \neq y$ and $\mathbf{G}_k^{(d)} \geq \mathbf{0}$ for $1 \leq k \leq K_d$), MAPs (an MMAP with $K_d = 1$), ME distributions (with $K_d = 1$ and $\mathbf{G}_1^{(d)} = -\mathbf{G}_0^{(d)}\mathbb{1}\pi^{(d)}$) and PH distributions (a ME distribution where $\mathbf{G}_0^{(d)}(x,y) \geq 0$ for $x \neq y$ and $\mathbf{G}_1^{(d)} \geq \mathbf{0}$).

Since automata are composed to build a network, the definition of timing information for synchronized transitions is often crucial. One common solution is to define one automaton as a master automaton that determines the timing of an event and the remaining automata only react (see e.g., (Sbeity et al., 2008)). The disadvantage of this approach is that only the complete model has an interpretation as a process since the behavior of isolated automata according to transitions where the automaton is not the master is not defined. Thus, we choose a slightly different approach by associating an exponential distribution with rate 1 to those events where the timing is triggered by the environment. The exponential distribution with rate 1 acts as a neutral element according to the synchronized composition of MRAPs which is the basic operation to compose automata in a network. Let $(\pi, \mathbf{G}_0, \ldots, \mathbf{G}_K)$ be a MRAP and let $(\phi = (1), \mathbf{H}_0 = (-K), \mathbf{H}_1 = (1), \ldots, \mathbf{H}_K = (1))$, then the synchronized composition via $\mathcal{C} = \{1,\ldots,K\}$ where the initial vector is computed as the Kronecker

product of the initial vectors is given by

$$\begin{array}{lll} \eta & = \pi \otimes (1) & = \pi \\ \mathbf{F}_0 & = \mathbf{G'}_0 \oplus (0) + \sum_{k=1}^{K} \hat{\mathbf{G}}_k \otimes (1) & = \mathbf{G}_0 \\ \mathbf{F}_k & = \mathbf{G}_k \otimes (1) & = \mathbf{G}_k \end{array}$$

For a location $x \in \mathcal{L}$ let $Act(x) \subseteq \mathcal{D}$ be the set of MRAPs that are *active* in x. Each class $k \in Class(Act(x))$ is assigned to exactly one $e \in Ena(x)$ but it is possible that more than one class k is assigned to an event. For a valid specification at least one k has to be assigned to each event. Let $Assign(x, e) \subseteq Class(Act(x))$ be the set of classes of MRAPs assigned to event e in location x. We usually use the notation (d, k) for elements of $Assign(x, e)$ where d is the MRAP and k denotes the class. Furthermore, we define for a location, event pair, (x, e), where $e \in Ena(x)$, the set $Reset(x, e) \subseteq Act(x)$ which defines the set of MRAPs that are reset when event e occurs in location x.

3.3. The Stochastic Process described by a RA

We denote the skeleton automaton plus the timing information as a rational automaton (RA). The behavior of the RA is as follows: It starts in an initial location x_{ini} by initializing all $d \in Act(x_{ini})$ according to the initial vector $\pi^{(d)}$. All MRAPs from $Act(x_{ini})$ run in parallel. If a MRAP generates an event that is associated to event $e \in Ena(x)$, the corresponding transition e occurs in the automaton and the successor location is chosen according to the probability distribution defined in row x_{ini} of matrix $\mathbb{P}_e$. Let y be the successor location. Then upon entering y all MRAPs $d \in Reset(x, e) \cap Act(y)$ and all newly activated MRAPs $d \in Act(y) \setminus Act(x)$ are reinitialized according to $\pi^{(d)}$. The remaining MRAPs $d \in (Act(y) \cap Act(x)) \setminus Reset(x, e)$ keep their phases.

This behavior will now be described formally. The state space of a RA is defined by expanding the location space by the state vectors of the active MRAPs in each location. Thus,

$$\mathcal{S} = \bigcup_{x \in \mathcal{L}} \mathcal{S}_x = \bigcup_{x \in \mathcal{L}} \bigtimes_{d \in Act(x)} \mathcal{S}^{(d)}$$

where $\mathcal{S}^{(d)}$ is the set space of MRAP d and $\mathcal{S}_x$ is the set of states belonging to location x. The size of the state space equals

$$n = |\mathcal{S}| = \sum_{x \in \mathcal{L}} \prod_{d \in Act(x)} n^{(d)}.$$

We assume that the elements in $\mathcal{D}$ are ordered such that the enumeration of elements of $\mathcal{D}$ is well defined.

The automata matrices $\mathbb{P}_e$ are expanded by considering the phases of the MRAPs. We denote the resulting matrices by $\mathbf{Q}_e$. Each matrix $\mathbf{Q}_e$ is structured into $|\mathcal{L}| \times |\mathcal{L}|$ submatrices such that $\mathbf{Q}_e[x, y]$ describes the transitions between location x and location y. The blocks are built as follows.

$$\mathbf{Q}_e[x,y] = \begin{cases} \bigoplus_{d \in Act(x)} \mathbf{G}_0^{(d)} & \text{for } e = \varepsilon \text{ and } x = y, \\ \mathbb{P}_e(x,y) \sum_{(d,k) \in Assign(x,e)} \mathbf{V}[d,k,x,y] & \text{otherwise} \end{cases} \tag{2}$$

where $\mathbf{V}[d, k, x, y] = \otimes_{c \in Act(x)} \mathbf{V}_c[d, k, x, y]$ and

$$\mathbf{V}_c[d,k,x,y] = \begin{cases} \mathbf{G}_k^{(d)} & \text{if } d = c \text{ and } d \in Act(y) \\ \mathbf{G}_k^{(d)} \mathbb{1}_{n^{(d)}} & \text{if } d = c \text{ and } d \notin Act(y) \\ \mathbf{I}_{n^{(c)}} & \text{if } d \neq c \text{ and } c \in (Act(x) \cap Act(y)) \setminus Reset(x,a) \\ \mathbb{1}_{n^{(c)}} \pi^{(d)} & \text{if } d \neq c \text{ and } \ c \in Act(x) \cap Act(y) \cap Reset(x,a) \\ \mathbb{1}_{n^{(c)}} & \text{if } d \neq c \text{ and } c \in Act(x) \setminus Act(y) \\ \pi^{(c)} & \text{if } d \neq c \text{ and } c \in Act(y) \setminus Act(x) \\ 1 & \text{otherwise} \end{cases}$$

where $\mathbf{I}_n$ and $\mathbb{1}_n$ are identity matrix and the unit column vector of dimension n, respectively. As before let $\mathbf{Q'}_\varepsilon = \mathbf{Q}_\varepsilon + \sum_{e \in \mathcal{E}_s} \hat{\mathbf{Q}}_e$ with $\hat{\mathbf{Q}}_e = diag(\mathbf{Q}_e \mathbb{1})$.

$\mathbf{V}_c[d, k, x, y]$ describe the role of event k of MRAP d in the transition from x to y when MRAP c is active. Starting from the top the expression for $\mathbf{V}_c[d, k, x, y]$ describes the following cases:

- The transition is due to MRAP c and it remains active.
- The transition is due to MRAP c and it becomes inactive.
- The transition is not due to MRAP c, it remains active and it is not reset.
- The transition is not due to MRAP c, it remains active and it is reset (re-initialized with vector $\pi^{(d)}$).
- The transition is not due to MRAP c and it becomes inactive.
- MRAP c was inactive and becomes active after the transition.
- MRAP c was inactive and remains inactive after the transition.

In the following we describe a rational automaton (RA) by its matrices, i.e., $RA = (\mathbf{Q}_\varepsilon, \mathbf{Q}_e(e \in \mathcal{E}_s))$ is a RA. A vector describing the state of a RA contains one entry per state (i.e., has length n) and can be decomposed into subvectors $\pi[x]$ according to the locations

of the skeleton automaton. For state vector ν vector $\tilde{\nu}$ is an aggregation that contains one entry per location and is computed as

$$\tilde{\nu}[x] = \sum_{y \in \mathcal{S}_x} \nu[x](y),$$

where $\nu[x](y)$ is the entry for state y of location x. The initial vector π of a RA is defined by assigning the initial vectors of the MRAPs to the states belonging to locations x_{ini}; i.e.,

$$\pi[x] = \begin{cases} \bigotimes_{d \in Act(x_{ini})} \pi^{(d)} & \text{if } x = x_{ini}, \\ \mathbf{0} & \text{otherwise.} \end{cases} \tag{3}$$

As it is summarized in the following theorem, a RA in isolation has a well defined behavior which might contain negative elements at the state level but at the location level the behavior is characterized by non-negative probabilities. For example, $\nu[x](y)$ might be negative, but $\tilde{\nu}[x]$ is non-negative.

Theorem 5 *Let $(\pi, \mathbf{Q}_\varepsilon, \mathbf{Q}_e (e \in \mathcal{E}_s))$ be a RA, then*

1. *for any $t > 0$ the vector $\tilde{\nu}_t$ that results from $\nu_t = \pi e^{\mathbf{Q}t}$ with $\mathbf{Q} = \sum_{e \in \mathcal{E}} \mathbf{Q}_e$ represents a probability distribution over the locations (i.e., $\tilde{\nu}_t[x] \geq 0$ for all $x \in \mathcal{L}$ and $\sum_{x \in \mathcal{L}} \tilde{\nu}_t[x] = 1$), and*

2. *it is a MRAP, if $\eta \mathbf{Q} = \mathbf{0}$ and $\eta \mathbb{1} = 1$ has a unique solution and $\lim_{t \to \infty} \pi e^{\mathbf{Q}t} = \eta$.*

Before we consider the computation of transient or stationary vectors in the following section, networks of RAs are introduced and it is shown that a network can again be interpreted as a RA.

Examples

We extend the examples for the skeleton automata presented in Example 3-5 by introducing timing information. Although we do not explicitly define for an event a master automaton that determines the time, as it is done in other approaches, we have implicitly such a concept in our models using the Poisson signal with rate 1 and assuming that the timing of an event is defined in one automaton.

Example 3 *In Example 3 we assume that automaton* 1 *defines the timing for both events* r *and* s*. We assume that the event type* s *has an exponential duration with rate* λ*. The event* r *has a duration with a low coefficient of variation. We take one of the ME distributions from (Éltető et al., 2006) for example the distribution with* 3 *phases and squared coefficient of variation* 0.2009 *presented in Example 1. Let* $(\pi, \mathbf{G}_0)$ *be the representation of the distribution. We furthermore assume that event* r *is reset when* s *occurs. Automaton* 1 *is then described by the following matrices*

$$\mathbf{Q}_\varepsilon = \mathbf{G}_0 - \lambda\mathbf{I}, \quad \mathbf{Q}_s = \lambda \begin{pmatrix} \pi \\ \pi \\ \pi \end{pmatrix}, \quad \mathbf{Q}_r = -\mathbf{G}_0\mathbb{1}\pi \ .$$

The automaton observes the conditions of Theorem 4 and describes therefore a MRAP for all $\lambda > 0$*.*

For the second type of automata events r *and* s *are locally driven by an exponential distribution with rate* 1 *that acts as a neutral element in composition as shown above. Timing of the local events is given by a PH distribution* $(\phi, \mathbf{H}_0)$ *which is expanded into a MMAP with* 2 *classes as* $(\mathbf{H}_0, (1-p)\mathbf{h_0}\phi, p\mathbf{h_0}\phi)$ *with* $0 < p < 1$ *the probability that a computation is not successful, where* $\mathbf{h_0} = -\mathbf{H}_0\mathbb{1}$*. We assume that the occurrence of an* r *event in location* 0 *resets local events. The following matrices describe the second type of automata.*

$$\mathbf{Q}_\varepsilon = \begin{pmatrix} \mathbf{H}_0 - \mathbf{I} & (1-p)\mathbf{h_0} & p\mathbf{h_0} \\ \mathbf{0} & -2 & 0 \\ \mathbf{0} & 0 & -1 \end{pmatrix}, \ \mathbf{Q}_r = \begin{pmatrix} \mathbb{1}\phi & \mathbf{0} & \mathbf{0} \\ \phi & 0 & 0 \\ \phi & 0 & 0 \end{pmatrix}, \ \mathbf{Q}_s = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \phi & 0 & 0 \\ \mathbf{0} & 0 & 0 \end{pmatrix} \ .$$

Example 4 *For Example 4 automaton 1 describes the arrivals which are realized by a RAP* $(\pi, \mathbf{G}_0, \mathbf{G}_1)$*. We assume that a failure in one of the queues resets the arrival process such that*

$$\mathbf{Q}_\varepsilon = \mathbf{G}_0 - 3\mathbf{I}, \ \mathbf{Q}_a = \mathbf{G}_1, \ \mathbf{Q}_s = \mathbf{I}, \ \mathbf{Q}_{f_1} = \mathbf{Q}_{f_2} = \mathbb{1}\upsilon,$$

where υ *is the state vector after the RAP has been reset. The failure time distribution of the second automaton is given by ME distribution* $(\phi, \mathbf{F}_0)$ *and the service time is exponentially*

distributed with rate μ_1. The matrices for automaton 2 are as follows.

$$\mathbf{Q}_\varepsilon = \begin{pmatrix} -2 & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_0-(\mu_1+2)\mathbf{I} & \mathbf{0} & \cdots & \vdots \\ \vdots & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & & \vdots \\ \vdots & & \dots & \mathbf{F}_0-(\mu_1+2)\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \cdots & \cdots & \mathbf{0} & \mathbf{F}_0-(\mu_1+2)\mathbf{I} \end{pmatrix},$$

$$\mathbf{Q}_a = \begin{pmatrix} 0 & \phi & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \mathbf{0} \\ \vdots & & & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \cdots & \cdots & \cdots & \mathbf{I} \end{pmatrix}, \quad \mathbf{Q}_s = \begin{pmatrix} 0 & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mu_1\mathbb{1} & \mathbf{0} & \mathbf{0} & \ddots & \vdots \\ \mathbf{0} & \mu_1\mathbf{I} & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mu_1\mathbf{I} & \mathbf{0} \end{pmatrix},$$

$$\mathbf{Q}_{f_1} = \begin{pmatrix} 0 & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{f}_0 & \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ \mathbf{f}_0 & \mathbf{0} & \cdots & \cdots & \mathbf{0} \end{pmatrix}, \quad \mathbf{Q}_{f_2} = \begin{pmatrix} 1 & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbb{1} & \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ \mathbb{1} & \mathbf{0} & \cdots & \cdots & \mathbf{0} \end{pmatrix},$$

where $\mathbf{f}_0 = -\mathbf{F}_0\mathbb{1}$. The failure time distribution for the third automaton is given by a ME distribution $(\eta, \mathbf{H}_0)$ and the internal event of this automaton represents the service with rate μ_2. The automaton is described by the following matrices.

$$\mathbf{Q}_\varepsilon = \begin{pmatrix} -3 & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mu_2\mathbb{1} & \mathbf{H}_0-(\mu_2+3)\mathbf{I} & \mathbf{0} & \cdots & \vdots \\ \mathbf{0} & \mu_2\mathbf{I} & \mathbf{H}_0-(\mu_2+3)\mathbf{I} & & \vdots \\ \vdots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & & \mu_2\mathbf{I} & \mathbf{H}_0-(\mu_2+3)\mathbf{I} \end{pmatrix},$$

$$\mathbf{Q}_s = \begin{pmatrix} 0 & \eta & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \mathbf{0} \\ \vdots & & & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \cdots & \cdots & \cdots & \mathbf{I} \end{pmatrix}, \quad \mathbf{Q}_{f_1} = \begin{pmatrix} 1 & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbb{1} & \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ \mathbb{1} & \mathbf{0} & \cdots & \cdots & \mathbf{0} \end{pmatrix},$$

$$\mathbf{Q}_{f_2} = \begin{pmatrix} 0 & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{h}_0 & \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ \mathbf{h}_0 & \mathbf{0} & \cdots & \cdots & \mathbf{0} \end{pmatrix},$$

where $\mathbf{h}_0 = -\mathbf{H}_0\mathbb{1}$ *and* $\mathbf{Q}_a = \mathbf{I}$.

Example 5 *In Example 5 we assume that for automaton 1 all event times are exponentially distributed. Local events occur with rate* μ*,* d *events with rate* λ *and* r_i *(*$i = 2, \ldots, J$*) events with rate* ω*. This defines the following matrices for automaton 1.*

$$\mathbf{Q}_\varepsilon = \begin{pmatrix} (1-J)(\omega+1)-\lambda & 0 & 0 \\ \mu & -\mu-\lambda-J+1 & 0 \\ 0 & 0 & 1-J \end{pmatrix},$$
$$\mathbf{Q}_d = \begin{pmatrix} 0 & 0 & \lambda \\ 0 & 0 & \lambda \\ 0 & 0 & 0 \end{pmatrix},\ \mathbf{Q}_{r_i} = \begin{pmatrix} 0 & \omega & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix},\ \mathbf{Q}_{f_i} = \mathbf{I}.$$

Note that the matrices satisfy $(\mathbf{Q}_\varepsilon + \mathbf{Q}_d + \sum_{i=2}^{J}(\mathbf{Q}_{r_i} + \mathbf{Q}_{f_i}))\mathbb{1} = \mathbf{0}$.

For the second automata type describing automaton j*,* $j \in \{2, \ldots, J\}$*, we assume that the timing of failures is defined by a MAP* $(\pi, \mathbf{G}_0, \mathbf{G}_1)$*. In automaton* j*, this failure process is assigned to the* ε *transition in location* 0 *and to* f_j *transitions in location* 1*. It is not enabled in location* 2 *but it remains enabled and is not reset when entering location* 0 *from* 1 *by a local* ε *transition. Observe that this modeling assures that consecutive failure times that correspond to transitions from location* 0 *to* 1 *and from* 1 *to* 2 *are correlated as defined by the RAP. Local transitions in location* 1 *have an exponential distribution with rate* δ*. To all other events* f_i*,* r_i *(*$i \in \{2, \ldots, J\} \setminus \{j\}$*) and* d *an exponential distribution with rate* 1 *is assigned. The following matrices describe the process.*

$$\mathbf{Q}_\varepsilon = \begin{pmatrix} \mathbf{G}_0 + (4-2J)\mathbf{I} & \mathbf{G}_1 & \mathbf{0} \\ \delta\mathbf{I} & \mathbf{G}_0 + (4-2J-\delta)\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -2J \end{pmatrix},\ \mathbf{Q}_{r_j} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \pi & \mathbf{0} & 0 \end{pmatrix},$$
$$\mathbf{Q}_{r_i} = \mathbf{I},\ \mathbf{Q}_{f_j} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{G}_1\mathbb{1} \\ \mathbf{0} & \mathbf{0} & 0 \end{pmatrix},\ \mathbf{Q}_{f_i} = \begin{pmatrix} (1-c)\mathbf{I} & c\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 1 \end{pmatrix},\ \mathbf{Q}_d = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 1 \end{pmatrix}.$$

Note that $(\mathbf{Q}_\varepsilon + \mathbf{Q}_d + \mathbf{Q}_{f_j} + \mathbf{Q}_{r_j} + \sum_{i=2, i\neq j}^{J}(\mathbf{Q}_{f_i} + \mathbf{Q}_{r_i}))\mathbb{1} = \mathbf{0}$*, since* $(\mathbf{G}_0 + \mathbf{G}_1)\mathbb{1} = \mathbf{0}$.

4. Networks of Rational Automata

RAs can be combined to rational automata networks (RANs) which are an extension of SANs. In this section we first describe the composition of RAs and show that this defines again a RA before we introduce the representation of the transition matrix using Kronecker products and sums. Finally, we briefly consider the computation of the transient and stationary distribution for the skeleton automaton.

4.1. Composition of RAs

A rational automata network (RAN) results from the synchronous composition of J RAs. We use the notation $(\pi^{(j)}, \mathbf{Q}_e^{(j)}(e \in \mathcal{E}))$ for RA j, denote the location and state space of RA j by $\mathcal{L}^{(j)}$ and $\mathcal{S}^{(j)}$ and assume that all RAs are defined for the same event set $\mathcal{E}$. Automata with differing event sets are first transformed by adding additional events as it is described above Theorem 3. The composition is formally defined as

$$(\pi^{(1)}, \mathbf{Q}_\varepsilon^{(1)}, \mathbf{Q}_e^{(1)}(e \in \mathcal{E}_s)) \| _{\mathcal{E}_s} \ldots \| _{\mathcal{E}_s} (\pi^{(J)}, \mathbf{Q}_\varepsilon^{(J)}, \mathbf{Q}_e^{(J)}(e \in \mathcal{E}_s)) \tag{4}$$

All RAs perform events labeled with ε independently, all remaining events can only occur synchronously in all automata, that is $\mathcal{C} = \mathcal{E}_s$. This is the same behavior as it is defined for SANs. For the composition the results of Theorem 3 have to be considered. Thus, if event e is associated to a MRAP that is not a MMAP, then event e has to be always enabled by other automata or it is slowed down/speeded up and the scaled matrices still describe a stochastic process. For MMAPs we do not have such restrictions.

The state and location space of a RAN is J-dimensional such that states and locations can be described by vectors. Vector $(x_1, \ldots, x_J)$ where $x_j \in \mathcal{L}^{(j)}$ describes a location and $(y_1, \ldots, y_J)$ with $y_j \in \mathcal{S}^{(j)}$ a state. Of course, a location represents a set of states. The value of the transition matrix element from state $(y_1, \ldots, y_J)$ to $(z_1, \ldots, z_J)$ via event e equals $\prod_{j=1}^{J} \mathbf{Q}_e^{(j)}(y_j, z_j)$. The value of the initial vector element for state $(y_1, \ldots, y_J)$ equals $\prod_{j=1}^{J} \pi^{(j)}(y_j)$. The number of locations of a RAN equals $\prod_{j=1}^{J} |\mathcal{L}^{(j)}|$ and the number of states equals $\prod_{j=1}^{J} |\mathcal{S}^{(j)}|$.

Examples

We briefly describe the behavior of the composed automata networks.

Example 3 *In Example 3 the automata* 2 *through* J *run in parallel and perform their computations that are either successful or unsuccessful. If all processes reach a successful end, the joint* s *event starts a new cycle. Automaton* 1 *controls the system by resetting the processes after a timeout expired and the* s *event did not occur. Whenever the* s *event takes place the timeout is reset.*

Example 4 *The example is a* $RAP/M/1/N_1 \rightarrow ./M/1/N_2$ *queuing system with failures that reset the whole system.*

Example 5 *In the example we have* $J-1$ *components subject to failures. From a single failure a component is able to recover. However, if a second failure occurs during recovery, the component requires repair by a repair unit. If a component goes down, this initiates with probability* c *a simultaneous failure in any other operating component. If all components are down, the repair unit might also go down into an absorbing state which means that the whole system is down.*

4.2. Structure of the Descriptor

The descriptor of a RAN is an $n \times n$ matrix $\mathbf{Q}$ which can be described by a sum of Kronecker products and sums as follows.

$$\begin{aligned} \mathbf{Q} &= \mathbf{Q'}_{\varepsilon} + \sum_{e \in \mathcal{E}_s} \mathbf{Q}_e - \sum_{e \in \mathcal{E}_s} \mathbf{D}_e \\ \text{where} \quad \mathbf{Q'}_{\varepsilon} &= \bigoplus_{j=1}^{J} \mathbf{Q'}_{\varepsilon}^{(j)}, \qquad \mathbf{Q}_e = \bigotimes_{j=1}^{J} \mathbf{Q}_e^{(j)}, \\ \mathbf{D}_e &= \bigotimes_{j=1}^{J} \hat{\mathbf{Q}}_e^{(j)}, \qquad \hat{\mathbf{Q}}_e^{(j)} = diag(\mathbf{Q}_e^{(j)} \mathbb{1}) \end{aligned} \tag{5}$$

The initial vector is given by

$$\pi = \bigotimes_{j=1}^{J} \pi^{(j)}. \tag{6}$$

Computation of the descriptor is completely analogous to the computation of the descriptor for a SAN in (Plateau, 1985; Plateau and Fourneau, 1991), the only difference is that at the moment we do not consider state dependent transitions and generalized Kronecker products. However, the introduction of rates depending on the location of an automaton is possible but requires in general a proof that the resulting matrices still specify a stochastic process.

Theorem 6 *If* $\mathbf{Q}$ *is irreducible and conditions of composing MRAPs given in Theorem 3 have been observed, then* $(\pi, \mathbf{Q}'_{\varepsilon} - \sum_{e\in\mathcal{E}_s} \mathbf{D}_e, \mathbf{Q}_e(e \in \mathcal{E}_s))$ *is a MRAP.*

Examples

The representation of the descriptor matrices for Examples 3-5 are defined above. We now briefly argue why the conditions of Theorem 3 yield a valid composition.

Example 3 *We have* $\mathcal{E}_s = \{r, s\}$ *and* $\mathbf{Q}_s^{(1)}\mathbb{1} = \lambda\mathbb{1}$. *Since the automata* $2, \ldots, J$ *are MMAPs they can be composed and the resulting automaton is a MMAP. For this automaton* $\mathbf{Q}_r^{(2,\ldots,J)}\mathbb{1} = \mathbb{1}$ *holds. Consequently, the automaton can be composed with automaton 1 and the conditions of Theorem 3 case (3) are met.*

Example 4 *For automaton* 1 *and event a the conditions of Theorem 3 case (3) hold since* $\mathbf{Q}_a^{(2)}\mathbb{1} = \mathbf{Q}_a^{(3)}\mathbb{1} = \mathbb{1}$. *Similarly, for automaton* 2 *and events* $e \in \{f_1, s\}$ $\mathbf{Q}_e^{(1)}\mathbb{1} = \mathbf{Q}_e^{(3)}\mathbb{1} = \mathbb{1}$ *and for automaton* 3 $\mathbf{Q}_{f_2}^{(1)}\mathbb{1} = \mathbf{Q}_{f_2}^{(2)}\mathbb{1} = \mathbb{1}$ *hold. Thus, the conditions of Theorem 3 case (3) are observed with* $c_e = 1$ *for* $e \in \{a, s, f_1, f_2\}$ *which are all events in the RAN. Consequently, the composed model is a RA if the components are RAs.*

Example 5 *Since all components are MMAPs, the composition is also a MMAP according to Theorem 3 case (1).*

It should be noted that Examples 3-4 define irreducible descriptor matrices. In Example 5, location 2 of the first automaton is absorbing. Since location 2 in the first automaton can only be entered via a d transition which is only enabled in location 2 of the automata $2, \ldots, J$, states belonging to locations $(2, x_2, \ldots, x_J)$ with at least one $x_j \neq 2$ are unreachable. This implies that the network reaches from the initial state a single absorbing state and the descriptor contains unreachable states such that methods as developed for the analysis of Markov models in (Buchholz et al., 2000; Buchholz, 1999) may be applied for analysis. Of course, since the model is absorbing, only transient analysis makes sense.

4.3. Numerical Analysis of RANs

For the lack of space, we cannot go too much into detail concerning numerical analysis methods for RANs. We briefly present some solution approaches that are applicable for transient or stationary analysis of RANs. The problem that $\mathbf{Q}$ might not be irreducible is not

explicitly considered. However, this problem occurs also in SANs and has been solved there (see for example (Buchholz et al., 2000; Buchholz, 1999)). Exactly the same approaches can be applied for RANs such that the following methods can be easily extended to the case where $\mathbf{Q}$ is not irreducible. Here we assume that for stationary analysis $\mathbf{Q}$ is irreducible.

For transient analysis the differential equation

$$\frac{d}{dt}\nu_t = \nu_t \mathbf{Q} \quad \text{with} \quad \nu_0 = \pi \tag{7}$$

has to be solved. An overview of different methods for the solution of (7) when $\mathbf{Q}$ is the generator matrix of a Markov chain can be found in (Stewart, 1994, Chapter 8). Most of the methods can also be applied if $\mathbf{Q}$ is the matrix of a RP. A notable exception is the uniformization approach (Stewart, 1994, Chapter 8.1) which is based on the correspondence between continuous time and discrete time MCs that does not hold for more general matrices. However, the different ODE solvers that are presented in (Stewart, 1994; Cellier and Kofman, 2010) can all be applied for the solution of (7). Of particular importance are Runge-Kutta methods which are based on the following iteration to compute ν_{t+h} from ν_t.

$$\nu_{t+h} = \nu_t + \sum_{l=1}^{L} b_l v^{(l)} \text{ where } v^{(l)} = \left(\nu_t + \sum_{i=1}^{l-1} a_{li} v^{(i)}\right) \mathbf{Q} \tag{8}$$

a_{li}, b_l are constants and L is the number of stages. Runge-Kutta methods compute the first L' ($\geq L$) terms of the Taylor series expansion. Different variants exist (see Cellier and Kofman, 2010). In our implementations we use the Runge-Kutta-Fehlberg (4,5) method (RKF45) and the Runge-Kutta Prince-Dormand (8,9) (RKPD89) both combined with an adaptive step-size control to set h according to a predefined error bound. All Runge-Kutta algorithms, like most other algorithms for transient analysis, are based on the repeated computation of vector matrix products which can be implemented using the compact representation of the descriptor matrix defined in (5) without building the whole matrix (e.g., Plateau and Fourneau, 1991; Buchholz et al., 2000).

Apart from Runge-Kutta methods other transient analysis techniques given in (Stewart, 1994) like implicit integration methods or projection methods may as well be used for the transient analysis of RANs. However, Runge-Kutta methods are often the most efficient among these approaches but are outperformed by uniformization if the system is not too stiff (Reibman and Trivedi, 1988). This implies that there is the decision to use a SAN with more states and a more efficient solver or to use a RAN with a slightly less efficient solver

but with less states. It depends, of course, on the concrete model which of both alternatives is preferable.

Stationary analysis requires the solution of the set of linear equations

$$\psi \mathbf{Q} = \mathbf{0} \text{ subject to } \psi \mathbb{1} = 1. \tag{9}$$

Again matrix $\mathbf{Q}$ is not necessarily a generator matrix since it may contain negative off-diagonal entries and more important for stationary analysis, the absolute value of a diagonal element is not necessarily larger than the absolute values of the off-diagonal elements in the row. If the state space is of a moderate size, then matrix $\mathbf{Q}$ can be generated from the Kronecker representation (2) and a direct solution method like LU-decomposition (Stewart, 1994, Chapter 2) can be applied. In contrast to generator matrices where a pivot step is not necessary, general matrices $\mathbf{Q}$ require the use of a pivot step to compute the solution.

For larger state spaces, direct methods which produce fill-in in the sparse matrix are not appropriate, instead iterative methods have to be used. For generator matrices of MCs a large number of different iterative solution methods is proposed in (Stewart, 1994). The classical iterative methods like the Power method, JOR or SOR rely on the structure of generator matrices where the absolute value of the diagonal element is at least as large as the largest off-diagonal element in the row. Since this property does no longer hold for $\mathbf{Q}$ matrices resulting form RANs the mentioned methods cannot be applied directly. The same holds for advanced numerical methods that are based on SOR or block SOR like aggregation/disaggregation methods (Stewart, 1994). Fortunately, projection methods like GMRES (Saad and Schultz, 1986) or BiCGStab (van der Vorst, 1992) can be applied for general matrices and have been used for MCs and even for SANs in (Stewart, 1994). Consequently, projection methods may as well be used for $\mathbf{Q}$ matrices resulting from RANs and, since they require no modification of the matrix $\mathbf{Q}$, they can be implemented without building matrix $\mathbf{Q}$. If $\mathbf{Q}$ is generated as a sparse matrix, then projection methods can be combined with preconditioners like incomplete LU factorizations. We made good experiences with the ILUTH type preconditions. For details about the algorithms we refer to (Stewart, 1994).

Although some numerical methods are available for the computation of the stationary vector of a RAN, more research is required to develop new and efficient numerical methods for this kind of matrices.

After the transient or stationary vector has been computed, result measures are derived. Usually these measures are defined at the level of the RAN and often they are defined according to the locations and not the detailed states. Thus, it is sufficient to consider the vectors $\tilde{\psi}$ and $\tilde{\nu}_t$ as aggregated stationary and transient vectors, respectively. According to Theorem 5 $\tilde{\psi}$ and $\tilde{\nu}_t$ contain a probability distribution. Let $\mathbf{r}^{(j)}$ be a column vector of length $|\mathcal{L}^{(j)}|$ that assigns a reward to each location of RA j. Then the stationary reward can be computed as

$$\tilde{\psi} \bigotimes_{j=1}^{J} \mathbf{r}^{(j)} \text{ or } \tilde{\psi} \bigoplus_{j=1}^{J} \mathbf{r}^{(j)}, \text{ and the transient reward as } \tilde{\nu}_t \bigotimes_{j=1}^{J} \mathbf{r}^{(j)} \text{ or } \tilde{\nu}_t \bigoplus_{j=1}^{J} \mathbf{r}^{(j)}.$$

Kronecker products assure that rewards are multiplied, e.g., to compute the probability that all RAs are in a subset of their location spaces, Kronecker sums assure that rewards are added, e.g., to compute the joint population. Other operations like minimum or maximum can also be defined to combine rewards of the RAs. Since the vectors $\tilde{\psi}$ and $\tilde{\nu}_t$ contain probability distributions, the standard approaches to compute expectations, higher moments or distributions of rewards can still be applied.

For the computation of throughputs, detailed state vectors are required. The stationary and transient throughput of event $e \in \mathcal{E}_s$ is given by

$$\psi \bigotimes_{j=1}^{J} \mathbf{Q}_e^{(j)} \mathbb{1} \text{ and } \nu_t \bigotimes_{j=1}^{J} \mathbf{Q}_e^{(j)} \mathbb{1}.$$

Observe that although the vectors and matrices may contain negative elements in these computations, the resulting throughput will be non-negative.

Examples

Example 3 *For Example 3 assume that $J = 3$, i.e., we have the control process and two other processes. The probability that a computation is not successful is $p = 0.1$ and the mean duration of a computation is $\phi(-\mathbf{H}_0)^{-1}\mathbb{1} = 1$. The r events are triggered by the ME distribution, $(\pi, \mathbf{G}_0)$, with three states and scv of* 0.2009. *Obsrve that an Erlang distribution of order 5 would be necessary to reach such a low coefficient of variation. We analyze the system for exponential, Erlang 2 and Cox 2 (with scv = 10 and equally occupied phases) distributed computation times and different mean values for the interevent time of the r events. As result measure we take the steady state throughput of s events (i.e., successful*

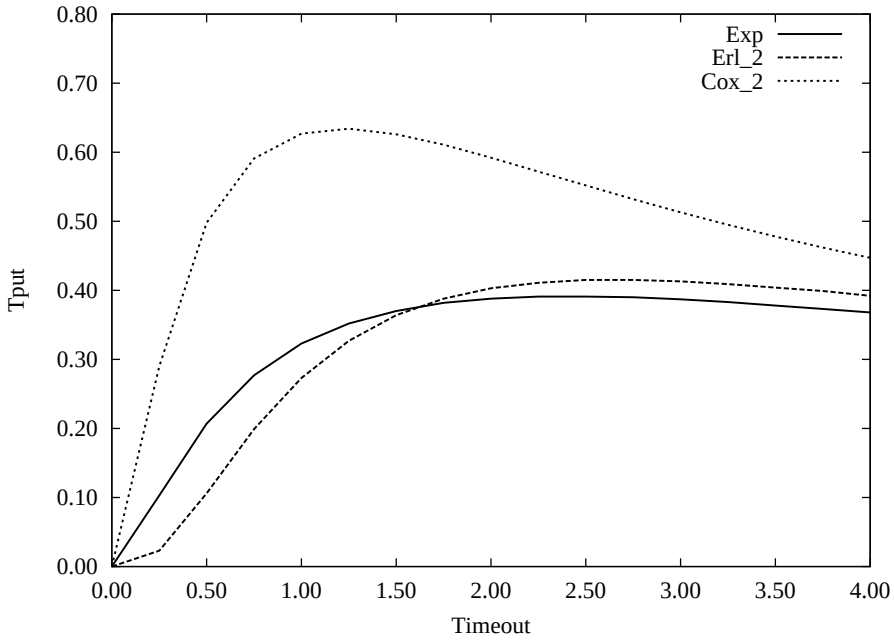

Figure 4: Throughput of Example 3

computations). Results are shown in Fig. 4. It can be seen that for the Coxian distribution with scv $= 10$*, a high throughput can be reached by resetting the computations frequently since the restart and the high coefficient of variation assure that small computation times are likely. The duration of the reset time has only a small effect for exponentially distributed computation times if the reset time is not too small. For Erlang distributed computation times it is better to choose a slightly longer reset time since small computation times are rarely observed and restarting a computation gives no benefits.*

To compare the solution times with different techniques we analyze the model with Coxian distributed computation times and J between 3 *and* 10*. Table 1 includes the size of the state space and the CPU time required for the computation of the steady state vector using an LU-decomposition, GMRES or BiCGStab with ILUTH preconditioning, and GMRES or BiCGStab without preconditioning. The LU-decomposition and the preconditioned projection techniques use a sparse matrix representation, whereas the projection techniques without preconditioning exploit the Kronecker representation and do not build matrix* $\mathbf{Q}$*. The iterative techniques are stopped if the residual becomes smaller than* 10^{-10}*. All experiments are performed on a standard PC with* 2.50 *GHz QuadCore CPU and 4 GB main memory. It can be noticed that the LU-decomposition can only be used for small configurations and the preconditioned projection techniques are clearly outperformed by the projection techniques without*

preconditioning that work with the Kronecker representation of matrix $\mathbf{Q}$.

J	states	CPU time in seconds				
		LU	GMRES	GMRES+ILUTH	BiCGS	BiCGS+ILUTH
3	48	0.00	0.00	0.00	0.00	0.00
4	192	0.00	0.00	0.00	0.00	0.00
5	768	0.63	0.02	0.03	0.01	0.03
6	3072	43.7	0.03	0.22	0.04	0.21
7	12288	2577	0.18	2.18	0.25	2.06
8	49152	−−	1.20	35.0	0.76	33.8
9	196608	−−	6.92	523	3.57	504
10	786432	−−	26.7	−−	21.6	−−

Table 1: Solution times for Example 3

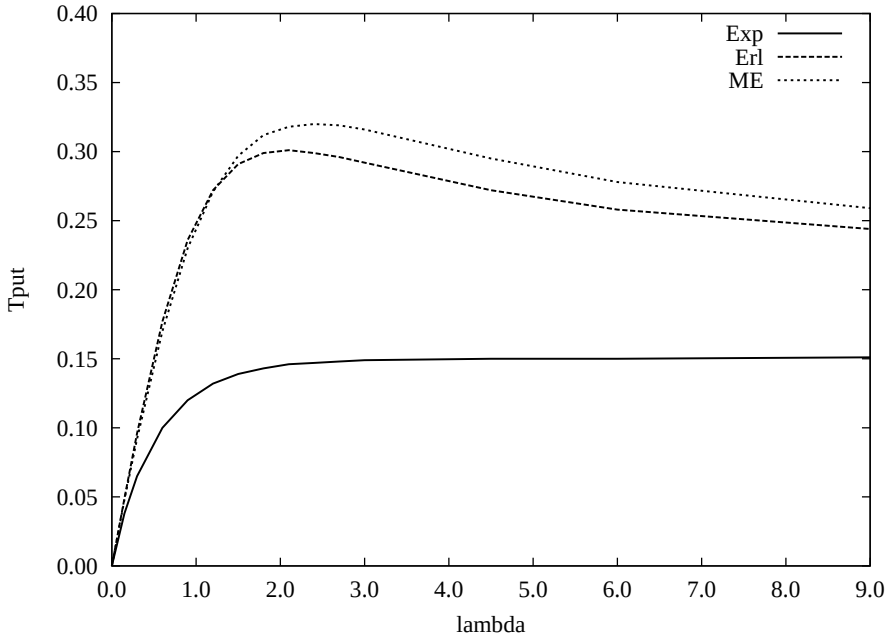

Figure 5: Mean throughput for different arrival rates λ for Example 4

Example 4 *For Example 4 we consider a configuration where arrivals are generated by a switched Poisson process with mean arrival rate* $\lambda/3$*, mean on time* 20 *and mean off time* 40*, i.e.,* $\mathbf{G}_0 = \begin{pmatrix} -\lambda - 0.05 & 0.05 \\ 0.025 & -0.025 \end{pmatrix}$, $\mathbf{G}_1 = \begin{pmatrix} \lambda & 0 \\ 0 & 0 \end{pmatrix}$. *If the arrival process is reset by a failure, it is switched off which implies that* $\upsilon = (0, 1)$. *Service times are exponentially distributed with rates* $\mu_1 = \mu_2 = 2$. *The buffer size is* 3 *for both queues. Buffer failures are modeled by different distributions with mean* 10. *We consider an exponential, an Erlang 3 and the ME distribution with* 3 *phases and low coefficient of variation in Example 1.*

We first perform a stationary analysis for different arrival rates λ and determine the throughput of the second queue which equals the system throughput. Results are shown in Figure 5. For exponentially distributed failure times, the throughput increases monotonically with the arrival rate. For the Erlang or ME distribution an optimal arrival rate exists that yields maximal throughput. This behavior results from the resetting of the failure times whenever the buffer becomes empty. The ME distribution with the smaller scv results in a higher throughput than the Erlang 3 distribution.

To compare solution times we consider a larger model with buffer size 10 for both queues. With ME distributed failure times the process has 1922 states. Computation of the transient distribution at $t = 100$ with RKF45 requires 2.06 second, RKPD89 needs 1.34 seconds. Alternatively, one can model the failure times with Erlang 5 distributions which have a similar coefficient of variation than the ME distributions. The state space of the resulting model includes 5202 states. However, uniformization requires only 0.95 seconds for the computation of the transient distribution at $t = 100$. Consequently, for this example and a transient analysis at $t = 100$ the state space reduction due to the substitution of Erlang by ME distribution gives no benefits. The situation is different for stationary analysis with BiCGStab which is the fastest solver for this model. For the model with ME distributions 254 iterations are required and the computation needs 0.07 seconds. With Erlang distributions 328 iterations and 0.24 seconds are needed.

Example 5 *Finally, we consider the transient analysis of Example 5 with $J = 3$ automata. Failure times are modeled by a two state MAP that generates positively correlated failure times, $\mathbf{G}_0 = \begin{pmatrix} -1.95 & 0.05 \\ 0.95 & -1.05 \end{pmatrix}$, $\mathbf{G}_1 = \begin{pmatrix} 1.9 & 0 \\ 0 & 0.1 \end{pmatrix}$, or by a PH distribution with the same failure time distribution but uncorrelated failures. The mean time to failure is set to 1.0, for the remaining parameters we choose $\mu = 2$, $\delta = 10$, $\lambda = 10$, $\omega = 5$ and $c = 0.1$. Figure 6 shows the distribution function of the time until the first automaton reaches state 2 which is absorbing and describes a system failure. It can be seen that with the introduction of correlations, the time to system failure is drastically reduced. The computation time is in all cases negligible since the state space contains only 75 states.*

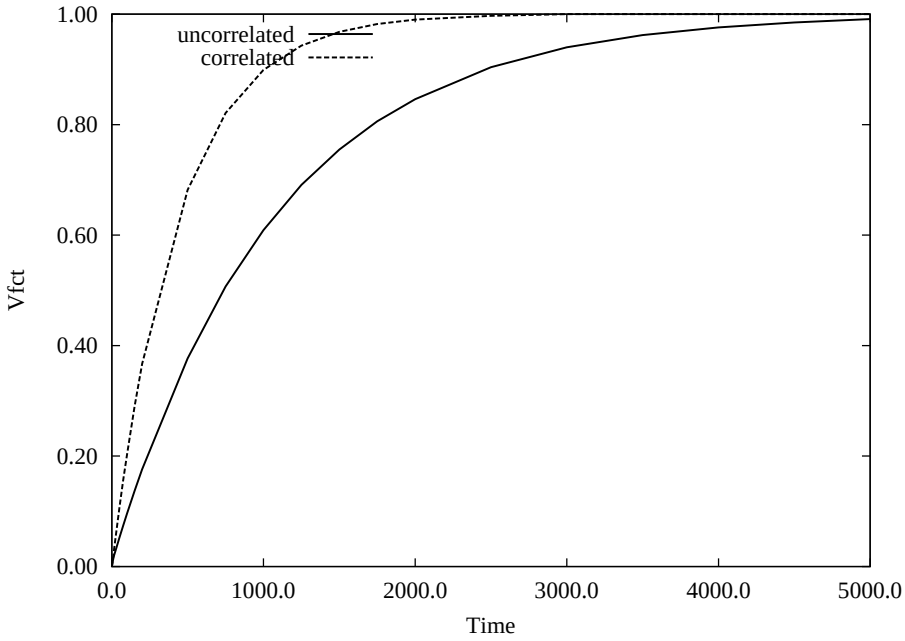

Figure 6: Distribution function of the time to system failure for Example 5

5. Conclusion and Possible Extensions

This paper introduces a new class of stochastic models, namely rational automata networks (RANs) that uses ME distributions and rational arrival processes as extensions of the well established PH distributions and MAPs for defining timing in automata models. It is shown that the descriptor matrix of a RAN can be represented in compact form like the descriptor matrix of a stochastic automata network that describes a Markov chain. Although stochastic processes described by RANs are non-Markovian, they can be analyzed numerically similar to Markov chains and results at the automata level are stochastic distributions that can be used for the computation of result measures via rewards.

There are several aspects that require more investigations. In particular, the parametrization of ME distributions and RAPs is an important task, although some progress has been made in this area recently. Furthermore, specifically tailored numerical solution techniques for the new class of stochastic processes are required that make use of the compact representation of the descriptor matrix. The model class can be extended by location dependent transition rates. It is, of course, also possible to define model checking or equivalences like bisimulations for rational automata. However, this is more a long time goal.

References

Asmussen, S., M. Bladt. 1999. Point processes with finite-dimensional conditional probabilities. *Stochastic Processes and their Application* **82** 127–142.

Bean, N. G., B. F. Nielsen. 2010. Quasi-birth-and-death processes with rational arrival process components. *Stochastic Models* **26** 309–334.

Bladt, M., M. F. Neuts. 2003. Matrix-exponential distributions: Calculus and interpretation via flows. *Stochastic Models* **19** 113–124.

Buchholz, P. 1992. Numerical solution methods based on structured descriptions of Markovian models. G. Balbo, G. Serazzi, eds., *Computer Performance Evaluation - Modelling Techniques and Tools*. Elsevier, 251–267.

Buchholz, P. 1999. Hierarchical structuring of superposed GSPNs. *IEEE Trans. Software Eng.* **25** 166–181.

Buchholz, P., G. Ciardo, S. Donatelli, P. Kemper. 2000. Complexity of Kronecker operations and sparse matrices with applications to the solution of Markov models. *INFORMS Journal on Computing* **12** 203–222.

Buchholz, P., M. Telek. 2010. Stochastic Petri nets with matrix exponentially distributed firing times. *Performance Evaluation* **67** 1373–1385.

Cellier, F. E., E. Kofman. 2010. *Continuous System Simulation*. Springer.

Dayar, T., O. Pentakalos, A. B. Stephens. 1997. Analytical modeling of robotic tape libraries using stochastic automata. *Technical Report TR-97-189, Center of Excellence in Space Data and Information Systems, NASA/Goddard Space Flight Center*.

Donatelli, S. 1994. Superposed stochastic automata: a class of stochastic Petri nets amenable to parallel solution. *Performance Evaluation* **18** 21–36.

El-Rayes, A., M. Kwiatkowska, G. Norman. 1999. Solving infinite stochastic process algebra models through matrix-geometric methods. J. Hillston, M. Silva, eds., *Proc. 7th Process Algebras and Performance Modelling Workshop (PAPM'99)*. 41–62.

Éltető, T., S. Rácz, M. Telek. 2006. Minimal coefficient of variation of matrix exponential distributions. *2nd Madrid Conference on Queueing Theory*. Madrid, Spain. Abstract.

Fackrell, M. 2005. Fitting with matrix-exponential distributions. *Stochastic Models* **21** 377–400.

Fackrell, M. W. 2003. Characterization of matrix-exponential distributions. Ph.D. thesis, School of Applied Mathematics, The University of Adelaide.

He, Qi-Ming, Marcel Neuts. 1998. Markov arrival processes with marked transitions. *Stochastic Processes and their Applications* **74** 37–52.

Hermanns, H. 2002. *Interactive Markov Chains: The Quest for Quantified Quality*, *Lecture Notes in Computer Science*, vol. 2428. Springer.

Hillston, J. 1995. Compositional Markovian modelling using a process algebra. W. J. Stewart, ed., *Computations with Markov Chains*. Kluwer Academic Publishers, 177–196.

Horváth, A., S. Rácz, M. Telek. 2009. Moments characterization of the class of order 3 matrix exponential distributions. *16th Int. Conf. on Analytical and Stochastic Modeling Techniques and Applications (ASMTA)*, *LNCS*, vol. 5513. Springer, Madrid, Spain, 174 – 188.

Horváth, Gábor, Miklós Telek. 2009. On the canonical representation of phase type distributions. *Performance Evaluation* **66** 396 – 409.

Lipsky, L. 2008. *Queueing Theory: A Linear Algebraic Approach*. Springer.

Neuts, M. F. 1979. A versatile Markovian point process. *Journal of Applied Probability* **16** 764–779.

Neuts, M. F. 1981. *Matrix-geometric solutions in stochastic models*. Johns Hopkins University Press.

Plateau, B. 1985. On the stochastic structure of parallelism and synchronisation models for distributed algorithms. *ACM Performance Evaluation Review* **13** 142–154.

Plateau, B., J. M. Fourneau. 1991. A methodology for solving Markov models of parallel systems. *J. Parallel Distrib. Comput.* **12** 370–387.

Reibman, Andrew L., Kishor S. Trivedi. 1988. Numerical transient analysis of markov models. *Computers & OR* **15** 19–36.

Saad, Y., M. Schultz. 1986. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific Computing* **7** 856–869.

Sbeity, I., L. Brenner, B. Plateau, W. J. Stewart. 2008. Phase-type distributions in stochastic automata networks. *European Journal of Operational Research* **186** 1008–1028.

Stewart, W. J. 1994. *Introduction to the numerical solution of Markov chains*. Princeton University Press.

van der Vorst, H. A. 1992. Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM Journal on Scientific Computing* **13** 631–644.

Appendix

Proof of Theorem 1

Since matrix $\mathbf{G} = \mathbf{G}_0 + \sum_{k=1}^{K} \mathbf{G}_k$ we have $\mathbf{G}\mathbb{1} = \mathbf{0}$ such that

$$\nu_t \mathbb{1} = \nu_0 e^{\mathbf{G}t}\mathbb{1} = \nu_0 \sum_{i=0}^{\infty} \frac{(\mathbf{G}t)^i}{i!}\mathbb{1} = \nu_0 \mathbb{1} = 1.$$

Proof of Theorem 2

We have to prove that $f_{(\mathbf{H}_0,\ldots,\mathbf{H}_K)}(t_1, k_1, \ldots, t_j, k_j)$ is a valid joint density.

First we introduce the joint density of absolute times $\hat{f}_{(\mathbf{H}_0,\ldots,\mathbf{H}_K)}(\tau_1, k_1, \ldots, \tau_j, k_j) = f_{(\mathbf{H}_0,\ldots,\mathbf{H}_K)}(t_1, k_1, \ldots, t_j, k_j)$, where $\tau_i = \sum_{\ell=1}^{i} t_\ell$. Define for the sequences $T_j = (\tau_1, k_1, \ldots, \tau_j, k_j)$ with $\tau_{i-1} < \tau_i$ and $k_i \in \mathcal{K} = \{1, \ldots, K\}$ and $S_h = (\omega_1, m_1, \ldots, \omega_h, m_h)$ with $\omega_{i-1} < \omega_i$, $\omega_h < \tau_j$ and $m_i \in \mathcal{K}$ an ordered set $Expand(T_j, S_h)$ containing the elements of T_j and S_h, and the elements are ordered according to their time information (τ_i and ω_i). Then we obtain

$$\hat{f}_{(\mathbf{H}_0,\ldots,\mathbf{H}_K)}(T_j) = \sum_{h=1}^{\infty} \int_{\underline{\omega}} \sum_{\underline{m} \in \mathcal{K}^h} Prob(T_j, S_h) \hat{f}_{(\mathbf{G}_0,\ldots,\mathbf{G}_K)}(Expand(T_j, S_h)) d\underline{\omega}$$

where T_j and S_h stand for the accepted and discarded arrivals, respectively,

$$Prob(T_j, S_h) = \prod_{i=1}^{j} p_{k_i} \prod_{i=1}^{h} (1 - p_{m_i}),$$

the second summation is according to all combination of types, $\int_{\underline{\omega}} = \int_{\omega_1=0}^{\tau_j} \int_{\omega_2=\omega_1}^{\tau_j} \dots \int_{\omega_h=\omega_{h-1}}^{\tau_j}$ and $d\underline{\omega} = d\omega_h \dots d\omega_1$. This implies $\hat{f}_{(\mathbf{H}_0,\dots,\mathbf{H}_K)}(T_j) \geq 0$ and consequently $f_{(\mathbf{H}_0,\dots,\mathbf{H}_K)}(t_1, k_1, \dots, t_j, k_j) \geq 0$.

Further more, from $\sum_{k=0}^{K} \mathbf{G}_k \mathbb{1} = \mathbf{0}$ we have $\sum_{k=0}^{K} \mathbf{H}_k \mathbb{1} = \mathbf{0}$, which ensures the unit integral of the joint density.

Proof of Theorem 3

We have to show that $(\eta, \mathbf{F}_0, \dots, \mathbf{F}_M)$ describes a valid density. The unit sum of the joint density $f_{\mathbf{F}_0,\dots,\mathbf{F}_M}(t_1, k_1, \dots, t_j, k_j)$ is provided by $\sum_{k=0}^{M} \mathbf{F}_k \mathbb{1} = \mathbf{0}$, since

$$\begin{aligned}
&\int_{t_1} \dots \int_{t_j} \sum_{k_1} \dots \sum_{k_j} f_{(\mathbf{F}_0,\dots,\mathbf{F}_M)}(t_1, k_1, \dots, t_j, k_j) dt_j \dots dt_1 = \\
&\eta \int_{t_1} e^{\mathbf{F}_0 t_1} dt_1 \left(\sum_{k_1} \mathbf{F}_{k_1} \right) \dots \int_{t_j} e^{\mathbf{F}_0 t_j} dt_j \left(\sum_{k_j} \mathbf{F}_{k_j} \right) \mathbb{1} = \\
&\eta (-\mathbf{F}_0)^{-1} \left(\sum_{k_1} \mathbf{F}_{k_1} \right) \dots (-\mathbf{F}_0)^{-1} \left(\sum_{k_j} \mathbf{F}_{k_j} \right) \mathbb{1} = 1,
\end{aligned}$$

where $\mathbb{1} = (-\mathbf{F}_0)^{-1} \sum_{k=1}^{M} \mathbf{F}_k \mathbb{1}$ is used in the last step. The non-negativity of the density function remained to prove.

In case 1, we have two MMAPs and show that the resulting process is also a MMAP which automatically shows that the density is non-negative. For MMAPs $\mathbf{G'}_0$ and $\mathbf{H'}_0$ have non-negative non-diagonal elements and negative diagonal elements. This implies that the same holds for

$$\mathbf{G'}_0 \oplus \mathbf{H'}_0 = (\mathbf{G'}_0 \otimes \mathbf{I}) + (\mathbf{I} \otimes \mathbf{H'}_0) \ .$$

Furthermore, since $\mathbf{G}_k, \mathbf{H}_k \geq \mathbf{0}$, the same holds for $\mathbf{G}_k \oplus \mathbf{H}_k$ and $\mathbf{G}_k \otimes \mathbf{H}_k$. $\hat{\mathbf{G}}_k \oplus \hat{\mathbf{H}}_k$ and $\hat{\mathbf{G}}_k \otimes \hat{\mathbf{H}}_k$ are non negative diagonal matrices. This implies that $\mathbf{F}_0$ has non-negative non-diagonal elements and $\mathbf{F}_k$ is non-negative which together assures that $(\mathbf{F}_0, \dots, \mathbf{F}_K)$ is a MMAP.

In case 2, $\mathcal{C} = \emptyset$, i.e., the two MRAPs run in parallel without any synchronization. Observe that in this case

$$\mathbf{F}_0 = \mathbf{G'}_0 \oplus \mathbf{H'}_0 - \sum_{k=1}^{K} \hat{\mathbf{G}}_k \oplus \hat{\mathbf{H}}_k = (\mathbf{G'}_0 - \sum_{k=1}^{K} \hat{\mathbf{G}}_k) \oplus (\mathbf{H'}_0 - \sum_{k=1}^{K} \hat{\mathbf{H}}_k) = \mathbf{H}_0 \oplus \mathbf{G}_0$$

since $\hat{\mathbf{G}}_k$ and $\hat{\mathbf{H}}_k$ are diagonal matrices. Additionally, we have

$$\begin{aligned} e^{\mathbf{G}\oplus\mathbf{H}t} &= \sum_{i=0}^{\infty} \tfrac{((\mathbf{G}\oplus\mathbf{H})t)^i}{i!} &&= \sum_{i=0}^{\infty} \tfrac{((\mathbf{G}\otimes\mathbf{I}+\mathbf{I}\otimes\mathbf{H})t)^i}{i!} &&= \\ \sum_{i=0}^{\infty} \tfrac{1}{i!} \sum_{j=0}^{i} \binom{i}{j} (\mathbf{G}t)^j \otimes (\mathbf{H}t)^{i-j} &= \sum_{i=0}^{\infty} \tfrac{(\mathbf{G}t)^i}{i!} \otimes \sum_{j=0}^{\infty} \tfrac{(\mathbf{H}t)^j}{j!} &&= e^{\mathbf{G}t} \otimes e^{\mathbf{H}t} \end{aligned}$$

such that both processes run independently. Consequently, $(\eta, \mathbf{F}_0, \ldots, \mathbf{F}_K)$ is composed as the superposition of two independent MRAPs and it is straightforward to show that the superposition of two independent stochastic processes is again a stochastic process.

For case 3, without loss of generality we assume that $\mathcal{C} = \{1, \ldots, \kappa\}$ $(0 \leq \kappa \leq K)$, where $\kappa = 0$ means $\mathcal{C} = \emptyset$ and $\mathcal{F} = \{1, \ldots, \iota\}$ $(0 \leq \iota \leq \kappa)$ such that $\mathbf{H}_k \mathbb{1} = d_k \mathbb{1}$ which implies $\hat{\mathbf{H}}_k = d_k \mathbf{I}$ for $k \in \{1, \ldots, \iota\}$ and $\mathbf{G}_k \mathbb{1} = c_k \mathbb{1}$ which implies $\hat{\mathbf{G}}_k = c_k \mathbf{I}$ for $k \in \{\iota + 1, \ldots, \kappa\}$. According to this numbering of classes we have $\tilde{\mathbf{G}}_0 = \mathbf{G'}_0 - \sum_{k=1}^{\iota} d_k \hat{\mathbf{G}}_k - \sum_{k=\iota+1}^{K} \hat{\mathbf{G}}_k$ and $\tilde{\mathbf{H}}_0 = \mathbf{H'}_0 - \sum_{k=1}^{\iota} \hat{\mathbf{H}}_k - \sum_{k=\iota+1}^{\kappa} c_k \hat{\mathbf{H}}_k - \sum_{k=\kappa+1}^{K} \hat{\mathbf{H}}_k$ and by assumption $(\tilde{\mathbf{G}}_0, d_1\mathbf{G}_1, \ldots, d_\iota \mathbf{G}_\iota, \mathbf{G}_{\iota+1}, \ldots, \mathbf{G}_K)$ and
$(\tilde{\mathbf{H}}_0, \mathbf{H}_1, \ldots, \mathbf{H}_\iota, c_{\iota+1}\mathbf{H}_{\iota+1}, \ldots, c_\kappa \mathbf{H}_\kappa, \mathbf{H}_{\kappa+1}, \ldots, \mathbf{H}_K)$ are MRAPs.

The composed process is

$$\begin{aligned} \mathbf{F}_0 &= \mathbf{G'}_0 \oplus \mathbf{H'}_0 - \sum_{k=1}^{\kappa} \hat{\mathbf{G}}_k \otimes \hat{\mathbf{H}}_k - \sum_{k=\kappa+1}^{K} \hat{\mathbf{G}}_k \oplus \hat{\mathbf{H}}_k = \\ &= \mathbf{G'}_0 \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{H'}_0 - \sum_{k=1}^{\iota} \hat{\mathbf{G}}_k \otimes d_k \mathbf{I} - \sum_{k=\iota+1}^{\kappa} c_k \mathbf{I} \otimes \hat{\mathbf{H}}_k - \sum_{k=\kappa+1}^{K} \hat{\mathbf{G}}_k \otimes \mathbf{I} - \sum_{k=\kappa+1}^{K} \mathbf{I} \otimes \hat{\mathbf{H}}_k = \\ &= (\mathbf{G'}_0 - \sum_{k=1}^{\iota} d_k \hat{\mathbf{G}}_k - \sum_{k=\kappa+1}^{K} \hat{\mathbf{G}}_k) \otimes \mathbf{I} + \mathbf{I} \otimes (\mathbf{H'}_0 - \sum_{k=\iota+1}^{\kappa} c_k \hat{\mathbf{H}}_k - \sum_{k=\kappa+1}^{K} \hat{\mathbf{H}}_k) = \\ &= (\mathbf{G'}_0 - \sum_{k=1}^{\iota} d_k \hat{\mathbf{G}}_k - \sum_{k=\kappa+1}^{K} \hat{\mathbf{G}}_k) \oplus (\mathbf{H'}_0 - \sum_{k=\iota+1}^{\kappa} c_k \hat{\mathbf{H}}_k - \sum_{k=\kappa+1}^{K} \hat{\mathbf{H}}_k) = \\ &= (\tilde{\mathbf{G}}_0 + \sum_{k=\iota+1}^{\kappa} \hat{\mathbf{G}}_k) \oplus (\tilde{\mathbf{H}}_0 + \sum_{k=1}^{\iota} \hat{\mathbf{H}}_k) = (\tilde{\mathbf{G}}_0 + (\kappa - \iota)\mathbf{I}) \oplus (\tilde{\mathbf{H}}_0 + \iota \mathbf{I}). \end{aligned}$$

It means that $e^{\mathbf{F}_0 t} = e^{(\tilde{\mathbf{G}}_0 + (\kappa-\iota)\mathbf{I})t} \otimes e^{(\tilde{\mathbf{H}}_0 + \iota\mathbf{I})t}$ and $e^{(\tilde{\mathbf{G}}_0 + (\kappa-\iota)\mathbf{I})t} = e^{(\kappa-\iota)t} e^{\tilde{\mathbf{G}}_0 t}$ as well as $e^{(\tilde{\mathbf{H}}_0 + \iota\mathbf{I})t} = e^{\iota t} e^{\tilde{\mathbf{H}}_0 t}$. This way

$$\begin{aligned} &f_{\mathbf{F}_0,\ldots,\mathbf{F}_K}(t_1, k_1, \ldots, t_j, k_j) = \eta \left(\prod_{i=1}^{j} e^{\mathbf{F}_0 t_i} \mathbf{F}_{k_i} \right) \mathbb{1} = \\ &(\pi \otimes \phi) \left(\prod_{i=1}^{j} \left(e^{(\tilde{\mathbf{G}}_0 + (\kappa-\iota)\mathbf{I})t_i} \otimes e^{(\tilde{\mathbf{H}}_0 + \iota\mathbf{I})t_i} \right) (\mathbf{G}_{k_i} \otimes \mathbf{H}_{k_i}) \right) \mathbb{1} = \\ &\left(\pi \left(\prod_{i=1}^{j} e^{(\kappa-\iota)t_i} e^{\tilde{\mathbf{G}}_0 t_i} \mathbf{G}_{k_i} \right) \mathbb{1} \right) \otimes \left(\phi \left(\prod_{i=1}^{j} e^{\iota t_i} e^{\tilde{\mathbf{H}}_0 t_i} \mathbf{H}_{k_i} \right) \mathbb{1} \right) = \\ &f_{\tilde{\mathbf{G}}_0, d_1\mathbf{G}_1, \ldots, d_\iota \mathbf{G}_\iota, \mathbf{G}_{\iota+1}, \ldots, \mathbf{G}_K}(t_1, k_1, \ldots, t_j, k_j) \\ &\qquad f_{\tilde{\mathbf{H}}_0, \mathbf{H}_1, \ldots, \mathbf{H}_\iota, c_{\iota+1}\mathbf{H}_{\iota+1}, \ldots, c_\kappa \mathbf{H}_\kappa, \mathbf{H}_{\kappa+1}, \ldots, \mathbf{H}_K}(t_1, k_1, \ldots, t_j, k_j) e^{\kappa t^*} \geq 0 \end{aligned}$$

where $t^* = \sum_{i=1}^{j} t_i$.

Proof of Theorem 4

The unit integral of the joint density follows from $\mathbf{G}_0\mathbb{1} = \sum_{k=1}^{K} \mathbf{G}_k\mathbb{1}$ which also holds after adding $\lambda\mathbb{1}$ on both sides. Thus it remains that show that the joint density is non negative.

Case $\mathbf{G}_{K+1} = \lambda\mathbf{I}$:

MRAP $(\phi, \mathbf{G}_0 - \lambda\mathbf{I}, \ldots, \mathbf{G}_K, \lambda\mathbf{I})$ arises by the independent composition, $\mathcal{C} = \emptyset$, of two MRAPs $(\phi, \mathbf{H}_0, \ldots, \mathbf{H}_K, \mathbf{0})$ and $(-\lambda, 0, \ldots, 0, \lambda)$, where $\mathbf{H}_k = \mathbf{G}_k$ for $k = 1, \ldots, K$. According to case 2 of Theorem 3 the composed process is a MRAP too.

Let us extract a consequence of this fact. Let $k_1 = \ldots = k_{j-1} \leq K$ and $k_j = K+1$, then

$$\begin{aligned} &f_{\phi,\mathbf{G}_0-\lambda\mathbf{I},\ldots,\mathbf{G}_K,\mathbf{G}_{K+1}}(t_1, k_1, \ldots, t_j, k_j) = \\ &\phi\left(\prod_{i=1}^{j} e^{(\mathbf{G}_0-\lambda\mathbf{I})t_i}\mathbf{G}_{k_i}\right)\mathbb{1} = e^{-\lambda t^*}\phi\left(\prod_{i=1}^{j} e^{\mathbf{G}_0 t_i}\mathbf{G}_{k_i}\right)\mathbb{1} = \\ &\lambda e^{-\lambda t^*}\phi\left(\prod_{i=1}^{j-1} e^{\mathbf{G}_0 t_i}\mathbf{G}_{k_i}\right)e^{\mathbf{G}_0 t_j}\mathbb{1} \geq 0\ . \end{aligned} \tag{10}$$

Case $\mathbf{G}_{K+1} = \lambda\mathbb{1}\phi$:

Let $z \leq j$ and $t_{a_1}, \ldots, t_{a_z}$ the instances when the process generates type $K+1$ signals, i.e., $k_{a_1} = \ldots = k_{a_z} = K+1$. Then

$$\begin{aligned} &f_{\phi,\mathbf{G}_0-\lambda\mathbf{I},\ldots,\mathbf{G}_K,\mathbf{G}_{K+1}}(t_1, k_1, \ldots, t_j, k_j) = \\ &\phi\left(\prod_{i=1}^{j} e^{(\mathbf{G}_0-\lambda\mathbf{I})t_i}\mathbf{G}_{k_i}\right)\mathbb{1} = e^{-\lambda t^*}\phi\left(\prod_{i=1}^{j} e^{\mathbf{G}_0 t_i}\mathbf{G}_{k_i}\right)\mathbb{1} = \\ &\lambda^z e^{-\lambda t^*}\prod_{\ell=1}^{z}\phi\left(\prod_{i=a_{\ell-1}}^{a_\ell-1} e^{\mathbf{G}_0 t_i}\mathbf{G}_{k_i}\right)e^{\mathbf{G}_0 t_{a_\ell}}\mathbb{1} \geq 0\ , \end{aligned}$$

where $a_0 = 0$, $t^* = \sum_{i=1}^{j} t_i$ and (10) is used in the last step.

Proof of Theorem 5

Let $\mathcal{D} = \{(\pi^{(d)}, \mathbf{G}_0^{(d)}, \ldots, \mathbf{G}_K^{(d)})\}$ be the set of MRAPs that are associated with the RA. In each location $x \in \mathcal{L}$, the set $Act(x)$ is the set of active MRAPs in x which means that each class from $Class(Act(x))$ is assigned to exactly one event $e \in Ena(x)$. This implies that all MRAPs from $Act(x)$ are composed without synchronization. We define $\mathbf{H}_0 = \oplus_{d\in Act(x)}\mathbf{G}_0^{(d)}$ and $\mathbf{H}_{e,(d,k)} = \left(\otimes_{c\in Act(x),c<d}\mathbf{I}_{n^{(c)}}\right) \otimes \mathbf{G}_k^{(d)} \otimes \left(\otimes_{c\in Act(x),c>d}\mathbf{I}_{n^{(c)}}\right)$ for $(d,k) \in Assign(x,e)$. For a proof we consider a sample path that visits different locations and show that the sojourn time in each location is a valid density and the successor location is chosen probabilistically.

Assume that location x has been entered and the state vector equals $\phi = \otimes_{d\in Act(x)}\phi^{(d)}$ with $\phi^{(d)} \in \mathcal{V}_{(\pi^{(d)},\mathbf{G}_0^{(d)},\ldots,\mathbf{G}_K^{(d)})}$. The density of observing $e \in Ena(x)$ initiated by an event $(d,k) \in Assign(x,e)$ is then given by

$$f_{(\phi,\mathbf{H}_0,\mathbf{H}_f(f\in Ena(x))}(t,e,(d,k)) = \phi e^{\mathbf{H}_0 t}\mathbf{H}_{e,(d,k)}\mathbb{1} =$$
$$\left(\bigotimes_{d\in Act(x)} \phi^{(d)}\right) e^{\oplus_{d\in Act(x)}\mathbf{G}_0^{(d)}t} \left(\bigotimes_{c\in Act(x),c<d} \mathbf{I}_{n^{(c)}}\right) \otimes \mathbf{G}_k^{(d)} \otimes \left(\bigotimes_{c\in Act(x),c>d} \mathbf{I}_{n^{(c)}}\right)\mathbb{1} =$$
$$\left(\bigotimes_{c\in Act(x),c<d} \phi^{(c)}e^{\mathbf{G}_0^{(c)}t}\right) \otimes \left(\phi^{(d)}e^{\mathbf{G}_0^{(d)}t}\mathbf{G}_k^{(d)}\right) \otimes \left(\bigotimes_{c\in Act(x),c>d} \phi^{(c)}e^{\mathbf{G}_0^{(c)}t}\right)\mathbb{1} \geq 0$$

Furthermore,

$$f_{(\phi,\mathbf{H}_0,\mathbf{H}_f(f\in Ena(x))}(t,e) = \sum_{(d,k)\in Assign(x,e)} f_{(\phi,\mathbf{H}_0,\mathbf{H}_f(f\in Ena(x))}(t,e,(d,k)) \geq 0$$

is the density for event e. This implies that we obtain a valid density if the initial vector is given by a Kronecker product of vectors from $\mathcal{V}_{(\pi^{(d)},\mathbf{G}_0^{(d)},\ldots,\mathbf{G}_K^{(d)})}$. This obviously holds for the initial location since the initial vector is defined as $\otimes_{d\in Act(x_{ini})}\pi^{(d)}$.

We now show that the state vector upon entering location y from x via $(c,k) \in Assign(x,e)$ can be represented as $\otimes_{d\in Act(y)}\eta^{(d)}$ with $\eta^{(d)} \in \mathcal{V}_{(\pi^{(d)},\mathbf{G}_0^{(d)},\ldots,\mathbf{G}_K^{(d)})}$ if location x has been entered with a state vector $\otimes_{d\in Act(x)}\phi^{(d)}$ with $\phi^{(d)} \in \mathcal{V}_{(\pi^{(d)},\mathbf{G}_0^{(d)},\ldots,\mathbf{G}_K^{(d)})}$. We have to distinguish the different possibilities how the matrices of RA are built in (2) and consider the resulting vector for $d \in Act(y)$. If d in $Act(y) \setminus Act(x)$ or in $Act(y) \cap Reset(x,e)$, then $\eta^{(d)} = \pi^{(d)} \in \mathcal{V}_{(\pi^{(d)},\mathbf{G}_0^{(d)},\ldots,\mathbf{G}_K^{(d)})}$. If $d = c$, then

$$\eta^{(c)} = \phi^{(c)}e^{\mathbf{G}_0^{(c)}t}\mathbf{G}_k^{(c)} \in \mathcal{V}_{(\pi^{(c)},\mathbf{G}_0^{(c)},\ldots,\mathbf{G}_K^{(c)})}$$

and for all other $d \in Act(y) \cap Act(x)$, we have $\eta^{(c)} = \phi^{(c)}e^{\mathbf{G}_0^{(c)}t} \in \mathcal{V}_{(\pi^{(c)},\mathbf{G}_0^{(c)},\ldots,\mathbf{G}_K^{(c)})}$ according to (10), where t is the sojourn time in location x.

Consequently, it follows by induction that a new location x is always entered with a vector that is given by the Kronecker product of vectors $\phi^{(d)} \in \mathcal{V}_{(\pi^{(c)},\mathbf{G}_0^{(c)},\ldots,\mathbf{G}_K^{(c)})}$ for all $d \in Act(x)$ and the sojourn time in the states of a location is ME distributed. After leaving a location, the successor location is chosen probabilistically according to the corresponding row in matrix $\mathbb{P}_e$ if event e occurs. This defines a stochastic process with state space $\mathcal{L}$ where the distribution at each time is well defined. Thus, the first part of the theorem is proved.

The above result also shows that the density of each sample path $x_1 = x_{ini},(d_1,k_1),t_1,x_2,(d_2,k_2),t_2,\ldots$ where $x_i \in \mathcal{L}$, $(d_i,k_i) \in Assign(x_i,e)$ for $e \in$

$Ena(x_i)$ and $t_i \geq 0$ is non-negative. This implies that also the density of a sample path $x_1 = x_{ini}, e_1, t_1, x_2, e_2, t_2, \ldots$ for $e_i \in Ena(x_i)$ is non-negative and therefore $f_{(\pi,\mathbf{Q}_\varepsilon,\mathbf{Q}_e(e\in\mathcal{E}_s))}(t_1, e_1, \ldots, t_j, e_j) \geq 0$. The unit integral of the density follows from $\mathbf{Q}_\varepsilon\mathbb{1} - \sum_{e\in\mathcal{E}_s}\mathbf{Q}_e\mathbb{1} = \mathbf{0}$ which can be easily proved from the representation of the matrices defined in (2). To have a valid MRAP we only have to show that the vector after an arbitrary event from $\mathcal{E}_s$ is unique. Since the solution $\eta\mathbf{Q} = \mathbf{0}\ \eta\mathbb{1} = 1$ is unique, the vector ϕ after an arbitrary event can then be computed as

$$\phi = \sum_{e\in\mathcal{E}_s} \frac{\eta\mathbf{Q}_e}{\eta\sum_{e\in\mathcal{E}_s}\mathbf{Q}_e\mathbb{1}}.$$

Proof of Theorem 6

The computation of the matrices in (5) equals the computation of the composed MRAP matrices in Theorem 3 if $\mathcal{C} = \mathcal{E}$. Since each RA is a MRAP and the rules for composing MRAPs are observed the theorem follows if π can be computed uniquely, which follows from the irreducibility of the descriptor of a RAN.