A Fourth-Order Algorithm with Automatic Stepsize Control for the Transient Analysis of DSPNs

Armin Heindl, Reinhard German

Abstract— **This paper presents an efficient and numerically reliable method for the transient analysis of deterministic and stochastic Petri nets. The transient behavior is described by state equations derived by the method of supplementary variables. Significant features of the proposed solution algorithm of fourth order are an automatic stepsize control and a two-stage relative error control. Furthermore, a formal way of dealing with discontinuities in the transient state equations is developed. This resolves the problems posed by initially enabled deterministic transitions and also improves the accuracy of numerical results. Experiments with a queueing system with failure and repair illustrate the efficiency (with respect to both CPU-time and memory space) and the numerical quality of the new algorithm.**

Keywords— **Deterministic and stochastic Petri nets, transient analysis, adaptive numerical method.**

I. Introduction

STOCHASTIC *Petri nets* (SPNs) are well suited for performance and dependability modeling. The firing times of the transitions of an SPN usually have to be exponentially distributed. For more realistic models, non-exponential timing is also required. In *Markov regenerative stochastic Petri nets* (MRSPNs, e.g. [1], [2], [3]), transitions with general firing time distributions are allowed provided that in each marking at most one transition of this kind may be enabled. In this paper, we concentrate on the transient analysis of *deterministic and stochastic Petri nets* (DSPNs, [4]). DSPNs are a particular class of MRSPNs in the sense that timed transitions may only fire after a constant delay or, of course, an exponentially distributed time. In the following, we will refer to timed transitions as deterministic and exponential transitions, respectively. For both stationary and transient analysis of DSPNs and MRSPNs, researchers have pursued the approaches based on *Markov renewal theory* (e.g. [4], [5], [1], [2], [3], [6], [7]) and on the method of *supplementary variables* [8], [9], [10], [11]. A comparison of both approaches in [12] suggested that the incorporation of an adaptive stepsize control into the algorithms should improve performance and accuracy. More specifically, the time efficiency of the approach based on supplementary variables is highly sensitive to the chosen step size. To this end, the transient analysis algorithm and its implementation in the software tool `TimeNET` [13], [10] have been redesigned, which is the subject of this paper.

The new algorithm for DSPNs is still based on transient state equations derived by the method of supplementary variables [11], and its basic structure resembles that of the one described in [10]. However, significant elements have been changed: an automatic stepsize control has been introduced and various subalgorithms have been replaced by their higher-order equivalents resulting in an adaptive fourth-order algorithm (with respect to the step size). Substantial savings in run time and required memory space could be achieved. A two-stage error control together with a transition from absolute to relative precision account for the enhanced accuracy. Furthermore, a formal way of dealing with discontinuities in the state equations has been developed. Thus, a restriction of the earlier version of the analysis algorithm could be removed: it is now possible to analyse DSPNs in which deterministic transitions are initially enabled. Such transitions lead to discontinuities in the state probabilities. Possibly, discontinuities only occur in the derivatives of the state probability functions. The careful consideration of all kinds of discontinuities results in an improved numerical quality of the algorithm. The older version of the transient analysis component of `TimeNET` has been replaced by the version presented in this paper and is now available to all users of the software tool.

The remainder of this paper is organized as follows. In the next section, DSPNs are briefly reviewed and the transient state equations are given. We will also discuss the impact of discontinuities on these equations. Section 3 explains the design of the analysis algorithm, and Section 4 examines various aspects of the algorithm. Experimental results for a queueing system with failure and repair are shown in Section 5, followed by conclusions in Section 6.

II. DSPNs and the transient state equations

We assume that the reader is familiar with the class of *deterministic and stochastic Petri-Nets* (DSPNs, [4]). The restrictions required for the analysis in this paper are that at most one deterministic transition is enabled in each marking and that the firing policy is *preemptive repeat different*. This term – common in both SPN and queueing literature – means that, when a timed transition is preempted, its already performed work is lost. In the SPN literature, this policy is also referred to as *race with enabling memory* (see [14], [3], [6] for the definition of firing policies). Figure 1 shows the model of a ΣMMPP/D/1/K-queueing system with failure and repair, which is also used for the experiments in Section V.

The left part of the DSPN, consisting of the places P1 and P2 and the transitions T1 through T4, models the arrival of customers as a superposition of N *Markov modulated Poisson processes* (MMPPs). As indicated by the number of tokens in P1, these N processes start in the same state. The firing delays of each of the four transitions

Both authors are with the Institut für Technische Informatik, Technische Universität Berlin, Franklinstr. 28/29, 10587 Berlin, Germany. E-mail: {heindl,rge}@cs.tu-berlin.de

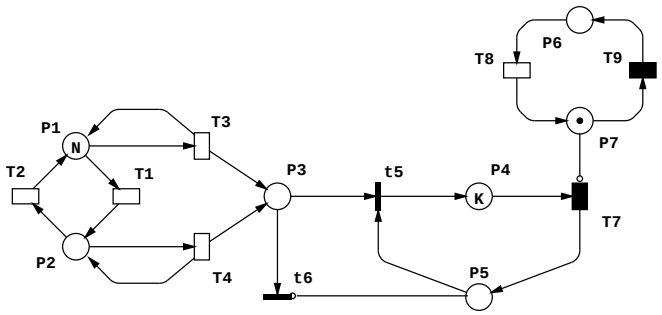


Fig. 1. A queueing system with failure and repair

are exponentially distributed with rates $\rho_1, \rho_2, \lambda_1$, and λ_2, respectively, with T3 and T4 having 'infinite server' semantics. The queueing system is represented by the lower central subnet. The number of tokens in P5 stands for the free buffer places; initially, all K places are occupied. The immediate transitions $t5$ and $t6$ model entrance and rejection of customers for the queue represented by P4. The upper right subnet controls the service facility T7 with deterministic service time τ by means of the inhibitor arc starting in P7. Failure, modelled by the exponential transition T8 with rate μ, can occur at any time (even if the system is not utilized) and once failed, it takes the constant time δ associated with the deterministic transition T9 to repair the system. Note that the deterministic transitions T7 and T9 are never concurrently enabled.

A. The transient state equations

We will now describe the dynamic behavior of DSPNs by means of the transient state equations as derived in [9] and [11] by the method of supplementary variables. For an explicit handling of the discontinuities, which will also result in slight modifications of the state equations, we start with the age distribution functions (instead of the age density functions used in [9], [11]). Let T^D be the set of all deterministic transitions which we will denote by the letters c and d ($c, d \in T^D$). After the elimination of vanishing markings due to immediate transitions, the tangible markings of the DSPN constitute the (finite) state space $\mathcal{S}$. According to the restriction that at most one deterministic transition may be enabled in each state, $\mathcal{S}$ can be partitioned into the sets $\mathcal{S}^E$, whose elements are the states in which only exponential activities are enabled, and $\mathcal{S}^d$ (for each $d \in T^D$ exists a separate set $\mathcal{S}^d$). $\mathcal{S}^d$ consists of the states in which the deterministic transition d is enabled. For any state i in $\mathcal{S}^d$ ($d \in T^D$), a supplementary variable $X(t)$ records how long the transition d has been enabled uninterruptedly. Together with the discrete random variable $N(t)$ giving the tangible marking at time t, we write

$$\{(N(t), X(t)), t \in \mathbb{R}_0^+\}$$

for the underlying stochastic process, where the continuous random variable $X(t)$ is undefined for $N(t) \in \mathcal{S}^E$. For $N(t) \in \mathcal{S}^d$ ($d \in T^D$), we define a (defective) *age distribution function* for any fixed $t \geq 0$ and x in $[0, \tau^d]$ with τ^d being the constant delay of transition d:

$$P_i^d(t,x) \equiv Pr\,(N(t) = i,\ X(t) \leq x)\,.$$

In the papers cited above, the assumption that no deterministic transitions should initially be enabled guarantees the continuity of the age distribution function, and its derivative could at least be defined piecewise forming one-sided limits on both sides of any interval. Dropping this assumption, $P_i^d(t,x)$ might be discontinuous, and the generalized version of its derivative - the *age density function* $p_i^d(t,x)$ - consists of piecewise continuous and of discrete components:

$$p_i^d(t,x) \equiv \frac{d}{dx} P_i^d(t,x) = r_i^d(t,x) + q_i^d(t,x) \cdot \delta_x\,. \qquad (1)$$

The symbol δ_x denotes a Dirac impulse at location x on a vertical axis in the t-x-plain. The jumps of $P_i^d(t,\cdot)$ for a fixed t result in Dirac impulses in the age density functions. Their essential information, namely their coordinates in the t-x-plain and the associated areas, is recorded in the function $q_i^d(t,x)$ ($i \in \mathcal{S}^d, d \in T^D$). The function $q_i^d(t,x)$ is defined for all $t \geq 0$ and $0 \leq x \leq \tau^d$, although the domain on which $q_i^d(t,x)$ takes on values different from zero is restricted to a series of straight lines. Later in connection with the investigation of impulse propagation on the t-x-plains, we will show how these lines can be determined. It should be mentioned that, whenever $q_i^d(t,x) = 0$, the term $q_i^d(t,x) \cdot \delta_x$ is supposed to vanish. Whereas δ_x symbolizes the discrete nature of the second term, the function $r_i^d(t,x)$ represents the continuous components of $p_i^d(t,x)$. Opposed to $r_i^d(t,x)$, $q_i^d(t,x)$ is a probability.

In order to facilitate generalizations concerning Dirac impulses and the resulting discontinuities in the state probability functions, we give the transient state equations [1] as derived in [11] with slightly altered initial conditions and integral equations. As in [11], we introduce the vectors $\mathbf{p}^d(t,x)$ ($d \in T^D$) so that the ith component of $\mathbf{p}^d(t,x)$ equals $p_i^d(t,x)$, if $i \in \mathcal{S}^d$, and zero otherwise. Correspondingly, the vectors of the transient state probabilities $\boldsymbol{\pi}^E(t)$ and $\boldsymbol{\pi}^d(t)$ ($d \in T^D$) are defined. For example, $\pi_i^E(t) = Pr(N(t) = i)$, if $i \in \mathcal{S}^E$, and zero otherwise. Similarly, all other vectors like $\mathbf{P}^d(t,x)$, $\mathbf{q}^d(t,x)$, $\mathbf{r}^d(t,x)$, etc., are composed of the appropriate elements. All vectors are of dimension $|\mathcal{S}|$, the following matrices of dimension $|\mathcal{S}| \times |\mathcal{S}|$. These matrices can be divided into $\mathbf{Q}$-matrices and $\boldsymbol{\Delta}$-matrices:

- $\mathbf{Q}$-matrices: $\mathbf{Q}^{E,E}$, $\mathbf{Q}^{E,d}$, $\mathbf{Q}^{d,E}$, $\mathbf{Q}^{d,c}$, $\mathbf{Q}^{d}$
- $\boldsymbol{\Delta}$-matrices: $\boldsymbol{\Delta}^{d,E}$, $\boldsymbol{\Delta}^{d,c}$

for $c, d \in T^D$.

The entries of the $\boldsymbol{\Delta}$-matrices are *branching probabilities*, with which the process jumps from one state to the other after the firing of a deterministic transition. For example,

[1] In [11], age density functions, denoted by $\pi_i^d(t,x)$, and age rate functions, denoted by $p_i^d(t,x)$, were defined. Since both functions are equal for transitions with deterministic firing times, we do not differentiate between them in this paper.

for $i \in \mathcal{S}^d$ and $j \in \mathcal{S}^E$ the element δ_{ij} of matrix $\mathbf{\Delta}^{d,E}$ is the probability that the firing of the deterministic transition d in state i leads to state j, given that d fires in i. In the queueing system of Figure 1, all these probabilities are either unity or zero, but in general immediate transitions which become concurrently enabled after the firing of a deterministic transition may cause branching probabilities of any value in $[0,1]$. In the $\mathbf{\Delta}$-matrices as well as in the $\mathbf{Q}$-matrices, the superscripts indicate the subsets of the state space between which the state transitions take place. Elements whose indices do not belong to the indicated subsets are set to zero. Of course, all sparsely populated vectors and matrices are represented by sparse data structures in the implementation of the algorithm. The $\mathbf{Q}$-matrices describe the state changes due to the exponential transitions by containing the associated rates. In addition, the diagonal elements of $\mathbf{Q}^{E,E}$ and $\mathbf{Q}^d$ $(d \in T^D)$ equal the negative sum of all outgoing rates in state $i \in \mathcal{S}^E$ and $\mathcal{S}^d$ $(d \in T^D)$, respectively. Note the difference between $\mathbf{Q}^{d,d}$ and $\mathbf{Q}^d$. In the latter case the state changes caused by the exponential transitions do not disable d, whereas an exponential transition represented in $\mathbf{Q}^{d,d}$ preempts the deterministic transition d, but enables it again immediately afterwards. The transient state equations for DSPNs are given by:

- **initial conditions**

$$\begin{aligned} \boldsymbol{\pi}^E(0) = \boldsymbol{\pi}_0^E \quad &, \quad \boldsymbol{\pi}^d(0) = \boldsymbol{\pi}_0^d \\ \mathbf{p}^d(0,0) = \boldsymbol{\pi}_0^d \cdot \delta_0 \quad &, \quad \mathbf{p}^d(0,x) = \mathbf{0} \end{aligned}$$

where $x > 0$ and $\boldsymbol{\pi}_0^E$, $\boldsymbol{\pi}_0^d$ $(d \in T^D)$ represent the initial state occupancy distribution,

- **partial differential equations**

$$\frac{\partial}{\partial t}\mathbf{p}^d(t,x) + \frac{\partial}{\partial x}\mathbf{p}^d(t,x) = \mathbf{p}^d(t,x) \cdot \mathbf{Q}^d \tag{2}$$

$(0 \le x < \tau^d)$, which can be transformed into ordinary differential equations by the method of characteristics [9], the solution of which is given by

$$\mathbf{p}^d(t+h, x+h) = \mathbf{p}^d(t,x) \cdot e^{\mathbf{Q}^d h}, \tag{3}$$

- **integral equations**

$$\boldsymbol{\pi}^d(t) = \int_0^{\tau^d} d_x \mathbf{P}^d(t,x) = \int_0^{\tau^d} \mathbf{p}^d(t,x)dx, \tag{4}$$

- **ordinary differential equations**

$$\begin{aligned} \frac{d}{dt}\boldsymbol{\pi}^E(t) \quad = \quad & \boldsymbol{\pi}^E(t) \cdot \mathbf{Q}^{E,E} + \sum_{d \in T^D} \boldsymbol{\pi}^d(t) \cdot \mathbf{Q}^{d,E} \\ & + \sum_{d \in T^D} \mathbf{p}^d(t,\tau^d) \cdot \mathbf{\Delta}^{d,E}, \end{aligned} \tag{5}$$

- **boundary conditions**

$$\begin{aligned} \mathbf{p}^d(t,0) \quad = \quad & \boldsymbol{\pi}^E(t) \cdot \mathbf{Q}^{E,d} + \sum_{c \in T^D} \boldsymbol{\pi}^c(t) \cdot \mathbf{Q}^{c,d} \\ & + \sum_{c \in T^D} \mathbf{p}^c(t,\tau^c) \cdot \mathbf{\Delta}^{c,d}. \end{aligned} \tag{6}$$

Note that, whenever d appears on the left side of an equation, one equation exists for every $d \in T^D$. A detailed derivation of the equations can be found in [9]. Following [8], difference equations are used to express the change of the time-dependent quantities in the time interval $(t, t+dt)$. Equations (2) and (5) are the differential versions of these equations. Equations (2) express the change of the age density functions $\mathbf{p}^d(t,x)$, when deterministic transition d has been enabled for x time units. Note that both the time parameter t and the age variable x increase with the same speed. Consequently, on characteristic lines with slope one the partial differential equations (2) can be reduced to ordinary differential equations with the solution given in (3). Equations (5) express the change of the state probability functions $\boldsymbol{\pi}^E(t)$. The terms on the right side correspond to the following types of state transitions: transitions inside $\mathcal{S}^E$, preemptions of deterministic transitions, and the firings of deterministic transitions. The boundary conditions (6) capture the act of the enabling of a deterministic transition. The three right-hand side terms correspond to the same types of state transitions as in the ordinary differential equations. Here, however, the destination states are in $\mathcal{S}^d$ and not in $\mathcal{S}^E$, as indicated by the superscripts.

B. Separating continuous and discrete parts

In this subsection, we will show that both continuous and discrete components of the age density functions can be dealt with separately to a certain degree. By inserting $\mathbf{p}^d(t,x) = \mathbf{r}^d(t,x) + \mathbf{q}^d(t,x) \cdot \delta_x$ and splitting continuous and discrete parts of the resulting state equations into autonomous equations, we obtain two sets of transient state equations, which describe the evolution of either the functions $\mathbf{r}^d(t,x)$ or $\mathbf{q}^d(t,x)$ in the t-x-plains and which are coupled via the state probabilities.

The continuous parts

According to the initial conditions, the age density function $p_i^d(0,x)$ for each $i \in \mathcal{S}^d$ consists of a single impulse at $x = 0$ so that its continuous parts $r_i^d(0,x)$ are set to zero for all $x \in [0, \tau^d]$. Equations (3) and the boundary conditions (6) are split as proposed above. The integral over the continuous parts of the age density function gives an intermediate result vector $\boldsymbol{\varrho}^d(t)$. The continuous-form version of the ordinary differential equations (5) is valid at instants of time without impulses on the lines $x = \tau^d$ $(d \in T^D)$ (i.e. $\mathbf{q}^d(t,\tau^d) = \mathbf{0}$ $\forall d \in T^D$) and obtained by replacing $\mathbf{p}^d(t,\tau^d)$ by $\mathbf{r}^d(t,\tau^d)$. Thus, we have the following set of equations relating to the continuous parts of all functions:

$$\boldsymbol{\pi}^E(0) = \boldsymbol{\pi}_0^E,\ \boldsymbol{\pi}^d(0) = \boldsymbol{\pi}_0^d,\ \mathbf{r}^d(0,x) = \mathbf{0}\ (x \ge 0), \tag{7}$$

$$\mathbf{r}^d(t+h, x+h) = \mathbf{r}^d(t,x) \cdot e^{\mathbf{Q}^d h}, \tag{8}$$

$$\boldsymbol{\varrho}^d(t) = \int_0^{\tau^d} \mathbf{r}^d(t,x)dx. \tag{9}$$

If $\mathbf{q}^d(t,\tau^d) = \mathbf{0}\ \ \forall d \in T^D$:

$$\frac{d}{dt}\boldsymbol{\pi}^E(t) \quad = \quad \boldsymbol{\pi}^E(t) \cdot \mathbf{Q}^{E,E} + \sum_{d \in T^D} \boldsymbol{\pi}^d(t) \cdot \mathbf{Q}^{d,E}$$

$$+ \sum_{d \in T^D} \mathbf{r}^d(t,\tau^d) \cdot \mathbf{\Delta}^{d,E}, \qquad (10)$$

$$\mathbf{r}^d(t,0) = \boldsymbol{\pi}^E(t) \cdot \mathbf{Q}^{E,d} + \sum_{c \in T^D} \boldsymbol{\pi}^c(t) \cdot \mathbf{Q}^{c,d} + \sum_{c \in T^D} \mathbf{r}^c(t,\tau^c) \cdot \mathbf{\Delta}^{c,d}. \qquad (11)$$

If no deterministic transitions are initially enabled, i.e. $\boldsymbol{\pi}_0^d = \mathbf{0}\ \forall d \in T^D$, obviously no impulses at all appear on the t-x-plains, i.e. $\mathbf{q}^d(t,\tau^d) = \mathbf{0}$ for $t \geq 0, 0 \leq x \leq \tau^d$ $(\forall d \in T^D)$, so that the vector $\boldsymbol{\varrho}^d(t)$ equals $\boldsymbol{\pi}^d(t)$ and the condition $\mathbf{q}^d(t,\tau^d) = \mathbf{0}$ for all $d \in T^D$ for (10) is always true. The set of equations in the following subsection becomes meaningless in that case such that (7) through (11) coincide with the original state equations (by setting $\mathbf{p}^d(t,x) = \mathbf{r}^d(t,x)$).

The discrete parts

With deterministic transitions being initially enabled, impulses do exist on the t-x-plains. At time $t = 0$, the function $\mathbf{q}^d(0,x)$ vanishes for all $x \geq 0$, except for $x = 0$, where it takes on the value $\boldsymbol{\pi}_0^d$, the vector that contains the areas of the initial impulses at the origin of the t-x-plains. The discrete counterparts of (8) and (11) are found by separating the impulse terms of (3) and (6), after (1) has been inserted. The separate integration of the impulses along a line parallel to the x-axis for a fixed t results in a sum of areas denoted by $\boldsymbol{\varphi}^d(t)$. Let us now consider the ordinary differential equations (5) in the alternative case that $\mathbf{q}^d(t,\tau^d) \neq \mathbf{0}$ for at least one $d \in T^D$. Inserting (1) and forming the single-point integral over the instant of time t yields (15). The superscripts $-$ and $+$ indicate that the respective quantity, here $\boldsymbol{\pi}^E(t)$, may be subject to a jump with different values before and after the discontinuity. Below, we have listed all the discrete counterparts of Equations (7) through (11):

initial impulses:

$$\mathbf{q}^d(0,0) \cdot \delta_0 = \boldsymbol{\pi}_0^d \cdot \delta_0, \quad \mathbf{q}^d(0,x) \cdot \delta_x = \mathbf{0} \quad (x > 0), \qquad (12)$$

impulse propagation:

$$\mathbf{q}^d(t+h, x+h) \cdot \delta_{x+h} = \mathbf{q}^d(t,x) \cdot \delta_x \cdot e^{\mathbf{Q}^d h}, \qquad (13)$$

summation of impulses:

$$\boldsymbol{\varphi}^d(t) = \sum_{\forall x \mid \mathbf{q}^d(t,x) \neq \mathbf{0}} \mathbf{q}^d(t,x). \qquad (14)$$

absorption of impulses (if $\mathbf{q}^d(t,\tau^d) \neq \mathbf{0}$ for at least one $d \in T^D$)**:**

$$\boldsymbol{\pi}^E(t^+) = \boldsymbol{\pi}^E(t^-) + \sum_{d \in T^D} \mathbf{q}^d(t,\tau^d) \cdot \mathbf{\Delta}^{d,E}, \qquad (15)$$

displacement of impulses:

$$\mathbf{q}^d(t,0) \cdot \delta_0 = \sum_{c \in T^D} \mathbf{q}^c(t,\tau^c) \cdot \delta_{\tau^c} \cdot \mathbf{\Delta}^{c,d}. \qquad (16)$$

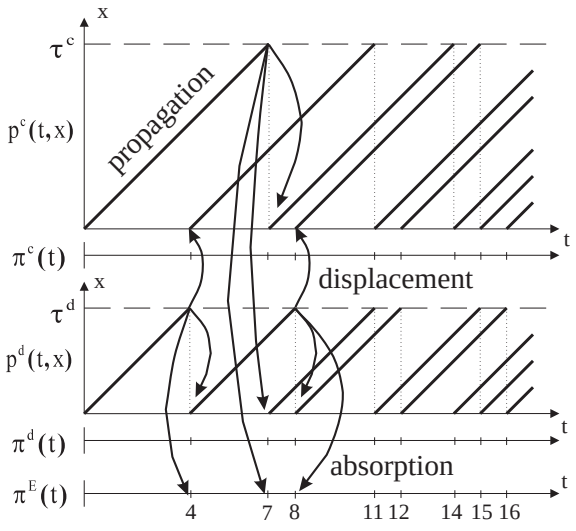


Fig. 2. Determining the instants of discontinuity

One recognizes that (13) and (16) describe the evolution of the impulses on the t-x-plains completely independently from the state probability functions and the continuous parts of the age density functions. On the contrary, the impulses directly influence the curves of the state probabilities and thus indirectly the development of the continuous segments of the age densities. Whereas for $\boldsymbol{\pi}^d(t)$ $(d \in T^D)$ this impact lasts throughout the whole time domain via the *combined integral equation*

$$\boldsymbol{\pi}^d(t) = \boldsymbol{\varrho}^d(t) + \boldsymbol{\varphi}^d(t), \qquad (17)$$

the functions $\boldsymbol{\pi}^E(t)$ are directly affected by the impulses only at certain instants of time via (15). We also want to point out that (15) could have been defined for all instants of time, i.e. also if $\mathbf{q}^d(t,\tau^d) = \mathbf{0}\ \ \forall d \in T^D$. Equation (15) would then simply mean that the function $\boldsymbol{\pi}^E(t)$ is continuous at time t. Also, (16) turns out to be significant only for the instants of time at which $\mathbf{q}^d(t,\tau^d) \neq \mathbf{0}$ for at least one $d \in T^D$. Similarly, one must keep in mind that in (13) and (14) $\mathbf{q}^d(t,x)$ differs from zero just for a certain number of x-arguments.

C. Impulse propagation and the instants of discontinuity

In order to determine the instants of time at which the condition $\mathbf{q}^d(t,\tau^d) \neq \mathbf{0}$ for at least one $d \in T^D$ is possibly valid, we investigate the impulse propagation on the t-x-plains. We will identify the *instants of discontinuity*, where the state probability functions or their derivatives *may* be subject to jumps. The following considerations will also reveal of how many terms the sum in (14) is composed and where the corresponding impulses are located. One should keep in mind that, whenever the age density functions $\mathbf{p}^d(t,\cdot)$ can theoretically encounter an impulse, their continuous parts may undergo discontinuities, possibly only in their derivatives. Thus, impulse propagation also means discontinuity propagation of the functions $\mathbf{r}^d(t,x)$.

Equation (13) expresses that the existing Dirac impulses move along their characteristic lines with slope one. During time propagation, the impulses might spread to other states in $\mathcal{S}^d$ according to the structure of the matrix $\mathbf{Q}^d$. Whenever the impulses reach the line $x = \tau^d$ (meaning that the deterministic transition d fires with a probability greater than zero), they are displaced according to the structures of the $\mathbf{\Delta}$-matrices and give rise to new impulses on the t-axes of the t-x-plains for the same and/or other deterministic transitions (see (16)) or are absorbed by the probabilities $\boldsymbol{\pi}^E(t)$ (see (15)). Starting from the origins of the t-x-plains, Figure 2 shows the characteristic lines (as thick straight lines with slope one) on which impulses might be found for a DSPN with two deterministic transitions having the constant delays $\tau^d = 4$ and $\tau^c = 7$. Beside the two t-x-plains on parts of which the age density functions $\mathbf{p}^d(t,x)$ and $\mathbf{p}^c(t,x)$ are defined respectively, three time axes for the state probabilities $\boldsymbol{\pi}^E(t)$, $\boldsymbol{\pi}^d(t)$, and $\boldsymbol{\pi}^c(t)$ are depicted, for which the instants of discontinuity have to be found. The thick arrows represent either the displacement or the absorption of impulses depending on whether they point to a t-x-plain or the axis for $\boldsymbol{\pi}^E(t)$. Note that $\mathbf{q}^d(t,x)$ is zero off the countable number of highlighted characteristic lines. The instants when any of these characteristic lines crosses the respective line $x = \tau^d$ are called the *instants of discontinuity* – in Figure 2, they are marked on the t-axes of the state probabilities. However, even on the traced characteristic lines, $\mathbf{q}^d(t,x)$ may be zero depending on the initial state occupancy distribution and the $\mathbf{\Delta}$-matrices. As a consequence, the state probability functions will not jump at the instants of discontinuity, but may take sharp bends due to the discontinuities of the functions $\mathbf{r}^d(t,x)$. Continuity of the state probability functions is then preserved, but not their differentiability. This is also the reason why the instants of discontinuity have to be determined according to Figure 2 for DSPNs with two deterministic transitions, even though these transitions must not be concurrently enabled. In the case that impulses are actually displaced, i.e. $\exists\, d \mid \mathbf{q}^d(t,\tau^d) \neq \mathbf{0}$, at least one of the state probabilities must jump at this instant of discontinuity. The right-hand sides of $\boldsymbol{\pi}^E(t)$ are computed by means of (15), whereas those of $\boldsymbol{\pi}^d(t)$ $(d \in T^D)$ in (17) can be determined by:

$$\boldsymbol{\pi}^d(t^+) = \boldsymbol{\pi}^d(t^-) - \mathbf{q}^d(t,\tau^d) + \mathbf{q}^d(t,0). \tag{18}$$

Equation (18) saves a new evaluation of the integral Equations (9) and (14) and has to be applied after evaluating (16). We point out that in many cases $\mathbf{q}^d(t,\tau^d)$ might differ from zero for only a single d in T^D, but this is not necessarily the case as Figure 2 shows for $t = 11$ and $t = 15$. Further, we notice that only the state probabilities of states in which a deterministic transition is enabled may jump in both positive and negative directions at the instants of discontinuity, whereas the probabilities $\boldsymbol{\pi}^E(t)$ can only increase due to jumps.

For those x-arguments where impulses may occur in the age density functions, the left-hand and right-hand sides of $\mathbf{r}^d(t,x)$ must be distinguished. So, at an instant of discontinuity, the boundary conditions (11) have to be applied twice, once before and once after the displacement of impulses. Upon arrival at an instant of discontinuity, (11) reads:

$$\begin{aligned}\mathbf{r}^d(t,0^+) &= \boldsymbol{\pi}^E(t^-)\cdot\mathbf{Q}^{E,d} + \sum_{c\in T^D}\boldsymbol{\pi}^c(t^-)\cdot\mathbf{Q}^{c,d} \\ &\quad + \sum_{c\in T^D}\mathbf{r}^c(t,\tau^{c+})\cdot\mathbf{\Delta}^{c,d}.\end{aligned} \tag{19}$$

After (15), (16), and (18) have been performed, the new left-hand sides of $\mathbf{r}^d(t,x)$ at the point $x = 0$ are given by:

$$\begin{aligned}\mathbf{r}^d(t,0^-) &= \boldsymbol{\pi}^E(t^+)\cdot\mathbf{Q}^{E,d} + \sum_{c\in T^D}\boldsymbol{\pi}^c(t^+)\cdot\mathbf{Q}^{c,d} \\ &\quad + \sum_{c\in T^D}\mathbf{r}^c(t,\tau^{c-})\cdot\mathbf{\Delta}^{c,d}.\end{aligned} \tag{20}$$

Due to the terms $\mathbf{r}^c(t,\tau^{c+})$ and $\mathbf{r}^c(t,\tau^{c-})$, (19) and (20) have to be observed at *every* instant of discontinuity, even if no impulses are actually displaced, i.e. $\boldsymbol{\pi}^E(t^-) = \boldsymbol{\pi}^E(t^+)$ and $\boldsymbol{\pi}^d(t^-) = \boldsymbol{\pi}^d(t^+)$ $(d \in T^D)$. This explains the occurrence of discontinuities in the functions $\mathbf{r}^d(t,x)$ also in the case that no deterministic transitions have initially been enabled.

III. Design of the analysis algorithm

In [9] and [10], an explicit one-step method with fixed step size has been proposed for the numerical analysis of the transient state equations. The general idea is to discretize the equations and to compute the state probabilities and age densities successively. In one step, the values at the instant of time t_{i+1} are computed based on the values at the instant of time t_i. The difference $t_{i+1} - t_i$ is called the step size h_{i+1}. Each step can be divided into substeps, in which different state equations are evaluated, i.e. the partial differential equations, the integral equations, the ordinary differential equations and the boundary conditions.

The algorithm described in this paper has the same basic structure. [2] However, it has been redesigned in order to

- improve the efficiency (with respect to both time and space),
- enhance the accuracy, and
- remove restrictions of the algorithm (it is now possible that deterministic transitions are initially enabled).

In detail, the new aspects of the redesigned algorithm include:

- an automatic stepsize control instead of a fixed step size,
- a mechanism for the control of the relative error, which is organized in two stages,
- well-suited higher-order numerical subalgorithms for the integration, inter-, and extrapolation required in the various substeps such that the error of a main step is of fourth order with respect to the chosen step size,

[2] In Section VI we will discuss the limitations of our algorithm due to this agreement.

• explicit dealing with discontinuities, which includes the tracing of possible locations of impulses in order to detect the instants of discontinuity, the careful incorporation of jumps in the probability functions or in their derivatives, the consideration of integration bounds caused by discontinuities, and a starting procedure after an instant of discontinuity.

A. Structure of the algorithm

The structure of the redesigned algorithm is as follows:

Initialization step $t_0 = 0$

Initialization of the state probabilities and the age density functions according to the initial conditions (7) and (12). t_0 is an instant of discontinuity, after which the starting procedure is used for the next steps.

Main step $t_i \rightarrow t_{i+1}$

substep 1: Eventual truncation of the step size h_{i+1} in order not to skip an instant of discontinuity.

substep 2: Computing the next layer of nodes of the age density functions at time t_{i+1} by means of the solution of the partial differential equations (8) and (13) employing *Jensen's method*, also called *randomization* or *uniformization* [15].

substep 3: Computing the state probabilities $\boldsymbol{\pi}^d(t_{i+1})$ $(d \in T^D)$ by numerical integration of the age densities according to Equations (9), (14), and (17). Unknown points of $\mathbf{r}^d(t_{i+1}, x)$ at the boundaries of the integrals are obtained by inter- and extrapolation.

substep 4: Computing the state probabilities $\boldsymbol{\pi}^E(t_{i+1})$ by integrating the ordinary differential equations (10) by means of an explicit embedded Runge-Kutta method. This adaptive solver includes the first stage of the error control (step acceptance/rejection) and a suggestion of the next step size h_{i+2}.

substep 5: Checking whether the computed quantities satisfy the boundary conditions (11). Otherwise, an iteration over substeps 3, 4 and 5 is performed.

substep 6: Step rejection, if the sum of all state probabilities differs from 1.0 significantly (second stage of the error control).

substep 7: (to be performed only, if t_{i+1} is an instant of discontinuity) Eventual displacement of impulses according to (16) and determination of the new right-hand sides of $\boldsymbol{\pi}^E(t_{i+1})$ and $\boldsymbol{\pi}^d(t_{i+1})$ $(d \in T^D)$ by (15) and (18). In any case, $\mathbf{r}^d(t, 0^-)$ $(d \in T^D)$ must be computed by means of (20) and the starting procedure must be initiated.

Figure 3 illustrates the progression of the algorithm between two instants of discontinuity for a DSPN with only a single deterministic transition. Only the major substeps 2 through 5 are shown. The symbols stand for specific numerical values of different quantities associated with the t-axes and the t-x-plain. For example, the two filled triangles represent $\boldsymbol{\pi}^E(t_i)$ and $\boldsymbol{\pi}^E(t_{i+1})$, respectively, whereas all circles along the vertical line $t = t_i$ on the t-x-plain depict the known values of the continuous components $\mathbf{r}^d(t_i, x)$ of the age density functions $\mathbf{p}^d(t_i, x)$. The thick arrow between the filled and the empty circle on this line is the

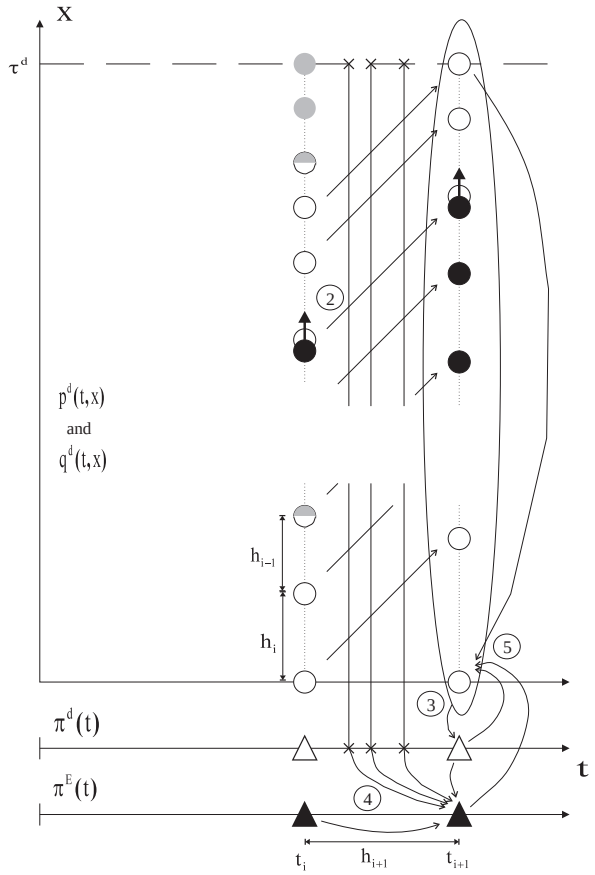


Fig. 3. The major substeps of a main step

symbol for a Dirac impulse in the age density functions $\mathbf{p}^d(t_i, x)$. The different fillings of the circles indicate that neighboring nodes of the functions $\mathbf{r}^d(t, x)$ with the same filling belong to the same integration interval. Some circles may be part of two adjacent integration intervals shown by two fillings each occupying one half of the circle. The thin one-directional arrows illustrate the necessary computations of the related substeps – e.g., all arrows with slope one on the t-x-plain are assigned to substep 2. The values at the beginning of the arrows flow into the computations of the values that the corresponding arrow points to. Further explanations, which clarify the meaning of Figure 3, will be given in the following, where we describe the substeps in detail.

B. Detailed description of the substeps

The automatic stepsize control in substep 4 might reject the main step and have the same procedure repeated with a step size smaller than the original h_{i+1}. Furthermore, $(t_i, 0)$ is a starting point for another characteristic line on every t-x-plain, on which the values of $\mathbf{r}^d(t_{i+1}, h_{i+1})$, $\mathbf{r}^d(t_{i+2}, h_{i+1} + h_{i+2})$, $\mathbf{r}^d(t_{i+3}, h_{i+1} + h_{i+2} + h_{i+3})$ and so on can be computed from $\mathbf{r}^d(t_i, 0)$ by Jensen's method (substep 2) for all $d \in T^D$. Thus, the automatic stepsize control of substep 4 is also responsible for the distances between the points at which the age density functions $\mathbf{p}^d(t_{i+1}, x)$ are known at time t_{i+1}. For substep 3, an interpolation

quadrature is chosen for numerical integration. However, the discontinuities of the functions $\mathbf{r}^d(t,x)$ $(d \in T^D)$ have to be taken into account.

Polynomial inter- and extrapolation (substeps 3 and 4)

In order to obtain a satisfactory value for $\boldsymbol{\pi}^d(t_{i+1})$, we have to compute numerical values for $\mathbf{r}^d(t_{i+1},\tau^d)$ and $\mathbf{r}^d(t_{i+1},0)$. In the latter case, this can be done by extrapolating the age density functions along the axis $t = t_{i+1}$ (we employ the *Aitken-Neville* algorithm, see [16]), whereas interpolation along the axis $x = \tau^d$ yields the desired result for $\mathbf{r}^d(t_{i+1},\tau^d)$. To this end, we will have to determine some interpolation points of the function $\mathbf{r}^d(t,\tau^d)$, which is continuous between two instants of discontinuity, by applying Jensen's method for some suitable known values of $\mathbf{r}^d(t_i,x)$. Figure 3 depicts the unlikely situation that $\mathbf{r}^d(t_{i+1},\tau^d)$ is already located on a characteristic line on which the values of the age density functions can be found without interpolation [3]. The costly procedure described above for computing $\mathbf{r}^d(t_{i+1},\tau^d)$ is only justified by the fact that higher-order embedded Runge-Kutta (RK) methods for the integration of the ordinary differential equations do require values of the functions $\mathbf{r}^d(t,\tau^d)$ (as well as $\boldsymbol{\pi}^d(t)$) at some intermediate points in $[t_i,t_{i+1}]$ so that the interpolation points for $\mathbf{r}^d(t,\tau^d)$ determined in substep 3 can be used again to estimate $\mathbf{r}^d(t,\tau^d)$ for certain $t \in [t_i,t_{i+1}]$ (or more precisely at $t_i + c_l h_{i+1}$, see (22)). For the interpolation of $\boldsymbol{\pi}^d(t)$ at these intermediate points indicated by the symbol $\times$ in Figure 3, we are forced to fall back on the values $\boldsymbol{\pi}^d(t_{i+1})$, $\boldsymbol{\pi}^d(t_i)$, $\boldsymbol{\pi}^d(t_{i-1})$ and so on depending on the desired quality of the approximation.

Iteration (substep 5)

Suppose that the error control of the embedded RK-method results in the acception of substep 4, then all relevant quantities - state probabilities and age density functions - are available in the selected order of precision at time t_{i+1} even without having made use of the boundary conditions. Comparing $\mathbf{r}^d(t_{i+1},0)$ as obtained from an evaluation of the boundary conditions with the previously determined value of this quantity (see extrapolation) gives a hint whether the computations have been accurate enough or whether one should try to improve the results further. If the deviations of the values of $\mathbf{r}^d(t_{i+1},0)$ cannot be tolerated, $\mathbf{r}^d(t_{i+1},0)$ is replaced by the value computed from the boundary conditions and the substeps 3, 4 and 5 are repeated, until the deviations lie within a given margin.

The two-stage stepsize control (substeps 4 and 6)

Given that the embedded RK-method in substep 4 does not lead to a step rejection, it proposes the step size h_{i+2} for the next step from t_{i+1} to t_{i+2}. But this is only the first stage of the employed two-stage stepsize control. Reducing the size of h_{i+2} in a second stage at the end of any main step may seem reasonable, since it only reflects the local error of the vector $\boldsymbol{\pi}^E(t_{i+1})$ and completely ignores that the numerical integration of substep 3 and the various interpolations also impose restrictions on the step sizes in order to ensure a sufficient precision. We hope to detect the errors resulting from integration and interpolation by watching the difference between the sum of all state probabilities and the ideal value 1.0. If this difference increases unreasonably from time t_i to t_{i+1}, we will cut the proposed next step size by a factor proportional to this increase. As a result, the nodes in the t-x-plains are spread more uniformly, which should reduce the errors in the numerical integration and interpolations ahead. A deviation exceeding a given tolerance will even prompt us to reject the complete step and repeat the last main step with a smaller step size than h_{i+1}. Due to the possibility of step refusions we have to save all the data of time t_i temporarily.

Reaching an instant of discontinuity and the starting procedure (substeps 7 and 1)

If a main step has successfully been undertaken and t_{i+1} is *not* an instant of discontinuity, the computations start again with substep 1 at time t_{i+1} and with step size h_{i+2}. The algorithm again has to make sure that $t_{i+1} + h_{i+2}$ is not beyond the next instant of discontinuity by possibly cutting the proposed step size h_{i+2} so that the algorithm hits the instant of discontinuity exactly.

Suppose, t_{i+1} is an instant of discontinuity. If we additionally have to deal with impulses due to initially enabled transitions, we will in case displace the impulses from the straight lines $x = \tau^d$ to the t-axes of the t-x-plains according to Equation (16) and compute the right-hand sides of the functions $\boldsymbol{\pi}^E(t)$ and $\boldsymbol{\pi}^d(t)$ via Equations (15) and (18). Whereas this is only necessary, if $\mathbf{q}^d(t_{i+1},\tau^d) \neq \mathbf{0}$ for at least one $d \in T^D$, we must not ignore (20) and the starting procedure as described in the following paragraph.

We have described a main step of the algorithm from t_i to t_{i+1} independently of the chosen order of the algorithm. For a fourth-order algorithm, the method of Fehlberg [17] can be selected as the embedded RK-formula of substep 4. However, we made a decision in favor of the method of Dormand/Prince [17], which is of fifth order, but requires only few more operations. All extrapolations and interpolations as well as the numerical integration have to be compatible with the order of substep 4 (to be discussed in the next section). Unfortunately, the first two main steps at $t = 0$ and after every instant of discontinuity cannot be carried out with fourth order, since the extrapolation of $\mathbf{r}^d(t_{i+1},0)$ and the interpolation of the function $\boldsymbol{\pi}^d(t)$ at the intermediate points do not achieve the desired quality of approximation due to the lack of appropriate nodes. So the two starting steps after every instant of discontinuity (including $t = 0$) will only be of second and third order, respectively. In these steps, we will refrain from an automatic stepsize control by employing two simple explicit RK-methods of an adapted order in substep 4 and select the step sizes sufficiently small so that the numerical errors can be neglected. The step size of the third main step will then be chosen by an educated guess of the computer in

[3] For example, this situation occurs, if t_{i+1} is an instant of discontinuity and some of the displaced impulses belong to states in $\mathcal{S}^d$.

analogy to an algorithm given in [17].

C. Simplifications in special cases

The description of the algorithm above concentrated on the 'worst case', that means that we considered initially enabled deterministic transitions as well as exponential transitions which might preempt deterministic transitions in any situation. In the following, we point out special cases in which the complexity of the algorithm can be reduced:

- If deterministic transitions are not initially enabled, the discrete-form versions of the Equations (12) through (16) together with (17) and (18) become obsolete (but not (19) and (20)). The result is the same, as if all impulses $\mathbf{q}^d(t,x)$ were set to zero.
- On the assumption that $\mathbf{Q}^{d,E} = \mathbf{0}\ \forall d \in T^D$ – in words, an enabled deterministic transition can never be preempted by an exponential transition such that the process enters a state in $\mathcal{S}^E$ –, substep 4 can be skipped *within an iteration*. Since $\boldsymbol{\pi}^d(t)$ vanishes in the ordinary differential equations (10) for all $d \in T^D$, these equations are invariant to the iterations possibly caused by checking the boundary conditions in substep 5.
- If in addition to $\mathbf{Q}^{d,E} = \mathbf{0}$ $(\forall d \in T^D)$ the matrices $\mathbf{Q}^{c,d}$ $(\forall c,d \in T^D)$ contain zero elements only – in words, an enabled deterministic transition can *never* be preempted –, we no longer need any kind of iteration at all. The substeps 2 through 5 are reorganized as follows:

$$\text{substep } 2 \rightarrow \text{substep } 4 \rightarrow \text{substep } 5 \rightarrow \text{substep } 3.$$

Note that the values of $\mathbf{r}^d(t_{i+1},0)$ do not have to be extrapolated any more. Instead, they are determined from the boundary conditions in substep 5. This differs from the algorithm described in the previous section inasmuch as originally the boundary conditions only served to improve the obtained results further. We mention again that the algorithm in its general version does not need the boundary conditions (and iteration) thanks to the high-order extrapolation.

- If $\mathcal{S}^E$ is the empty set, while deterministic transitions *can* be preempted, the ordinary differential equations do not exist, and the deviation of the sum of all state probabilities from the value 1.0 provides the only clues for an automatic stepsize control. However, if furthermore a fixed sequence of deterministic transitions which fires repeatedly can be identified, a specialized and more efficient algorithm has been described in [7], [9] and [10].

IV. Numerical aspects of the algorithm

In Section III, we have outlined the basic structure of the proposed algorithm. We now want to discuss some aspects related to the accuracy and order of the algorithm.

A. Relative precision

As opposed to earlier implementations of `TimeNET`, the new algorithm interpretes the input parameter *precision* ε as a relative precision. The parameter ε is used at various points of the algorithm:

- The step sizes of the two starting main steps will be guessed according to the orders of these steps and with respect to ε. For example, this means for the first step size h_1: $h_1 \approx \sqrt{\varepsilon}$. In our experiments, it proved to be efficient to reuse the same values in the starting procedure after every instant of discontinuity.
- Whenever Jensen's method has to be applied – as in substep 2 or substep 3 for obtaining interpolation points along the line $x = \tau^d$ –, the employed method of Fox/Glynn [18] requires ε in order to determine the upper and lower bounds for the computation of the relevant Poisson probabilities. We have slightly altered Jensen's method so that, after a period h, the relative errors in the components of the age density functions and of the impulses are in the order of magnitude of the parameter ε (see [19]).
- In the code of the method of Dormand/Prince for the integration of the ordinary differential equations (substep 4), the local error *err* of the numerical values of $\boldsymbol{\pi}^E(t+h)$ is estimated by a norm of the difference of two such approximations of different order. Opposed to the original code in [17], we have to operate with relatively precise quantities. If substep 4 can be accepted, i.e. $err \leq \varepsilon$, the new step size is roughly given by $h_{new} = h_{old} \cdot (\frac{\varepsilon}{err})^{\frac{1}{5}}$.
- According to the idea of the two-stage stepsize control, the proposed new step size h_{new} is subject to a reduction dependent on the deviation of the sum of all state probabilities from the ideal value 1.0 in substep 6. We determine the correcting factor for the main step from t_i to t_{i+1} by $\frac{|\sum_{d\in T^D} \|\boldsymbol{\pi}^d(t_{i+1})\|_\infty + \|\boldsymbol{\pi}^E(t_{i+1})\|_\infty - 1.0|}{i\cdot|T^D|\cdot\varepsilon}$. The main step is rejected, if the correcting factor is greater than 4.5, and h_{new} remains unchanged, if the factor is less than 1.0. For all other values, h_{new} is divided by the correcting factor.
- In substep 5, the newly computed components of the vector $\mathbf{r}^d(t_{i+1},0)$ are compared with their previously obtained equivalents. A relative error greater than ε in any entry will lead to another iteration of the substeps 3, 4 and 5.

We also point out that the measures listed above do not ensure that the quantities at the final instant of time are available in relative precision, since we only attempt to keep the *local* errors within the given tolerance. Unfortunately, to the best of the authors' knowledge, there seems to be no reasonable way to quantify the final errors for problems of the complexity considered here.

B. Choice of the numerical subalgorithms

In Section III-B, we have already remarked that all quantities of state probabilities and age density functions that flow into the integration of the ordinary differential equations by the method of Dormand/Prince ([17], abbreviated as DOPRI5 from now on) must be approximated in a sufficient quality so that the order of DOPRI5 is sustained. We will now examine the effects of this requirement on all other subalgorithms in more detail. Given the vector $\boldsymbol{\pi}^E(t_0)$, the numerical solution $\mathbf{y}_1$ for $\boldsymbol{\pi}^E(t_0+h)$ after one

TABLE I

EMPLOYED SUBALGORITHMS

application	*employed algorithm*
substep 4: - integration of the ordinary differential equations - approximation of $\mathbf{r}^d(t,\tau^d)$ and $\boldsymbol{\pi}^d(t)$ at different intermediate points ($d \in T^D$)	 method of Dormand/Prince (for the starting steps: lower-order explicit RK-methods without stepsize control) Newton interpolation ([16], allowing convenient replacement of one or two interpolation points)
substep 2: - multiplication with matrix exponential	 Jensen's method (also called randomization/uniformization)
substep 3: - extrapolation of $\mathbf{r}^d(t_{i+1},0)$ ($d \in T^D$) - approximation of $\mathbf{r}^d(t_{i+1},\tau^d)$ ($d \in T^D$) - integration of the age density functions	 Aitken-Neville algorithm Newton interpolation ([16]) segmented interpolation quadrature

step with size h is computed by

$$\mathbf{y}_1 = \boldsymbol{\pi}^E(t_0) + h \cdot \sum_{i=1}^{7} b_i \mathbf{k}_i \tag{21}$$

with $\mathbf{k}_i$ being successively determined from

$$\begin{aligned} \mathbf{k}_i \quad \equiv \quad & \boldsymbol{\pi}^E(t_0) \cdot \mathbf{Q}^{E,E} + h \cdot \left(\sum_{l=1}^{i-1} a_{il} \mathbf{k}_l \right) \cdot \mathbf{Q}^{E,E} \\ + \quad & \sum_{d \in T^D} \boldsymbol{\pi}^d(t_0 + c_i h) \cdot \mathbf{Q}^{d,E} \\ + \quad & \sum_{d \in T^D} \mathbf{r}^d(t_0 + c_i h, \tau^d) \cdot \boldsymbol{\Delta}^{d,E} \end{aligned} \tag{22}$$

(a_{ij}, c_i and b_i are the coefficients of the 7-stage, explicit RK-method DOPRI5, see [17], page 178).

Suppose that the functions $\mathbf{r}^d(t,\tau^d)$ and $\boldsymbol{\pi}^d(t)$ were known exactly, $\mathbf{y}_1$ locally represents a sixth-order approximation of $\boldsymbol{\pi}^E(t_0+h)$. So, due to the multiplication with h in (21), fifth-order approximations of the probabilities $\boldsymbol{\pi}^d(t)$ and the age density functions would be desirable at the intermediate points, but can only be obtained by exorbitant numerical efforts. Therefore, we will be satisfied with fourth-order approximations thus accepting a global order of four for the proposed algorithm. Jensen's method always provides sufficiently accurate values of the age density functions, for the input parameter *relative precision* makes it a method of variable order. Special care must be taken, when it comes to interpolation, extrapolation, and numerical integration. In all these cases, one has to make sure that the discontinuities of the age density functions do not coincide with an inner point of the interpolation intervals. The same rule must be observed for extrapolation intervals and integration intervals, and analogously for the instants of discontinuity and the functions $\boldsymbol{\pi}^d(t)$. In order to obtain fourth-order approximations for $\boldsymbol{\pi}^d(t_{i+1})$, we form integration intervals which generally stretch out over four subsequent nodes. On these intervals, cubic polynomials will be integrated instead of the true function. At the boundaries $x = \tau^d$ and $x = 0$ and in the neighborhood of discontinuities, we might be forced to restrict the integration intervals to three nodes allowing an integration of quadratic polynomials only. It can be shown that these exceptions do not impair the quality of the approximation due to their boundedness in number along the axis $t = t_{i+1}$. In Figure 3, we depicted this *segmented interpolation quadrature* by assigning white, grey and black fillings, respectively, to the interpolation points of the same integration interval. Without proof, we state that the order of approximation of $\boldsymbol{\pi}^d(t_{i+1})$ does not deteriorate when extrapolating $\mathbf{r}^d(t_{i+1},0)$ with the help of the *Aitken-Neville algorithm* [16] using the three nodes $\mathbf{r}^d(t_{i+1},h_{i+1})$, $\mathbf{r}^d(t_{i+1},h_{i+1}+h_i)$, and $\mathbf{r}^d(t_{i+1},h_{i+1}+h_i+h_{i-1})$. On the other hand, since the values of the functions $\mathbf{r}^d(t,\tau^d)$ and $\boldsymbol{\pi}^d(t)$ at the intermediate points are directly involved in (22), we always need four interpolation points to compute approximations of order four for these quantities by evaluating the corresponding interpolation polynomial at the intermediate points. The efficient administration of the interpolation points in both cases suggests to employ *Newton interpolation* [16] which enables us to replace one or two interpolation points with little additional costs.

In Table I, we summarize the subalgorithms which are applied in a main step from time t_i to t_{i+1}.

V. EXPERIMENTAL RESULTS

In this section, numerical experiments are performed in order to investigate the behavior of the analysis algorithm presented in this paper. The ΣMMPP/D/1/K-queueing system with failure and repair of Section II is used in all experiments. First, the transient curves of two measures illustrate a typical outcome of the analysis component. Second, the accuracy of the results is examined. Third, comparisons of the new algorithm and the former algorithm with fixed step size are given. Finally, the impact of stiffness on the solution complexity is demonstrated. If not stated otherwise, the parameters of the model are set as follows: $\rho_1 = 0.01$, $\rho_2 = 1.0$, $\lambda_1 = 0.2$, $\lambda_2 = 0.5$, $\tau = 3.0$, $\delta = 10.0$, and $\mu = 0.05$. The solution parameter *precision* ε equals 10^{-7}, and the transient analysis is performed up to the time $t = 120$ in V-B or $t = 100$ in V-C. All experiments are carried out on a Sun 4/50 workstation with

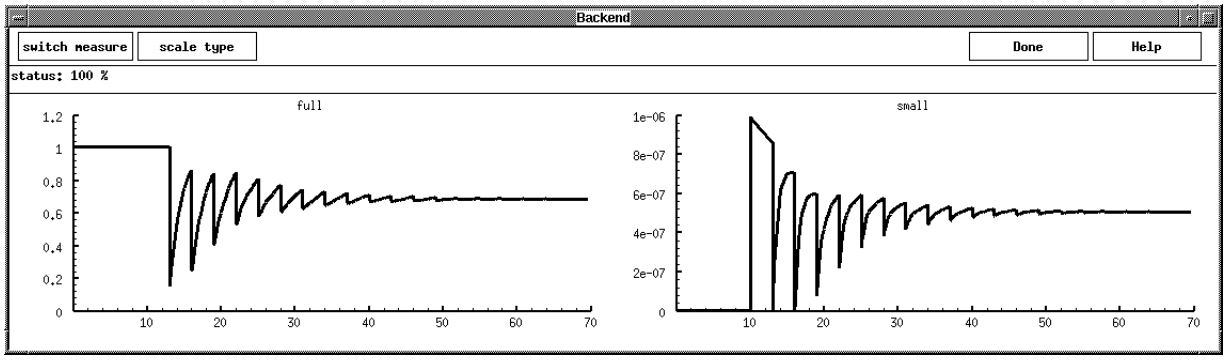


Fig. 4. *full* and *small* if a deterministic transition is initially enabled

TABLE II

A COMPARISON OF RESULTS OF THE TRANSIENT ANALYSIS AND OF THE STATIONARY ANALYSIS (SOFTWARE-TOOL **TimeNET**)

result measure	*transient analysis*		*stationary analysis*
	without initially enabled determ. trans.	*with initially enabled determ. trans.*	
small	$5.0044054 \cdot 10^{-7}$	$5.0043982 \cdot 10^{-7}$	$5.0060473 \cdot 10^{-7}$
full	0.6794014	0.6794001	0.6794134
sum	0.999998030	0.9999998961	1.00000000000

64 MB RAM running SunOS 4.1. (exception: the experiments for stiff problems are carried out on a DEC-Alpha 3000 APX workstation with 567 MB RAM running OSF2).

A. Illustration of transient results

For the model above with $K = 2$ and $N = 3$, i.e. three tokens are initially in place P1, two in place P4 and one token in place P7, the following result measures are considered:

- $full \equiv Pr\{\#tokens\ in\ \mathrm{P5} = 0\}$
- $small \equiv Pr\{\#tokens\ in\ \mathrm{P1} = 0 \wedge \#tokens\ in\ \mathrm{P5} = 0 \wedge \#tokens\ in\ \mathrm{P6} = 1\}$.

The measure *small* is an artificially chosen measure, which is particularly small. Figure 4 shows the results (**TimeNET** draws the curves in a pop-up window during computation).

B. Accuracy

Figure 4 suggests that, at time $t = 120.0$, the process will (almost) have reached its steady state so that it seems reasonable to compare the numerical results of the transient analysis at this final instant of time with those of the stationary analysis [4]. The results are given in Table II, which also lists the results, if no deterministic transition is initially enabled. In this case, both tokens of place P4 were moved to place P5 and the one of place P7 was put into place P6.

Sum denotes the sum of the numerical values of all state probabilities. As opposed to the algorithm presented in this paper, the stationary analysis forces *sum* to take on the ideal value 1.0 by normalization and therefore contains no clue on the quality of the numerical computations. We mention that, especially for small measures as *small*, additional numerical errors may be introduced, since an absolute precision of 10^{-16} has been adopted, which should not be confused with the user-defined parameter *precision*. In Figure 5, a (typical) curve is presented which shows the deviation of *sum* from its ideal value (with a deterministic transition being initially enabled).

Also, the adjustment to relative precision has significantly improved the reliability of the proposed algorithm. Compared to absolute precision, calculating the quantities in relative precision seems to be more expensive at first sight, because substep 2 requires more operations, substep 5 results in more iterations, and the step size control delivers more cautious guesses for the step sizes. Nevertheless, as the Figures 6 and 7 display, the algorithm in relative precision needs unsignificantly more processor time and memory space; in cases where more precision is desired substantially less resources are consumed. Apart from the indications given in Section IV, this is also due to the fact that fewer steps are rejected. The better handling of the transient state equations can also be expressed in terms of the deviation of *sum* from its ideal value at time $t = 100.0$: For *precision* equal to 10^{-7}, we found a deviation of $6.12 \cdot 10^{-7}$ as compared to $1.56 \cdot 10^{-4}$ in the absolute case.

Further investigations provided the insight that, as long as the state probabilities are subject to considerable changes, the second stage of the stepsize control in substep 6 does decrease the proposed step size from substep 4 more often, whereas after this phase the stepsize control of substep 4 basically remains unaffected. A possible explanation is that then the errors due to interpolation and numerical integration are less significant than in the beginning.

[4]Stationary solutions were provided by the stationary analysis component of **TimeNET** employing an algorithm based on an embedded Markov chain [4].

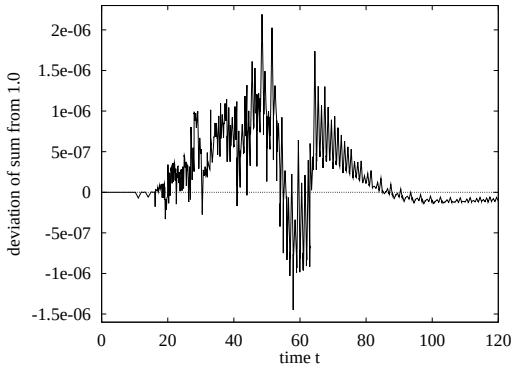


Fig. 5. Deviation of *sum* from its ideal value 1.0 ($N = 3$, $K = 2$, $precision = 10^{-7}$)

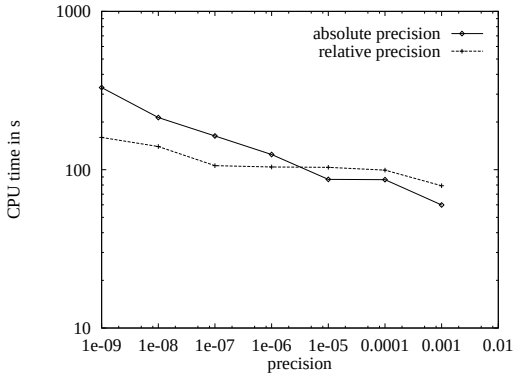


Fig. 6. CPU time versus precision ($K = 10, N = 3$)

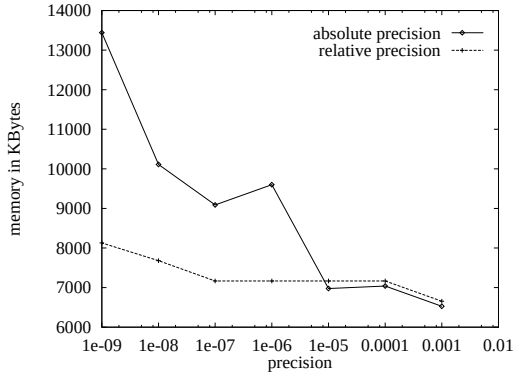


Fig. 7. Memory requirements versus precision ($K = 10, N = 3$)

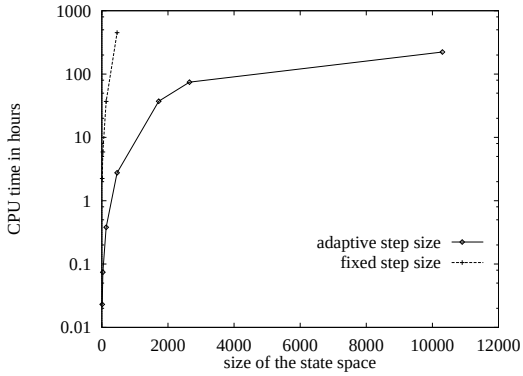


Fig. 8. CPU-time versus state space

C. A comparison with the fixed step size algorithm

With respect to accuracy, the transient analysis component without automatic stepsize control cannot compete with the new implementation. However, an empirical comparison of time and space efficiency is still interesting. We choose a fixed stepsize of 0.01, a choice that favors the former algorithm, since the step size would have to be selected much smaller, if error propagation was investigated thoroughly.

In Figures 8 and 9, the CPU-time and memory requirements are given for different sizes of the state space for the old and the new implementation, respectively. Both parameters K and N are varied to increase the state space with a constant ratio $\frac{K}{N}$.

Clearly, the adaptive algorithm performs much better than the one with fixed step size with respect to both aspects. The processing time could be reduced by about two orders of magnitude. When removing the subnet of the ΣMMPP/D/1/K-queueing system that models failure and repair of the system, the specified results for models with up to 100,000 states would be obtained within comparable processing times. This is due to the fact that any additional deterministic transition adds substantially to the computational effort. It should be mentioned that, in accordance with measurements in [10], substep 2 – the application of Jensen's method – accounts for almost all the costs of the algorithm.

D. Behavior for stiff problems

The proposed algorithm was designed for the general class of DSPNs as described in Section II. In many cases, an explicit one-step method is well suited for the integration of the transient state equations. However, when dealing with stiff problems, the proposed algorithm might turn out to be inefficient. Stiffness occurs, if there are negative eigenvalues of the matrices $\mathbf{Q}^d$ ($d \in T^D$) or $\mathbf{Q}^{E,E}$ that differ by some orders of magnitude. For our queueing system with failure and repair, stiffness increases with a growing rate ρ_2, while ρ_1 is fixed at 0.01. In other words, the ratio $\frac{\rho_2}{\rho_1}$, which is given on the horizontal axis of Figure 10, may serve as a measure of stiffness in the example. The parameters K and N are set to 1 and 2.

Step rejections, which will take place more frequently, if the problem is stiff, lead to much longer processing times. But as opposed to the algorithm with fixed stepsize, the adaptive algorithm still guarantees result measures in the expected accuracy.

VI. Conclusions

We have proposed a fourth-order algorithm with automatic stepsize control for the transient analysis of DSPNs. It is based on the numerical analysis of the transient state equations derived by the method of supplementary variables. A careful investigation of the influence of initially

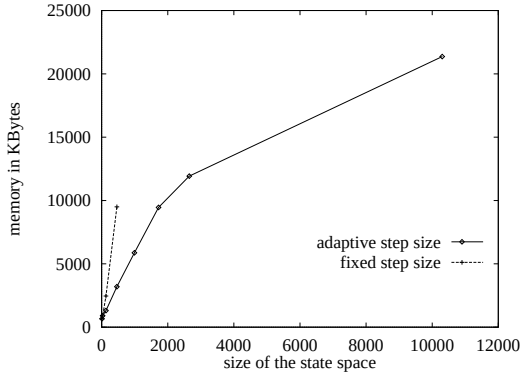


Fig. 9. Memory requirements versus state space

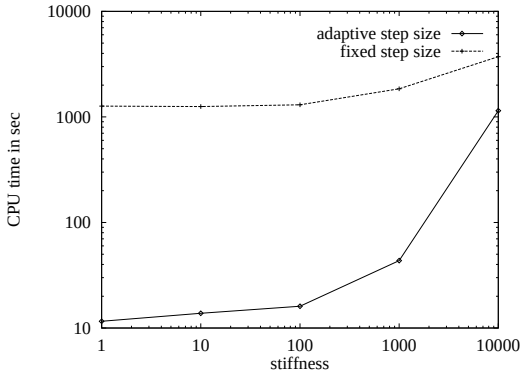


Fig. 10. CPU-time versus stiffness ($K = 1$, $N = 2$)

enabled deterministic transitions resulted in a splitting up of the state equations into a continuous and a discrete part. The discrete part allows us to deal with discontinuities in a formal way. Irregularities, like discontinuities and Dirac impulses, may lead to jumps of the state probabilities or their derivatives and have to be taken into account by a reliable algorithm.

An implementation of the algorithm replaced the former transient analysis component of the software tool `TimeNET` [13]. Apart from now being able to deal with initially enabled deterministic transitions, this new implementation performs much better than its predecessor in every aspect, as we have illustrated by the example of a ΣMMPP/D/1/K-queueing system with failure and repair. The two-stage automatic stepsize control substantially reduces processing times as well as storage requirements while enhancing the reliability of the results at the same time. The explicit consideration of the discontinuities, the choice of well-suited numerical subalgorithms, the core of which is an explicit embedded Runge-Kutta method of Dormand/Prince, and the adjustment to relative precision helped to improve efficiency and accuracy of the adaptive algorithm. Our experiments suggest that the new implementation performs very efficiently, at least as long as the encountered problems do not become too stiff.

During the development of the algorithm, we have taken care that it can be adapted to the transient analysis of Markov regenerative stochastic Petri nets (in which deterministic transitions are substituted by transitions with generally distributed firing times). However, since the number of instants of discontinuity drastically increases in this case, we recommend to lower the order of the proposed algorithm to three.

Acknowledgments

The authors would like to thank R. D. Grigorieff for fruitful discussions on various numerical topics.

References

[1] H. Choi, V. G. Kulkarni, and K. S. Trivedi, "Markov Regenerative Stochastic Petri Nets," *Performance Evaluation*, vol. 20, pp. 337–357, 1994.

[2] G. Ciardo, R. German, and C. Lindemann, "A Characterization of the Stochastic Process Underlying a Stochastic Petri Net," *IEEE Trans. Softw. Engin.*, vol. 20, pp. 506–515, 1994.

[3] A. Bobbio and M. Telek, "Markov Regenerative SPN with Non-Overlapping Activity Cycles," in *Proc. IEEE Int. Performance and Dependability Symp.*, Erlangen, Germany, 1995, pp. 124–133.

[4] M. Ajmone Marsan and G. Chiola, "On Petri Nets with Deterministic and Exponentially Distributed Firing Times," in *Advances in Petri Nets*, G. Rozenberg, Ed., vol. 266 of *LNCS*, pp. 132–145. Springer Verlag, 1987.

[5] H. Choi, V. G. Kulkarni, and K. S. Trivedi, "Transient Analysis of Deterministic and Stochastic Petri Nets," in *Proc. 14th Int. Conf. on Application and Theory of Petri Nets*, Chicago, IL, USA, 1993, vol. 691 of *LNCS*, pp. 166–185, Springer Verlag.

[6] M. Telek, A. Bobbio, and A. Puliafito, "Steady State Solution of MRSPNs with Mixed Preemption Policies," in *Proc. IEEE Int. Performance and Dependability Symp.*, Urbana-Champaign, IL, USA, 1996, pp. 106–115.

[7] D. Logothetis, K. S. Trivedi, and A. Puliafito, "Markov Regenerative Models," in *Proc. IEEE Int. Performance and Dependability Symp.*, Erlangen, Germany, 1995, pp. 134–142.

[8] D. R. Cox, "The Analysis of Non-Markovian Stochastic Processes by the Inclusion of Supplementary Variables," *Proc. Camb. Phil. Soc. (Math. and Phys. Sciences)*, vol. 51, pp. 433–441, 1955.

[9] R. German, "New Results for the Analysis of Deterministic and Stochastic Petri Nets," in *Proc. IEEE Int. Performance and Dependability Symp.*, Erlangen, Germany, 1995, pp. 114–123.

[10] R. German and J. W. Mitzlaff, "Transient Analysis of Deterministic and Stochastic Petri Nets with TimeNET," in *Proc. Joint Conferences 8th Int. Conf. on Modelling Techniques and Tools for Performance Evaluation and 8th GI/ITG Conf. on Measuring, Modelling and Evaluating Computing and Communication Systems*, Heidelberg, Germany, 1995, vol. 977 of *LNCS*, pp. 209–223, Springer Verlag.

[11] R. German, A. P. A. van Moorsel, M. A. Qureshi, and W. H. Sanders, "Expected Impulse Rewards in Markov Regenerative Stochastic Petri Nets," in *Proc. 17th Int. Conf. on Application and Theory of Petri Nets*, Osaka, Japan, 1996, vol. 1091 of *LNCS*, pp. 172–191, Springer Verlag.

[12] R. German, D. Logothetis, and K. S. Trivedi, "Transient Analysis of Markov Regenerative Stochastic Petri Nets: A Comparison of Approaches," in *Proc. 6th Int. Workshop on Petri Nets and Performance Models - PNPM'95*, Durham, NC, USA, Oct. 1995, pp. 103–112, IEEE-CS Press.

[13] R. German, C. Kelling, A. Zimmermann, and G. Hommel, "TimeNET: A Toolkit for Evaluating Non-Markovian Stochastic Petri Nets," *Performance Evaluation*, vol. 24, pp. 69–87, 1995.

[14] M. Ajmone Marsan, G. Balbo, A. Bobbio, G. Chiola, G. Conte, and A. Cumani, "The Effect of Execution Policies on the Semantics of Stochastic Petri Nets," *IEEE Trans. Softw. Engin.*, vol. 15, pp. 832–846, 1989.

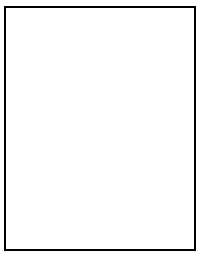
[15] W. K. Grassmann, "Transient Solutions in Markovian Queueing Systems," *Comput. & Ops Res.*, vol. 4, no. 1, pp. 47–53, 1977.

[16] G. Hämmerlin and K. H. Hoffmann, *Numerische Mathematik*, Springer Verlag, 1991 (in German).

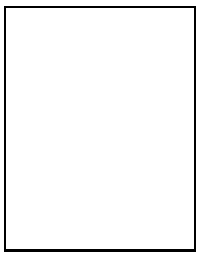
[17] E. Hairer, S. P. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations I*, Springer Verlag, 1993.

[18] B. L. Fox and P. W. Glynn, "Computing Poisson Probabilities," *Comm. of the ACM*, vol. 31, pp. 440–445, 1988.

[19] A. Heindl, "Entwicklung eines numerischen adaptiven Lösungsverfahrens zur transienten Analyse von stochastischen Petri-Netzen," M.S. thesis, TU Berlin, Computer Science Department, Germany, 1996.



Armin Heindl studied applied mathematics at the University of Kaiserslautern and the Technische Universität Berlin where he received the degree Diplom-Technomathematiker in 1996. In his master's thesis he worked on the transient analysis of DSPNs.
Currently, he is a Ph.D. student in a research project on the performance evaluation of wireless local area networks funded by the German Research Council (DFG). His research interests include applied probability theory, performance and dependability modeling, and stochastic networks.



Reinhard German received the MS in computer science in 1991 and the PhD in engineering in 1994, both from the Technische Universität Berlin. He received a doctoral fellowship from the German Research Council (DFG) and worked on an industrial research project funded by the Siemens Corporate, Research and Development.
He is currently an assistant professor in the Department of Computer Science at the Technische Universität Berlin. His research interests include performance and dependability modeling, stochastic Petri nets, analysis of non-Markovian systems, and application to communication systems. He is a member of the German Society for Computer Science (GI) and of the ACM.