Cumulative measures of absorbing joint Markov chains and an application to Markovian process algebras

Brenner, Freimut

In: ICB Research Reports - Forschungsberichte des ICB / 2007

Source: ICB-Research Report No. 12, February 2007

ICB
Institut für Informatik und Wirtschaftsinformatik

Freimut Brenner

Cumulative Measures of Absorbing Joint Markov Chains and an Application to Markovian Process Algebras

ICB-RESEARCH REPORT

ICB-Research Report No.12
February 2007
Universität Duisburg-Essen

Die Forschungsberichte des Instituts für Informatik und Wirtschaftsinformatik dienen der Darstellung vorläufiger Ergebnisse, die i. d. R. noch für spätere Veröffentlichungen überarbeitet werden. Die Autoren sind deshalb für kritische Hinweise dankbar.

The ICB Research Reports comprise preliminary results which will usually be revised for subsequent publications. Critical comments would be appreciated by the authors.

Authors' Address:

Freimut Brenner

Institut für Informatik und
Wirtschaftsinformatik (ICB)
Universität Duisburg-Essen
D-45117 Essen

Freimut.Brenner@icb.uni-due.de

Managing Assistant and Contact:

Jürgen Jung

Institut für Informatik und
Wirtschaftsinformatik (ICB)
Universität Duisburg-Essen
Universitätsstr. 9
45141 Essen
Germany

Email: icb@uni-duisburg-essen.de

ISSN 1860-2770

Preleminary Remark

The work presented in this note is planned to evolve into the theoretical core of the author's doctoral thesis and should be regarded as work in progress. Objections and suggestions are welcome.

Abstract

Markov Models are of outstanding importance in the performance and reliability evaluation of computer systems and communication networks. In this paper we aim at contributing to the field of Markovian Process Algebras (MPAs). An MPA model is (or may be) the composition of several concurrent sub-components (each of which describes an underlying Markov chain) which may interact with each other through synchronisation. On the one hand the existence of sub-components implies the possibility of the state space explosion problem, i.e. the size of the state space of the Markov chain underlying the composite component grows exponentially in the number of sub-components. On the other hand the interaction of sub-components in general negates the property of independence of their underlying Markov chains, and hence forbids a product-form solution for steady state probabilities.

Our target quantities are single steady state probabilities of the Markov chain underlying the composite component. We consider composite components which possess only global synchronisations, i.e. every sub-component is involved in every synchronisation. For this class of MPA models the behaviour of the composite component between two successive synchronisations can be described by the joint process of several absorbing Markov chains.

First, a new result on cumulative measures of absorbing joint Markov chains is presented. We compute the mean time to absorption and the mean time the joint Markov chain spends in a certain set before absorption. Our computations do not operate on the state space of the joint Markov chain, and hence the problem of state space explosion is avoided. The computational effort of our method rather depends on convergence properties of the joint Markov chain.

Afterwards, this result is applied to compute steady state probabilities for a class of composite components specified as PEPA models which are popular ambassadors of MPAs. It is easily understood that these results carry over from PEPA to other MPA variants.

Contents

1 Introduction

The analysis of stochastic systems is a branch of applied research which spreads across many disciplines of science. Areas of application include performance and reliability of computing systems and telecommunication networks, and also fields like reaction kinetics in physical chemistry and financial risk theory, to name but a few.

Often a model of such a stochastic system is forced to fit into a Markovian framework, i.e. a Markov chain can be extracted from that model. As a consequence the model can be analysed by means provided by the rich and often times elegant theory of Markov chains. In addition, formalisms to build (or describe) Markovian models exist. These formalisms constitute the advantage that they (may or may not) equip the model, or certain activities or states of that model, with an intuitive meaning. Queueing stations (and networks), stochastic Petri nets and Markovian process algebras are outstanding examples for formalisms which have shown their usefulness in the areas of performance and reliability evaluation of computer and communication networks. For an extended overview and application examples see e.g. [BGMT98], [CaTu02], [HLR00] and [BHK00], or the latest proceedings of the conferences MMB ([GeHe06]), QEST ([AMR06]) and Performance ([MKS06]).

When dealing with Markov chain formalisms, the first two questions should be: a) Is that formalism useful from the modellers point of view? and b) Can the special structure which the formalism induces on the underlying Markov chain be exploited to derive efficient (or elegant) solutions of that Markov chain? For instance, the solution of a birth-death process is given by a simple symbolic expression; BCMP networks possess a product-form solution ([BCMP75]); nearly completely decomposable Markov chains can be partitioned into suited subchains, where transitions inside of subchains and transitions between subchains can be treated separately ([Cour77]).

In this paper we aim at contributing to the field of Markovian Process Algebras (MPAs). An MPA model consists of several concurrent components which may interact with each other through synchronisation. On the one hand the existence of concurrent components imply the possibility of the state space explosion problem, and on the other hand the interaction of components in general negates the property of independence of components, and hence forbids a product-form solution. We will make use of the fact that for a certain class of MPA models the concurrent components behave independently of each other between successive points of synchronisation.

The paper is organised as follows: In chapter 2 of this paper we present a new result on absorbing continuous time Markov chains which are built of several marginal absorbing Markov chains. This result is applied to compute steady state probabilities for a class of PEPA components in chapter 3 (PEPA is a popular variant of an MPA).

ad chapter 2: For a Markov chain $U = (U_1, \ldots, U_m)$, with the marginal absorbing Markov chains U_i, $i = 1 \ldots m$, we propose a new method to compute:

(a) The mean time to absorption.

(b) The mean time which the Markov chain U spends in some set A before absorption. The set A is restricted to possess the form $A = \times_{i=1}^{m} A_i$, where the A_i are subsets of the marginal state spaces.

Unlike classical approaches to compute these two quantities our method does not carry out operations on the global state space, i.e. on the state space of U. Note, that the global state space grows exponentially in the number m of involved marginal processes.

Without diving into detail at this point, we explain the basic idea of our method. Let $\mathcal{A} = \times_{i=1}^{m} \mathcal{A}_i$ be a subset of the state space of U, where the $\mathcal{A}_i$ are subsets of the marginal state spaces. The key will be to reformulate the probabilities $\nu(n)[\mathcal{A}]$, $n \geq 0$, that the discrete time Markov chain (DTMC) embedded in U – more precisely this DTMC is obtained from U by uniformisation – is in some set $\mathcal{A}$ in the $n-$th step. At first the marginal CTMCs are uniformised, which yields embedded discrete time Markov chains. These DTMCs are then solved in isolation. In this context the solution is considered the discrete function $\nu_i(\cdot)[\mathcal{A}_i]$, where for fixed i and $\mathcal{A}_i$ the quantity $\nu_i(n)[\mathcal{A}_i]$ is the probability that the DTMC embedded in U_i is in set $\mathcal{A}_i$ in the $n-$th step. By a convolution-like operator $\star$ these solutions are combined to the solution of the DTMC embedded in U. That means we obtain a function $\nu(\cdot)[\mathcal{A}]$, where on the one hand $\nu(n)[\mathcal{A}] = (\nu_1[\mathcal{A}_i] \star \cdots \star \nu_m[\mathcal{A}_m])(n)$ and on the other hand $\nu(n)[\mathcal{A}]$ is the probability that the DTMC embedded in the joint Markov chain U is in set $\mathcal{A}$ in the $n-$th step. Provided that the expected values in (a) and (b) exist, they can be expressed by infinite and converging series involving the functions $\nu[S]$ (i.e. take $\mathcal{A} := S$, set of absorbing states of U) and $\nu[A]$ (i.e. take $\mathcal{A} := A$).

The method we propose is an exact method, where, of course, the computation of the series just mentioned requires truncation at a certain index. The time complexity is $O(Nmd^2 + mN^2)$, where m is the number of marginal CTMCs, d is the maximal size of the marginal state spaces and N is the truncation index of the infinite series which depends on convergence properties (eigenvalues) of the joint CTMC and the desired accuracy.

ad chapter 3: Stochastic process algebras have become popular since the formalism was proposed by Herzog in [Herz90]. In particular, Markovian Process Algebras (MPAs) have drawn much attention due to the the fact that the quantitative solution of an MPA happens to be the solution of the underlying Markov chain. Examples for MPAs involve PEPA ([Hill96]), EMPA ([BDG94], [BeGo96]), MTIPP ([HeRe02]) and IMC ([Herm02]).

On the one hand MPAs allow to define components in isolation whose behaviour is determined by underlying Markov chains. On the other hand these components can be combined to a composite component by some cooperation operator. This operation forces components to interact or cooperate with each other via synchronisation. Basically, all components act independently of each other until they reach some point, where they are forced to synchronise with other components. Components that are ready to synchronise must wait for the other involved components to become ready on their part. When all processes have reached a point where they are ready to synchronise the synchronisation is executed and afterwards the components again evolve independently of each other. Depending on the specific MPA the synchronisation itself can possess an exponentially distributed duration or it can be timeless. In either case all synchronising components begin to synchronise at a common time instant and they end the synchronisation at a common time instant (here, we interpreted a timeless synchronisation to possess the duration 0).

Obviously, waiting times (until synchronisation can take place) of components depend on the behaviour of other components, and hence synchronising components are not independent

of each other. In general, this permits product-form solutions for the Markov chain underlying the composite component. Since the state space of this Markov chain grows exponentially with the number of sub-components, the computation of quantitative measures is subject to the state space explosion problem. An overview of different methods which tackle this challenge can be found in [Hill99].

In his dissertation [Bohn02] Bohnenkamp took on a view on MPAs which had not been investigated up to that point. He considered a special class of MPA components where there exist only global synchronisations, i.e. every sub-component must participate in every synchronisation. Then, points of synchronisation define an embedded DTMC of the composite component. Together with the first passage times from embedded states to their embedded successor states the embedded DTMC defines an embedded semi-Markov chain. Solving the embedded DTMC and computing the expected values of these first passage times yields the steady state distribution of the semi-Markov chain. If in addition some more computations are carried out on local components, **local** steady state probabilities can be determined.

In [Bohn02] Bohnenkamp exploited the fact that such a first passage time is the maximum of, say m, phase-type distributions. In other words, it is the mean value of the maximum of the times to absorption of m absorbing Markov chains. An algorithm to compute this mean time is given in the cited work.

In chapter 3 we take on the view of Bohnenkamp and apply it to PEPA. We realise that the maximum of the times to absorption of m absorbing Markov chains is just the time to absorption of the joint CTMC of these absorbing Markov chains. We compute the mean time to absorption of the joint CTMC, and **in addition** we will also compute the mean time the joint CTMC spends in some set A before absorption. For these computations we employ the method developed in chapter 2, and hence we never operate on the global state space of the composite PEPA component. Due to computing these additional quantities, we will be able to derive the **global** steady state probability that the composite PEPA component is in set A.

In chapter 4 we conclude.

2 Cumulative Measures of Absorbing Joint Markov Chains

This chapter deals with cumulative measures of an absorbing Markov chain U which is the joint process of m marginal absorbing CTMCs U_i, $i = 1 \dots m$. In this context the mean time to absorption and the mean time spent in some set A before absorption will be of particular interest to us.

We set out with some basic properties of absorbing joint Markov chains in section 2.1. Methods that compute the two quantities mentioned above are briefly discussed. Furthermore, basic convergence properties of the joint process U and the strongly related topic of eigenvalues are addressed.

Sections 2.2 and 2.3 contain the actual accomplishment of this chapter. At first we show how transient probabilities of a discrete time Markov chain embedded in U can be obtained in a compositional way by transient probabilities of DTMCs embedded[1] in the marginal CTMCs U_i, $i = 1 \dots m$, and hence the global state space (i.e. the state space of U) needs not to be constructed nor being operated on. This results in formulas for the computation of the desired two mean values (the mean time to absorption and the mean time spent in A before absorption) whose computation time does not depend on the size of the global state space. The computation times of these formulas depend on the number of marginal CTMCs, the sizes of the marginal state spaces and the convergence speed of the joint CTMC U. The algorithm which computes the two mean values is the topic of section 2.3.

Finally, a list containing most of the notation used throughout this chapter can be found in section 2.4.

[1] The embedded DTMCs refer to uniformised variants of U and the U_i, $i = 1 \dots m$.

2.1 Basic Properties of Absorbing Joint Markov Chains

Let $U_i = (U_i(t))_{t\in\mathbb{R}_{\geq 0}}$, $i = 1,\ldots,m$, be m independent homogeneous absorbing continuous time Markov chains, where U_i, $1 \leq i \leq m$, is defined by the finite state space E_i, the starting state $s_i' \in E_i$, the set of absorbing states $S_i \subset E_i$,and the generator matrix $Q_i = (Q_i(j,\ell))_{j,\ell\in E_i}$.

Now define the Markov chain U as the joint process of $U_1,\ldots,U_m$, i.e. $U = (U(t))_{t\in\mathbb{R}_{\geq 0}} := (U_1,\ldots,U_m)$. Then the state space of U is given by $E = \times_{i=1}^m E_i$, the starting state of U is $s' = (s_1',\ldots,s_m')$ and the set of absorbing states is given by $S = \times_{i=1}^m S_i$. It is well-known that the generator matrix Q of the joint CTMC U can be represented by the Kronecker sum of the generator matrices of the marginal CTMCs, i.e.

$$Q = \oplus_{i=1}^m Q_i. \tag{2.1}$$

The transient probability distribution $p(t)$ of U at time t is then given by the matrix exponential

$$p(t) = p(0)e^{Qt}, \tag{2.2}$$

where $p(0)$ denotes the initial distribution of U. Alternatively, the distribution $p(t)$ can be expressed as

$$p(t) = \sum_{n=0}^{\infty} \frac{(qt)^n}{n!} e^{-qt} p(0) P^n, \tag{2.3}$$

where $P = I + \frac{1}{q}Q$ for some $q \geq \max_{k\in E}\{|Q(k,k)|\}$. Defining $\nu(0) := p(0)$ and $\nu(n) = \nu(n-1)P = p(0)P^n$, for $n > 0$, we also have

$$p(t) = \sum_{n=0}^{\infty} \frac{(qt)^n}{n!} e^{-qt} \nu(n). \tag{2.4}$$

Algorithms building on equation (2.3) or (2.4) to compute transient distributions are widely known as uniformisation method or Jensen's method. Note, that it is recommended to choose $q > \max_{k\in E}\{|Q(k,k)|\}$, in order to avoid periodicities.

Assumptions Throughout This Paper. Throughout this paper we impose the following restrictions on the CTMC U.

- The marginal state spaces E_i, $i = 1\ldots m$, are finite. This implies that the state space E of U is finite.
- $P = I + \frac{1}{q}Q$ is aperiodic. Note, that $q > \max_{k\in E}\{|Q(k,k)|\}$ implies aperiodicity.
- All states not contained in S are transient. That means U possesses exactly $|S|$ recurrent classes, where each absorbing state forms a recurrent class.

2.1.1 Convergence Speed and Eigenvalues

Eigenvalues of P. The following properties about the eigenvalues of the matrix P are well-known (see e.g. [Stew94]).

- P possesses $dim(P)$ eigenvalues, if counting multiplicities.
- P is a stochastic matrix, and hence the multiplicity of the eigenvalue 1 equals the number of recurrent classes of U, i.e. the number of absorbing states $|S|$.
- Since P is aperiodic, there are no other eigenvalues than 1 with modulus 1.

In the following assume that P possesses d distinct eigenvalues, i.e. we are not counting multiplicities, and assume the following indexing

$$1 = |\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_d|. \tag{2.5}$$

Convergence Speed of U. It is a well-known fact that the convergence speed of a discrete time Markov chain is strongly connected with the eigenvalues of the transition matrix P. To understand this we consider the following iterative procedure that computes the stationary distribution π of this Markov chain.

$$\nu(k) = \nu(k-1)P, \quad k \geq 1, \tag{2.6}$$

$$\lim_{k\to\infty} \nu(k) \to \pi, \tag{2.7}$$

where $\nu(0)$ is the initial probability distribution. We see that one iteration step of (2.6) is exactly one step of the power method to compute the eigenpair (λ_1, π) of P, where λ_1 is the largest eigenvalue in modulus of P and π is the corresponding left eigenvector. Of course $\lambda_1 = 1$, since P is stochastic and π is a stationary distribution of the discrete time Markov chain.

The speed of convergence of (2.6) is connected to the eigenvalue λ_2 by

$$\| \nu(k) - \pi \| = O\left(\left| \frac{\lambda_2}{\lambda_1} \right|^k \right). \tag{2.8}$$

Hence, there exists a positive constant c', with

$$\lim_{k\to\infty} \| \nu(k) - \pi \| = c' \cdot |\lambda_2|^k. \tag{2.9}$$

From this follows that there exists also a positive constant c'', such that with the set of absorbing states S

$$\lim_{k\to\infty} |\nu(k)(S) - \pi(S)| = c'' \cdot |\lambda_2|^k. \tag{2.10}$$

With $\pi(S) = 1$, $\nu(k)(S) \leq 1$, this can be written as

$$\lim_{k\to\infty} \frac{1 - \nu(k)(S)}{|\lambda_2|^k} = c''. \tag{2.11}$$

This implies the important relation

$$\lim_{k\to\infty} \frac{1 - \nu(k+1)(S)}{1 - \nu(k)(S)} = |\lambda_2|. \tag{2.12}$$

Provided that the values $\nu(k)(S)$ are known up to some suited index K, $|\lambda_2|$ and the constant c'' can be estimated by exploiting equations (2.12) and (2.11). Of course, for a suited constant $c > c''$ the following bound can be given

$$1 - \nu(k)(S) \leq c \cdot |\lambda_2|^k, \quad \text{for } k > K. \tag{2.13}$$

Eigenvalues and the Marginal CTMCs. In the preceding paragraph we have seen that the second largest eigenvalue modulus $|\lambda_2|$ of P can be estimated if the quantities $\nu(n)(S)$ are known up to a certain index K. Here, we briefly point out how this eigenvalue modulus can be bounded in terms of eigenvalues of the marginal generator matrices Q_i, $i = 1 \ldots m$.

Corollary 1. *For the generator matrices Q_i, $i = 1 \dots m$, and $Q = \oplus_{i=1}^{m} Q_i$ let $P = I + 1/qQ$ and $P_i = I + 1/q_i Q_i$, where $q_i \geq max_j\{|Q_i(j,j)|\}$, $i = 1 \dots m$, and $q \geq max_j\{|Q(j,j)|\}$. Assume that P and the P_i, $i = 1 \dots m$, are aperiodic. Then the following is true.*

(U.1) λ is an eigenvalue of $Q \iff \exists\, \lambda^{(i)}, i = 1 \dots m$, with $\lambda^{(i)}$ eigenvalue of Q_i and $\lambda = \sum_i \lambda^{(i)}$.

(U.2) λ is an eigenvalue of $Q \iff \lambda' := \lambda/q + 1$ is an eigenvalue of P. This result also holds if replacing Q, P and q by Q_i, P_i and q_i.

(U.3) $\exists\, \lambda^{(i)}, i = 1 \dots m$, with $\lambda^{(i)}$ eigenvalue of Q_i and $\lambda' = 1 + 1/q \sum_i \lambda^{(i)} \iff \lambda'$ is an eigenvalue of P.

(U.4) Q and Q_i, $i = 1 \dots m$, possess the eigenvalue 0.

(U.5) Every eigenvalue $\lambda^{(i)} \neq 0$ of Q_i lies in the complex plane within a circle with center $-q_i$ and radius q_i.

(U.6) Every eigenvalue $\lambda^{(i)}$ of Q_i satisfies $|q_i + \lambda^{(i)}| \leq q_i$. If $\lambda^{(i)} \neq 0$ then we have the strict inequality $|q_i + \lambda^{(i)}| < q_i$

Proof. (U.1) is a basic property of the Kronecker sum by which Q can be represented (cf. equation (2.1)). (U.2) is a direct consequence of $P = I + \frac{1}{q}Q$. (U.3) follows by combining (U.1) and (U.2). (U.4) results from Q and the Q_i, $i = 1 \dots m$, being singular matrices. (U.5) follows from the fact that every eigenvalue $1 + 1/q_i\lambda^{(i)} \neq 1$ of the stochastic and aperiodic matrix $P_i := I + 1/q_i Q_i$ lies within the unit circle in the complex plane, and hence transforming P_i into Q_i implicates that the eigenvalues of P_i are shifted to the left by 1 and afterwards scaled by the factor q_i. (U.6) follows from (U.5) by noting that for every eigenvalue $\lambda^{(i)} \neq 0$ of Q_i the number $q_i + \lambda^{(i)}$ lies in the complex plane within a circle with center 0 and radius q_i.

□

The following corollary bounds the second eigenvalue modulus $|\lambda_2|$ of P in terms of eigenvalues of the Q_i, $i = 1 \dots m$.

Corollary 2. *For the generator matrices Q_i, $i = 1 \dots m$, and $Q = \oplus_{i=1}^{m} Q_i$ let $P = I + 1/qQ$, where $q_i \geq max_j\{|Q_i(j,j)|\}$, $i = 1 \dots m$, and $q = q_1 + \dots + q_m$. Assume that P is aperiodic. Let Λ_i be the set of distinct eigenvalues of Q_i, $i = 1 \dots m$, and define the values max and $i(max)$ as*

$$max = \max_{\substack{1 \leq i \leq m \\ \lambda^{(i)} \in \Lambda_i \setminus \{0\}}} \{|q_i + \lambda^{(i)}|\} \quad \textit{and} \quad i(max) = i \iff max = |q_i + \lambda^{(i)}|. \tag{2.14}$$

Then the second largest eigenvalue modulus of P satisfies

$$|\lambda_2| \leq \frac{1}{q} \left(max + \sum_{\substack{1 \leq i \leq m \\ i \neq i(max)}} q_i \right). \tag{2.15}$$

Proof. First note that by the representation of Q as the Kronecker sum of the Q_i, from $q_i \geq max_j\{|Q_i(j,j)|\}$, $i = 1 \dots m$, follows $q = q_1 + \dots + q_m \geq max_j\{|Q(j,j)|\}$. Thus, P is indeed a stochastic matrix. According to (U.1) for every eigenvalue λ' of P there exist $\lambda^{(i)} \in \Lambda_i$, $i = 1 \dots m$, with

$$\lambda' = 1 + \frac{1}{q} \sum_{i=1}^{m} \lambda^{(i)} = \frac{1}{q} \sum_{i=1}^{m} q_i + \lambda^{(i)}, \tag{2.16}$$

where the last term is obtained by obeying $q = q_1 + \cdots + q_m$. λ' is a complex number, and hence application of the triangle equality yields

$$|\lambda'| = |1 + \frac{1}{q}\sum_{i=1}^{m} \lambda^{(i)}| \leq \frac{1}{q}\sum_{i=1}^{m} |q_i + \lambda^{(i)}|. \tag{2.17}$$

(U.4) and (U.6) state that the sum on the right-hand side becomes maximal if and only if all of the $\lambda^{(i)}$ equal 0. In this case both $|\lambda'|$ and the sum on the right-hand side become 1.

Hence in the representation (2.16) of the eigenvalue λ_2 of P in terms of eigenvalues $\lambda^{(i)}$, $i = 1 \ldots m$, at least one of the $\lambda^{(i)}$ must not equal 0. Clearly the maximum value of $\frac{1}{q}\sum_{i=1}^{m} |q_i + \lambda^{(i)}|$, where at least one of the $\lambda^{(i)}$ not equal 0, is given by (2.15).

□

2.1.2 The Mean Time to Absorption and Related Quantities

Two quantities of U will be of interest to us throughout the rest of this chapter: the mean time to absorption and the mean time spent in some set A before absorption. Obviously, the latter quantity is a fraction of the mean time to absorption.

At first we wish to give some insight into classical approaches to compute these two quantities.

The Mean Time to Absorption: A Classical Approach. Let H be a random variable for the time until absorption of the CTMC U. Classic approaches to compute the mean time $\mathbb{E}[H]$ rely on the fact that H is a phase-type distributed random variable whose generator matrix can be written in the form

$$Q^{\circ} = \begin{pmatrix} T & T_0 \\ 0 & 0 \end{pmatrix}, \tag{2.18}$$

where Q° results from Q be rearranging the states such that the block T contains the transition rates between the transient states and T_0 contains the rates from transient to absorbing states.

Let $\alpha = (\alpha_1, \ldots, \alpha_{dim(T)})$ be the sub-vector of the starting distribution of U, which contains the initial probabilities of the transient states. Then the phase-type distribution can be characterised by its representation (α, T). It is well known that the $n-$th moment of a phase-type distributed random variable is given by (see e.g. [Neut81])

$$\mathbb{E}[H^n] = (-1)^n n! \alpha T^{-n} \mathbb{1}, \tag{2.19}$$

where $\mathbb{1}$ is a column vector of size $dim(T)$ consisting of ones. In particular, the mean hitting time is given by

$$\mathbb{E}[H] = -1 \alpha T^{-1} \mathbb{1}, \tag{2.20}$$

which could be solved by an explicit matrix inversion. Alternatively, with the solution of $xT = \alpha$, one obtains $\mathbb{E}[H] = -x\mathbb{1}$. Since the state space of U, and hence also the dimension of the matrix T, grows exponentially in the number m of marginal CTMCs, the computation of $\mathbb{E}[H]$ according to (2.20) would in general only be feasible for small values of m.

Fractions of the Mean Time to Absorption: A Classical Approach. From the mean time to absorption, which can also be seen as the mean time the CTMC U spends in any state before absorption, we turn to the mean time which U spends in some set A before absorption. With respect to requirements needed later on in this work, we restrict the set A to possess the

structure

$$A = A_1 \times A_2 \times \cdots \times A_m, \tag{2.21}$$

where $A_i \subseteq E_i$, $i = 1 \ldots m$. We impose the restriction that A does not contain absorbing states, i.e. $A \cap S = \emptyset$, or equivalently $\exists i \in \{1, \ldots, m\} : A_i \cap S_i = \emptyset$.

Let H_A be a random variable for the time that U spends in set A before absorption and let π_A be the relative amount of time that U spends in A before absorption. It is clear that the mean time $\mathbb{E}[H_A]$ is given by

$$\mathbb{E}[H_A] = \pi_A \mathbb{E}[H]. \tag{2.22}$$

Now, we concentrate on the computation of π_A. First, we assume that U possesses exactly one starting state. W.l.o.g. assume that the first row and first column of Q° correspond to that starting state. Now, we construct a regenerative process $\mathfrak{U}$, where one regeneration cycle of that new process is described by the absorbing Markov chain U, i.e. if U becomes absorbed, the process $\mathfrak{U}$ steps into a new regeneration cycle. In other words, we manipulate U such that each transition which would lead to an absorbing state is directed to to the starting state instead – this new process we call $\mathfrak{U}$.

Of course, the steady state probability $\mathbb{P}(\mathfrak{U} \in A)$ equals the fraction of time spent in A during one regeneration cycle, and hence we have $\mathbb{P}(\mathfrak{U} \in A) = \pi_A$. More formally, if $\mathfrak{T}$ is the generator matrix[2] of $\mathfrak{U}$, then

$$\pi_A = y(A), \qquad \text{where } \; y\mathfrak{T} = 0, \;\; \| \, y \, \|_1 = 1. \tag{2.23}$$

The Mean Time to Absorption: Uniformisation. An alternative approach to compute the mean time to absorption is based on the uniformisation method. The starting point is the following representation of the mean time to absorption

$$\mathbb{E}[H] = \int_0^\infty p(t)(E \setminus S)dt. \tag{2.24}$$

Employing equation (2.4) we obtain

$$\begin{aligned} \mathbb{E}[H] &= \int_0^\infty \sum_{n=0}^{\infty} \frac{(qt)^n}{n!} e^{-qt} \nu(n)(E \setminus S)dt && (2.25) \\ &= \frac{1}{q} \sum_{n=0}^{\infty} \nu(n)(E \setminus S) && (2.26) \\ &= \frac{1}{q} \sum_{n=0}^{\infty} 1 - \nu(n)(S), && (2.27) \end{aligned}$$

where $\nu(0) = p(0)$ and $\nu(n) = p(0)P^n = \nu(n-1)P$, for $n \geq 1$.

For practical computations it is necessary to truncate this sum after some index N, which introduces the absolute error

$$err = \frac{1}{q} \sum_{n=N+1}^{\infty} 1 - \nu(n)(S). \tag{2.28}$$

In section 2.1.1 we saw that $1 - \nu(k)(S)$ can be bounded by $c|\lambda_2|^k$, for all $k > K$, where λ_2 is the second largest eigenvalue in modulus of P and c is a suited positive constant. Hence, if

[2] Note that $\mathfrak{T}$ can easily be derived from the matrices T and T_0.

c and λ_2 are known the error can be bounded by

$$err < \frac{1}{q} \sum_{n=N+1}^{\infty} c|\lambda_2|^n = \frac{c}{q} \frac{|\lambda_2|^{N+1}}{1-|\lambda_2|}. \tag{2.29}$$

Fractions of the Mean Time to Absorption: Uniformisation. The fraction of time spent in the set A before absorption is given by

$$\mathbb{E}[H_A] = \int_0^{\infty} p(t)(A)dt, \quad \text{for } A \cap S = \emptyset \tag{2.30}$$

In analogy to the transformations concerning the mean time to absorption in the preceding paragraph, we at first adopt the notation $\nu(n) = p(0)P^n$, for $n \geq 0$, and finally obtain, for $A \cap S = \emptyset$,

$$\mathbb{E}[H_A] = \int_0^{\infty} \sum_{n=0}^{\infty} \frac{(qt)^n}{n!} e^{-qt} \nu(n)(A)dt = \frac{1}{q} \sum_{n=0}^{\infty} \nu(n)(A). \tag{2.31}$$

Since A contains no absorbing states, we have $\nu(n)(A) \leq \nu(n)(E \setminus S) = 1 - \nu(n)(S)$, and hence

$$\frac{1}{q} \sum_{n=N+1}^{\infty} \nu(n)(A) \leq \frac{c}{q} \frac{|\lambda_2|^{N+1}}{1-|\lambda_2|}. \tag{2.32}$$

2.1.3 Outline

A straight forward way to evaluate the formulas

$$\mathbb{E}[H] = \frac{1}{q} \sum_{n=0}^{\infty} 1 - \nu(n)(S), \tag{2.33}$$

$$\mathbb{E}[H_A] = \frac{1}{q} \sum_{n=0}^{\infty} \nu(n)(A) \tag{2.34}$$

would be to compute the distributions $\nu(n)$ up to some truncation index N, where $\nu(n)$ could be obtained by exploiting the relation $\nu(n) = \nu(0)P^n = \nu(n-1)P$, and afterwards compute the above sums. But since the state space of $U = (U_1, \ldots, U_m)$ grows exponentially in the number m of marginal CTMCs, the dimension of P, as well as the size of the distribution vector $\nu(n)$, is exponential in m. Thus, computing $\mathbb{E}[H]$ and $\mathbb{E}[H_A]$ by the procedure sketched above is only feasible for small values of m.

The next section 2.2 deals with finding a different method to compute the quantities $\nu(n)(A)$ and $\nu(n)(S)$ which circumvents the state space explosion problem. The idea is to express $\nu(n)(A)$ and $\nu(n)(S)$ by means of akin marginal quantities $\nu_i(k)(A_i)$ and $\nu_i(k)(S_i)$, $k = 0 \ldots n$, $i = 1 \ldots m$, which are gained from computations on the marginal chains only.

In section 2.3 we introduce a new method to compute $\mathbb{E}[H]$ and $\mathbb{E}[H_A]$. This method follows formulas (2.33) and (2.34), but employs a special technique for the computation of the $\nu(n)(A)$ and $\nu(n)(S)$, $n = 0 \ldots N$.

2.2 A Non-Product Form Representation for Transient State Probabilities

Let $A \subseteq E$ be any subset of the state space of the joint CTMC U, with

$$A = \times_{i=1}^{m} A_i, \tag{2.35}$$

where $A_i \subseteq E_i$, $i = 1 \dots m$.

2.2.1 Transient State Probabilities in Product Form

Transient Probabilities of the Marginal CTMCs. Let $p_i(t) = (p_i(t)(u))_{u \in E_i}$ be the transient probability distribution of the chain U_i at time t. For a given initial distribution $p_i(0)$, some $q_i \geq \max_{j \in E_i}\{|Q_i(j,j)|\}$ and with

$$P_i := I + 1/q_i Q_i, \tag{2.36}$$

the probability $p_i(t)(A_i)$ can be expressed as the series

$$p_i(t)(A_i) = \sum_{k=0}^{\infty} \frac{(q_i t)^k}{k!} e^{-q_i t} \nu_i(k)[A_i], \tag{2.37}$$

where $\nu_i(0) = p_i(0)$ and $\nu_i(k+1) = \nu_i(k)P_i$, for $k \geq 0$. In (2.37) we use the convention

$$\nu_i[A] = (\nu_i(k)[A_i])_{k \in \mathbb{N}_0} := (\nu_i(k)(A_i))_{k \in \mathbb{N}_0}, \quad \text{for } i = 1 \dots m, \tag{2.38}$$

i.e. we interpret $\nu_i[A_i]$ as a function of n.

Transient Probabilities of the Joint CTMC U. For some $q \geq \max_{j \in E}\{|Q(j,j)|\}$, define $P = I + \frac{1}{q}Q$. Let $\nu(0) = p(0)$ be the initial distribution of U and $\nu(n) = \nu(0)P^n = \nu(n-1)P$, for $n \geq 1$.

Analogously to the preceding paragraph, for $\nu(n)(A)$ we subsequently write $\nu(n)[A]$ to stress the fact that A is fixed and $\nu[A]$ can be seen as a function of n.

On the one hand, the transient probability $p(t)(A)$, that U is in A at time t, can be obtained by the uniformisation equation

$$p(t)(A) = \sum_{n=0}^{\infty} \frac{(qt)^n}{n!} e^{-qt} \nu(n)[A]. \tag{2.39}$$

On the other hand, by independence of the marginal CTMCs U_i, $i = 1 \dots m$, $p(t)(A)$ possesses the product form

$$p(t)(A) = p_1(t)(A_1) \cdot p_2(t)(A_2) \cdots p_m(t)(A_m), \tag{2.40}$$

where $p_i(t)(A_i) = \mathbb{P}(U_i(t) \in A_i)$, $i = 1 \dots m$.

Although, this product-form relation exists for transient probabilities, a similar product-form result for the cumulative measure $\int_0^\infty p(t)(A)dt$ is not available.

The remainder of this section deals with a reformulation of (2.40). Based on this reformulation, we will be able to compute cumulative measures of the above kind by operating on the marginal CTMCs U_i, $i = 1 \dots m$, only.

2.2.2 Transient State Probabilities in Non-Product Form

First recall that $q_i \geq \max_{j\in E_i}\{|Q_i(j,j)|\}$ and $q \geq \max_{j\in E}\{|Q(j,j)|\}$. Since $Q = \oplus_{i=1}^m Q_i$, we have

$$\max_{j\in E}\{|Q(j,j)|\} = \sum_{i=1}^m \max_{j\in E_i}\{|Q_i(j,j)|\}. \tag{2.41}$$

Hence for valid uniformisation rates q_i of U_i, $i = 1\dots m$, $\sum_i q_i$ is a valid uniformisation rate of U. Vice versa, for every valid uniformisation rate q of U, there exist valid uniformisation rates q_i of U_i, $i = 1\dots m$, with $q = \sum_i q_i$.

With the substitution

$$q := q_1 + q_2 + \cdots + q_m \tag{2.42}$$

and equations (2.40) and (2.37) the following equalities are obtained

$$\begin{aligned}
p(t)(A) &= \left(\sum_{n_1=0}^{\infty} \frac{(q_1 t)^{n_1}}{n_1!} e^{-q_1 t}\nu_1(n_1)[A_1]\right)\cdots\left(\sum_{n_m=0}^{\infty} \frac{(q_m t)^{n_m}}{n_m!} e^{-q_1 t}\nu_m(n_m)[A_m]\right) && (2.43)\\
&= \sum_{n_1=0}^{\infty}\cdots\sum_{n_m=0}^{\infty}\left(t^{(n_1+\cdots+n_m)}e^{-qt}\prod_{i=1}^m \frac{q_i^{n_i}}{n_i!}\nu_i(n_i)[A_i]\right) && (2.44)\\
&= \sum_{n=0}^{\infty} t^n e^{-qt}\sum_{n_1+\cdots+n_m=n}\prod_{i=1}^m \frac{q_i^{n_i}}{n_i!}\nu_i(n_i)[A_i] && (2.45)\\
&= \sum_{n=0}^{\infty}\frac{(qt)^n}{n!}e^{-qt}\sum_{n_1+\cdots+n_m=n}\left[n!\prod_{i=1}^m \frac{q_i^{n_i}}{n_i! q^{n_i}}\right]\prod_{i=1}^m \nu_i(n_i)[A_i]. && (2.46)
\end{aligned}$$

Since $n_1 + \cdots + n_m = n$, the above term in brackets is a multinomial probability. We denote this term by

$$M(n, n_1, \dots, n_m) := \frac{n!}{\prod_{i=1}^m n_i!}\prod_{i=1}^m \left(\frac{q_i}{q}\right)^{n_i} \tag{2.47}$$

and write

$$p(t)(A) = \sum_{n=0}^{\infty}\frac{(qt)^n}{n!}e^{-qt}\sum_{n_1+\cdots+n_m=n} M(n, n_1, \dots, n_m)\prod_{i=1}^m \nu_i(n_i)[A_i]. \tag{2.48}$$

Comparing (2.48) to (2.39), it might be supposed that the inner sum equals $\nu(n)[A]$ which is the probability that U is in $A = (A_1, \dots, A_m)$ at the $n-$th step. That this is indeed the case is proved in the following corollary.

Corollary 3. *The following assertion holds.*

$$\nu(n)[A] = \sum_{n_1+\cdots+n_m=n} M(n, n_1, \dots, n_m)\prod_{i=1}^m \nu_i(n_i)[A_i]. \tag{2.49}$$

Proof. We provide a probabilistic proof. At first, let us recall that the marginal processes U_i, $i = 1\dots m$, are uniformised with rate q_i, i.e. the sequence of steps of U_i forms a poisson stream with rate q_i. Then, the sequence of steps of the joint process U is the superposition of m poisson streams, and hence its rate is given by $q = q_1 + \cdots + q_m$.

Let $\#(U)$ be the number of steps of U , and accordingly let $\#(U_i)$ be the number of steps

of U_i. Then, the probability $\nu(n)[A]$ that U is in A in the $n-$th step is given by

$$\begin{aligned}
\nu(n)[A] &= \mathbb{P}(U \in A | \#(U) = n) && (2.50)\\
&= \mathbb{P}(U \in A | \#(U_1) + \cdots + \#(U_m) = n) && (2.51)\\
&= \sum_{n_1,\ldots,n_m \geq 0} \mathbb{P}(\#(U_1) = n_1, \ldots, \#(U_m) = n_m | \sum_i n_i = n) \cdot && (2.52)\\
&\qquad \mathbb{P}(U \in A | \#(U_1) = n_1, \ldots, \#(U_m) = n_m, \sum_i n_i = n) && (2.53)
\end{aligned}$$

Expressing $\{U \in A\}$ in the second term by means of marginal quantities and with the insight that the condition $\{\sum_i n_i = n\}$ in the second term is unnecessary, we obtain

$$\begin{aligned}
\nu(n)[A] &= \sum_{n_1,\ldots,n_m \geq 0} \mathbb{P}(\#(U_1) = n_1, \ldots, \#(U_m) = n_m | \sum_i n_i = n) \cdot && (2.54)\\
&\qquad \mathbb{P}(U_1 \in A_1, \ldots, U_m \in A_m | \#(U_1) = n_1, \ldots, \#(U_m) = n_m) && (2.55)\\
&= \sum_{n_1,\ldots,n_m \geq 0} \mathbb{P}(\#(U_1) = n_1, \ldots, \#(U_m) = n_m | \sum_i n_i = n) \prod_i \nu_i(n_i)[A_i] && (2.56)
\end{aligned}$$

The last line results from the independence of the marginal processes U_i, $i = 1 \ldots m$, and the fact that $\nu_i(n_i)[A_i] = \mathbb{P}(U_i \in A_i | \#(U_i) = n_i)$. Now, we realise that due to the condition $\{\sum_i n_i = n\}$ we only need to sum over values of the n_i, $i = 1 \ldots m$, which sum up to n.

$$\begin{aligned}
\nu(n)[A] &= \sum_{n_1+\cdots+n_m=n} \mathbb{P}(\#(U_1) = n_1, \ldots, \#(U_m) = n_m | \#(U) = n) \prod_i \nu_i(n_i)[A_i] && (2.57)\\
&= \sum_{n_1+\cdots+n_m=n} \frac{\prod_i \frac{q_i^{n_i}}{n_i!} e^{-q_i}}{\frac{q^n}{n!} e^{-q}} \prod_i \nu_i(n_i)[A_i] && (2.58)\\
&= \sum_{n_1+\cdots+n_m=n} \frac{n!}{\prod_i n_i!} \prod_i \left(\frac{q_i}{q}\right)^{n_i} \prod_i \nu_i(n_i)[A_i] && (2.59)
\end{aligned}$$

The last two lines make use of the fact that $\#(U_i)$ is poisson distributed with rate q_i, $i = 1 \ldots m$, and $\#(U)$ is poisson distributed with rate $q = q_1 + \cdots + q_m$. □

The interpretation of the statement of corollary 3 is that, in order to determine $\nu(n)[A]$, we consider all possible combinations of the number of marginal steps, which sum up to n. This is due to the fact that the sum of the marginal steps is the number of steps of the joint process U. Given some combination of marginal steps $(n_1, \ldots, n_m)$ we can immediately state that the conditional probability of being in A is given by $\prod_i \nu_i(n_i)[A_i]$. As it turns out, the probability that a certain combination of marginal steps occurs is a multinomial probability. Thus, $\nu(n)[A]$ is obtained by deconditioning $\prod_i \nu_i(n_i)[A_i]$ from $(n_1, \ldots, n_m)$.

The next section reformulates (2.49) as a convolution-like expression, which can be used to compute $\nu(n)[A]$ in an iterative way.

2.2.3 Transient Probabilities and the $\star$-Convolution

Definition 4. For two discrete functions $f, g : \mathbb{N}_0 \to \mathbb{R}_0$ and constant values $c_f, c_g \in \mathbb{R}_{>0}$ assigned to these functions, define

$$(f \star g)(n) = \sum_{k=0}^{n} \binom{n}{k} \frac{c_f^k c_g^{n-k}}{(c_f + c_g)^n} f(k) g(n-k) \tag{2.60}$$

$$\text{and } c_{f\star g} = c_f + c_g. \tag{2.61}$$

The proof of the following corollary is found in the appendix A.1.

Corollary 5. *The $\star-$operator is commutative and associative.*

Theorem 6. *With the functions $\nu_i[A_i], 1 \le i \le m$, and the values $c_{\nu_i[A_i]} := q_i$ assigned to them, the following assertion holds*

$$(\nu_1[A_1] \star \nu_2[A_2] \star \cdots \star \nu_m[A_m])(n) = \sum_{n_1+\cdots+n_m=n} M(n, n_1, \ldots, n_m) \prod_{i=1}^{m} \nu_i(n_i)[A_i]. \tag{2.62}$$

Proof.

$$\sum_{n_1+\cdots+n_m=n} M(n, n_1, \ldots, n_m) \prod_{i=1}^{m} \nu_i(n_i)[A_i] = n! \sum_{n_1+\cdots+n_m=n} \prod_{i=1}^{m} \underbrace{\frac{q_i^{n_i}}{n_i! q^{n_i}} \nu_i(n_i)[A_i]}_{a_i(n_i)} \tag{2.63}$$

$$= n!(a_1 * \cdots * a_m)(n), \tag{2.64}$$

where $a_i(n_i) = \frac{q_i^{n_i}}{n_i! q^{n_i}} \nu_i(n_i)[A_i]$ and $*$ is the discrete convolution operator.

Now, it is sufficient to show that

$$(\nu_1[A_1] \star \nu_2[A_2] \star \cdots \star \nu_m[A_m])(n) = n!(a_1 * a_2 * \cdots * a_m)(n). \tag{2.65}$$

Bearing in mind that $q = q_1 + \cdots + q_m$ it suffices to show that for every $\ell \le m$ the relation

$$\frac{(q_1 + \cdots + q_\ell)^n}{q^n} (\nu_1[A_1] \star \cdots \star \nu_\ell[A_\ell])(n) = n!(a_1 * \cdots a_\ell)(n) \tag{2.66}$$

holds. We proof this by induction over ℓ. For $\ell = 1$ the assertion is trivially true and for $\ell = 2$ we have

$$(\nu_1[A_1] \star \nu_2[A_2])(n) = \sum_{k=0}^{n} \binom{n}{k} \frac{q_1^k q_2^{n-k}}{(q_1 + q_2)^n} \nu_1(k)[A_1] \nu_2(n-k)[A_2] \tag{2.67}$$

$$= n! \frac{q^n}{(q_1 + q_2)^n} (a_1 * a_2)(n), \tag{2.68}$$

where we multiplied the sum with $\frac{q^n}{q^n}$, in order to obtain the last line. This provides us with a

valid induction hypothesis. For the induction step we obtain

$$
\begin{aligned}
(\nu_1[A_1]\star \quad \cdots \quad \star\nu_{\ell+1}[A_{\ell+1}])(n) = & \qquad (2.69)\\
= \quad \sum_{k=0}^{n} \binom{n}{k} \frac{(q_1+\cdots+q_\ell)^k q_{\ell+1}^{n-k}}{(q_1+\cdots+q_{\ell+1})^n} \underbrace{(\nu_1[A_1]\star\cdots\star\nu_\ell[A_\ell])(k)}_{\substack{=k!\frac{q^k}{(q_1+\cdots+q_\ell)^k}(a_1*\cdots*a_\ell)(k)\\ \textit{per induction hypothesis}}} \nu_{\ell+1}(n-k)[A_{\ell+1}] & \qquad (2.70)\\
= \quad \sum_{k=0}^{n} k! \binom{n}{k} \frac{q_{\ell+1}^{n-k}}{(q_1+\cdots+q_{\ell+1})^n} q^k (a_1*\cdots*a_\ell)(k)\nu_{\ell+1}(n-k)[A_{\ell+1}] & \qquad (2.71)\\
= \quad \sum_{k=0}^{n} \frac{n!}{(n-k)!} \frac{q_{\ell+1}^{n-k}}{(q_1+\cdots+q_{\ell+1})^n} \frac{q^n}{q^{n-k}} (a_1*\cdots*a_\ell)(k)\nu_{\ell+1}(n-k)[A_{\ell+1}] & \qquad (2.72)\\
= \quad n! \frac{q^n}{(q_1+\cdots+q_{\ell+1})^n} \sum_{k=0}^{n} \frac{q_{\ell+1}^{n-k}}{(n-k)!q^{n-k}} (a_1*\cdots*a_\ell)(k)\nu_{\ell+1}(n-k)[A_{\ell+1}] & \qquad (2.73)\\
= \quad n! \frac{q^n}{(q_1+\cdots+q_{\ell+1})^n} (a_1*\cdots*a_{\ell+1})(n) & \qquad (2.74)
\end{aligned}
$$

□

The following theorem summarises corollary 3 and theorem 6.

Theorem 7. *The following is true:*

$$\nu(n)[A] = (\nu_1[A_1]\star\nu_2[A_2]\star\cdots\star\nu_m[A_m])(n) = \sum_{n_1+\cdots+n_m=n} M(n,n_1,\ldots,n_m)\prod_{i=1}^{m}\nu_i(n_i)[A_i]. \quad (2.75)$$

Proof. Corollary 3 and theorem 6. □

To conclude this subsection, we take up equation (2.48) and in consideration of theorem 7 obtain

$$p(t)[A] = \sum_{n=0}^{\infty} \frac{(qt)^n}{n!} e^{-qt} (\nu_1[A_1]\star\nu_2[A_2]\star\cdots\star\nu_m[A_m])(n). \quad (2.76)$$

2.2.4 Cumulative Measures by Means of the $\star$-Convolution

We have seen that the transient probability $\nu(n)[A]$ can be related to the (marginal) functions $\nu_i[A_i]$, $i = 1\ldots m$, via the $\star$-convolution.

As a direct consequence, the cumulative measures

$$\mathbb{E}[H] = \int_0^\infty 1 - p(t)(S)dt \quad \text{and} \quad \mathbb{E}[H_A] = \int_0^\infty p(t)(A)dt, \quad \text{for } A\cap S = \emptyset \quad (2.77)$$

can also be expressed by means of the $\star$-convolution.

Theorem 8. *For $A_i \subseteq E_i$, $i = 1\ldots m$, $A = \times_{i=1}^m A_i$ and $A\cap S = \emptyset$, the expected time $\mathbb{E}[H_A]$ which U spends in A before absorption is given by*

$$\mathbb{E}[H_A] = \frac{1}{q}\sum_{n=0}^{\infty}(\nu_1[A_1]\star\nu_2[A_2]\star\cdots\star\nu_m[A_m])(n). \quad (2.78)$$

Proof. The assertion follows from equation (2.31) in combination with theorem 7. □

Theorem 9. *The expected time until absorption $\mathbb{E}[H]$ of U is given by*

$$\mathbb{E}[H] = \frac{1}{q}\sum_{n=0}^{\infty} 1 - (\nu_1[S_1] \star \nu_2[S_2] \star \cdots \star \nu_m[S_m])(n). \tag{2.79}$$

Proof. The assertion follows from equation (2.27) in combination with theorem 7.

□

2.3 Algorithms for $\mathbb{E}[H]$ and $\mathbb{E}[H_A]$

This section aims at giving all means needed for an implementation of a programme which computes $\mathbb{E}[H]$ and $\mathbb{E}[H_A]$.

Therefore, we provide a short compilation of the key results (section 2.3.1) which have been the subject of previous sections, followed by the actual algorithm and a detailed complexity analysis (section 2.3.2).

2.3.1 A Short Summarization

In order to compute the two quantities

$$\mathbb{E}[H] = \frac{1}{q}\sum_{n=0}^{\infty} 1 - \nu(n)[S], \tag{2.80}$$

$$\mathbb{E}[H_A] = \frac{1}{q}\sum_{n=0}^{\infty} \nu(n)[A] \tag{2.81}$$

it is necessary to truncate the sum after a certain index N, i.e. we actually compute

$$\mathbb{E}[H] \approx \frac{1}{q}\sum_{n=0}^{N} 1 - \nu(n)[S], \tag{2.82}$$

$$\mathbb{E}[H_A] \approx \frac{1}{q}\sum_{n=0}^{N} \nu(n)[A]. \tag{2.83}$$

With the second largest eigenvalue in modulus λ_2 of P and a suited constant c the absolute errors can be bounded by

$$\frac{1}{q}\sum_{n=N+1}^{\infty} 1 - \nu(n)[S] < \frac{c}{q}\frac{|\lambda_2|^{N+1}}{1-|\lambda_2|} =: error \tag{2.84}$$

$$\frac{1}{q}\sum_{n=N+1}^{\infty} \nu(n)[A] < \frac{c}{q}\frac{|\lambda_2|^{N+1}}{1-|\lambda_2|} =: error. \tag{2.85}$$

The absolute value of the eigenvalue λ_2 can be computed by exploiting the fact that

$$\lim_{n\to\infty} \frac{1-\nu(n+1)[S]}{1-\nu(n)[S]} = |\lambda_2|.$$

After $|\lambda_2|$ has been determined (or estimated) the constant c can be estimated by considering

$$\lim_{n\to\infty} \frac{1-\nu(n)[S]}{|\lambda_2|^n} = constant. \tag{2.86}$$

That means we chose some $c > constant$.

2.3.2 Algorithm

We now present a detailed algorithm in pseudo-code (Procedure CM()) which computes approximations of $\mathbb{E}[H_A]$ and $\mathbb{E}[H]$.

The input of this procedure comprises of the generator matrices Q_i, $i = 1 \dots m$, of the marginal CTMCs, the marginal starting states s'_i, $i = 1 \dots m$, and an upper error bound ϵ.

The predicate $precision()$ becomes true if the current values of $|\lambda_2|$ and c fulfil a given precision. Although, not explicitly implemented in the algorithm below, it is clear how an implementation of such a predicate could look like. For example, the values in lines 20 and 21 could be stored (for a while), and the relative changes of these values determine the predicate $precision$.

Procedure CM($Q_1, \ldots, Q_m; s'_1, \ldots, s'_m; A_1, \ldots, A_m; \epsilon$)

```
   /* ---------------- Uniformise all marginal processes ----------------  */
1  for i = 1...m do
2      choose q_i ≥ max_{j∈E_i}{|Q_i(j,j)|};          /* Uniformisation factor */
3      P_i = I + (1/q_i) Q_i;                          /* One-step transition matrix */
4      ν_i(0)(s'_i) = 1;                               /* Initial state */

   /* --------------------- Some initial assignments --------------------  */
5  ν(0)[S] = (ν_1[S_1] ⋆ ν_2[S_2] ⋆ ··· ⋆ ν_m[S_m])(0);
6  ν(0)[A] = (ν_1[A_1] ⋆ ν_2[A_2] ⋆ ··· ⋆ ν_m[A_m])(0);
7  q = q_1 + q_2 + ··· + q_m;
8  n = 0;
9  choose W ≥ 1;
10 error = 2ε;                                         /* initially chosen such that error > ε */

   /* ---------- Compute marginal distributions and convolution ----------  */
11 while not precision() and error ≥ ε do
12     for i = 1...m do
13         for k = n+1...n+W do
14             ν_i(k) = ν_i(k−1) P_i;                  /* distribution at the k-th step */
15             compute ν_i(k)[A_i];                    /* the aggregated state probabilities */
16             compute ν_i(k)[S_i];                    /* ν_i(·)[A_i] and ν_i(·)[S_i] are stored permanently */
17     for k = n+1...n+W do
18         ν(k)[S] = (ν_1[S_1] ⋆ ν_2[S_2] ⋆ ··· ⋆ ν_m[S_m])(k);
19         ν(k)[A] = (ν_1[A_1] ⋆ ν_2[A_2] ⋆ ··· ⋆ ν_m[A_m])(k);
20         |λ_2| = (1−ν(k)[S]) / (1−ν(k−1)[S]);        /* Approximation of |λ_2| */
21         c = (1−ν(k)[S]) / |λ_2|^k;
22     n = n + W;
23     error = (c/q) · |λ_2|^{n+1} / (1−|λ_2|);

   /* ------------------- Compute cumulative measures -------------------  */
24 N = n;
25 a = Σ_{k=0}^{N} (1/q) ν(k)[A];                       /* Approximation to E[H_A] */
26 b = Σ_{k=0}^{N} (1/q) (1 − ν(k)[S]);                 /* Approximation to E[H] */
```

In lines 1 to 4 the m marginal CTMCs are uniformised. P_i is the one-step transition matrix resulting from the generator Q_i. Initially, all probability mass of the $i-$th CTMC is gathered in the state s'_i.

In lines 5 to 10 some initial computations are carried out.

The interesting work of the algorithm is carried out in the while loop beginning in line 11. This while-loop is iterated through until the error of the desired quantities and the estimated parameters $|\lambda_2|$ and c that are used to compute the error fulfil a given precision. During each pass of the while-loop W more elements of the functions $\nu[A]$ and $\nu[S]$ are computed, i.e. in the first pass $\nu(1)[A], \ldots, \nu(W)[A]$ are computed, in the second pass $\nu(W+1)[A], \ldots, \nu(2W)[A]$, and so on ($\nu(\cdot)[S]$ accordingly). Note, that the values of $\nu(\ell)[A]$ and $\nu(\ell)[S]$ can indeed be computed, since all the values $\nu_i(n)[A_i]$ and $\nu_i(n)[S_i]$, $n = 0 \ldots \ell$, $i = 1 \ldots m$, have been

determined before (in the current pass and previous passes of the while-loop).

One might wonder why we don't just take $W = 1$ and compute only one convolution element in each pass of the while loop. Indeed, if all the marginal transition matrices P_i, $i = 1 \dots m$, and the values $\nu_i(n)[A_i]$, $n = 0 \dots N$, $i = 1 \dots m$, can be kept in the main memory at the same time, $W = 1$ would be the first choice. If, on the other hand, the capacity of the main memory is limited, such that we are forced to keep the P_i that are currently not needed, in a secondary memory, then every time we are required to operate on a matrix P_ℓ, it must be loaded from the secondary to the main memory. Of course, loading (or copying) a huge matrix into the main memory, is a time intensive procedure, especially if it has to be done frequently. It is clear that the number of times a matrix must be copied decreases as W increases.

We note that at the beginning of the algorithm an upper bound for the second largest eigenvalue modulus $|\lambda_2|$ of P could be computed according to corollary 2, in order to get an idea of the number of times the while loop has to be cycled through.

Storage Requirements. Essentially the storage requirements are assembled by the space needed to store the transition matrices P_i, $i = 1 \dots m$, and the space to store the functions $\nu_i[A_i]$ and $\nu_i[S_i]$, $i = 1 \dots m$. With $dim(P_i) =$*'dimension of* P_i*'*, let $d = \max_i\{dim(P_i)\}$. Then the overall storage requirement of the procedure CM lies in $O(m(d^2 + N))$.

Time Complexity. To determine the time complexity of the algorithm, we can set $W = 1$. Then, the while-loop is iterated through N times. In the $k-$th pass of the while-loop we have to do the following:

- Compute the m marginal distributions at the $k-$th step, which is in $O(md^2)$.
- Compute $\nu_i(k)[A_i]$ and $\nu_i(k)[S_i]$, $i = 1 \dots m$, which is in $O(2md)$.
- Compute the values $\nu(k)(A)$ and $\nu(k)(S)$, which according to the representation as the star convolution can be done with $O(2mk)$ operations.

After N iteration steps, we arrive at $N \cdot O(md^2) + N \cdot O(2md) + O(m(N^2 - N)) = O(Nmd^2 + mN^2)$.

For Later Reference... For later reference in chapter 3, we supply two more procedures. Procedure CM_MTTA() only computes the mean time to absorption $\mathbb{E}_{s'}[H]$ and procedure CM_MTABA() only computes the mean time spent in set A before absorption: $\mathbb{E}_{s'}[H_A]$.

Procedure `CM_MTTA`$(Q_1, \dots, Q_m; s'_1, \dots, s'_m; A_1, \dots, A_m; \epsilon)$

Procedure `CM_MTABA`$(Q_1, \dots, Q_m; s'_1, \dots, s'_m; A_1, \dots, A_m; \epsilon)$

2.4 Notation and Important Formulas

Marginal Absorbing Markov Chains $i \in \{1, \ldots, m\}$:

U_i	absorbing Markov chain
E_i	state space of U_i
$S_i \subset E_i$	set of absorbing states of U_i
$s_i' \in E_i \setminus S_i$	starting state of U_i
Q_i	generator matrix of U_i
$q_i \geq \max_{j \in E_i}\{\lvert Q_i(j,j)\rvert\}$	uniformisation rate of U_i
$P_i := I + 1/q_i Q_i$	(uniformised) one-step transition matrix
$\nu_i(n) = \nu_i(0)P_i^n = \nu_i(n-1)P_i$, for $n \geq 1$	distribution of the uniformised CTMC U_i at the $n-$th step, with the initial distribution $\nu_i(0)$
$\mathsf{A}_i \subseteq E_i$	some subset of E_i
$A_i \subseteq E_i$	some subset of E_i
$\nu_i(n)[\mathsf{A}_i] := \sum_{x \in \mathsf{A}_i} \nu_i(n)(x)$	aggregated state probability at the $n-$th step

Absorbing Joint Markov Chain

$U = (U_1, \ldots, U_m)$	joint absorbing Markov chain
$E := \times_{i=1}^m E_i$	state space of U
$S := \times_{i=1}^m S_i$	set of absorbing states of U
$Q = \oplus_{i=1}^m Q_i$	generator matrix of U
$q := q_1 + \cdots + q_m$	uniformisation rate of U
$P = I + \frac{1}{q}Q$	(uniformised) one-step transition matrix of U
$\nu(n) = \nu(0)P^n = \nu(n-1)P$, for $n \geq 1$	distribution of the uniformised CTMC U at the $n-$th step, with the initial distribution $\nu(0)$
$\mathsf{A} := \times_{i=1}^m \mathsf{A}_i \subseteq E$	a certain subset of E
$A := \times_{i=1}^m A_i \subseteq E$	a certain subset of E
$\nu(n)[\mathsf{A}] = \sum_{x \in \mathsf{A}} \nu(n)(x)$	aggregated state probability of the (uniformised) CTMC U at the $n-$th step

Important Formulas for the Absorbing Joint Markov Chain U

$\nu(n)[\mathsf{A}] = (\nu_1[\mathsf{A}_1] \star \cdots \star \nu_m[\mathsf{A}_m])(n)$	
$\mathbb{E}[H_A] = \sum_{n=0}^{\infty} \frac{1}{q}\nu(n)[A]$, for $A \cap S = \emptyset$	expected time that U spends in A before absorption
$\mathbb{E}[H] = \sum_{n=0}^{\infty} \frac{1}{q}(1 - \nu(n)[S])$	expected time until absorption of U

Random Variables

H	r.v. for the time until absorption of U
H_A	r.v. for the time that U spends in the set A before absorption.

3 Application to Markovian Process Algebra Models

In the preceding chapter we introduced a new method to compute the mean time to absorption and the mean time spent in a certain set A of a CTMC which is the joint process of a number of marginal absorbing CTMCs. In this chapter we apply this method to compute steady state probabilities of models specified in the Markov process algebra language PEPA.

In PEPA components can be defined whose behaviour is characterised by activities with exponentially distributed durations, and hence the stochastic process underlying such a component is a continuous time Markov chain. Among other things PEPA allows to combine several components via a cooperation combinator such that the resulting composite component is again a continuous time Markov chain.

The basic idea underlying this chapter goes back to Bohnenkamp who in [Bohn02] considered a special class of processes specified in the language $\mathcal{YAWN}$, where there exist only global synchronisations, i.e. every component is involved in every synchronisation. Bohnenkamp was able to derive local steady state probabilities for this class of processes. Instead of $\mathcal{YAWN}-$processes we consider processes specified in PEPA. In similarity to the work of Bohnenkamp we restrict ourselves to processes which possess only global synchronising activities. We succeed in deriving not only local but also global steady state probabilities of the composite PEPA process.

The chapter is organised as follows:

In section 3.1 we give a short overview of the Markov process algebra PEPA which was introduced by Hillston in [Hill96].

Section 3.2 gives a short compilation of some key results from the theory of semi-regenerative processes.

In section 3.3 we compute certain steady state probabilities for a class of PEPA processes by at first decomposing the PEPA component into a number of absorbing Markov chains which themselves consist of several marginal absorbing Markov chains. Cumulative measure of the joint absorbing CTMCs are then computed with the method introduced in chapter 2. Afterwards the results are combined by means of a relation from the theory of semi-regenerative processes.

Section 3.4 gathers the notations used frequently throughout this chapter.

3.1 PEPA

The language defined by the following grammar is the set of all possible PEPA expressions.

$$C \;:=\; (\alpha, r).C \mid C + D \mid C/L \mid C \underset{L}{\bowtie} D \mid M. \tag{3.1}$$

Each expression from this language describes a process whose behaviour is determined by the following rules:

Prefix: The process $(\alpha, r).C$ executes the activity (α, r) – which possesses the action type α and an exponentially distributed duration with rate r, $r \in \mathbb{R}_{>0}$ – and afterwards behaves like C. It is also possible to leave the rate r unspecified, in which case we use the symbol $\top$.

Choice: In a process $C + D$ all currently enabled activities in C and D are involved in a race condition. The activity to win this race is executed. Due to the memoryless property of the exponential distribution all other activities are reset. If for example in the process $(\alpha, r).C + (\beta, v).D$ the activity (α, r) wins, then afterwards the process behaves like $C + (\beta, v).D$.

Cooperation: $C \underset{L}{\bowtie} D$ denotes the situation where the components C and D must synchronise over activities which are of an action type contained in the synchronisation set L. Activities of this kind are called shared activities. C and D evolve independently of each other (i.e. in parallel) until the first of the two components, say C, *reaches* a shared activity. From this time instant on this shared activity becomes blocked in C until also D reaches a shared activity of the same action type. If this happens the shared activity is executed simultaneously by C and D. The rate of the shared activity is determined by the smallest rate of all activities involved in the synchronisation. If one or more activities involved in the synchronisation possess an unspecified rate, then these activities can be regarded as passive – they are not taken into account when determining the rate of the shared activity.

Hiding: C/L has the meaning that the action type of all activities in C which are of an action type contained in L are hidden to the *outside* of C. Hidden activities are not executed in cooperation with other components. Nevertheless, inside of component C these actions are still visible.

Constant: Constants are components whose meaning is given by a defining equation. For two constants M and E, $M \stackrel{def}{=} E$ assigns M the behaviour of component E.

Definition 10. The apparent rate $r_\alpha(C)$ of an activity of action type α in component C is defined by

$$\begin{aligned}
r_\alpha((\beta, r).C) &= \begin{cases} r, & \text{if } \alpha = \beta \\ 0, & \text{if } \alpha \neq \beta \end{cases} \\
r_\alpha(C/L) &= \begin{cases} r_\alpha(C), & \text{if } \alpha \notin L \\ 0, & \text{if } \alpha \in L \end{cases} \\
r_\alpha(C + D) &= r_\alpha(C) + r_\alpha(D) \\
r_\alpha(C \underset{L}{\bowtie} D) &= \begin{cases} r_\alpha(C) + r_\alpha(D), & \text{if } \alpha \notin L \\ \min\{r_\alpha(C), r_\alpha(D)\}, & \text{if } \alpha \in L \end{cases}
\end{aligned}$$

The apparent rate $r_\alpha(C)$ returns the sum of the rates of all activities of type α which are currently enabled in C. In other words, $r_\alpha(C)$ is the overall rate at which activities of type α are currently observed in C.

The semantics for the PEPA language is given by the structured operational semantics rules (SOS-rules) in figure 3.1. A rule of the form $\frac{B}{C}A$ is read as: Given A, B implies C. If A is missing, then there is no precondition. If B is missing, then C holds, provided A.

Prefix :

$$\frac{}{(\alpha, r).C \xrightarrow{(\alpha,r)} C}$$

Choice :

$$\frac{C \xrightarrow{(\alpha,r)} C'}{C + D \xrightarrow{(\alpha,r)} C'} \qquad \frac{D \xrightarrow{(\alpha,r)} D'}{C + D \xrightarrow{(\alpha,r)} D'}$$

Cooperation :

$$\frac{C \xrightarrow{(\alpha,r)} C'}{C \underset{L}{\bowtie} D \xrightarrow{(\alpha,r)} C' \underset{L}{\bowtie} D}(\alpha \notin L) \qquad \frac{D \xrightarrow{(\alpha,r)} D'}{C \underset{L}{\bowtie} D \xrightarrow{(\alpha,r)} C \underset{L}{\bowtie} D'}(\alpha \notin L)$$

$$\frac{C \xrightarrow{(\alpha,r_1)} C' \quad D \xrightarrow{(\alpha,r_2)} D'}{C \underset{L}{\bowtie} D \xrightarrow{(\alpha,R)} C' \underset{L}{\bowtie} D'}(\alpha \in L) \qquad \text{where } R = \frac{r_1}{r_\alpha(C)} \frac{r_2}{r_\alpha(D)} \min(r_\alpha(C), r_\alpha(D))$$

Hiding :

$$\frac{C \xrightarrow{(\alpha,r)} C'}{C/L \xrightarrow{(\alpha,r)} C'/L}(\alpha \notin L) \qquad \frac{C \xrightarrow{(\alpha,r)} P'}{C/L \xrightarrow{(\tau,r)} C'/L}(\alpha \in L)$$

Constant :

$$\frac{C \xrightarrow{(\alpha,r)} C'}{D \xrightarrow{(\alpha,r)} C'}(D \overset{def}{=} C)$$

Figure 3.1: SOS-rules of PEPA

Definition 11. Let C be a PEPA process.

- The one-step derivative set $ds^{(1)}(C)$ of C contains all the processes C' which can be reached by applying the SOS-rules to C.
- The k-step derivative set $ds^{(k)}(C)$ of C, $k \geq 2$, is given by $ds^{(k)}(C) = \bigcup_{C' \in ds^{(k-1)}(C)} ds^{(1)}(C')$.
- The derivative set $ds(C)$ of C is given by $ds(C) = \bigcup_{k=1}^{\infty} ds^{(k)}(C)$.

Starting with a process C the *derivation graph* can be constructed by successive application of the SOS-rules. The elements of the derivative set $ds(C)$ form the nodes of this graph. There exists an edge from node C_1 to C_2, $C_1, C_2 \in ds(C)$, iff $C_2 \in ds^{(1)}(C_1)$, i.e. iff there exists an activity (α, r) which causes C_1 to evolve into C_2. Edges are labelled by the corresponding activities. Note that the derivation graph might be a multigraph since C_1 may evolve into C_2 through different activities.

A CTMC is obtained from the derivation graph by

1. considering nodes as states
2. abstraction from action types within activities
3. amalgamation of multiedges into a single edge, where the activity rates are summed up.

Definition 12. A PEPA component C is said to be cyclic, or irreducible, if $C \in ds(C')$ for all $C' \in ds(C)$.

The importance of cyclic PEPA components arises from the fact that the CTMC associated with this component is irreducible if and only if the PEPA component is cyclic. Since PEPA components define only finite state CTMCs, irreducibility implies positive recurrence of that CTMC.

In [Hill96] it is shown that a necessary condition for a PEPA component to be cyclic is that all choices must occur within cooperating PEPA components. Thus, every cyclic PEPA component can be constructed out of the following grammar.

$$\text{sequential components} \qquad R := (\alpha, r).R \mid R + R \mid M \tag{3.2}$$

$$\text{model components} \qquad C := R \mid C/L \mid C \underset{L}{\bowtie} C. \tag{3.3}$$

3.2 Elements of the Theory of Semi-Regenerative Processes

This section gives a compilation of some key results of the theory of semi-regenerative processes. The contents of this section is taken from [Cinl75]. A process $Z = (Z_t)_{t \geq 0}$ is called *regenerative* if there exist certain random times $T_0, T_1, \ldots$ at which the future of Z becomes a probabilistic replica of the process itself. If in addition the future development of Z after time instants of regeneration depends also on the state of a Markovian renewal process embedded in Z at time instants of regeneration, then Z is called a semi-regenerative process.

Definition 13. Let $Z = (Z_t)_{t \geq 0}$ be a stochastic process with topological state space E and the càdlàg[1] property. Z is called semi-regenerative if there exists an embedded Markov renewal process (X, T) with infinite lifetime, with

- T_n is a stopping time of Z for each $n \in \mathbb{N}_0$
- X_n is determined by the history $\{Z_u : u \leq t\}$ for all $n \in \mathbb{N}_0$
- For every $n \in \mathbb{N}_0$, $k \geq 1$ and every positive function $f : F^k \to \mathbb{R}_{>0}$

$$E(f(Z_{T_n+t_1}, \ldots, Z_{T_n+t_k}) | Z_u : u \leq T_n) = E(f(Z_{t_1}, \ldots, Z_{t_k}) | X_n),$$

for $0 \leq t_1 < t_2 < \cdots < t_k$.

[1] càdlàg: continue à droite, limite à gauche. A function with the càdlàg property is right-continuous and has left limits everywhere.

The kernel of a semi-regenerative process Z with embedded Markov renewal process (X, T) is defined as the set of conditional probabilities

$$K_t(s, A) = P_s(Z_t \in A, T_1 > t) \tag{3.4}$$

for $s \in S$, $t \geq 0$ and $A \subseteq E$, where S is the state space of X.

The family $Q = \{Q_{ij}(t) : i, j \in S, t \geq 0\}$, with

$$Q_{ij}(t) = P(X_{n+1} = j, T_{n+1} - T_n \leq t | X_n = i), \tag{3.5}$$

defines the semi-Markov kernel of (X, T).

With the recursive definition for $n \geq 0$

$$Q^{n+1}(s, k, t) = \sum_{j \in S} \int_0^t Q(s, j, \xi) dQ^n(j, k, t - \xi) \tag{3.6}$$

the following sum defines the renewal function of (X, T):

$$R_t(i, j) = \sum_{n=0}^{\infty} Q_{ij}^n(t). \tag{3.7}$$

The family $R = \{R_t(i, j) : i, j \in S, t \geq 0\}$ is called the Markov renewal kernel of (X, T).

We cite the following two key results from the theory of semi-regenerative processes. The proofs for both theorems can be found in [Cinl75].

Theorem 14. *For a semi-regenerative process $Z = (Z_t)_{t \geq 0}$ with state space F and embedded Markov renewal process (X, T) let Q be the semi-Markov kernel and R be the Markov renewal kernel of (X, T). Then, for any subset $A \subseteq E$*

$$P_s(Z_t \in A) = \sum_{j \in S} \int_0^t R_\xi(s, j) dK_{t-\xi}(j, A). \tag{3.8}$$

Theorem 15. *Let $Z = (Z_t)_{t \geq 0}$ be a semi-regenerative process with state space F and embedded Markov renewal process (X, T). Suppose that (X, T) is irreducible and aperiodic recurrent. Furthermore, let $\pi = (\pi(0), \pi(1), \ldots)$ be an invariant measure for X and define the column vector $m = (m_0, m_1, \ldots)$ by $m_s = \mathbb{E}_s[T_1]$. Provided that the function $t \to K_t(s, A)$ is Riemann integrable for each $s \in S$ the following assertion holds:*

$$\lim_{t \to \infty} P_s(Z_t \in A) = \frac{1}{\pi m} \sum_{h \in S} \pi(h) \int_0^\infty K_t(h, A) dt. \tag{3.9}$$

3.3 Steady State Probabilities for a Class of PEPA Models

The aim of this section is to provide a technique to compute steady state probabilities for a restricted class of PEPA components, or more precisely steady state probabilities of the underlying Markov chains. In order to do so, we at first formulate the underlying Markov chain as a semi-regenerative process which is defined by a family of stopping times, an embedded discrete time Markov chain and its kernel.

In section 3.3.1 we list the restrictions which we impose on PEPA models to be solved by our method.

Let C be a such PEPA component and assume C is the composition of several sequential components. In section 3.3.2 we reformulate the CTMC Y underlying this composite component as a semi-regenerative process.

The first step to determine steady state probabilities of a semi-regenerative process Y, is to compute the steady state distribution of the embedded DTMC. This requires knowledge of the transition matrix P of this embedded DTMC. Section 3.3.3 shows how this transition matrix can be gained from the sequential components.

The second step to determine the steady state probabilities is to compute certain elements of the Kernel of Y. These quantities can be expressed by related quantities of the sequential components. This is done in sections 3.3.4 and 3.3.5, where in the latter section the results about cumulative measures of absorbing joint Markov chains come into play.

Section 3.3.6 summarises the results in algorithmic form.

3.3.1 Requirements and the Target Quantity

We now consider PEPA processes which possess the following three properties:

(a) the process is cyclic

(b) all concurrent components synchronise over the same synchronisation set L

(c) there is no choice between synchronising and non-synchronising activities, i.e. language terms of the form $(\alpha, r_1).C_1 + (\beta, r_2).C_2$, with $\alpha \in L$ and $\beta \notin L$ are prohibited

Condition (a) assures that the CTMC underlying the PEPA component, say C, is ergodic, and hence possesses a unique stationary distribution. Furthermore, this condition implies that the PEPA component C can be expressed as the cooperation of a number, say m, of sequential PEPA components $C_1, \ldots, C_m$. Together with condition (b) C is of the form

$$C = C_1 \underset{L}{\bowtie} C_2 \underset{L}{\bowtie} \cdots \underset{L}{\bowtie} C_m. \tag{3.10}$$

For simplicity we assume that C starts immediately after some shared activity. The m components start to evolve independently of each other. Every component which reaches an activity that requires synchronisation has to wait until all of the remaining components have reached an activity of that same type. The fact that components *must* wait is a consequence of condition (c). Once all m components are ready to synchronise, the shared activity is executed. After that the m components again start to evolve independently of each other.

Our target quantity is the steady state probability that the CTMC, say Y, underlying the composite component C is in some set A, i.e.

$$\mathbb{P}(Y \in A). \tag{3.11}$$

We require that

- either A contains synchronising states only, i.e. states which possess outgoing synchronising transitions
- or A is the cross product of sets $A_1, \ldots, A_m$ which are subsets of the state spaces of the Markov chains underlying the m sequential PEPA components, and A contains no synchronising states.

3.3.2 The Underlying CTMC as a Semi-Regenerative Process

Let the PEPA process C be the composition of m sequential PEPA components $C_1, \ldots, C_m$, where C fulfils the requirements stated in the preceding section. Let Y be the CTMC which underlies the composite component C.

In this section we aim at formulating the CTMC Y as a semi-regenerative process. Since the component C is the result of the cooperation of the m components C_i, $i = 1 \ldots m$, every state of Y is an $m-$tupel. The projection of Y onto its $i-$th component yields the marginal process Y_i.

In the following we consider three types of stochastic processes:

(a) The CTMCs underlying the PEPA components (i.e. in isolation).
(b) The CTMC $Y = (Y_1, \ldots, Y_m)$ underlying the composite PEPA component C.
(c) The marginal processes Y_i, $i = 1 \ldots m$, which are projections of the CTMC Y underlying the composite PEPA component.

(a) The Component CTMCs in Isolation

Every PEPA component C_i can be viewed as the specification of a CTMC in isolation. That means we pretend that the component under consideration is not part of a composite process. Ignoring synchronisation of C_i with other components implies that in C_i a waiting condition can not occur. The CTMC underlying a single sequential PEPA component C_i is given by its starting state and the generator matrix G_i. Both can be gained from the specification of the PEPA component. Note that the G_i, $i = 1 \ldots m$, might contain unspecified transition rates. Denote the state space of this CTMC by E_i.

(b) The CTMC Y Underlying the Composite PEPA Component

Let Y be the CTMC underlying the composite PEPA component C with state space E. Let S denote the set of synchronising states and S' the set of states which can be occupied immediately after a synchronisation.

Define the family of random variables $T = \{T_0, T_1, T_2, \ldots\}$, where T_k, $k \geq 1$, denotes the time instant of the $k-$th visit of Y to a state contained in S or S'. T_0 can be viewed as the initial time instant of Y. For simplicity we assume $T_0 = 0$. This comes w.l.o.g. since for the semi-regenerative process Y, $\mathbb{P}(Y(t) \in A | Y(T_0) = x) = \mathbb{P}(Y(t - T_0) \in A | Y(0) = x)$, for any $A \subseteq E$ and $x \in S \cup S'$.

Define the DTMC $X = (X(n))_{n \in \mathbb{N}}$ as the discrete time Markov chain embedded in Y at time instants of entering states contained in $S \cup S'$, i.e.

$$X(n) = Y(T_n). \tag{3.12}$$

The pair (X, T) is a Markov renewal process embedded in Y. The kernel elements of the semi-regenerative process Y, with respect to the set A, are given by

$$K_t(s, A) := \mathbb{P}_s(Y(t) \in A, T_1 > t), \text{ for } s \in S \cup S'. \tag{3.13}$$

By theorem 15 the following values must be known in order to determine the steady state probability $\mathbb{P}(Y \in A)$.

- the steady state distribution ν of the embedded DTMC X
- the mean times $\mathbb{E}_s[T_1]$, for all $s \in S \cup S'$
- the values $\int_0^\infty K_t(s, A)dt$, with $s \in S \cup S'$.

(c) The Marginal Processes Y_i

Y is the Markov chain underlying the composite PEPA component C. Since C is the composition of m concurrent sequential components, Y can be expressed as an m-dimensional process $Y = (Y_1, \cdots, Y_m)$, where Y_i is the projection of Y onto its $i-$th component, $i = 1 \dots m$. Assume that during the evolution of the composite PEPA component C, we only observe the behaviour of the component C_i. Then, this behaviour is described by the the process Y_i. We call the processes Y_i, $i = 1 \dots m$, the marginal processes of Y.

The state space of Y_i is E_i (same as for the corresponding CTMC in isolation). With $S_i \subseteq E_i$ we denote the set of synchronising states, i.e. states which possess outgoing synchronising transitions. With $S_i' \subseteq E_i$ we denote the set of states which may be occupied immediately after a synchronising state has been left. Note that $S \cap S' \neq \emptyset$ is possible.

Y_i differs from the corresponding CTMC in isolation in two points. On the one hand, in synchronising states of Y_i waiting times are inserted. These waiting times result from Y_i waiting for the other marginal processes to become ready to synchronise. On the other hand, the duration of the synchronising transition which follows a waiting time is the same for all marginal processes. Thus, the marginal processes are not independent of each other.

For each CTMC Y_i define the stopping time $T_1^{Y_i} \geq T_0$ as follows

- if $Y(T_0) \in S' \setminus S$: $T_1^{Y_i}$ is the time instant $\geq T_0$ where Y_i enters a state contained in S_i for the first time.
- if $Y(T_0) \in S$: $T_1^{Y_i}$ is the time instant $\geq T_0$ where Y_i enters a state contained in S_i' for the first time.

Relating Y to the Marginal Processes Y_i, $i = 1 \dots m$

Obviously, we have

$$E \subseteq \times_{i=1}^{m} E_i, \quad \text{state space} \tag{3.14}$$

$$S \subseteq \times_{i=1}^{m} S_i, \quad \text{set of synchronising states} \tag{3.15}$$

$$S' \subseteq \times_{i=1}^{m} S_i'. \quad \text{set of states occupied after a synchronisation} \tag{3.16}$$

For a starting state $s' \in S' \setminus S$, T_1 is the time instant, where Y reaches the next embedded state (i.e. a state $\in S \cup S'$). This next embedded state must be a synchronising state (i.e. $\in S$). This synchronising state is reached iff all of the marginal processes Y_i, $i = 1 \dots m$, have reached their next synchronising state. Thus, T_1 is the maximum of the $T_1^{Y_i}$, $i = 1 \dots m$.

$$T_1 = \max_{i \in \{1,\dots,m\}} \{T_1^{Y_i}\}. \tag{3.17}$$

For a starting state $s \in S$, i.e. Y starts in a synchronising state, T_1 is the end of that first synchronising transition. In this case, obviously $T_1 = T_1^{Y_i}$, $i = 1 \dots m$. This relationship between stopping times of Y and its marginal processes Y_i, $i = 1 \dots m$, is illustrated in the Figures 3.2 – 3.4 (for $m = 3$).

For the DTMC $X = (X(n))_{n \in \mathbb{N}_0}$ embedded in Y at time instants of entering states contained in $S \cup S'$ we have

$$X(n) = Y(T_n) = (Y_1(T_n), \dots, Y_m(T_n)). \tag{3.18}$$

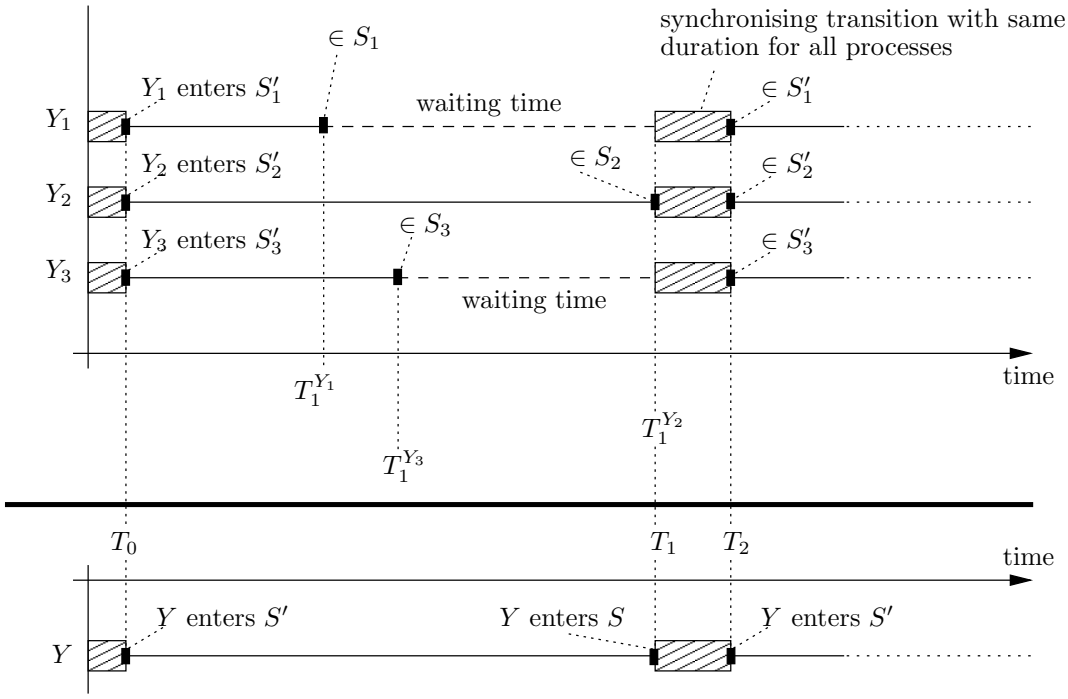


Figure 3.2: *Relationship between stopping times of Y and the stopping times of the marginal processes. Here, the case $Y(T_0) \in S' \setminus S$ is illustrated. Then $T_1^{Y_i}$ is the time instant of entering the next state $\in S_i$, $i = 1 \dots m$. It is seen that $T_1 = \max_{i\in\{1,\dots,m\}}\{T_1^{Y_i}\}$.*

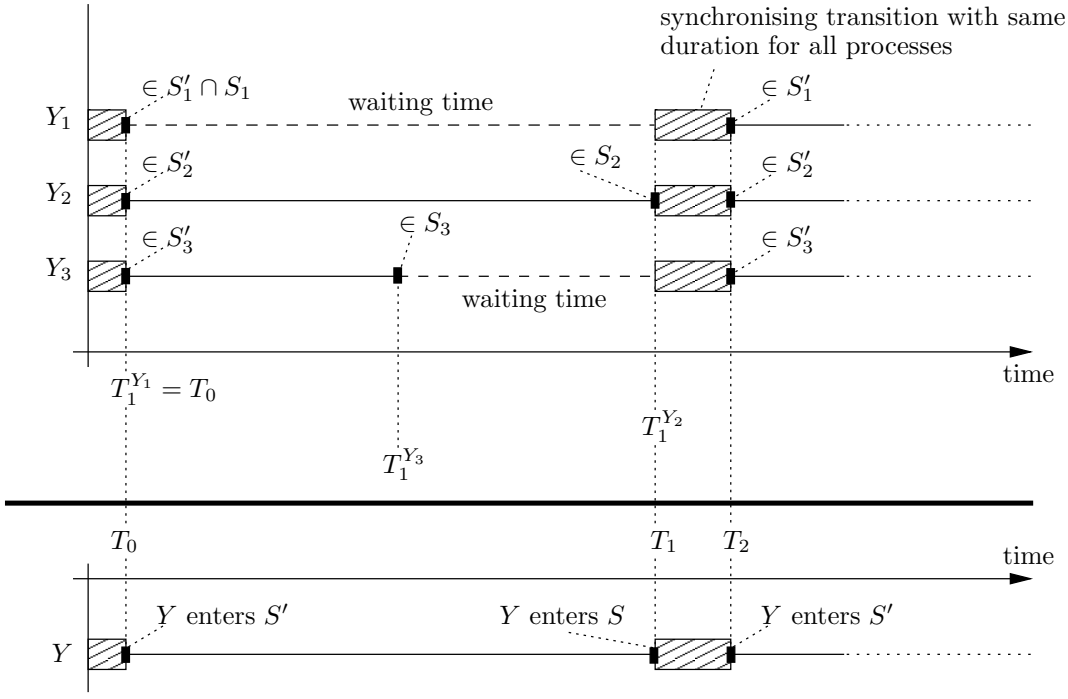
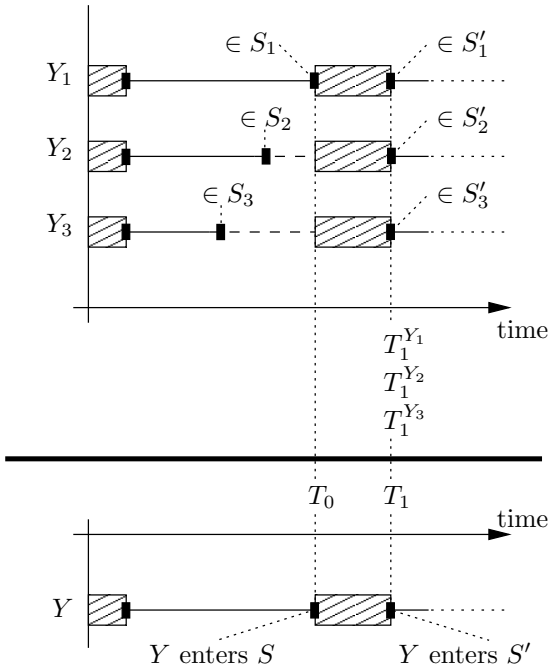
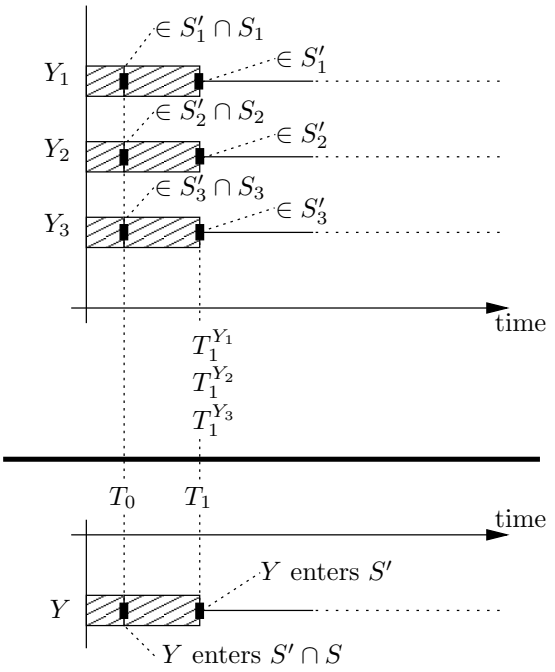


Figure 3.3: *Again, the case $Y(T_0) \in S' \setminus S$ is illustrated. Since in this example the marginal process Y_1 is already in a synchronising state at time T_0 we have $T_1^{Y_1} = T_0$. And, of course, $T_1 = \max_{i\in\{1,\dots,m\}}\{T_1^{Y_i}\}$.*



(a) Here, the case $Y(T_0) \in S \setminus S'$ is illustrated. The duration of the synchronisation is the same for all marginal processes, and hence $T_1 = T_1^{Y_i}$, $i = 1 \dots m$.



(b) Here, the case $Y(T_0) \in S \cap S'$ is illustrated. The duration of the synchronisation is the same for all marginal processes, and hence $T_1 = T_1^{Y_i}$, $i = 1 \dots m$. It is seen that states which are contained in both S and S' are always treated as synchronising states.

Figure 3.4: *Stopping times for the case $Y(T_0) \in S$.*

3.3.3 Solving the Embedded DTMC X

X is an m-dimensional DTMC, and hence it can be written as $X = (X_1, \dots, X_m)$, where X_i, $i = 1 \dots m$, is the projection of X onto its $i-$th component. The solution of $X = (X_1, \dots, X_m)$ is given by the probability vector π, which (uniquely) satisfies $\pi P = \pi$. Were the projections X_i, $i = 1 \dots m$, independent of each other and discrete time Markov chains, π could be obtained from the marginal steady state probabilities in a compositional way. But unfortunately, in general neither of the two properties is given.

The Projections X_i are not Independent. Consider the two components $D = ((\alpha, r_1) + (\beta, r_2)).D'$ and $F = ((\alpha, r_3) + (\beta, r_4)).F'$ and the composite component

$$C = D \underset{\{\alpha,\beta\}}{\bowtie} F. \tag{3.19}$$

Independence would imply that every pair from $\{(\alpha, r_1), (\beta, r_2)\} \times \{(\alpha, r_3), (\beta, r_4)\}$ could be used to derive a valid shared transition. But e.g. the two activities (α, r_1) and (β, r_4) can never form a shared activity, since their types do not match.

In this example we have $m = 2$, i.e. we consider the CTMC $Y = (Y_1, Y_2)$ and the embedded DTMC $X = (X_1, X_2)$. The transition from D to D' corresponds to the transition from some state $\in S_1$ to some state $\in S_1'$ in the projection X_1 (corresponding to the component D). Analogously, the transition from F to F' corresponds to the transition from some state $\in S_2$ to some state $\in S_2'$ in the projection X_2 (corresponding to the component F). Thus, the projections X_1 and X_2 are not independent of each other.

The Projections X_i are not DTMCs. Consider the components $D = (\alpha, r_1).(\alpha, r_2).D$ and $F = (\alpha, r_3).(\beta, r_4).(\alpha, r_5).F$ and the composite component

$$C = D \underset{\{\alpha\}}{\bowtie} F. \tag{3.20}$$

Now, interpret derivations of components as states. In isolation the components D and F asynchronously travel through the following sequence of transitions

$$D \to \overbrace{(\alpha, r_2).D}^{a} \to D \tag{3.21}$$

$$F \to \underbrace{(\beta, r_4).(\alpha, r_5).F}_{y} \to \underbrace{(\alpha, r_5).F}_{z} \to F. \tag{3.22}$$

The cooperation combinator forces the two components to synchronise in the shared states (D, F) and (a, z).

$$D \to \overbrace{(\alpha, r_2).D}^{a} \to D \tag{3.23}$$

$$F \to \underbrace{(\beta, r_4).(\alpha, r_5).F}_{y} \to \underbrace{(\alpha, r_5).F}_{z} \to F. \tag{3.24}$$

Since $m = 2$, i.e. we consider the CTMC $Y = (Y_1, Y_2)$ and the embedded DTMC $X = (X_1, X_2)$. (D, F) and (a, z) are synchronising states, i.e. $S = \{(D, F), (a, z)\}$, and (a, y) is a state to be occupied immediately after a synchronising state, i.e. $S' = \{(a, z)\}$. The CTMC Y is in this example given by the transition system

$$(D, F) \overset{min\{r_1, r_3\}}{\longrightarrow} (a, y) \overset{r_4}{\longrightarrow} (a, z) \overset{min\{r_2, r_5\}}{\longrightarrow} \text{to } (D, F). \tag{3.25}$$

Since all states belong to the set of embedded states $S \cup S'$ the embedded Markov chain X is given by

$$(D, F) \overset{1}{\longrightarrow} (a, y) \overset{1}{\longrightarrow} (a, z) \overset{1}{\longrightarrow} \text{to } (D, F), \tag{3.26}$$

where of course the labels of arrows are one-step transition probabilities. For the projection X_1 we obtain the sequence of transitions (equipped with transition probabilities)

$$D \overset{1}{\longrightarrow} a \overset{1}{\longrightarrow} a \overset{1}{\longrightarrow} \text{to } D. \tag{3.27}$$

Without additional information this sequence can not be described by a discrete time Markov chain.

Solving X. In order to solve the DTMC X, its transition matrix $P = (\mathbb{P}(X(n+1) = y | X(n) = x))_{x,y \in S \cup S'}$ has to be constructed. We don't elaborate too much on how this is done but provide the rough idea. First, for a PEPA component D and a state x of the underlying Markov chain, let $D(x)$ be the derivation of D which corresponds to the state x. Now, consider the two rules:

- Suppose $s = (s_1, \ldots, s_m)$ is a synchronising state (i.e $s \in S$). Then the transition probabilities to the next states (which in any case are embedded states) are determined by evaluating $C_1(s_1) \underset{L}{\bowtie} \cdots \underset{L}{\bowtie} C_m(s_m)$. Note that by evaluation of this expression we also obtain the embedded successor states. These are the states to be occupied immediately after the synchronising state s. It is possible that some (or all) of these embedded successor states are synchronising states at the same time.

- Suppose $s' = (s'_1, \ldots, s'_m)$ is a state to be occupied immediately after a synchronisation and s' is not a synchronising state itself (i.e. $s' \in S' \backslash S$). Since the behaviour of component C_i from state $C_i(s'_i)$ to any next synchronising state is independent of the other components, the transition probability from s' to a synchronising state, say $s = (s_1, \ldots, s_m)$, is simply the product $\prod_{i=1}^{m} \mathbb{P}(Y_i(T_1^{Y_i}) = s_i | Y_i(T_0) = s'_i)$. For every s'_i the transition probabilities can be computed from the specification of the component C_i in isolation.

Starting with some (valid) state s' it is clear that with the above two rules the set of embedded states $S \cup S'$, as well as the transition probabilities $\mathbb{P}(X(n+1) = y | X(n) = x)$, $x, y \in S \cup S'$ can be computed.

One might wonder whether the explicit construction of P is contrary to our initial wish to circumvent the state space explosion problem when solving MPA models. To answer this question, notice that the dimension of P equals the number of embedded states $S \cup S'$. This number can be small, even if the state space E of the CTMC Y (underlying the composite PEPA component C) is huge.

3.3.4 Determining $\int_0^\infty K_t(s, A)dt$ and $\mathbb{E}_s[T_1]$, for $s \in S$

Under the condition that Y starts at time $T_0 = 0$ in the synchronising state $s \in S$, the time instant T_1 of entering the next embedded state is just the time instant of leaving the state s, and hence T_1 is the state holding time in s. This yields

$$K_t(s, A) = \mathbb{P}_s(Y(t) \in A, T_1 > t) = \begin{cases} \mathbb{P}_s(T_1 > t), & \text{if } s \in A \\ 0, & \text{if } s \notin A \end{cases} \tag{3.28}$$

$\mathbb{P}_s(T_1 > t)$ is the complementary cumulative probability distribution of the state holding time in the synchronising state s. This state holding time is exponentially distributed. Its rate $\lambda(s)$ can by gained by applying the apparent rate function r to $C(s)$ – the derivative of the composite PEPA component C corresponding to the state s – and all action types $\alpha \in L$ which determine synchronising activities, i.e.

$$\lambda(s) = \sum_{\alpha \in S} r_\alpha(C(s)). \tag{3.29}$$

Since T_1 is the state holding time of state s, it follows that

$$\mathbb{E}_s[T_1] = \frac{1}{\lambda(s)}. \tag{3.30}$$

With

$$\mathbb{P}_s(T_1 > t) = e^{-\lambda(s)t}, \tag{3.31}$$

we obtain

$$\int_0^\infty K_t(s, A)dt = \begin{cases} \frac{1}{\lambda(s)}, & \text{if } s \in A \\ 0, & \text{if } s \notin A \end{cases}. \tag{3.32}$$

3.3.5 Determining $\int_0^\infty K_t(s', A)dt$ and $\mathbb{E}_{s'}[T_1]$, for $s' \in S' \setminus S$

The computation of $\int_0^\infty K_t(s, A)dt$ and $\mathbb{E}_s[T_1]$, for states $s \in S$ has been treated in the previous section. Thus, here we only need to consider the computation of these two quantities for the remaining embedded states, i.e. for all $s' \in S' \setminus S$.

Definition 16. For a real valued random variable D and the time instant t define

$$t\,[D] = \begin{cases} t, & \text{if } t \leq D \\ D, & \text{if } t > D \end{cases} . \tag{3.33}$$

Then $Y(t\,[T_1])$ is a process which behaves like $Y(t)$ until $t = T_1$. For $t \geq T_1$ the process $Y(t\,[T_1])$ stays in the state that it occupied at the time instant T_1, i.e. once $Y(t\,[T_1])$ has reached its next embedded state, it stays there forever. Analogous considerations apply to the processes $Y_i\left(t\left[T_1^{Y_i}\right]\right)$, $i = 1 \ldots m$. Note, that for a global starting state $s' \in S' \setminus S$, i.e. $Y(T_0) = s$, the marginal process Y_ℓ is completely independent of the other marginal processes until it reaches the next marginal embedded state (which must be a synchronising state). Hence, the $Y_i\left(t\left[T_1^{Y_i}\right]\right)$, $i = 1 \ldots m$, are independent of each other.

The following corollary states that the kernel element $K_t(s', A)$ is the probability that the process $Y(t\,[T_1])$ is in A at time t.

Corollary 17. *For $A \cap S = \emptyset$ and $s' \in S' \setminus S$ the following relation holds:*

$$K_t(s', A) = \mathbb{P}_{s'}(Y(t\,[T_1]) \in A). \tag{3.34}$$

Proof. Appendix A.2. □

Corollary 18. *For $i = 1 \ldots m$ and $Y(T_0) = s' \in S' \setminus S$ the following equivalence holds*

$$\{Y_i\,(t\,[T_1]) \in A_i\} \Longleftrightarrow \{Y_i\left(t\left[T_1^{Y_i}\right]\right) \in A_i\}. \tag{3.35}$$

Proof. Appendix A.3. □

The following theorem relates the kernel elements $K_t(s', A)$, $s' \in S'$, of the composite CTMC Y to quantities of the processes $Y_i\left(t\left[T_1^{Y_i}\right]\right)$, $i = 1 \ldots m$.

Theorem 19. *Let Y and Y_i, $1 \leq i \leq m$, be the CTMCs from above. Then for any $A = \times_{i=1}^{m} A_i$, with $A \cap S = \emptyset$, and $s' \in S' \setminus S$*

$$K_t(s', A) = \prod_{i=1}^{m} \mathbb{P}_{s'_i}\left(Y_i\left(t\left[T_1^{Y_i}\right]\right) \in A_i\right). \tag{3.36}$$

Proof. With corollary 17, we have

$$\begin{aligned} K_t(s', A) &= \mathbb{P}_{s'}(Y(t\,[T_1]) \in A) && (3.37)\\ &= \mathbb{P}_{s'}\left(Y_1\,(t\,[T_1]) \in A_1, \ldots, Y_m\,(t\,[T_1]) \in A_m\right) && (3.38) \end{aligned}$$

Reformulation of the last line according to corollary 18 yields

$$K_t(s', A) = \mathbb{P}_{s'}\left(Y_1\left(t\left[T_1^{Y_1}\right]\right) \in A_1, \ldots, Y_m\left(t\left[T_1^{Y_m}\right]\right) \in A_m\right). \tag{3.39}$$

The assertion results from the independence of the processes $Y_i\left(t\left[T_1^{Y_i}\right]\right)$, $i = 1 \ldots m$. □

The $Y_i\left(t\left[T_1^{Y_i}\right]\right)$, $i = 1 \ldots m$, are Absorbing Markov Chains

Note, that the Markov chain $Y_i\left(t\left[T_1^{Y_i}\right]\right)$ remains in the state which it enters at time $T_1^{Y_i}$. If at time T_0 we have $Y_i(T_0) = s_i' \in S_i'$, we know that $T_1^{Y_i}$ is the first time $\geq T_0$ at which Y_i enters the set S_i (cf. section 3.3.2, paragraph (c)). That means Y_i is an absorbing Markov chain with starting state $s_i' \in S_i'$ and the set of absorbing states S_i.

It may be that for Y_i the stopping time $T_1^{Y_i} = T_0$. This is the case if the marginal starting state s_i' is at the same time the next synchronising state, i.e. $s_i' \in S_i \cap S_i'$. Thus, the process $Y_i(t[T_1^{Y_i}])$ is immediately absorbed. This is not an exception to the above said but a remark.

Now, consider any trajectory of Y_ℓ, $\ell \in \{1, \ldots, m\}$, which starts in the initial state $s_\ell' \in S_\ell'$. Then, this trajectory evolves completely independent of the other processes Y_i, $i \in \{1, \ldots, m\} \setminus \{\ell\}$, until it reaches the next state contained in S_ℓ. Thus, the generator matrix Q_ℓ of the absorbing Markov chain $Y_\ell\left(t\left[T_1^{Y_\ell}\right]\right)$ can by gained from the generator matrix G_ℓ of the CTMC underlying the corresponding PEPA component C_ℓ in isolation by declaring all states of S_ℓ as absorbing states.

$Y\left(t\left[T_1\right]\right)$ is an Absorbing Joint Markov Chain

From theorem 19 and the fact that the Markov chains $Y_i\left(t\left[T_1^{Y_i}\right]\right)$, $i = 1 \ldots m$, are independent it is clear that $K_t(s', A)$ is the joint probability of the absorbing Markov chains $Y_i\left(t\left[T_1^{Y_i}\right]\right)$, $i = 1 \ldots m$.

Computing $\int_0^\infty K_t(s, A)dt$ and $\mathbb{E}_s[T_1]$, for $s' \in S' \setminus S$. If Y starts in a state $s' \in S' \setminus S$, T_1 is the time instant where Y enters a state contained in S for the first time. Thus, T_1 is the time to absorption of the absorbing Markov chain $Y(t[T_1])$. Hence, the mean time to absorption is given by

$$\mathbb{E}_{s'}[T_1] = \int_0^\infty 1 - \mathbb{P}_{s'}(Y(t[T_1]) \in S)dt. \tag{3.40}$$

Since $Y(t[T_1])$ is the joint CTMC of the absorbing CTMCs $Y_i(t[T_1^{Y_i}]$, $i = 1 \ldots m$, this mean time can be computed with the method introduced in chapter 2.

In order to compute $\int_0^\infty K_t(s', A)dt$, we make a case distinction:

- $A \cap S = \emptyset$: In this case we have $K_t(s', A) = \mathbb{P}_{s'}(Y(t[T_1]) \in A)$, and hence

$$\int_0^\infty K_t(s', A)dt = \int_0^\infty \mathbb{P}_{s'}(Y(t[T_1]) \in A)dt. \tag{3.41}$$

  $Y(t[T_1])$ is an absorbing Markov chain, and hence the above expression is the mean time this CTMC spends in A before absorption. $Y(t[T_1])$ is the joint CTMC of the absorbing CTMCs $Y_i(t[T_1^{Y_i}])$, $i = 1 \ldots m$, and $A \cap S = \emptyset$. Thus, the mean time spent in A before absorption can be computed with the method introduced in chapter 2

- $A \subseteq S$: In this case $\int_0^\infty K_t(s', A)dt = 0$. This is due to the fact that $\mathbb{P}_{s'}(Y(t) \in S, T_1 > t) = 0$.

3.3.6 Algorithm

Let $A \subseteq E$ be a subset of the state space of Y, with $A = \times_{i=1}^m A_i$, where $A_i \subset E_i$. For $A \cap S = \emptyset$ or $A \subseteq S$, the following algorithm (Procedure SteadyStateProb) shows a way to compute the

steady state probability $\mathbb{P}(Y \in A)$ of the CTMC Y which is the CTMC underlying the composite PEPA component C.

The procedures $CM\_MTTA(\cdot)$ and $CM\_MTABA(\cdot)$ which were introduced in section 2.3.2 are made use of in lines 9 and 11. These procedures compute the values of $\int_0^\infty K_t(s', A)dt$ and $E_{s'}[T_1]$, $s' \in S' \setminus S$, with the maximal absolute error of ϵ.

Procedure `SteadyStateProb`

`/* --------------- Solve embedded DTMC (section 3.3.3) ---------------  */`

1 construct the transition matrix P of X;
2 compute the steady state distribution π of X;

`/* ----- Compute` $\int_0^\infty K_t(s, A)dt$ `and` $\mathbb{E}_s[T_1]$`, for` $s \in S$ `(section 3.3.4) -----  */`

3 **for** $s \in S$ **do**
4 Compute $\int_0^\infty K_t(s, A)dt$;
5 Compute $\mathbb{E}_s[T_1]$;
6 **end**

`/* -- Compute` $\int_0^\infty K_t(s', A)dt$ `and` $\mathbb{E}_{s'}[T_1]$`, for` $s' \in S' \setminus S$ `(section 3.3.5) --  */`

7 for $i = 1 \ldots m$: construct Q_i by replacing all rows in G_i associated with states $\in S_i$ with zero-vector;

8 **for** $s' \in S' \setminus S$ **do**
9 $CM\_MTTA(Q_1, \ldots, Q_m, s'_1, \ldots, s'_m; A_1, \ldots, A_m; \epsilon)$; `/* compute` $\mathbb{E}_{s'}[T_1]$ `*/`
10 **if** $A \cap S = \emptyset$ **then**
11 $CM\_MTABA(Q_1, \ldots, Q_m, s'_1, \ldots, s'_m; A_1, \ldots, A_m; \epsilon)$; `/*` $\int_0^\infty K_t(s', A)dt$ `*/`
12 **else if** $A \subseteq S$ **then**
13 $\int_0^\infty K_t(s', A)dt = 0$;
14 **else**
15 output: set A is invalid;
16 **end**
17 **end**

`/* --------------------- Application of theorem 15---------------------  */`

18 Compute $\lim_{t\to\infty} P_s(Y(t) \in A) = \frac{\sum_{h\in S\cup S'} \pi(h) \int_0^\infty K_t(h,A)dt}{\sum_{h\in S\cup S'} \pi(h)\mathbb{E}_h[T_1]}$;

Time Complexity. The time complexity of this algorithm mainly depends on the complexity of the procedures $CM\_MTTA(\cdot)$, $CM\_MTABA(\cdot)$ respectively, and the number of embedded states, i.e. $|S \cup S'|$.

The complexity of $CM\_MTTA(\cdot)$ and $CM\_MTABA(\cdot)$ mainly depends on the second largest eigenvalue in modulus of the matrix $\oplus_{i=1}^m Q_i$ (cf. section 2.3.2) and the number m of sequential PEPA components.

Thus, it can be concluded that the above algorithm is suited to compute steady state probabilities, if the matrices Q_i are *good-natured*[2] and if the number of embedded states $S \cup S'$ is sufficiently small. In this context note that although the size of the state space of the CTMC Y is exponential in the number m of concurrent PEPA components, the number $|S \cup S'|$ can be small. This depends on the concrete model under consideration.

[2] In this context we will speak of the matrices Q_i, $i = 1 \ldots m$, as *good-natured* if their eigenvalues result in a small second largest eigenvalue in modulus of $\oplus_{i=1}^m Q_i$.

3.4 Notation and Important Formulas

PEPA Components

C_i, $i = 1 \dots m$	sequential PEPA components
$C = C_1 \underset{L}{\bowtie} \cdots \underset{L}{\bowtie} C_m$	composite PEPA component

Markov Chain Underlying the Composite PEPA Component C

$Y = (Y(t))_{t \in \mathbb{R}_{\geq 0}}$	CTMC underlying the PEPA component C
E	state space of Y
S	set of absorbing synchronising states of Y
S'	set of possible successors of synchronising states
T_n, $n = 0, 1, \dots$	T_n is the time instant of entering a state contained in $S \cup S'$ for the $n-$th time
$X = (X(n)) = (Y(T_n))$	embedded DTMC in Y
$A = \times_{i=1}^{m} A_i \subseteq E$	subset of the state space (for the A_i see below)
$K_t(x, A)$, $x \in S \cup S'$	$= \mathbb{P}_x(Y(t) \in A, T_1 > t)$, element of the kernel of Y

Marginal Processes of Y $i \in \{1, \dots, m\}$:

$Y_i = (Y_i(t))_{t \in \mathbb{R}_{\geq 0}}$	projection of Y onto its $i-$th component
E_i	state space of Y_i
S_i	set of absorbing synchronising states of Y_i
S_i'	set of possible successors of synchronising states
$T_1^{Y_i}$	$T_1^{Y_i}$ is the time instant where Y_i enters a state (a) contained in S_i for the first time provided that $Y(T_0) \in S' \backslash S$ (b) contained in S_i' for the first time provided that $Y(T_0) \in S$.
$A_i \subseteq E_i$	a certain subset of the marginal state space

CTMCs of components in isolation For the PEPA component C_i in isolation let G_i be the generator matrix of the underlying CTMC.

Relationships Between Y and the Y_i, $i = 1 \dots m$, (Sets and Stopping Times)

$Y = (Y_1, \dots, Y_m), \quad E \subseteq \times_{i=1}^{m} E_i, \quad S \subseteq \times_{i=1}^{m} S_i, \quad S' \subseteq \times_{i=1}^{m} S_i'$

$T_1 = \max_{i \in \{1,\dots,m\}} \{T_1^{Y_i}\}$, for $Y(T_0) \in S' \setminus S$

$T_1 = T_1^{Y_i}$, $i = 1 \dots m$, for $Y(T_0) \in S$

Relationships Between Y and the Y_i, $i = 1 \dots m$, (Kernel Elements)

Definition: $t\,[D] = \begin{cases} t, & \text{if } t \leq D \\ D, & \text{if } t > D \end{cases}$, where t is a continuous time parameter and D is a continuous random variable.

$K_t(s', A) = \prod_{i=1}^{m} \mathbb{P}_{s_i'} \left(Y_i \left(t \left[T_1^{Y_i} \right] \right) \in A_i \right)$, for $s' = (s_1', \dots, s_m') \in S' \setminus S$ and $(A \cap S = \emptyset \vee A \subseteq S)$.

$\int_0^\infty K_t(s, A) dt = \begin{cases} \frac{1}{\lambda(s)}, & \text{if } s \in A \\ 0, & \text{if } s \notin A \end{cases}$, for $s \in S$, where $\lambda(s)$ is the rate out of state s.

4 Conclusion And Future Work

4.1 Conclusion

We have presented a new method to compute the following two cumulative measures of an absorbing joint Markov chain $U = (U_1, \ldots, U_m)$, with state space E and set of absorbing states S:

- $\int_0^\infty \mathbb{P}(U(t) \in A)dt$, for $A \cap S = \emptyset$ (mean time in A before absorption),
- $\int_0^\infty 1 - \mathbb{P}(U(t) \in S)dt$ (mean time to absorption),

where we require that A is the cross-product of subsets A_i of the marginal state spaces. Clearly, the first quantity is the mean time that U spends in A before absorption, and the second quantity is the mean time to absorption of U.

Our method consists of at first uniformising the marginal chains U_i. The resulting uniformised versions are then treated in isolation. For the uniformised version of U_i we obtain the function $\nu_i(\cdot)[A_i]$, where $\nu_i(n)[A_i]$ is the probability that the uniformised chain is in set A_i in the $n-$th step. We defined the convolution operator $\star$, in order to combine the solutions of the marginal (uniformised) CTMCs. We found that $\nu(n)[A] := (\nu_1[A_1] \star \cdots \star \nu_m[A_m])(n)$ is just the probability that the uniformised version of the joint CTMC U is in $A = \times_{i=1}^m A_i$ in the $n-$th step. Thus, together with the uniformisation rate q of U the transient probability $\mathbb{P}(U(t) \in A)$ can be expressed as $\mathbb{P}(U(t) \in A) = \sum_{n=0}^\infty \frac{(qt)^n}{n!} e^{-qt} \nu(n)[A]$, which is known as the uniformisation equation. For the set of absorbing states $S = \times_{i=1}^m S_i$, where the S_i are the marginal absorbing sets, of course, analogous results hold, i.e. $\nu(n)[S] := (\nu_1[S_1] \star \cdots \star \nu_m[S_m])(n)$. After replacing the probabilities under the above integrals by the series of the uniformisation equation the following is obtained

- $\sum_{n=0}^\infty \frac{1}{q} \nu(n)[A]$, for $A \cap S = \emptyset$ (mean time in A before absorption),
- $\sum_{n=0}^\infty \frac{1}{q}(1 - \nu(n)[S])$ (mean time to absorption).

This is straight forward and nothing new. But the fact that the functions $\nu[A]$ and $\nu[S]$ can be computed from akin marginal functions via the $\star-$operator is a new insight.

The method we propose is an exact method, where, of course, the computation of the above series requires truncation at a certain index. The time complexity is $O(Nmd^2 + mN^2)$, where m is the number of marginal CTMCs, d is the maximal size of the marginal state spaces and N is the truncation index of the infinite series which depends on convergence properties (eigenvalues) of the joint CTMC and the desired accuracy.

The above method for the computation of cumulative measures was applied to a class of composite PEPA models, in order to compute single steady state probabilities of the Markov chain underlying that model. A PEPA model of the considered class possesses only global synchronisations, i.e. all sequential components (of which the composite model is built) must participate in every synchronisation. Then, time instants of synchronisation (i.e. time instants of

the beginning and the end of every synchronisation) of the composite model define a probabilistic sequence of embedded states, which forms an embedded discrete time Markov chain. In order to determine steady state probabilities we treat the dynamics between embedded states (i.e. the embedded Markov chain) and the behaviour of the composite model between two successive embedded states in separation. Hence, the Markov chain underlying the composite PEPA model is treated as a semi-regenerative process. The behaviour of the composite model between two successive embedded states is either a) governed by the cooperation operator of the PEPA language or b) can be expressed by an absorbing joint Markov chain. Now, for each embedded state we compute the mean time until reaching the next embedded state and the mean time spent in some set A before reaching the next embedded state. For case b) these two quantities are just the mean time to absorption and the mean time spent in A before absorption of the said absorbing joint Markov chain. By application of a key result from the theory of semi-regenerative processes, these quantities can be combined with the steady state solution of the embedded discrete time Markov chain to the steady state probability that the composite PEPA model is in set A.

The size of the state space of the Markov chain underlying the composite PEPA model is exponential in the number of concurrent sequential PEPA components. The time complexity of our proposed method to compute steady state probabilities of PEPA models does not depend on the size of the state space of the composite model, but on the number of embedded states and the complexity of the method to compute the mean time to absorption and the mean time in A before absorption. Note that even for an exploding state space the number of embedded states may be small.

4.2 Future Work

Let Y be the Markov chain underlying some PEPA model. Our proposed method for the computation of the steady state probability $\mathbb{P}(Y \in A)$ is restricted by the following requirements:

- The set A needs to be the cross-product of sets A_i, $i = 1 \dots m$, where the A_i are subsets of the state spaces of the Markov chains underlying the m concurrent PEPA components. In addition we require either $A \cap S = \emptyset$ or $A \subseteq S$.
- All concurrent components synchronise over the same synchronisation set L.
- In none of the concurrent PEPA components there exists a choice between synchronising and non-synchronising activities.

Future work might aim at dropping one or more of these conditions. Surely, the first requirement cannot be dropped, since it is a consequence of how we computed cumulative measures of an absorbing joint Markov chain. As for the third requirement we have no idea yet. Maybe a workaround for PEPA models not satisfying this third condition can be found.

The second condition which requires that all concurrent sequential components must be involved in every synchronisation is a severe one. We have an idea of how to cope with such a model but do not intimate it at this point.

Up to now we have only considered steady state probabilities. Of course, the consequent question would be if our method could be enhanced such that it is also possible to compute transient state probabilities. A straight forward approach would be to evaluate (3.8) in theorem 14. This requires knowledge of the Markov renewal kernel R of (X, T), where X is the discrete time Markov chain embedded in the composite PEPA model and T is the family of random time instants which define this embedded Markov chain. However, the efficiency of such a procedure is questionable.

Bibliography

[AMR06] P. D'Argenio, A. Miner and G. Rubino (Editors), *Proceedings of the 3rd International Conference on the Quantitative Evaluation of SysTems (QEST 2006)* (IEEE Computer Society Press, 2006)

[BCMP75] F. Baskett, K. M. Chandy, R. R. Muntz, and G. Palacios, *Open, Closed, and Mixed Networks of Queues With Different Classes of Customers* (Journal od the ACM, 22(2), pages 248-260,1975)

[BDG94] M. Bernardo, L. Donatiello, R.Gorrieri, *Modeling and Analyzing Concurrent Systems with MPA* (Proceedings of the 2nd workshop on process algebras and performance modelling, vol. 27 Arbeitsberichte des IMMD, FAU Erlangen-Nürnberg, 1994)

[BeGo96] M. Bernardo, R.Gorrieri, *Extended Markovian Process Algebra* (Proceedings of the 7th International Conference on Concurrency Theory, 1996)

[Bohn02] H. Bohnenkamp, *Compositional Solution of Stochastic Process Algebra Models* (PhD thesis, Department of Computer Science, Rheinisch-Westfälische Technische Hochschule Aachen, Germany, 2002)

[BoHa02] H. Bohnenkamp, B. Haverkort, *The mean value of the maximum* (In Holger Hermanns and Roberto Segala, editors, Process Algebra and Probabilistic Methods (PAPM-ProbmiV 2002), volume 2399 of Lecture Notes in Computer Science, pages 37-56. Springer Verlag, 2002)

[BGMT98] G. Bolch, S. Greiner, H. de Meer, K. S. Trivedi, *Queueing Networks and Markov Chains – Modeling and Performance Evaluation with Computer Science Applications* (John Wiley & sons, Inc.,1998)

[BHK00] E. Brinksma, H. Hermanns, J.-P. Katoen (Editors), *Lectures on Formal Methods and Performance Analysis* (volume 2090 of Lecture Notes in Computer Science, Springer-Verlag, 2000)

[CaTu02] M. C. Calzarossa, S. Tucci (Editors), *Performance Evaluation of Complex Systems: Techniques and Tools* (volume 2459 of Lecture Notes in Computer Science, Springer-Verlag, 2002)

[Cinl75] E. Çinlar, *Introduction to Stochastic Processes* (Prentice-Hall, Inc., Englewood Cliffs, N.J., 1975)

[ClHi02] G. Clark, J. Hillston, *Product form solution for an insensitive stochastic process algebra structure* (Performance Evaluation, 50, pages 129-151, 2002)

[Cour77] P. J. Courtois, *Decomposability – Queueing and Computer Systems Applications* (Academic Press, 1977)

[GeHe06] R. German, A. Heindl (Editors), *Proceedings of the 13th GI/ITG Conference Measureing, Modelling and Evaluation of Computer and Communication Systems (MMB 2006)* (VDE Verlag, 2006)

[HLR00] G. Haring, C. Lindemann, M. Reiser (Editors), *Performance Evaluation: Origins and Directions* (volume 1769 of Lecture Notes in Computer Science, Springer-Verlag, 2000)

[Herm02] H. Hermanns, *Interactive Markov Chains and the Quest for Quantified Quality* (volume 2428 of Lecture Notes in Computer Science, Springer-Verlag, 2002)

[HeRe02] H. Hermanns, M. Rettelbach, *Syntax, Semantics, Equivalences, and Axioms for MTIPP* (Proceedings of the 2nd workshop on process algebras and performance modelling, vol. 27 Arbeitsberichte des IMMD, FAU Erlangen-Nürnberg, 1994)

[Herz90] U. Herzog, *Formal Description, Time and Performance Analysis – A Framework* (Technical Report 15/90, IMMD VII, Friedrich-Alexander-Universität, Erlangen-Nürnberg, Germany, 1990)

[Hill96] J. Hillston, *A Compositional Approach to Performance Modelling* (Distinguished Dissertations in Computer Science, vol. 12, Cambridge University Press, 1996)

[Hill99] J. Hillston, *Exploiting Structure in Solution: Decomposing Compositional Models* (Performance Evaluation, 35, pages 171-192, 1999)

[HiTh01] J. Hillston & N. Thomas, *Product Form Solution for a class of PEPA Models* (in Ed Brinksma, HolgerHermanns and Joost-Pieter Katoen, editors, Lectures on Formal Methods and Performance Analysis, volume 2090 of Lecture Notes in Computer Science, pages 278-314. Springer-Verlag, 2001)

[MKS06] R. A. Marie, P. B. Key, E. Smirni (Editors), *Proceedings of the Joint International Conference on Measurement and Modeling of Computer Systems, SIGMETRICS/Performance 2006* (ACM, 2006)

[Neut81] M. F. Neuts, *Matrix-Geometric Solutions in Stochastic Models – An Algorithmic Approach* (The Johns Hopkins University Press, Baltimore/London, 1981)

[Stew94] W. J. Stewart, *Introduction to the Numerical Solution of Markov Chains* (Princeton University Press, 1994)

A

A.1 Corollary 5

The $\star-$operator is commutative and associative.

Proof. The commutativity is self-evident. To proof associativity consider the functions $f, g, h : \mathbb{N}_0 \to \mathbb{R}_{\geq 0}$ with the values $c_f, c_g, c_h > 0$ assigned:

$$\begin{aligned}
((f \star g) \star h)(n) &= \sum_{k=0}^{n} \binom{n}{k} \frac{(c_f + c_g)^k c_h^{n-k}}{((c_f + c_g) + c_h)^n} (f \star g)(k) h(n-k) && \text{(A.1)} \\
&= \sum_{k=0}^{n} \binom{n}{k} \frac{\cancel{(c_f + c_g)^k} c_h^{n-k}}{(c_f + c_g + c_h)^n} \left[\sum_{\ell=0}^{k} \binom{k}{\ell} \frac{(c_f)^\ell c_g^{k-\ell}}{\cancel{(c_f + c_g)^k}} f(\ell) g(k-\ell) \right] h(n-k) && \text{(A.2)} \\
&= \frac{n!}{(c_f + c_g + c_h)^n} \sum_{k=0}^{n} \sum_{\ell=0}^{k} \frac{c_f^\ell c_g^{k-\ell} c_h^{n-k}}{(n-k)!\ell!(k-\ell)!} f(\ell) g(k-\ell) h(n-k). && \text{(A.3)}
\end{aligned}$$

With the substitution $\ell \mapsto n_1, \quad k - \ell \mapsto n_2, \quad n - k \mapsto n_3$ we obtain

$$\begin{aligned}
((f \star g) \star h)(n) &= \frac{n!}{(c_f + c_g + c_h)^n} \sum_{n_1 + n_2 + n_3 = n} \frac{c_f^{n_1} c_g^{n_2} c_h^{n_3}}{n_1! n_2! n_3!} f(n_1) g(n_2) h(n_3) && \text{(A.4)} \\
&= (f \star (g \star h))(n). && \text{(A.5)}
\end{aligned}$$

□

A.2 Corollary 17

For $A \cap S = \emptyset$ and $s' \in S' \setminus S$ the following relation holds:

$$K_t(s', A) = \mathbb{P}_{s'}(Y(t\,[T_1]) \in A). \tag{A.6}$$

Proof. With $K_t(s', A) = \mathbb{P}_{s'}(Y(t) \in A, T_1 > t)$, the equality

$$\mathbb{P}_{s'}(Y(t) \in A, T_1 > t) = \mathbb{P}_{s'}(Y(t\,[T_1]) \in A, T_1 > t). \tag{A.7}$$

is straight forward, because for $T_1 > t$ we have $Y(t) = Y(t\,[T_1])$. Since $A \cap S = \emptyset$ the implication $\{Y(t\,[T_1]) \in A\} \Longrightarrow \{T_1 > t\}$ is valid, and hence

$$\mathbb{P}_{s'}(Y(t\,[T_1]) \in A, T_1 > t) = \mathbb{P}_{s'}(Y(t\,[T_1]) \in A). \tag{A.8}$$

□

A.3 Corollary 18

For $i = 1 \dots m$ and $Y(T_0) = s' \in S' \setminus S$ the following equivalence holds

$$\{Y_i\left(t\left[T_1\right]\right) \in A_i\} \Longleftrightarrow \{Y_i\left(t\left[T_1^{Y_i}\right]\right) \in A_i\}. \tag{A.9}$$

Proof. For $Y(T_0) = s' \in S' \setminus S$, we have $T_1 = \max_{i\in\{1,\dots,m\}}\{T_1^{Y_i}\}$, and hence

$$T_1 \geq T_1^{Y_i}, \quad \text{for } i = 1 \dots m. \tag{A.10}$$

Now, consider three cases.

$\boldsymbol{t < T_1^{Y_i}}$**:** For $t < T_1^{Y_i}$ we have $t[T_1] = t\left[T_1^{Y_i}\right] = t$, and consequently $Y_i\left(t\left[T_1\right]\right) = Y_i\left(t\left[T_1^{Y_i}\right]\right)$.

$\boldsymbol{T_1^{Y_i} \leq t \leq T_1}$**:** For $T_1^{Y_i} \leq t \leq T_1$, the process Y_i is in a waiting condition, i.e. during the time interval $[T_1^{Y_i}, T_1]$, Y_i remains in the state that it occupied at time instant $T_1^{Y_i}$. Consequently, $Y_i\left(t\left[T_1\right]\right) = Y_i\left(t\left[T_1^{Y_i}\right]\right)$, for $T_1^{Y_i} \leq t \leq T_1$.

$\boldsymbol{t > T_1}$**:** From the preceding case follows $Y_i(T_1) = Y_i(T_1^{Y_i})$. For $t > T_1$ we have $t[T_1] = T_1$ and $t\left[T_1^{Y_i}\right] = T_1^{Y_i}$, which implies $Y_i\left(t\left[T_1\right]\right) = Y_i\left(t\left[T_1^{Y_i}\right]\right)$. □

Previously published ICB - Research Reports

2007

No 11 (February 2007)
Kirchner, Lutz: "Entwurf einer Modellierungssprache zur Unterstützung der Aufgaben des IT-Managements – Grundlagen, Anforderungen und Metamodell"

No 10 (February 2007)
Schauer, Carola; Strecker, Stefan: "Vergleichende Literaturstudie aktueller einführender Lehrbücher der Wirtschaftsinformatik: Bezugsrahmen und Auswertung"

No 9 (February 2007)
Strecker, Stefan; Kuckertz, Andreas; Pawlowski, Jan M.: "Überlegungen zur Qualifizierung des wissenschaftlichen Nachwuchses: Ein Diskussionsbeitrag zur (kumulativen) Habilitation"

No 8 (February 2007)
Frank, Ulrich; Strecker, Stefan; Koch, Stefan: "Open Model – Ein Vorschlag für ein Forschungsprogramm der Wirtschaftsinformatik (Langfassung)"

2006

No 7 (December 2006)
Frank, Ulrich: "Towards a Pluralistic Conception of Research Methods in Information Systems Research"

No 6 (April 2006)
Frank, Ulrich: "Evaluation von Forschung und Lehre an Universitäten – Ein Diskussionsbeitrag"

No 5 (April 2006)
Jung, Jürgen: "Supply Chains in the Context of Resource Modelling"

No 4 (February 2006)
Lange, Carola: "Development and status of the Information Systems / Wirtschaftsinformatik discipline: An interpretive evaluation of interviews with renowned researchers, Part III – Results Wirtschaftsinformatik Discipline"

2005

No 3 (December 2005)
Lange, Carola: "Development and status of the Information Systems / Wirtschaftsinformatik discipline: An interpretive evaluation of interviews with renowned researchers, Part II – Results Information Systems Discipline"

No 2 (December 2005)
Lange, Carola: "Development and status of the Information Systems / Wirtschaftsinformatik discipline: An interpretive evaluation of interviews with renowned researchers, Part I – Research Objectives and Method"

No 1 (August 2005)
Lange, Carola: „Ein Bezugsrahmen zur Beschreibung von Forschungsgegenständen und -methoden in Wirtschaftsinformatik und Information Systems"

The Institute for Computer Science and Business Information Systems (ICB), located at the Essen Campus, is dedicated to research and teaching in Applied Computer Science, Information Systems as well as Information Management. The ICB research groups cover a wide range of expertise:

Research Group	Core Research Topics
Prof. Dr. H. H. Adelsberger Information Systems for Production and Operations Management	E-Learning, Knowledge Management, Skill-Management, Simulation, Artificial Intelligence
Prof. Dr. P. Chamoni MIS and Management Science / Operations Research	Information Systems and Operations Research, Business Intelligence, Data Warehousing
Prof. Dr. F.-D. Dorloff Procurement, Logistics and Information Management	E-Business, E-Procurement, E-Government
Prof. Dr. K. Echtle Dependability of Computing Systems	Dependability of Computing Systems
Prof. Dr. S. Eicker Information Systems and Software Engineering	Process Models, Software-Architectures
Prof. Dr. U. Frank Information Systems and Enterprise Modelling	Enterprise Modelling, Enterprise Application Integration, IT Management, Knowledge Management
Prof. Dr. M. Goedicke Specification of Software Systems	Distributed Systems, Software Components, CSCW
Prof. Dr. T. Kollmann E-Business and E-Entrepreneurship	E-Business and Information Management,E-Entrepreneurship/ E-Venture, Virtual Marketplaces and Mobile Commerce, Online-Marketing
Prof. Dr. B. Müller-Clostermann Systems Modelling	Performance Evaluation, Modelling and Simulation, SAP Capacity Planning for R/3 and mySAP.com, Tools for Queueing Network Analysis and Capacity Planning, Communication Protocols and Distributed Systems, Mobile Systems
Prof. Dr. K. Pohl Software Systems Engineering	Requirements Engineering, Software Quality Assurance, Software-Architectures, Evaluation of COTS/Open Source-Components
Prof. Dr.-Ing. E. Rathgeb Computer Networking Technology	Computer Networking Technology
Prof. Dr. R. Unland Data Management Systems and Knowledge Representation	Data Management, Artificial Intelligence, Software Engineering, Internet Based Teaching
Prof. Dr. S. Zelewski Institute of Production and Industrial Information Mangement	Industrial Business Processes, Innovation Management, Information Management, Economic Analyses

For more information visit us on the Web: **http://www.icb.uni-due.de**

ISSN 1860-2770