RWTH Aachen

Department of Computer Science
Technical Report

Compositional Modeling and Minimization of Time-Inhomogeneous Markov Chains

Tingting Han and Joost-Pieter Katoen and Alexandru Mereacre

ISSN 0935–3232 · Aachener Informatik Berichte · AIB-2007-21

RWTH Aachen · Department of Computer Science · December 2007

The publications of the Department of Computer Science of *RWTH Aachen University* are in general accessible through the World Wide Web.

http://aib.informatik.rwth-aachen.de/

Compositional Modeling and Minimization of Time-Inhomogeneous Markov Chains

Tingting Han[1,2], Joost-Pieter Katoen[1,2], and Alexandru Mereacre[1]

[1] RWTH Aachen University, Software Modeling and Verification Group, Germany
{tingting.han,katoen,mereacre}@cs.rwth-aachen.de
[2] University of Twente, Formal Methods and Tools Group, The Netherlands

Abstract. This paper presents a compositional framework for the modeling of interactive continuous-time Markov chains with time-dependent rates, a subclass of communicating piecewise deterministic Markov processes. A poly-time algorithm is presented for computing the coarsest quotient under strong bisimulation for rate functions that are either piecewise uniform or (piecewise) polynomial. Strong as well as weak bisimulation are shown to be congruence relations for the compositional framework, thus allowing component-wise minimization. In addition, a new characterization of transient probabilities in time-inhomogeneous Markov chains with piecewise uniform rates is provided.

1 Introduction

Modeling large stochastic discrete-event dynamic systems is a difficult task that typically requires human intelligence and ingenuity. To facilitate this process, formalisms are needed that allow for the modeling of such systems in a compositional manner. This allows to construct models of simpler components—usually from first principles—that can be combined by appropriate composition operators to yield complete system models. In concurrency theory, process algebra [Mil89,Hoa85] has emerged as an important framework to achieve compositionality: it provides a formal apparatus for compositional reasoning about structure and behavior of systems, and features abstraction mechanisms allowing system components to be treated as black boxes.

Although originally aimed at purely functional behavior, process algebras for stochastic systems have been investigated thoroughly, see e.g., [HHK02,Her02]. In all these approaches, the dynamics of the stochastic models is assumed to be *time-homogeneous*, i.e., the probabilistic nature of mode transitions as well as the time-driven behavior are independent of the global time. This is, however, a serious drawback to adequately model random phenomena that occur in practice such as failure rates of hardware components (a bath-tub curve), software reliability (which reduces due to memory leaks and increases after a restart), and battery depletion (where the power extraction rate non-linearly depends on the remaining amount of energy [CJH07]), to mention a few. This paper attempts to overcome this deficiency by providing a process algebra for time-*inhomogeneous* continuous-time Markov chains (ICTMCs). This is a very versatile class of models and is a natural stepping-stone towards more full-fledged stochastic hybrid system models such as piecewise deterministic Markov processes (PDPs [Dav93]). We show that ICTMCs can be compositionally modeled by using a time-dependent adaptation of the framework of interactive Markov chains (IMCs) [Her02]. To facilitate this, ICTMCs are equipped with the potential for interaction, i.e., synchronization. Instrumental to this approach is the memoryless property of ICTMCs.

More importantly though, notions of strong and weak bisimulation are defined and shown to be congruences. Together with efficient quotienting algorithms this allows for the component-wise minimization of hierarchical ICTMC models. Finally, we present an axiomatization for strong and weak bisimulation which allows to simplify models by pure syntactic manipulations as opposed to performing minimization on the model level [3]. As a generalization of results on ordinary lumpability on Markov chains [Buc94], we show that strong bisimulation preserves transient and long-run state probabilities in ICTMCs. This allows to minimize symbolically ICTMCs prior to their analysis.

We present a bisimulation minimization algorithm to obtain the *coarsest* (and thus smallest) *strong bisimulation quotient* of a large class of interactive ICTMCs, viz. those that have piecewise uniform—rate $\mathbf{R}_k(t)$ on piece k is of the form $f_k(t)\cdot\mathbf{R}$ for integrable function $\mathbf{R}$—polynomial, or piecewise polynomial—where each polynomial is of degree three—rate functions. The worst-case time and space complexity is $\mathcal{O}\left(m_a \lg(n) + M m_r \lg(n)\right)$ and $\mathcal{O}\left(m_a + m_r\right)$, respectively, where $M{+}1$ is the number of pieces (or degrees of the polynomial), m_a is the number of action-labeled transitions and m_r the number of rate-labeled transitions in the ICTMC under consideration. This algorithm is based on the partition-refinement bisimulation algorithm for Markov chains by Derisavi *et al.* [DHS03] and Paige-Tarjan's algorithm for labeled transition systems (LTS) [PT87].

Related work. ICTMCs are related to piecewise deterministic Markov processes (PDPs), a more general class of continuous-time stochastic discrete-event dynamic systems proposed by Davis [Dav93]. The probabilistic nature of mode transitions in PDPs is as for ICTMCs; in fact, ICTMCs are a subclass of PDPs when the global time t has a clock dynamics i.e., $\dot{t} = 1$. The notion of parallel composition of ICTMCs corresponds to that for communicating PDPs (CPDPs) as introduced by Strubbe and van der Schaft [SvdS06,SJvdS03]. Alternative modeling formalisms for PDPs are, e.g., variants of colored Petri nets [EB05] but they lack a clear notion of compositionality. Compositional modeling formalisms for hybrid systems have been considered by, e.g., [AGLS06,AGH$^+$00]. Strong bisimulation has been proposed for several classes of (stochastic) hybrid systems, see e.g., [BLB05,HTP05,SvdS05]. Our notion of bisimulation is closely related to that for CPDPs [SvdS05] but differs in the fact that the maximal progress assumption—a race between one or more rates and a transition that is not subject to interaction with the environment is resolved in favor of the internal transition—is not considered in [SvdS05].

2 Inhomogeneous Continuous Time Markov Chains

Definition 1 (ICTMC). *An inhomogeneous continuous-time Markov chain is a tuple* $\mathcal{C} = (\mathbb{S}, \mathbf{R})$ *where:* $\mathbb{S} = \{1, 2, \ldots, n\}$ *is a finite set of states, and* $\mathbf{R}(t) = [R_{i,j}(t) \geq 0] \in \mathrm{I\!R}^{n \times n}$ *is a time-dependent rate matrix, where* $R_{i,j}(t)$ *is the rate between states* $i, j \in \mathbb{S}$ *at time* $t \in \mathrm{I\!R}_{\geq 0}$.

[3] The proofs for the congruence results and the axiomatization have been obtained by standard proof techniques and are omitted from the paper.

Let diagonal matrix $\mathbf{E}(t) = \mathrm{diag}\,[E_i(t)] \in \mathbb{R}^{n\times n}$, where $E_i(t) = \sum_{j\in\mathbb{S}} R_{i,j}(t)$ for all $i, j \in \mathbb{S}$, $i \neq j$ i.e., $E_i(t)$ is the total exit rate of state i at time t. Consider a non-homogeneous Poisson process $\{Z(t)|t \geq 0\}$ with rate $R(t)$. The probability of k arrivals in the interval $[t, t+\Delta t]$ is:

$$\Pr\{Z(t+\Delta t) - Z(t) = k\} = \frac{\left[\int_t^{t+\Delta t} R(\ell)d\ell\right]^k}{k!} e^{-\int_t^{t+\Delta t} R(\ell)d\ell}, \quad k = 0, 1, \ldots.$$

The probability that there will be no arrivals in the interval $[t, t+\Delta t]$ is:

$$\Pr\{Z(t+\Delta t) - Z(t) = 0\} = e^{-\int_t^{t+\Delta t} R(\ell)d\ell} = e^{-\int_0^{\Delta t} R(t+\ell)d\ell}. \tag{1}$$

Let the random variable $W_{i,j}(t)$ be the firing time of transition $i \to j$ ($i, j \in \mathbb{S}$) with rate $R_{i,j}(t)$ at time t. From (1) we obtain the cumulative probability distribution of the firing time of transition $i \to j$:

$$\Pr\{W_{i,j}(t) \leq \Delta t\} = 1 - \Pr\{Z(t+\Delta t) - Z(t) = 0\} = 1 - e^{-\int_0^{\Delta t} R_{i,j}(t+\ell)d\ell}. \tag{2}$$

Probability measures. For every ICTMC one can specify measures of interest. These measures are either related to the states or to the transitions of an ICTMC. Consider a random variable $W_i(t)$ which denotes the waiting time in state i.

Property 1.

$$\Pr\{W_i(t) \leq \Delta t\} = 1 - e^{-\int_0^{\Delta t} E_i(t+\ell)d\ell}. \tag{3}$$

An intuitive explanation of (3) is that the waiting time $W_i(t)$ in state i is determined by the minimal firing time of all k outgoing transitions from state i, i.e., $W_i(t) = \min\{W_{i,1}(t), \ldots, W_{i,k}(t)\}$. When $R_{i,j}(t) = R_{i,j}$ and $E_i(t) = E_i$ for all $t \in \mathbb{R}_{\geq 0}$, i.e., the ICTMC is a CTMC, $W_i(t)$ has the distribution $1 - e^{-E_i \Delta t}$. An interesting property is that the waiting time in any state i is *memoryless*, i.e.:

$$\Pr\left\{W_i(t) \leq t' + \Delta t | W_i(t) > t'\right\} = \Pr\left\{W_i(t+t') \leq \Delta t\right\}. \tag{4}$$

This can be shown as follows:

$$\begin{aligned}\Pr\left\{W_i(t) \leq t' + \Delta t | W_i(t) > t'\right\} &= \frac{e^{-\int_0^{t'} E_i(t+\ell)d\ell} - e^{-\int_0^{t'+\Delta t} E_i(t+\ell)d\ell}}{e^{-\int_0^{t'} E_i(t+\ell)d\ell}} \\ =1 - e^{-\int_0^{t'+\Delta t} E_i(t+\ell)d\ell + \int_0^{t'} E_i(t+\ell)d\ell} &= \Pr\left\{W_i(t+t') \leq \Delta t\right\}.\end{aligned}$$

Equation (4) will be of importance when we later define a calculus for ICTMCs.

Property 2. The probability $\Pr_{i,j}(t)$ to select transition $i \to j$ ($i \neq j, i, j \in \mathbb{S}$) with rate $R_{i,j}(t)$ at time t is:

$$\Pr_{i,j}(t) = \int_0^\infty R_{i,j}(t+\tau) e^{-\int_0^\tau E_i(t+\ell)d\ell} d\tau. \tag{5}$$

When rates are constant, the measure (5) takes the form $\Pr_{i,j} = \frac{R_{i,j}}{E_i}$ ($\Pr_{i,j}(t) = \Pr_{i,j}$ for all $t \in \mathbb{R}_{\geq 0}$), which corresponds to transition probability in CTMCs. An important relation which always holds for CTMCs is that the probability to leave a non-absorbing state i ($E_i(t) \neq 0$ for all $t \in \mathbb{R}_{\geq 0}$) equals $\sum_{j\in\mathbb{S}} \Pr_{i,j} = 1$. On the other hand, for ICTMCs, $\sum_{j\in\mathbb{S}} \Pr_{i,j}(t) = 1$ only holds when $\lim_{\tau\to\infty} \int_0^\tau E_i(\ell)d\ell = \infty$.

Property 3. The cumulative probability distribution $\mathrm{Pr}_{i,j}(t, \varDelta t)$ to move from state i to state j $(i \neq j)$ with rate $R_{i,j}(t)$ in $\varDelta t$ units of time starting at time t is:

$$\mathrm{Pr}_{i,j}(t, \varDelta t) = \int_0^{\varDelta t} R_{i,j}(t+\tau) e^{-\int_0^{\tau} E_i(t+\ell) d\ell} d\tau. \tag{6}$$

Notice that (6) is the same as (5) except that the range of the outer-most integral is $[0, \varDelta t]$. For CTMCs, equation (6) results ($\mathrm{Pr}_{i,j}(t, \varDelta t) = \mathrm{Pr}_{i,j}(\varDelta t)$ for all $t \in \mathbb{R}_{\geq 0}$) in $\mathrm{Pr}_{i,j}(\varDelta t) = \frac{R_{i,j}}{E_i}\left(1 - e^{-E_i \varDelta t}\right)$.

Transient probability distribution. One important measure which quantifies the probability to be in a specific state at some time point is the *transient probability distribution.* Consider an ICTMC described by the stochastic process $\{X(t) | t \geq 0\}$. The transient probability distribution $\Pr\{X(t+\varDelta t) = j\}$, denoted by $\pi_j(t+\varDelta t)$, is the probability to be in state j at time $t + \varDelta t$, and is described by the equation:

$$\pi_j(t+\varDelta t) = \sum_{i \in \mathbb{S}} \Pr\{X(t) = i\} \cdot \Pr\{X(t+\varDelta t) = j | X(t) = i\}. \tag{7}$$

Equation (7) can be expressed in matrix form as: $\boldsymbol{\pi}(t+\varDelta t) = \boldsymbol{\pi}(t)\boldsymbol{\Phi}(t+\varDelta t, t)$, where $\boldsymbol{\pi}(t) = [\pi_1(t), \ldots, \pi_n(t)]$ and $\boldsymbol{\Phi}(t+\varDelta t, t)$ represents the *transition probability matrix.* This equation represents the solution of a system of ODEs:

$$\frac{d\boldsymbol{\pi}(t)}{dt} = \lim_{\varDelta t \to 0} \frac{\boldsymbol{\pi}(t+\varDelta t) - \boldsymbol{\pi}(t)}{\varDelta t} = \lim_{\varDelta t \to 0} \boldsymbol{\pi}(t) \frac{[\boldsymbol{\Phi}(t+\varDelta t, t) - \mathbf{I}]}{\varDelta t}. \tag{8}$$

For the diagonal elements $q_{i,i}(t)$ of the matrix $\lim_{\varDelta t \to 0} \frac{[\boldsymbol{\Phi}(t+\varDelta t, t) - \mathbf{I}]}{\varDelta t}$ from (8), we obtain:

$$q_{i,i}(t) = \lim_{\varDelta t \to 0} \frac{\Pr\{X(t+\varDelta t) = i | X(t) = i\} - 1}{\varDelta t}.$$

As $\Pr\{X(t+\varDelta t) = i | X(t) = i\}$ denotes the probability to stay in state i for at least $\varDelta t$ units of time or the probability to return to state i in two or more steps, it follows:

$$q_{i,i}(t) = \lim_{\varDelta t \to 0} \frac{e^{-\int_0^{\varDelta t} E_i(t+\ell) d\ell} - 1 + o(\varDelta t)}{\varDelta t} = -E_i(t),$$

where $o(\varDelta t)$ denotes the probability to make two or more transitions in $\varDelta t$ units of time. Notice that $\lim_{\varDelta t \to 0} \frac{o(\varDelta t)}{\varDelta t} = 0$. For the off-diagonal elements $q_{i,j}(t)$ $(i \neq j)$ of matrix $\lim_{\varDelta t \to 0} \frac{[\boldsymbol{\Phi}(t+\varDelta t, t) - \mathbf{I}]}{\varDelta t}$, the relation is similar:

$$q_{i,j}(t) = \lim_{\varDelta t \to 0} \frac{\Pr\{X(t+\varDelta t) = j | X(t) = i\}}{\varDelta t} = \lim_{\varDelta t \to 0} \frac{\mathrm{Pr}_{i,j}(t, \varDelta t) + o(\varDelta t)}{\varDelta t},$$

which can be reduced using (6) to:

$$q_{i,j}(t) = \lim_{\varDelta t \to 0} \frac{\int_0^{\varDelta t} R_{i,j}(t+\tau) e^{-\int_0^{\tau} E_i(t+\ell) d\ell} d\tau + o(\varDelta t)}{\varDelta t} = R_{i,j}(t).$$

The resulting *infinitesimal generator* matrix $\mathbf{Q}(t)$ has the form:

$$\mathbf{Q}(t) = \lim_{\varDelta t \to 0} \frac{[\boldsymbol{\Phi}(t+\varDelta t, t) - \mathbf{I}]}{\varDelta t} = \mathbf{R}'(t) - \mathbf{E}(t),$$

where $\mathbf{R'}$ equals $\mathbf{R}$ except that $R'_{i,i}(t) = 0$. Plugging $\mathbf{Q}(t)$ into equation (8) yields the system of ODEs which describe the evolution of transient probability distribution over time (Chapman-Kolmogorov equations):

$$\frac{d\boldsymbol{\pi}(t)}{dt} = \boldsymbol{\pi}(t)\mathbf{Q}(t), \qquad \sum_{i=1}^{n} \pi_i(t_0) = 1, \tag{9}$$

where $\boldsymbol{\pi}(t_0)$ is the initial condition. From the literature (see [Kai80, pages 594–631]) it is known that the solution $\boldsymbol{\pi}(t)$ of (9), written as:

$$\boldsymbol{\pi}(t) = \boldsymbol{\pi}(t_0)\boldsymbol{\Phi}(t, t_0) \tag{10}$$

has the transition probability matrix given by the Peano-Baker series:

$$\boldsymbol{\Phi}(t, t_0) = \mathbf{I} + \int_{t_0}^{t} \mathbf{Q}(\tau_1)d\tau_1 + \int_{t_0}^{t} \mathbf{Q}(\tau_1) \int_{t_0}^{\tau_1} \mathbf{Q}(\tau_2)d\tau_2 d\tau_1 + \dots. \tag{11}$$

Note that if $\mathbf{Q}(\tau_1)\int_{t_0}^{\tau_1} \mathbf{Q}(\tau_2)d\tau_2 = \int_{t_0}^{\tau_1} \mathbf{Q}(\tau_2)d\tau_2\mathbf{Q}(\tau_1)$ then $\boldsymbol{\Phi}(t, t_0) = e^{\int_{t_0}^{t} \mathbf{Q}(\tau)d\tau}$. If the rate matrix $\mathbf{R}(t)$ is *piecewise constant* i.e., $\mathbf{R}(t) = \mathbf{R}_k$ or $\mathbf{Q}(t) = \mathbf{Q}_k$ for all $t \in [t_k, t_{k+1})$ and $k \le M \in \mathbb{N}$ ($M + 1$ is the total number of constant pieces), equation (10) can also be rewritten as (see [RWVT95]):

$$\boldsymbol{\pi}(t) = \begin{cases} \boldsymbol{\pi}(t_0)e^{\mathbf{Q}_0(t-t_0)} & \text{if } t \in [t_0, t_1) \\ \vdots & \vdots \\ \boldsymbol{\pi}(t_M)e^{\mathbf{Q}_M(t-t_M)} & \text{if } t \in [t_M, \infty) \end{cases} \quad \text{and } \boldsymbol{\pi}(t_k) = \boldsymbol{\pi}(t_{k-1})e^{\mathbf{Q}_{k-1}(t_k - t_{k-1})}.$$

The general case is when the rate matrix is *piecewise uniform* i.e., $\mathbf{R}(t) = \mathbf{R}_k(t) = f_k(t)\mathbf{R}_k$ or $\mathbf{Q}(t) = \mathbf{Q}_k(t) = f_k(t)\mathbf{Q}_k$ for any integrable function $f_k(t) : \mathbb{R}_{\ge 0} \to \mathbb{R}_{\ge 0}$ on time interval $[t_k, t_{k+1})$, constant matrices $\mathbf{R}_k$, $\mathbf{Q}_k$ and $k \le M$.

Theorem 1. *The transient probability distribution* $\boldsymbol{\pi}(t)$ *of an* ICTMC $\mathcal{C} = (\mathbb{S}, \mathbf{R})$ *with a piecewise uniform rate matrix* $\mathbf{R}(t)$ *and* $M{+}1$ *pieces is given by:*

$$\boldsymbol{\pi}(t) = \begin{cases} \boldsymbol{\pi}(t_0)e^{\mathbf{Q}_0 \int_{t_0}^{t} f_0(\tau)d\tau} & \text{if } t \in [t_0, t_1) \\ \vdots & \vdots \\ \boldsymbol{\pi}(t_M)e^{\mathbf{Q}_M \int_{t_M}^{t} f_M(\tau)d\tau} & \text{if } t \in [t_M, \infty) \end{cases}$$

where $\boldsymbol{\pi}(t_k) = \boldsymbol{\pi}(t_{k-1})e^{\mathbf{Q}_{k-1}\int_{t_{k-1}}^{t_k} f_{k-1}(\tau)d\tau}$.

Proof. When the rate matrix $\mathbf{R}(t)$ is piecewise uniform with $M{+}1$ pieces, the infinitesimal generator $\mathbf{Q}(t)$ takes the form:

$$\mathbf{Q}(t) = \begin{cases} \mathbf{Q}_0(t) & \text{if } t \in [t_0, t_1) \\ \vdots & \vdots \\ \mathbf{Q}_M(t) & \text{if } t \in [t_M, \infty) \end{cases}$$

where $\mathbf{Q}(t) = \mathbf{Q}_k(t) = f_k(t)\mathbf{Q}_k$ for all $t \in [t_k, t_{k+1})$ and $k \le M$. Therefore, we can describe the transient probability distribution as follows:

$$\boldsymbol{\pi}(t) = \begin{cases} \boldsymbol{\pi}(t_0)\boldsymbol{\Phi}(t, t_0) & \text{if } t \in [t_0, t_1) \\ \vdots & \vdots \\ \boldsymbol{\pi}(t_M)\boldsymbol{\Phi}(t, t_M) & \text{if } t \in [t_M, \infty) \end{cases}.$$

Notice that $\boldsymbol{\pi}(t_k) = \boldsymbol{\pi}(t_{k-1})\boldsymbol{\Phi}(t_k, t_{k-1})$. Now the question is: how can we compute $\boldsymbol{\Phi}$? As we already know, $\boldsymbol{\Phi}$ is given by the Peano-Baker series, cf. Equation (11). The characterization of $\boldsymbol{\Phi}$ in terms of the Peano-Baker series does not yield an effective computational recipe for $\boldsymbol{\Phi}$. Note however that the infinitesimal generator $\mathbf{Q}(t)$ commutes. Let us consider the k'th piece i.e., $\mathbf{Q}(t) = \mathbf{Q}_k(t)$ for all $t \in [t_k, t_{k+1})$. The fact that $\mathbf{Q}_k(t)$ commutes follows from:

$$\begin{aligned}
\mathbf{Q}_k(t)\int_{t_k}^{t}\mathbf{Q}_k(\tau)d\tau &= \int_{t_k}^{t}\mathbf{Q}_k(\tau)d\tau\mathbf{Q}_k(t) \Leftrightarrow \\
f_k(t)\mathbf{Q}_k\int_{t_k}^{t}f_k(t)\mathbf{Q}_k d\tau &= \int_{t_k}^{t}f_k(t)\mathbf{Q}_k d\tau f_k(t)\mathbf{Q}_k \Leftrightarrow \\
\mathbf{Q}_k^2 f_k(t)\int_{t_k}^{t}f_k(t)d\tau &= \int_{t_k}^{t}f_k(t)d\tau f_k(t)\mathbf{Q}_k^2.
\end{aligned}$$

Therefore, $\boldsymbol{\Phi}(t, t_k) = e^{\mathbf{Q}_k\int_{t_k}^{t}f_k(\tau)d\tau}$. The transient probability distribution becomes:

$$\boldsymbol{\pi}(t) = \begin{cases} \boldsymbol{\pi}(t_0)e^{\mathbf{Q}_0\int_{t_0}^{t}f_0(\tau)d\tau} & \text{if } t \in [t_0, t_1) \\ \vdots & \vdots \\ \boldsymbol{\pi}(t_M)e^{\mathbf{Q}_M\int_{t_M}^{t}f_M(\tau)d\tau} & \text{if } t \in [t_M, \infty) \end{cases}$$

3 Inhomogeneous Interactive Markov Chains

In order to facilitate the compositional modeling of ICTMCs, we equip these processes with the capability to allow for their mutual interaction. This is established by adding actions to ICTMCs. Let *Act* be the countable universe of actions. The aim of these actions is that certain actions can only be performed together with other processes.

Definition 2 (I^2MC). *An inhomogeneous interactive Markov chain* (I^2MC) *is a tuple* $\mathcal{I} = (\mathbb{S}, \mathit{Act}, \rightarrow, \mathbf{R}, s^0)$ *where* $\mathbb{S}$ *and* $\mathbf{R}$ *are as before,* $\rightarrow \subseteq \mathbb{S} \times \mathit{Act} \times \mathbb{S}$ *is a transition relation and* $s^0 \in \mathbb{S}$ *is the initial state.*

The semantic model of I^2MC represents the time-dependent variant of IMC [Her02].

Process algebra for I^2MC*.* Originally developed by Hoare and Milner (see [Mil89,Hoa85]), process algebras have been developed as a compositional framework for describing the functional behavior of the system. It allows for modeling complex systems in a component-wise manner by offering a set of *operators* to combine component models. Actions are the most elementary notions. The combination of several actions using the operators forms a *process*. We extend this framework by stochastic timing facilities.

Definition 3. *Let X be a process variable, $\lambda(t) \in \mathbb{R}_{\geq 0}$ with $t \in \mathbb{R}_{\geq 0}$, $A \subseteq \mathit{Act}$ and $a \in \mathit{Act}$. The syntax of inhomogeneous interactive Markov language* (I^2ML) *for* I^2MCs *is defined as follows:*

$$P ::= 0 \quad | \quad a.P \quad | \quad \lambda(t).P \quad | \quad P + P \quad | \quad P\|_A P \quad | \quad P \setminus A \quad | \quad X.$$

$\dfrac{}{a.P \xrightarrow{a} P}$	$\dfrac{}{\lambda(t).P \xrightarrow{\lambda(t)} P}$
$\dfrac{P \xrightarrow{a} P'}{P+Q \xrightarrow{a} P'}$	$\dfrac{P \xrightarrow{\lambda(t)} P'}{P+Q \xrightarrow{\lambda(t)} P'}$
$\dfrac{P \xrightarrow{a} P'}{P\Vert_A Q \xrightarrow{a} P'\Vert_A Q}\ (a \notin A)$	$\dfrac{P \xrightarrow{\lambda(t)} P'}{P\Vert_A Q \xrightarrow{\lambda(t)} P'\Vert_A Q}$
$\dfrac{P \xrightarrow{a} P' \quad \text{and} \quad Q \xrightarrow{a} Q'}{P\Vert_A Q \xrightarrow{a} P'\Vert_A Q'}\ (a \in A)$	$\dfrac{P \xrightarrow{\lambda(t)} P'}{P\backslash A \xrightarrow{\lambda(t)} P'\backslash A}$
$\dfrac{P \xrightarrow{a} P'}{P\backslash A \xrightarrow{a} P'\backslash A}\ (a \notin A)$	$\dfrac{E[X{:=}E/X] \xrightarrow{\lambda(t)} E'}{X{:=}E \xrightarrow{\lambda(t)} E'}$
$\dfrac{P \xrightarrow{a} P'}{P\backslash A \xrightarrow{\tau} P'\backslash A}\ (a \in A)$	$\dfrac{E[X{:=}E/X] \xrightarrow{a} E'}{X{:=}E \xrightarrow{a} E'}$

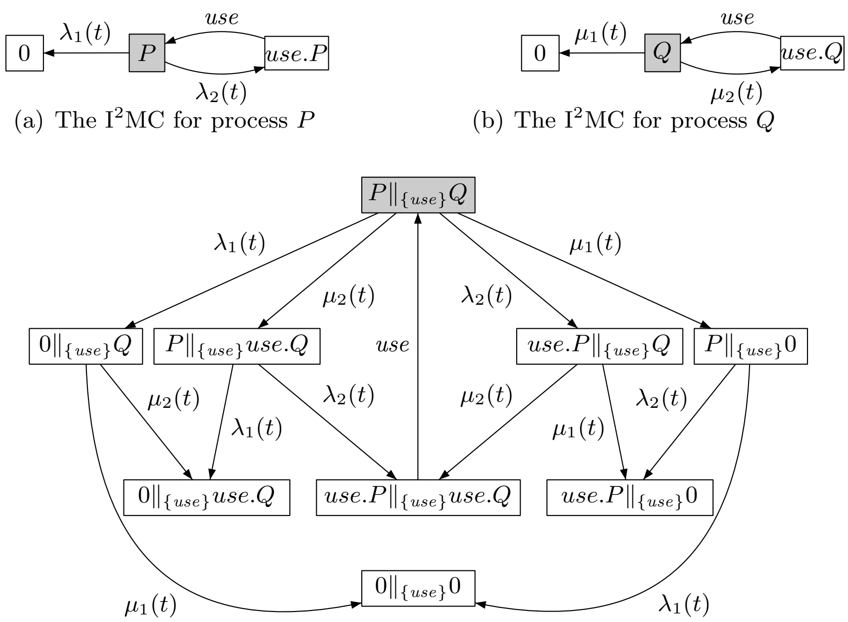
Table 1. Inference rules for the operational semantics of I^{2}ML.

Process variables are assumed to be defined by recursive equations of the form $X := P$, where P is an I^{2}ML term. The *null process* 0 is the deadlock process and cannot perform any action. The prefix operators are $a.P$ and $\lambda(t).P$ for actions and rates, respectively. The *choice* operator $P + Q$ chooses between processes P or Q. Process $P\Vert_A Q$ denotes the *parallel composition* of processes P and Q where synchronization is required only for actions in A; actions not in A are performed autonomously. The process $P\backslash A$ behaves like P except that all actions in A become unobservable to other processes; this is established by relabeling a by the distinguished action $\tau \in Act$. The operational semantics of I^{2}ML terms is defined by the inference rules in Table 1 where for the sake of conciseness symmetric rules are not shown.

A few remarks concerning time-prefix and choice are in order. The process $\lambda(t).P$ evolves into P within Δt time units with probability:

$$\Pr_{\lambda(t).P,P}(t, \Delta t) = \int_0^{\Delta t} \lambda(t+\tau) e^{-\int_0^\tau \lambda(t+\ell)d\ell} d\tau = 1 - e^{-\int_0^{\Delta t} \lambda(t+\ell)d\ell},$$

given that $\lambda(t).P$ is enabled at the global time t. The above relation can be easily proven from (6) by taking $i = \lambda(t).P$, $j = P$, $R_{i,j}(t+\tau) = \lambda(t+\tau)$ and $E_s(t+\ell) = \lambda(t+\ell)$. The process $\lambda(t).P + \mu(t).Q$ can evolve into P if the time delay generated by a stochastic process with rate $\lambda(t)$ is smaller than that generated by a different stochastic process with rate $\mu(t)$. By a symmetric argument it may evolve into Q. Therefore, from (3) it follows that the distribution of time until a choice is made is $\Pr\{W(t) \leq \Delta t\} = 1 - e^{-\int_0^{\Delta t} \lambda(t+\tau)+\mu(t+\tau)d\tau}$. For a choice between $|J|$ processes (J is a finite index set), the distribution of the waiting time becomes $\Pr\{W(t) \leq \Delta t\} = 1 - e^{-\int_0^{\Delta t} \sum_{i\in J} \lambda_i(t+\tau)d\tau}$. If the rates $\lambda_i(t)$ in the process $\sum_{i\in J} \lambda_i(t).P_i$ are constant ($\lambda_i(t) = \lambda_i$), then the waiting time is exponentially distributed with the sum of the rates λ_i i.e. $\Pr\{W(t) \leq \Delta t\} = 1 - e^{-\sum_{i\in J} \lambda_i \Delta t}$. This corresponds to the interpretation of choice in Markovian process algebras [HHK02]. It is important to note that when $P_i = P$ for all $i \in J$, the process $\sum_{i\in J} \lambda_i(t).P$ will evolve into P with rate $\sum_{i\in J} \lambda_i(t)$.



(a) The I^{2}MC for process P

(b) The I^{2}MC for process Q

(c) The I^{2}MC for the process $P\|_{\{use\}}Q$

Fig. 2. Two faulty components.

Parallel composition. When considering just actions the asynchronous parallel composition has the same functionality as that from basic process calculi. On the other hand when considering stochastic delays the composition is more involved.

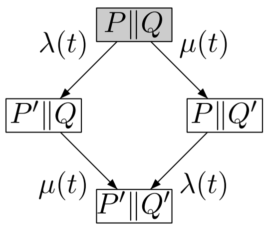


Fig. 1. $P\|Q$.

Consider $P := \lambda(t).P'$ and $Q := \mu(t).Q'$. They can evolve into P' and Q' after a time delay governed by a distribution with rate $\lambda(t)$ and $\mu(t)$, respectively. Since the waiting time in any state is memoryless (4), we can show the way by which processes P and Q are composed (Fig. 1). First consider that when both processes start their execution in initial state $P\|Q$ (the shadowed state) they probabilistically select a time delay, say, Δt_λ for P and Δt_μ for Q. If $\Delta t_\lambda < \Delta t_\mu$ then P finishes its execution first and evolves into P'. The same applies to Q when $\Delta t_\mu < \Delta t_\lambda$. By intuition we could think that when it is already in $P'\|Q$, $\Delta t_\lambda = 0$ and the remaining delay for process Q until it finishes its execution is $\Delta t_\mu - \Delta t_\lambda$. What really happens is that on entering state $P'\|Q$ both delays are set to zero i.e., $\Delta t_\lambda = \Delta t_\mu = 0$. As P' has no transitions, Δt_λ remains 0 but for Q its delay is initialized to a new value which might be different from $\Delta t_\mu - \Delta t_\lambda$ due to a probabilistic selection. Due to the memoryless property, however, the remaining delay for Q is fully determined by μ only.

Example 1. Consider two hardware components described by the equations $P := \lambda_1(t).0 + \lambda_2(t).use.P$ and $Q := \mu_1(t).0 + \mu_2(t).use.Q$, respectively. The underlying I^{2}MCs for both components are depicted in Fig. 2 (a) and (b). Each of the components may fail with rate $\lambda_1(t)$ and $\mu_1(t)$, respectively. As a result of the failure they evolve into process 0. On the other hand, the components may move to the working state with the rate $\lambda_2(t)$ and $\mu_2(t)$, respectively, where they can *use* some resources. If one of them fails then the entire system fails. Both components can use the resources at the same time if the system is working properly. Fig. 2 (c) depicts the I^{2}MC of $P\|_{\{use\}}Q$.

4 Strong and Weak Bisimulation

In order to compare the behavior of ICTMCs (and their interactive variants) we exploit the well-studied and widely accepted notion of bisimulation [Buc94,Mil89,Her02]. A classical bisimulation relation requires equivalent states to be able to mutually mimic their stepwise behavior. In the probabilistic setting this is interpreted as requiring equivalent states to have equal cumulative rates to move to any equivalence class. Bisimulation is considered as a natural notion of equivalent behavior, is equipped with quotienting algorithms, and has a clear correspondence to equivalence in terms of logical behavioral specifications. In this section, we will define strong bisimulation for I^{2}MC starting from a similar notion on ICTMCs. Some algebraic and probabilistic properties of bisimulation are investigated. The same applies to *weak* bisimulation that allows for the abstraction of internal, i.e., τ actions.

Bisimulation for ICTMCs.

Definition 4 (ICTMC strong bisimulation). *An equivalence $\mathcal{R} \subseteq \mathbb{S} \times \mathbb{S}$ is a strong bisimulation whenever for all $(P, Q) \in \mathcal{R}$, $t \in \mathbb{R}_{\geq 0}$ and $C \in \mathbb{S}/\mathcal{R}$:*

$$R(P, C, t) = R(Q, C, t),$$

where $R(P, C, t) = \sum_i \{|\lambda(t)|P \xrightarrow{\lambda(t)}_i P', P' \in C|\}$. P and Q are strongly bisimilar, denoted $P \sim Q$, if (P, Q) is contained in some strong bisimulation $\mathcal{R}$.

Here, $\{|\ldots|\}$ denotes a multiset. It follows that $\sim$ is the largest strong bisimulation, i.e., it contains any strong bisimulation. To be able to compare ICTMCs by bisimulation, let us equip an ICTMC with an initial state $s^0 \in \mathbb{S}$. Two ICTMCs $\mathcal{C}_P = (\mathbb{S}_P, \mathbf{R}_P, s^0_P)$ and $\mathcal{C}_Q = (\mathbb{S}_Q, \mathbf{R}_Q, s^0_Q)$ are bisimilar, denoted $\mathcal{C}_P \sim \mathcal{C}_Q$, iff their initial states are bisimilar, i.e., $s^0_P \sim s^0_Q$. The quotient of an ICTMC under $\sim$ is defined in the following way.

Definition 5 (Bisimulation quotient). *For the* ICTMC $\mathcal{C} = (\mathbb{S}, \mathbf{R}, s^0)$ *and $\sim$, the quotient $\mathcal{C}/\!\sim$ is defined by $\mathcal{C}/\!\sim = (\mathbb{S}/\!\sim, \mathbf{R}_\sim, s^0_\sim)$ where $s^0_\sim = [s^0]_\sim$ and $\mathbf{R}_\sim$ is defined by: $R_\sim([P]_\sim, [P']_\sim, t) = R(P, [P']_\sim, t)$* for all $t \in \mathbb{R}_{\geq 0}$.

Note that $\mathcal{C}$ is strongly bisimilar to $\mathcal{C}/\!\sim$. An important property of strong bisimulation is that it preserves transient probabilities; in particular, this means that there is a strong relationship between the transient probabilities in an ICTMC and its quotient.

Theorem 2. *Let $\mathcal{C} = (\mathbb{S}, \mathbf{R}, s^0)$ be an* ICTMC*. For every $C \in \mathbb{S}/\!\sim$, the transient probability distribution $\pi_C(t)$ of the state C in the quotient chain $\mathcal{C}/\!\sim$ is:*

$$\pi_C(t) = \sum_{s \in C} \pi_s(t) \quad \text{for all } t \in \mathbb{R}_{\geq 0},$$

where $\pi_s(t)$ is the transient probability distribution of state $s \in \mathbb{S}$ in $\mathcal{C}$.

Proof. The proof is carried out in two steps. First, we characterize the transient probability distribution by a system of ODEs for the aggregated chain $\mathcal{C}/\sim$ in terms of equivalence classes $C \in \mathbb{S}/\sim$. Then by suitable transformation we will derive an equivalent system of ODEs for the original ICTMC with state space $\mathbb{S}$. Note that if $\pi_C(t) = \sum_{s\in C} \pi_s(t)$, then $\frac{d\pi_C(t)}{dt} = \sum_{s\in C} \frac{d\pi_s(t)}{dt}$ due to the linearity of the differentiation operator. The system of ODEs for $\mathcal{C}/\sim$ is:

$$\begin{cases} \frac{d\pi_{C_1}(t)}{dt} = \sum_{C_v\in\mathbb{S}/\sim} R_\sim(C_v, C_1, t)\pi_{C_v}(t) - \sum_{C_k\in\mathbb{S}/\sim} R_\sim(C_1, C_k, t)\pi_{C_1}(t) \\ \vdots \\ \frac{d\pi_{C_i}(t)}{dt} = \sum_{C_v\in\mathbb{S}/\sim} R_\sim(C_v, C_i, t)\pi_{C_v}(t) - \sum_{C_k\in\mathbb{S}/\sim} R_\sim(C_i, C_k, t)\pi_{C_i}(t) \\ \vdots \\ \frac{d\pi_{C_n}(t)}{dt} = \sum_{C_v\in\mathbb{S}/\sim} R_\sim(C_v, C_n, t)\pi_{C_v}(t) - \sum_{C_k\in\mathbb{S}/\sim} R_\sim(C_n, C_k, t)\pi_{C_n}(t) \end{cases}$$

where $R_\sim(C_v, C_i, t)$ is the time-dependent rate between classes C_v and C_i in $\mathcal{C}/\sim$. The equation:

$$\frac{d\pi_{C_i}(t)}{dt} = \sum_{C_v\in\mathbb{S}/\sim} R_\sim(C_v, C_i, t)\pi_{C_v}(t) - \sum_{C_k\in\mathbb{S}/\sim} R_\sim(C_i, C_k, t)\pi_{C_i}(t)$$

shows that the change of probability mass in class C_i is equal to the difference between the in-flow and the out-flow for the state C_i. We assume that the rate $R_\sim(C_i, C_i, t) = 0$ for every $C_i \in \mathbb{S}/\sim$, as for ICTMCs the presence of such rate does not influence the transient probability distribution of the class C_i. The out-flow can be rewritten using Definition 5 as:

$$\sum_{C_k\in\mathbb{S}/\sim} R_\sim(C_i, C_k, t)\pi_{C_i}(t) = \pi_{C_i}(t) \sum_{C_k\in\mathbb{S}/\sim} R(s_m^{(i)}, C_k, t)$$

here $m = |C_i|$ and $s_m^{(i)}$ is the m'th state of class C_i (in fact the choice of $s_m^{(i)}$ is arbitrary; any state in C_i could be taken). By substituting $\pi_C(t) = \sum_{s\in C} \pi_s(t)$ in the above equation, we get:

$$\begin{aligned} \left(\pi_{s_1^{(i)}}(t) + \cdots + \pi_{s_m^{(i)}}(t)\right) \sum_{C_k\in\mathbb{S}/\sim} R(s_m^{(i)}, C_k, t) =& \pi_{s_1^{(i)}}(t) \sum_{C_k\in\mathbb{S}/\sim} R(s_m^{(i)}, C_k, t) + \cdots \\ &+ \pi_{s_m^{(i)}}(t) \sum_{C_k\in\mathbb{S}/\sim} R(s_m^{(i)}, C_k, t). \end{aligned}$$

As all states $s_m^{(i)} \in C_i$ are bisimilar we have for every equivalence class C_k the relation $R(s_1^{(i)}, C_k, t) = \cdots = R(s_m^{(i)}, C_k, t)$ which gives:

$$\pi_{s_1^{(i)}}(t) \sum_{C_k\in\mathbb{S}/\sim} R(s_1^{(i)}, C_k, t) + \cdots + \pi_{s_m^{(i)}}(t) \sum_{C_k\in\mathbb{S}/\sim} R(s_m^{(i)}, C_k, t).$$

By noting that $\bigcup_{j=1}^n C_j = \mathbb{S}$ for $|\mathbb{S}/\sim| = n$ we get the out-flow equation:

$$\sum_{C_k\in\mathbb{S}/\sim} R_\sim(C_i, C_k, t)\pi_{C_i}(t) = \pi_{s_1^{(i)}}(t)R(s_1^{(i)}, \mathbb{S}, t) + \cdots + \pi_{s_m^{(i)}}(t)R(s_m^{(i)}, \mathbb{S}, t).$$

Now we proceed with the in-flow equation:

$$\sum_{C_v \in \mathbb{S}/\sim} R_\sim(C_v, C_i, t)\pi_{C_v}(t) = R_\sim(C_1, C_i, t)\pi_{C_1}(t) + \cdots + R_\sim(C_n, C_i, t)\pi_{C_n}(t)$$

Using Definition 5, the above right-hand sum results in:

$$R(s_{v_1}^{(1)}, C_i, t)\pi_{C_1}(t) + \cdots + R(s_{v_n}^{(n)}, C_i, t)\pi_{C_n}(t),$$

where $v_\ell = |C_\ell|$ and $s_{v_\ell}^{(\ell)}$ is the v_ℓ'th states of class C_ℓ. By substituting $\pi_C(t) = \sum_{s \in C} \pi_s(t)$ we get:

$$\begin{aligned} &R(s_{v_1}^{(1)}, C_i, t)\left(\pi_{s_1^{(1)}}(t) + \cdots + \pi_{s_{v_1}^{(1)}}(t)\right) + \cdots + R(s_{v_n}^{(n)}, C_i, t)\left(\pi_{s_1^{(n)}}(t) + \cdots\right. \\ &\left.+\pi_{s_{v_n}^{(n)}}(t)\right) = R(s_{v_1}^{(1)}, C_i, t)\pi_{s_1^{(1)}}(t) + \cdots + R(s_{v_1}^{(1)}, C_i, t)\pi_{s_{v_1}^{(1)}}(t) + \cdots \\ &+ R(s_{v_n}^{(n)}, C_i, t)\pi_{s_1^{(n)}}(t) + \cdots + R(s_{v_n}^{(n)}, C_i, t)\pi_{s_n^{(n)}}(t). \end{aligned}$$

As all states $s^{(\ell)}$ in the equivalence class C_ℓ are bisimilar, $R(s_1^{(\ell)}, C_i, t) = \cdots = R(s_{v_\ell}^{(\ell)}, C_i, t)$ which yields:

$$\begin{aligned} &R(s_1^{(1)}, C_i, t)\pi_{s_1^{(1)}}(t) + \cdots + R(s_{v_1}^{(1)}, C_i, t)\pi_{s_{v_1}^{(1)}}(t) + \cdots + \\ &R(s_1^{(n)}, C_i, t)\pi_{s_1^{(n)}}(t) + \cdots + R(s_{v_n}^{(n)}, C_i, t)\pi_{s_{v_n}^{(n)}}(t) = \\ &\sum_{s \in \mathbb{S}} R(s, C_i, t)\pi_s(t) = \sum_{s \in \mathbb{S}} R(s, s_1^{(i)}, t)\pi_s(t) + \cdots + \sum_{s \in \mathbb{S}} R(s, s_m^{(i)}, t)\pi_s(t). \end{aligned}$$

Combining the equations for the in-flow and out-flow we obtain:

$$\begin{aligned} &\frac{d\sum_{j=1}^{m} \pi_{s_j^{(i)}}(t)}{dt} = \sum_{s \in \mathbb{S}} R(s, s_1^{(i)}, t)\pi_s(t) + \cdots + \sum_{s \in \mathbb{S}} R(s, s_m^{(i)}, t)\pi_s(t) - \\ &\pi_{s_1^{(i)}}(t)R(s_1^{(i)}, \mathbb{S}, t) - \cdots - \pi_{s_m^{(i)}}(t)R(s_m^{(i)}, \mathbb{S}, t). \end{aligned}$$

By linearity of the differential operator for the equivalence class C_i we obtain m equations:

$$\begin{cases} \frac{d\pi_{s_1^{(i)}}(t)}{dt} = \sum_{s \in \mathbb{S}} R(s, s_1^{(i)}, t)\pi_s(t) - \pi_{s_1^{(i)}}(t)R(s_1^{(i)}, \mathbb{S}, t) \\ \vdots \\ \frac{d\pi_{s_\ell^{(i)}}(t)}{dt} = \sum_{s \in \mathbb{S}} R(s, s_\ell^{(i)}, t)\pi_s(t) - \pi_{s_\ell^{(i)}}(t)R(s_\ell^{(i)}, \mathbb{S}, t) \\ \vdots \\ \frac{d\pi_{s_m^{(i)}}(t)}{dt} = \sum_{s \in \mathbb{S}} R(s, s_m^{(i)}, t)\pi_s(t) - \pi_{s_m^{(i)}}(t)R(s_m^{(i)}, \mathbb{S}, t) \end{cases} .$$

Forming a system of ODEs from the equations of all equivalence classes $C_i \in \mathbb{S}/\sim$ we obtain the transient probability distribution for the original ICTMC.

From Theorem 2 we may conclude that the steady state probability distribution (if it exists) is also preserved.

Corollary 1. *Let $\mathcal{C} = (\mathbb{S}, \mathbf{R}, s^0)$ be an* ICTMC*. For every $C \in \mathbb{S}/\!\sim$, the steady-state probability distribution π_C of the state C in the quotient chain $\mathcal{C}/\!\sim$ is:*

$$\pi_C = \lim_{t\to\infty} \pi_C(t) = \lim_{t\to\infty} \sum_{s\in C} \pi_s(t) = \sum_{s\in C} \pi_s,$$

where π_s is the steady-state probability distribution of state $s \in \mathbb{S}$.

In many cases it is reasonable to assume that two processes P and Q are equal up to time T. For this case we propose the *finite-horizon bisimulation.*

Definition 6. *An equivalence $\mathcal{R} \subseteq \mathbb{S} \times \mathbb{S}$ is a finite-horizon bisimulation whenever for all $(P, Q) \in \mathcal{R}$, $t \in [0, T]$ $(T \in \mathbb{R}_{\geq 0})$ and $C \in \mathbb{S}/\mathcal{R}$:*

$$R(P, C, t) = R(Q, C, t).$$

P and Q are finitely-horizon bisimilar, denoted $P \sim^T Q$, if (P, Q) is contained in some finite-horizon bisimulation $\mathcal{R}$.

Notice that the definition of finite-horizon bisimulation $\sim^T$ is the same except that the time t lies in the interval $[0, T]$. It is easy to see that finite-horizon bisimulation preserves the transient distribution up to time T.

Proposition 1. *For $0 < t_1 < \cdots < T < \cdots < \infty$ it holds:*

$$\sim^0 \subseteq \sim^{t_1} \subseteq \cdots \subseteq \sim^T \cdots \subseteq \sim^\infty = \sim$$

Thus , $P \sim^{t_i} Q$ implies $P \sim^{t_j} Q$ for every $t_j < t_i$. It follows that for $t_j < t_i$, the quotient under $\sim^{t_j}$ is coarser than under $\sim^{t_i}$.

Bisimulation for I^{2}MCs. So far, we have presented bisimulation for ICTMCs. In order to define bisimulation for I^{2}MCs, unobservable actions (i.e., τ) require special care. Consider the following diagram:

$$\begin{tikzcd} P_0 \arrow[r, "2\lambda(t)"] \arrow[dr, "\tau"'] & P_1 \arrow[d, dotted, dash, "\sim"] \\ & P_2 \end{tikzcd} \qquad \begin{tikzcd} Q_0 \arrow[r, "\lambda(t)"] \arrow[dr, "\lambda(t)"'] & Q_1 \arrow[d, dotted, dash, "\sim"] \\ & Q_2 \end{tikzcd}$$

where $P_1 \sim P_2 \sim Q_1 \sim Q_2$. At first sight, it seems natural to consider $P_0 \sim Q_0$ as $R(P_0, C, t) = R(Q_0, C, t) = 2\lambda(t)$. But, state P_0 can do something more. There is a transition $P_0 \xrightarrow{\tau} P_2$ which consumes no time as a τ-action is an internal one and is not prevented by the environment (maximal progress assumption). The probability that transition $P_0 \xrightarrow{2\lambda(t)} P_1$ will be taken in 0 time units is $\mathrm{Pr}_{P_0,P_1}(t, 0) = \int_0^0 2\lambda(t + \tau)e^{-\int_0^\tau 2\lambda(t+\ell)d\ell} d\tau = 0$. Thus, we may conclude that $P_0 \nsim Q_0$. When specifying the definition of bisimilarity we have to treat immediate actions (τ) in a special way. Let $\mathbb{S}$ be the state-space of an I^{2}MC.

Definition 7 (I^{2}MC strong bisimulation). *An equivalence $\mathcal{R} \subseteq \mathbb{S} \times \mathbb{S}$ is a strong bisimulation whenever for all $(P, Q) \in \mathcal{R}$, $t \in \mathbb{R}_{\geq 0}$, $a \in Act$ and $C \in \mathbb{S}/\mathcal{R}$:*

- *$P \xrightarrow{a} P'$ implies $Q \xrightarrow{a} Q'$ for some Q' and $(P', Q') \in \mathcal{R}$.*
- *$Q \xrightarrow{a} Q'$ implies $P \xrightarrow{a} P'$ for some P' and $(P', Q') \in \mathcal{R}$.*

$P + 0 = P$	$(P + Q) + R = P + (Q + R)$
$P + Q = Q + P$	$\lambda(t).P + \mu(t).P = (\lambda(t) + \mu(t)).P$
$a.P + a.P = a.P$	$\lambda(t).P + \tau.Q = \tau.Q$

Table 2. Sound and complete axioms for $\sim$ on the I^{2}ML sequential fragment.

– $P \stackrel{\tau}{\nrightarrow}$ *(or* $Q \stackrel{\tau}{\nrightarrow}$*) implies* $R(P, C, t) = R(Q, C, t)$.

P and Q are strongly bisimilar, denoted $P \sim Q$, *if* (P, Q) *is contained in some strong bisimulation* $\mathcal{R}$.

Example 2. Consider the I^{2}MC from Fig. 2 (c) and $\lambda_1(t) = \mu_1(t)$, $\lambda_2(t) = \mu_2(t)$. Its quotient under bisimulation is depicted in Fig. 3. The equivalence classes C_1, C_2 and C_3 contain the following states $C_1 = \{0\|_{\{use\}}Q, P\|_{\{use\}}0\}$, $C_2 = \{P\|_{\{use\}}use.Q, use.P\|_{\{use\}}Q\}$ and $C_3 = \{0\|_{\{use\}}use.Q, use.P\|_{\{use\}}0, 0\|_{\{use\}}0\}$.

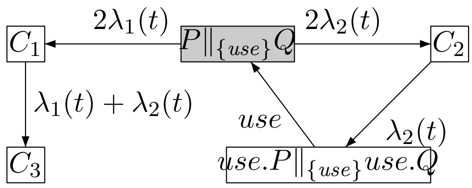


Fig. 3. Bisimulation quotient.

In a similar way as for ICTMCs, one can consider the quotient of an I^{2}MC. The compositional nature of I^{2}MC, however, allows in principle for obtaining such quotient in a component-wise manner, e.g., the quotient of $P\|_A Q$ can be obtained by first constructing the quotients of P and Q, then combine them, and quotienting the composition. The necessary requirement that $\sim$ needs to fulfill is that it is a *congruence* relation. The relation $\sim$ is a congruence whenever for processes P and Q it holds: $P \sim Q$ implies $\mathbf{C}[P] \sim \mathbf{C}[Q]$ where $\mathbf{C}[\cdot]$ is any context. (A context is basically a process term containing a hole that may be filled with any process.)

Theorem 3. $\sim$ *is a congruence with respect to all operators in* I^{2}ML.

Finite-horizon bisimulation is a congruence with one additional property.

Proposition 2. *For any processes* P, P', Q, Q' *and intervals* $[0, T_1]$ *and* $[0, T_2]$ *with* $T_1, T_2 \in \mathbb{R}_{\geq 0}$ *we have:*

$$P \sim^{T_1} P' \text{ and } Q \sim^{T_2} Q' \text{ implies } P\|_A Q \sim^{min(T_1,T_2)} P'\|_A Q' \text{ for all } A \subseteq Act.$$

As a next step, we consider the possibility to establish bisimulation symbolically, i.e., on the level of the syntax of the earlier introduced language I^{2}ML. This is facilitated by an axiomatization for $\sim$. The soundness of these axioms ensures that any two terms that are syntactically equal (denoted $=$) are bisimilar; formally, $P = Q \Rightarrow P \sim Q$. Whenever the axioms are complete, in addition, any strongly bisimilar processes can be represented by the same expressions in I^{2}ML, i.e., $P \sim Q \Rightarrow P = Q$. Summarizing, any bisimulation can be established syntactically, i.e., by just manipulating terms rather than I^{2}MCs, provided the axiom system is sound and complete. Let $\mathcal{A}_\sim$ be the set of axioms listed in Table

2 extended with the expansion law:

$$P\|_A Q = \sum_{i\in J_1} \lambda_i(t).\,(P_i\|_A Q) + \sum_{k\in J_3} \mu_k(t).\,(P\|_A Q_k) + \sum_{a_j=b_l\in A} a_j.\,(P_j\|_A Q_l) +$$
$$+ \sum_{a_j\notin A\wedge a_j\in J_2} a_j.\,(P_j\|_A Q) + \sum_{b_l\notin A\wedge b_l\in J_4} b_l.\,(P\|_A Q_l)$$

where $P := \sum_{i\in J_1} \lambda_i(t).P_i + \sum_{j\in J_2} a_j.P_j$ and $Q := \sum_{k\in J_3} \mu_k(t).Q_k + \sum_{l\in J_4} b_l.Q_l$ with the finite index sets J_1, J_2, J_3 and J_4. Then the following holds:

Theorem 4. *For any $P, Q \in \mathbf{RG}$, $\mathcal{A}_\sim \vdash P = Q$ if and only if $P \sim Q$.*

The term **RG** denotes the set of all *regular* (no parallel composition inside recursion) and *guarded* expressions. While $\mathcal{A}_\sim \vdash P = Q$ means that $P = Q$ can be deduced from the set of sound and complete axiom system $\mathcal{A}_\sim$. The axiom $\lambda(t).P + \mu(t).P = (\lambda(t) + \mu(t)).P$ is due to the fact that the sum of two Poisson processes with rates $\lambda(t)$ and $\mu(t)$ is a Poisson process with the rate $\lambda(t) + \mu(t)$, whereas the axiom $\lambda(t).P + \tau.Q = \tau.Q$ is due to the maximal progress assumption. Notice that $\mathcal{A}_\sim$ also contains all standard axioms which involve hiding and recursion operators which are standard and omitted here.

Bisimulation minimization. The previous sections have set the stage for bisimulation minimization. Experiments have shown that in the traditional [FV98] as well as in the stochastic setting [KKZJ07] exponential state space savings can be achieved. Given that $\sim$ is a congruence, individual processes can be replaced by their bisimilar quotient (under $\sim$) and the peak memory requirements can be reduced significantly. This all, however, requires an efficient bisimulation minimization algorithm. We adopt the *partition-refinement* paradigm to obtain a minimization algorithm for I^{2}MCs. As the problem for arbitrary rate functions is undecidable, we restrict to three classes of rate matrices $\mathbf{R}(t)$: piecewise uniform, polynomial ($\mathbf{R}(t) = t^{M+1}\mathbf{R}_{M+1} + \cdots + t\mathbf{R}_2 + \mathbf{R}_1$, where $\mathbf{R}_i$ with $i \leq M+1 \in \mathbb{N}$ are constant matrices) and piecewise polynomial (each piece is a polynomial of degree three). The same classes have been considered for the transient probability distribution, cf. Theorem 1. Rate comparisons and summations can easily be realized for these classes of functions. For rate matrix $\mathbf{R}$, let $M+1$ denote the total number of intervals for piecewise uniform $\mathbf{R}(t)$, the polynomial degree when $\mathbf{R}(t)$ is polynomial, and the number of polynomial pieces when $\mathbf{R}(t)$ is piecewise polynomial.

Our bisimulation minimization algorithm for I^{2}MCs is based on a generalization of the algorithm for obtaining the coarsest quotient of a Markov chain under bisimulation by Derisavi *et al.* [DHS03], and Paige-Tarjan's algorithm for LTS. The basic idea is to minimize iteratively over all pieces (or degrees of the polynomials). The bisimulation algorithm exploits an efficient data structure which groups all states with the same outgoing rate. This is in fact a binary tree where each *node* has four parameters: *node.left* and *node.right* - pointers to the left and right child, respectively, *node.sum* - stores the sum of the rates and *node.S* - stores all states with the same *node.sum*. Using such data structures, the time- and space complexity of bisimulation minimization for I^{2}MCs reduces to:

Theorem 5. *The coarsest quotient under $\sim$ of any I^{2}MC can be obtained in a worst-case time complexity $\mathcal{O}(m_a \lg(n) + M m_r \lg(n))$ and space complexity*

$\mathcal{O}(m_a + m_r)$, where m_a and m_r is the number of action-labeled and rate-labeled transitions, respectively.

Proof. The base algorithm for I^2MC bisimulation quotient computation based on partition refinement techniques is presented in Algorithm 1.

Algorithm 1 Bisimulation quotient computation for I^2MCs.

Require: I^2MC $(\mathbb{S}, Act, \to, \mathbf{R})$, M
Ensure: I^2MC $(\mathbb{S}/\!\sim, Act, \to_\sim, \mathbf{R}_\sim)$
1: $\Pi_\tau := \{s \in \mathbb{S} | s \xrightarrow{\tau}\}$ $\quad \Pi := \{s \in \mathbb{S} | s \not\xrightarrow{\tau}\}$
2: $L :=$ **push**$(\{\mathbb{S}\})$ {#Create the stack of splitters}
3: **while** $L \neq \varnothing$ **do**
4: $\quad C :=$ **pop**(L) {#Extract a splitter from L}
5: $\quad [\Pi, L] := \mathit{Refine}_a(\Pi, C, L)$
6: $\quad [\Pi_\tau, L] := \mathit{Refine}_a(\Pi_\tau, C, L)$
7: $\quad [\Pi, L] := \mathit{Refine}_r(\Pi, C, L, \mathbf{R}, M)$
8: **end while**
9: $\Pi := \Pi \cup \Pi_\tau$
10: **for all** $B \in \Pi$ **do**
11: $\quad$ **let** $s \in B$ {#Choose any state from block B}
12: $\quad$ **for all** $p \in Post(s)$ **do**
13: $\quad\quad B' := [p]_\Pi$
14: $\quad\quad$ **if** $(s, a, p) \subseteq\to$ **then**
15: $\quad\quad\quad (B, a, B') \subseteq\to_\sim$ {#Transition from B to B' labeled with a}
16: $\quad\quad$ **end if**
17: $\quad\quad$ **for** $i = 0$ to M **do**
18: $\quad\quad\quad R_\sim(B, B', i) := R_\sim(B, B', i) + R(s, p, i)$
19: $\quad\quad$ **end for**
20: $\quad$ **end for**
21: **end for**
22: **return** $[\Pi, \to_\sim, \mathbf{R}_\sim]$

The variables and functions used in Algorithm 1 have the following meaning:

- M - is the rate matrix $\mathbf{R}(t)$ parameter.
- Π - is the set of all states with no outgoing τ-transition.
- Π_τ - is the set of all states with at least one outgoing τ-transition.
- L - is the stack of all current splitters.
- C - is the current splitter which splits the partition Π and Π_τ.
- $\mathit{Refine}_a(\Pi, C, L)$ - is the function which refines partition Π using the current splitter C with respect to action-labeled transitions. It returns the refined partition and a new set of splitters.
- $\mathit{Refine}_r(\Pi, C, L, \boldsymbol{R}, M)$ - is the function which refines partition Π using the current splitter C with respect to rate-labeled transitions. It returns the refined partition and a new set of splitters.
- $[s]_\Pi$ - is the block of state s in partition Π.

The Algorithm 1 starts with two initial partitions Π and Π_τ (line 1). Then it successively refines them by using the functions Refine_a and Refine_r (line 5-7). It stops when there are no more splitters. The lines 10-21 have the role of creating the rate matrix $\mathbf{R}_\sim$ for the quotient state-space $\mathbb{S}/\!\sim$. The function Refine_a can be implemented using the adaptation of Fernandez (see [Fer89]) to Paige and

Tarjan's algorithm (see [PT87]) or the more efficient implementation of Dovier et al. [DPP04]. The time and space complexity is of order $\mathcal{O}(m_a \lg n)$ and $\mathcal{O}(m_a)$, respectively, where m_a is the total number of action-labeled transitions. It is important to notice that in function *Refine*$_a$ as well as *Refine*$_r$ the splitter with the largest size will be deleted from the set of potential splitters. This will give a total of $\lg(n+1)$ iterations for the cycle 3-8. The pseudo code of function *Refine*$_r$ is presented in Algorithm 2. The function *Refine*$_r$ uses the following variables

Algorithm 2 *Refine*$_r(\Pi, C, L, \mathbf{R}, M)$

Require: $\Pi, C, L, \mathbf{R}, M$
Ensure: Π, L
1: **for** $i = 0$ to M **do**
2:     $B_\Pi := \varnothing$
3:     **for all** $p \in C$ **do**
4:         **for all** $s \in Pre(p)$ **do**
5:             $s.sum := 0$
6:         **end for**
7:     **end for**
8:     **for all** $p \in C$ **do**
9:         **for all** $s \in Pre(p)$ **do**
10:             $s.sum := s.sum + R(s, p, i)$
11:         **end for**
12:     **end for**
13:     **for all** $s \in Pre(C)$ **do**
14:         $B' := [s]_\Pi$
15:         **delete** s *from* B'
16:         *Insert*$(\mathrm{T}(B'), s)$
17:         $B_\Pi := B_\Pi \cup \{B'\}$
18:     **end for**
19:     **for all** $B' \in B_\Pi$ **do**
20:         $\Pi := \Pi \cup \{\mathrm{T}(B')[1], \ldots, \mathrm{T}(B')[k]\}$
21:     **end for**
22: **end for**
23: **for all** $B' \in B_\Pi$ **do**
24:     $B_{max} :=$ *largest block of* $\{B', \mathrm{T}(B')[1], \ldots, \mathrm{T}(B')[k]\}$
25:     **push** $(L, \{B', \mathrm{T}(B')[1], \ldots, \mathrm{T}(B')[k]\} - \{B_{max}\})$
26: **end for**
27: **return** $[\Pi, L]$

and functions:

- $Pre(C)$ - is the set of predecessor states of splitter C.
- B_Π - is a set of blocks in partition Π.
- $R(s, p, i)$ - the rate from state s to state p with parameter i (interval, polynomial coefficient, etc).
- $\mathrm{T}(B')$ - binary tree of block B', with $\mathrm{T}(B')[j]$ being the j'th sub-block of tree $\mathrm{T}(B')$.
- *Insert*$(\mathrm{T}(B'), s)$ - the function inserts the state $s \in B'$ into the binary tree $\mathrm{T}(B')$ of block B'.

The Algorithm 2 has the same functionality as *Split* function from Derisavi et al. (see [DHS03]). The crucial difference is the line 1 which is a cycle over all time intervals or polynomial coefficients and the lines 19-21 which after each refinement create a new partition, where the new refined blocks are $\{\mathrm{T}(B')[1], \ldots, \mathrm{T}(B')[k]\}$. As the time and space complexity of Derisavi's base algorithm (the algorithm which calls the function *Split*) are bounded by $\mathcal{O}(m_r \lg n)$ and $\mathcal{O}(m_r)$, respectively, where m_r is the total number of rate-labeled transitions. We obtain in our case that the invocation of the base Algorithm 2 with $Refine_r$ function only will result in a time and space complexity of $\mathcal{O}(Mm_r \lg n)$ and $\mathcal{O}(m_r)$, respectively. Considering also $Refine_a$ function we obtain that the total time and space complexity of the bisimulation quotienting algorithm are $\mathcal{O}(m_a \lg n + Mm_r \lg n)$ and $\mathcal{O}(m_a + m_r)$.

Recall that ICTMCs are I^{2}MCs that contain no action-labeled transitions. As a side result, the above theorem yields that the coarsest bisimulation quotient of a time-inhomogeneous CTMC can be obtained with time and space complexity $\mathcal{O}(Mm_r \lg(n))$ and $\mathcal{O}(m_r)$, respectively. (The time complexity for homogeneous Markov chains is $\mathcal{O}(m_r \lg(n))$ [DHS03]). Given the results in this paper that $\sim$ preserves transient and steady state distributions, our algorithm can be used to minimize prior to any such analysis.

Weak bisimulation for I^{2}MCs. Strong bisimulation requires equivalent states to simulate their mutual stepwise behavior. While preserving the branching structure, strong bisimulation also requires mimicking of immediate actions (τ). As immediate actions consume no time it seems reasonable that two states will be equivalent regardless of the number of τ-steps in a sequence that they make. Therefore, the equivalence which will allow for the abstraction of sequences of immediate actions will be denoted as *weak bisimulation*. Let the transition $\stackrel{\tau}{\Longrightarrow}$ be the reflexive and transitive closure of $\stackrel{\tau}{\longrightarrow}^*$ and $\stackrel{a}{\Longrightarrow}$ a shorthand for $\stackrel{\tau}{\Longrightarrow}\stackrel{a}{\longrightarrow}\stackrel{\tau}{\Longrightarrow}$ $(a \neq \tau)$.

Definition 8 (I^{2}MC weak bisimulation). *An equivalence $\mathcal{R} \subseteq \mathbb{S} \times \mathbb{S}$ is a weak bisimulation whenever for all $(P, Q) \in \mathcal{R}$, $t \in \mathbb{R}_{\geq 0}$, $a \in Act$ and $C \in \mathbb{S}/\mathcal{R}$:*

- *$P \stackrel{a}{\longrightarrow} P'$ implies $Q \stackrel{a}{\Longrightarrow} Q'$ for some Q' and $(P', Q') \in \mathcal{R}$.*
- *$P \stackrel{\tau}{\not\rightarrow}$ implies $R(P, C, t) = R(Q'', C, t)$ for some $Q'' \stackrel{\tau}{\not\rightarrow}$ such that $Q \stackrel{\tau}{\Longrightarrow} Q''$ and $(P, Q'') \in \mathcal{R}$.*

For Q symmetric rules apply. P and Q are weakly bisimilar, denoted $P \approx Q$, if (P, Q) is contained in some weak bisimulation $\mathcal{R}$.

It seems intuitive that for the sequence $Q \stackrel{\tau}{\Longrightarrow} Q''$ the rates $R(P, C, t)$ and $R(Q'', C, t)$ have to be compared starting from time $t' = t + \Delta t$ where Δt is the time needed to make all τ in the sequence $Q \stackrel{\tau}{\Longrightarrow} Q''$. As τ transitions take no time the result will be the same even when the rates are compared from time t.

Example 3. Consider the I^{2}MC from Fig. 3 and its abstraction i.e. all actions are transformed into immediate ones (τ). The quotient under $\approx$ is depicted in Fig. 4, with C_1, C_2 and C_3 as in Fig. 3 and $C_0 = \{P\|_{\{use\}}Q,\ use.P\|_{\{use\}}use.Q\}$. It is important to note that after abstraction the transition labeled with use results in an immediate transition which gives the possibility to put the states $P\|_{\{use\}}Q$ and $use.P\|_{\{use\}}use.Q$ in the same equivalence class. Also note that the obtained I^{2}MC has no transitions labeled with actions, i.e., it is an ICTMC. This shows

$a.\tau.P = a.P$	$a.(P + \tau.Q) + a.Q = a.(P + \tau.Q)$
$P + \tau.P = \tau.P$	$\lambda(t).\tau.P = \lambda(t).P$

Table 3. Sound and complete axioms for $\backsimeq$ on the I^{2}ML sequential fragment.

that weak bisimulation may be an effective mechanism to turn an I^{2}MC into an ICTMC, which may be subject to analysis as discussed in Section 2.

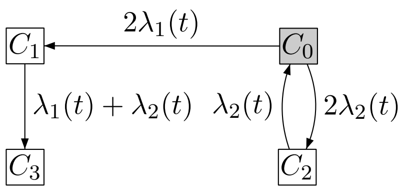


Fig. 4. Weak bisimulation quotient.

As in the case of strong bisimulation, weak bisimulation is also a congruence with respect to I^{2}ML operators. But there is an exception. Weak bisimulation is not a congruence with respect to the choice $(P + Q)$ operator [Mil89]. This is due to the fact that weak bisimulation will equate two processes whenever one can do $\stackrel{\tau}{\Longrightarrow}$ and the other one can do nothing. In order to cope with the choice operator one has to differentiate between $\stackrel{a}{\Longrightarrow}$ and $\stackrel{\tau}{\Longrightarrow}\stackrel{a}{\longrightarrow}\stackrel{\tau}{\Longrightarrow}$ when $a = \tau$ as follows:

Definition 9 (Weak congruence). *P and Q are weakly congruent, denoted by $P \backsimeq Q$, whenever for all $a \in Act$, $t \in \mathbb{R}_{\geq 0}$ and $C \in \mathbf{RG}/\approx$:*

- *$P \stackrel{a}{\longrightarrow} P'$ implies $Q \stackrel{\tau}{\Longrightarrow}\stackrel{a}{\longrightarrow}\stackrel{\tau}{\Longrightarrow} Q'$ for some Q' and $P' \approx Q'$.*
- *$Q \stackrel{a}{\longrightarrow} Q'$ implies $P \stackrel{\tau}{\Longrightarrow}\stackrel{a}{\longrightarrow}\stackrel{\tau}{\Longrightarrow} P'$ for some P' and $P' \approx Q'$.*
- *$P \stackrel{\tau}{\not\rightarrow}$ (or $Q \stackrel{\tau}{\not\rightarrow}$) implies $R(P, C, t) = R(Q, C, t)$.*

Theorem 6. *$\backsimeq$ is a congruence with respect to all operators in* I^{2}ML.

Consider the set of axioms from Table 2 and 3 together with axioms related to hiding and recursion operators as $\mathcal{A}_{\backsimeq}$. As for strong bisimulation the following also holds for weak congruence:

Theorem 7. *For any $P, Q \in \mathbf{RG}$, $\mathcal{A}_{\backsimeq} \vdash P = Q$ if and only if $P \backsimeq Q$.*

Recall that P and Q are regular and guarded process terms.

5 Concluding Remarks and Future Work

This paper presented a compositional formalism for time-inhomogeneous continuous-time Markov chains (ICTMCs), a subclass of piecewise deterministic Markov processes (PDPs). The main contributions are a full-fledged process algebra for interactive ICTMCs, congruence results for weak and strong bisimulation, and a polynomial-time quotienting algorithm. In addition, a new characterization of transient probabilities is provided for rate functions that are piecewise uniform. In contrast to works on communicating PDPs [SvdS06,SJvdS03,SvdS05], this paper considers weak bisimulation, congruence results and axiomatization, and, more importantly a notion of bisimulation which respects maximal progress. Current work consists of investigating improvements to the quotienting algorithm akin to [DPP04], model-checking algorithms [Kat06], and simulation relations for ICTMCs.

Acknowledgment. This research has been performed as part of QUPES project that is financed by the Netherlands Organization for Scientific Research (NWO).

References

[AGH⁺00] Rajeev Alur, Radu Grosu, Yerang Hur, Vijay Kumar, and Insup Lee. Modular Specification of Hybrid Systems in CHARON. In Nancy A. Lynch and Bruce H. Krogh, editors, *HSCC*, volume 1790 of *Lecture Notes in Computer Science*, pages 6–19. Springer, 2000.

[AGLS06] Rajeev Alur, Radu Grosu, Insup Lee, and Oleg Sokolsky. Compositional modeling and refinement for hierarchical hybrid systems. *J. Log. Algebr. Program.*, 68(1-2):105–128, 2006.

[BLB05] Manuela L. Bujorianu, John Lygeros, and Marius C. Bujorianu. Bisimulation for General Stochastic Hybrid Systems. In Morari and Thiele [MT05], pages 198–214.

[Buc94] Peter Buchholz. Exact and ordinary lumpability in finite Markov chains. *Journal of Applied Probability*, 31:59–75, 1994.

[CJH07] Lucia Cloth, Marijn R. Jongerden, and Boudewijn R. Haverkort. Computing Battery Lifetime Distributions. In *DSN*, pages 780–789. IEEE Computer Society, 2007.

[CL06] Christos G. Cassandras and John Lygeros, editors. *Stochastic Hybrid Systems: Recent Developments and Research Trends*, In Christos G. Cassandras, and John Lygeros, editors, Stochastic Hybrid Systems: Recent Developments and Research Trends. CRC Press, 2006.

[Dav93] Mark H. A. Davis. *Markov Models and Optimization.* Chapman and Hall, 1993.

[DHS03] Salem Derisavi, Holger Hermanns, and William H. Sanders. Optimal state-space lumping in Markov chains. *Inf. Process. Lett.*, 87(6):309–315, 2003.

[DPP04] Agostino Dovier, Carla Piazza, and Alberto Policriti. An efficient algorithm for computing bisimulation equivalence. *Theor. Comput. Sci.*, 311(1-3):221–256, 2004.

[EB05] Mariken Everdij and Henk Blom. Piecewise deterministic Markov processes represented by dynamically coloured Petri nets. *Stochastics*, 77(1):1–29, 2005.

[Fer89] Jean-Claude Fernandez. An Implementation of an Efficient Algorithm for Bisimulation Equivalence. *Sci. Comput. Program.*, 13(1):219–236, 1989.

[FV98] Kathi Fisler and Moshe Y. Vardi. Bisimulation Minimization in an Automata-Theoretic Verification Framework. In Ganesh Gopalakrishnan and Phillip J. Windley, editors, *FMCAD*, volume 1522 of *Lecture Notes in Computer Science*, pages 115–132. Springer, 1998.

[Her02] Holger Hermanns. *Interactive Markov Chains: The Quest for Quantified Quality*, volume 2428 of *Lecture Notes in Computer Science.* Springer, 2002.

[HHK02] Holger Hermanns, Ulrich Herzog, and Joost-Pieter Katoen. Process algebra for performance evaluation. *Theor. Comput. Sci.*, 274(1-2):43–87, 2002.

[Hoa85] Charles A. R. Hoare. *Communicating sequential processes.* Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1985.

[HTP05] Esfandiar Haghverdi, Paulo Tabuada, and George J. Pappas. Bisimulation relations for dynamical, control, and hybrid systems. *Theor. Comput. Sci.*, 342(2-3):229–261, 2005.

[Kai80] Thomas Kailath. *Linear Systems.* Prentice-Hall, Inc., 1980.

[Kat06] Joost-Pieter Katoen. Stochastic model checking. In Cassandras and Lygeros [CL06], pages 79–106.

[KKZJ07] Joost-Pieter Katoen, Tim Kemna, Ivan S. Zapreev, and David N. Jansen. Bisimulation Minimisation Mostly Speeds Up Probabilistic Model Checking. In Orna Grumberg and Michael Huth, editors, *TACAS*, volume 4424 of *Lecture Notes in Computer Science*, pages 87–101. Springer, 2007.

[Mil89] Robin J. Milner. *Communication and Concurrency.* Prentice-Hall, Inc., 1989.

[MT05] Manfred Morari and Lothar Thiele, editors. *Hybrid Systems: Computation and Control, 8th International Workshop, HSCC 2005, Zurich, Switzerland, March 9-11, 2005, Proceedings*, volume 3414 of *Lecture Notes in Computer Science.* Springer, 2005.

[PT87] Robert Paige and Robert Endre Tarjan. Three Partition Refinement Algorithms. *SIAM J. Comput.*, 16(6):973–989, 1987.

[RWVT95] Andy Rindos, Steven Woolet, Ioannis Viniotis, and Kishor S. Trivedi. Exact methods for the transient analysis of non-homogeneous continuous-time Markov chains. In *W. J. Stewart (ed.), Numerical Solution of Markov Chains (NSMC)*, pages 121–134. Kluwer Academic Publishers, 1995.

[SJvdS03] Stefan N. Strubbe, Agung A. Julius, and A. J. van der Schaft. Communicating piecewise deterministic Markov processes. In *IFAC Conf. on Analysis and Design of Hybrid Systems (ADHS)*, pages 349–354, 2003.

[SvdS05] Stefan N. Strubbe and A. J. van der Schaft. Bisimulation for Communicating Piecewise Deterministic Markov Processes (CPDPs). In Morari and Thiele [MT05], pages 623–639.

[SvdS06] Stefan N. Strubbe and A. J. van der Schaft. Compositional modelling of stochastic hybrid systems. In Cassandras and Lygeros [CL06], pages 47–77.

Aachener Informatik-Berichte

This is the list of all technical reports since 1987. To obtain copies of reports please consult http://aib.informatik.rwth-aachen.de/ or send your request to: Informatik-Bibliothek, RWTH Aachen, Ahornstr. 55, 52056 Aachen, Email: `biblio@informatik.rwth-aachen.de`

1987-01 * Fachgruppe Informatik: Jahresbericht 1986
1987-02 * David de Frutos Escrig, Klaus Indermark: Equivalence Relations of Non-Deterministic Ianov-Schemes
1987-03 * Manfred Nagl: A Software Development Environment based on Graph Technology
1987-04 * Claus Lewerentz, Manfred Nagl, Bernhard Westfechtel: On Integration Mechanisms within a Graph-Based Software Development Environment
1987-05 * Reinhard Rinn: Über Eingabeanomalien bei verschiedenen Inferenzmodellen
1987-06 * Werner Damm, Gert Döhmen: Specifying Distributed Computer Architectures in AADL*
1987-07 * Gregor Engels, Claus Lewerentz, Wilhelm Schäfer: Graph Grammar Engineering: A Software Specification Method
1987-08 * Manfred Nagl: Set Theoretic Approaches to Graph Grammars
1987-09 * Claus Lewerentz, Andreas Schürr: Experiences with a Database System for Software Documents
1987-10 * Herbert Klaeren, Klaus Indermark: A New Implementation Technique for Recursive Function Definitions
1987-11 * Rita Loogen: Design of a Parallel Programmable Graph Reduction Machine with Distributed Memory
1987-12 J. Börstler, U. Möncke, R. Wilhelm: Table compression for tree automata
1988-01 * Gabriele Esser, Johannes Rückert, Frank Wagner Gesellschaftliche Aspekte der Informatik
1988-02 * Peter Martini, Otto Spaniol: Token-Passing in High-Speed Backbone Networks for Campus-Wide Environments
1988-03 * Thomas Welzel: Simulation of a Multiple Token Ring Backbone
1988-04 * Peter Martini: Performance Comparison for HSLAN Media Access Protocols
1988-05 * Peter Martini: Performance Analysis of Multiple Token Rings
1988-06 * Andreas Mann, Johannes Rückert, Otto Spaniol: Datenfunknetze
1988-07 * Andreas Mann, Johannes Rückert: Packet Radio Networks for Data Exchange
1988-08 * Andreas Mann, Johannes Rückert: Concurrent Slot Assignment Protocol for Packet Radio Networks
1988-09 * W. Kremer, F. Reichert, J. Rückert, A. Mann: Entwurf einer Netzwerktopologie für ein Mobilfunknetz zur Unterstützung des öffentlichen Straßenverkehrs
1988-10 * Kai Jakobs: Towards User-Friendly Networking
1988-11 * Kai Jakobs: The Directory - Evolution of a Standard
1988-12 * Kai Jakobs: Directory Services in Distributed Systems - A Survey

1988-13 * Martine Schümmer: RS-511, a Protocol for the Plant Floor
1988-14 * U. Quernheim: Satellite Communication Protocols - A Performance Comparison Considering On-Board Processing
1988-15 * Peter Martini, Otto Spaniol, Thomas Welzel: File Transfer in High Speed Token Ring Networks: Performance Evaluation by Approximate Analysis and Simulation
1988-16 * Fachgruppe Informatik: Jahresbericht 1987
1988-17 * Wolfgang Thomas: Automata on Infinite Objects
1988-18 * Michael Sonnenschein: On Petri Nets and Data Flow Graphs
1988-19 * Heiko Vogler: Functional Distribution of the Contextual Analysis in Block-Structured Programming Languages: A Case Study of Tree Transducers
1988-20 * Thomas Welzel: Einsatz des Simulationswerkzeuges QNAP2 zur Leistungsbewertung von Kommunikationsprotokollen
1988-21 * Th. Janning, C. Lewerentz: Integrated Project Team Management in a Software Development Environment
1988-22 * Joost Engelfriet, Heiko Vogler: Modular Tree Transducers
1988-23 * Wolfgang Thomas: Automata and Quantifier Hierarchies
1988-24 * Uschi Heuter: Generalized Definite Tree Languages
1989-01 * Fachgruppe Informatik: Jahresbericht 1988
1989-02 * G. Esser, J. Rückert, F. Wagner (Hrsg.): Gesellschaftliche Aspekte der Informatik
1989-03 * Heiko Vogler: Bottom-Up Computation of Primitive Recursive Tree Functions
1989-04 * Andy Schürr: Introduction to PROGRESS, an Attribute Graph Grammar Based Specification Language
1989-05 J. Börstler: Reuse and Software Development - Problems, Solutions, and Bibliography (in German)
1989-06 * Kai Jakobs: OSI - An Appropriate Basis for Group Communication?
1989-07 * Kai Jakobs: ISO's Directory Proposal - Evolution, Current Status and Future Problems
1989-08 * Bernhard Westfechtel: Extension of a Graph Storage for Software Documents with Primitives for Undo/Redo and Revision Control
1989-09 * Peter Martini: High Speed Local Area Networks - A Tutorial
1989-10 * P. Davids, Th. Welzel: Performance Analysis of DQDB Based on Simulation
1989-11 * Manfred Nagl (Ed.): Abstracts of Talks presented at the WG '89 15th International Workshop on Graphtheoretic Concepts in Computer Science
1989-12 * Peter Martini: The DQDB Protocol - Is it Playing the Game?
1989-13 * Martine Schümmer: CNC/DNC Communication with MAP
1989-14 * Martine Schümmer: Local Area Networks for Manufactoring Environments with hard Real-Time Requirements
1989-15 * M. Schümmer, Th. Welzel, P. Martini: Integration of Field Bus and MAP Networks - Hierarchical Communication Systems in Production Environments
1989-16 * G. Vossen, K.-U. Witt: SUXESS: Towards a Sound Unification of Extensions of the Relational Data Model

1989-17 * J. Derissen, P. Hruschka, M.v.d. Beeck, Th. Janning, M. Nagl: Integrating Structured Analysis and Information Modelling

1989-18 A. Maassen: Programming with Higher Order Functions

1989-19 * Mario Rodriguez-Artalejo, Heiko Vogler: A Narrowing Machine for Syntax Directed BABEL

1989-20 H. Kuchen, R. Loogen, J.J. Moreno Navarro, M. Rodriguez Artalejo: Graph-based Implementation of a Functional Logic Language

1990-01 * Fachgruppe Informatik: Jahresbericht 1989

1990-02 * Vera Jansen, Andreas Potthoff, Wolfgang Thomas, Udo Wermuth: A Short Guide to the AMORE System (Computing Automata, MOnoids and Regular Expressions)

1990-03 * Jerzy Skurczynski: On Three Hierarchies of Weak SkS Formulas

1990-04 R. Loogen: Stack-based Implementation of Narrowing

1990-05 H. Kuchen, A. Wagener: Comparison of Dynamic Load Balancing Strategies

1990-06 * Kai Jakobs, Frank Reichert: Directory Services for Mobile Communication

1990-07 * Kai Jakobs: What's Beyond the Interface - OSI Networks to Support Cooperative Work

1990-08 * Kai Jakobs: Directory Names and Schema - An Evaluation

1990-09 * Ulrich Quernheim, Dieter Kreuer: Das CCITT - Signalisierungssystem Nr. 7 auf Satellitenstrecken; Simulation der Zeichengabestrecke

1990-11 H. Kuchen, R. Loogen, J.J. Moreno Navarro, M. Rodriguez Artalejo: Lazy Narrowing in a Graph Machine

1990-12 * Kai Jakobs, Josef Kaltwasser, Frank Reichert, Otto Spaniol: Der Computer fährt mit

1990-13 * Rudolf Mathar, Andreas Mann: Analyzing a Distributed Slot Assignment Protocol by Markov Chains

1990-14 A. Maassen: Compilerentwicklung in Miranda - ein Praktikum in funktionaler Programmierung (written in german)

1990-15 * Manfred Nagl, Andreas Schürr: A Specification Environment for Graph Grammars

1990-16 A. Schürr: PROGRESS: A VHL-Language Based on Graph Grammars

1990-17 * Marita Möller: Ein Ebenenmodell wissensbasierter Konsultationen - Unterstützung für Wissensakquisition und Erklärungsfähigkeit

1990-18 * Eric Kowalewski: Entwurf und Interpretation einer Sprache zur Beschreibung von Konsultationsphasen in Expertensystemen

1990-20 Y. Ortega Mallen, D. de Frutos Escrig: A Complete Proof System for Timed Observations

1990-21 * Manfred Nagl: Modelling of Software Architectures: Importance, Notions, Experiences

1990-22 H. Fassbender, H. Vogler: A Call-by-need Implementation of Syntax Directed Functional Programming

1991-01 Guenther Geiler (ed.), Fachgruppe Informatik: Jahresbericht 1990

1991-03 B. Steffen, A. Ingolfsdottir: Characteristic Formulae for Processes with Divergence

1991-04 M. Portz: A new class of cryptosystems based on interconnection networks

1991-05 H. Kuchen, G. Geiler: Distributed Applicative Arrays
1991-06 * Ludwig Staiger: Kolmogorov Complexity and Hausdorff Dimension
1991-07 * Ludwig Staiger: Syntactic Congruences for w-languages
1991-09 * Eila Kuikka: A Proposal for a Syntax-Directed Text Processing System
1991-10 K. Gladitz, H. Fassbender, H. Vogler: Compiler-based Implementation of Syntax-Directed Functional Programming
1991-11 R. Loogen, St. Winkler: Dynamic Detection of Determinism in Functional Logic Languages
1991-12 * K. Indermark, M. Rodriguez Artalejo (Eds.): Granada Workshop on the Integration of Functional and Logic Programming
1991-13 * Rolf Hager, Wolfgang Kremer: The Adaptive Priority Scheduler: A More Fair Priority Service Discipline
1991-14 * Andreas Fasbender, Wolfgang Kremer: A New Approximation Algorithm for Tandem Networks with Priority Nodes
1991-15 J. Börstler, A. Zündorf: Revisiting extensions to Modula-2 to support reusability
1991-16 J. Börstler, Th. Janning: Bridging the gap between Requirements Analysis and Design
1991-17 A. Zündorf, A. Schürr: Nondeterministic Control Structures for Graph Rewriting Systems
1991-18 * Matthias Jarke, John Mylopoulos, Joachim W. Schmidt, Yannis Vassiliou: DAIDA: An Environment for Evolving Information Systems
1991-19 M. Jeusfeld, M. Jarke: From Relational to Object-Oriented Integrity Simplification
1991-20 G. Hogen, A. Kindler, R. Loogen: Automatic Parallelization of Lazy Functional Programs
1991-21 * Prof. Dr. rer. nat. Otto Spaniol: ODP (Open Distributed Processing): Yet another Viewpoint
1991-22 H. Kuchen, F. Lücking, H. Stoltze: The Topology Description Language TDL
1991-23 S. Graf, B. Steffen: Compositional Minimization of Finite State Systems
1991-24 R. Cleaveland, J. Parrow, B. Steffen: The Concurrency Workbench: A Semantics Based Tool for the Verification of Concurrent Systems
1991-25 * Rudolf Mathar, Jürgen Mattfeldt: Optimal Transmission Ranges for Mobile Communication in Linear Multihop Packet Radio Networks
1991-26 M. Jeusfeld, M. Staudt: Query Optimization in Deductive Object Bases
1991-27 J. Knoop, B. Steffen: The Interprocedural Coincidence Theorem
1991-28 J. Knoop, B. Steffen: Unifying Strength Reduction and Semantic Code Motion
1991-30 T. Margaria: First-Order theories for the verification of complex FSMs
1991-31 B. Steffen: Generating Data Flow Analysis Algorithms from Modal Specifications
1992-01 Stefan Eherer (ed.), Fachgruppe Informatik: Jahresbericht 1991
1992-02 * Bernhard Westfechtel: Basismechanismen zur Datenverwaltung in strukturbezogenen Hypertextsystemen
1992-04 S. A. Smolka, B. Steffen: Priority as Extremal Probability
1992-05 * Matthias Jarke, Carlos Maltzahn, Thomas Rose: Sharing Processes: Team Coordination in Design Repositories

1992-06 O. Burkart, B. Steffen: Model Checking for Context-Free Processes

1992-07 * Matthias Jarke, Klaus Pohl: Information Systems Quality and Quality Information Systems

1992-08 * Rudolf Mathar, Jürgen Mattfeldt: Analyzing Routing Strategy NFP in Multihop Packet Radio Networks on a Line

1992-09 * Alfons Kemper, Guido Moerkotte: Grundlagen objektorientierter Datenbanksysteme

1992-10 Matthias Jarke, Manfred Jeusfeld, Andreas Miethsam, Michael Gocek: Towards a logic-based reconstruction of software configuration management

1992-11 Werner Hans: A Complete Indexing Scheme for WAM-based Abstract Machines

1992-12 W. Hans, R. Loogen, St. Winkler: On the Interaction of Lazy Evaluation and Backtracking

1992-13 * Matthias Jarke, Thomas Rose: Specification Management with CAD

1992-14 Th. Noll, H. Vogler: Top-down Parsing with Simultaneous Evaluation on Noncircular Attribute Grammars

1992-15 A. Schuerr, B. Westfechtel: Graphgrammatiken und Graphersetzungssysteme(written in german)

1992-16 * Graduiertenkolleg Informatik und Technik (Hrsg.): Forschungsprojekte des Graduiertenkollegs Informatik und Technik

1992-17 M. Jarke (ed.): ConceptBase V3.1 User Manual

1992-18 * Clarence A. Ellis, Matthias Jarke (Eds.): Distributed Cooperation in Integrated Information Systems - Proceedings of the Third International Workshop on Intelligent and Cooperative Information Systems

1992-19-00 H. Kuchen, R. Loogen (eds.): Proceedings of the 4th Int. Workshop on the Parallel Implementation of Functional Languages

1992-19-01 G. Hogen, R. Loogen: PASTEL - A Parallel Stack-Based Implementation of Eager Functional Programs with Lazy Data Structures (Extended Abstract)

1992-19-02 H. Kuchen, K. Gladitz: Implementing Bags on a Shared Memory MIMD-Machine

1992-19-03 C. Rathsack, S.B. Scholz: LISA - A Lazy Interpreter for a Full-Fledged Lambda-Calculus

1992-19-04 T.A. Bratvold: Determining Useful Parallelism in Higher Order Functions

1992-19-05 S. Kahrs: Polymorphic Type Checking by Interpretation of Code

1992-19-06 M. Chakravarty, M. Köhler: Equational Constraints, Residuation, and the Parallel JUMP-Machine

1992-19-07 J. Seward: Polymorphic Strictness Analysis using Frontiers (Draft Version)

1992-19-08 D. Gärtner, A. Kimms, W. Kluge: pi-Red^+ - A Compiling Graph-Reduction System for a Full Fledged Lambda-Calculus

1992-19-09 D. Howe, G. Burn: Experiments with strict STG code

1992-19-10 J. Glauert: Parallel Implementation of Functional Languages Using Small Processes

1992-19-11 M. Joy, T. Axford: A Parallel Graph Reduction Machine

1992-19-12 A. Bennett, P. Kelly: Simulation of Multicache Parallel Reduction

1992-19-13 K. Langendoen, D.J. Agterkamp: Cache Behaviour of Lazy Functional Programs (Working Paper)

1992-19-14 K. Hammond, S. Peyton Jones: Profiling scheduling strategies on the GRIP parallel reducer

1992-19-15 S. Mintchev: Using Strictness Information in the STG-machine

1992-19-16 D. Rushall: An Attribute Grammar Evaluator in Haskell

1992-19-17 J. Wild, H. Glaser, P. Hartel: Statistics on storage management in a lazy functional language implementation

1992-19-18 W.S. Martins: Parallel Implementations of Functional Languages

1992-19-19 D. Lester: Distributed Garbage Collection of Cyclic Structures (Draft version)

1992-19-20 J.C. Glas, R.F.H. Hofman, W.G. Vree: Parallelization of Branch-and-Bound Algorithms in a Functional Programming Environment

1992-19-21 S. Hwang, D. Rushall: The nu-STG machine: a parallelized Spineless Tagless Graph Reduction Machine in a distributed memory architecture (Draft version)

1992-19-22 G. Burn, D. Le Metayer: Cps-Translation and the Correctness of Optimising Compilers

1992-19-23 S.L. Peyton Jones, P. Wadler: Imperative functional programming (Brief summary)

1992-19-24 W. Damm, F. Liu, Th. Peikenkamp: Evaluation and Parallelization of Functions in Functional + Logic Languages (abstract)

1992-19-25 M. Kesseler: Communication Issues Regarding Parallel Functional Graph Rewriting

1992-19-26 Th. Peikenkamp: Charakterizing and representing neededness in functional loginc languages (abstract)

1992-19-27 H. Doerr: Monitoring with Graph-Grammars as formal operational Models

1992-19-28 J. van Groningen: Some implementation aspects of Concurrent Clean on distributed memory architectures

1992-19-29 G. Ostheimer: Load Bounding for Implicit Parallelism (abstract)

1992-20 H. Kuchen, F.J. Lopez Fraguas, J.J. Moreno Navarro, M. Rodriguez Artalejo: Implementing Disequality in a Lazy Functional Logic Language

1992-21 H. Kuchen, F.J. Lopez Fraguas: Result Directed Computing in a Functional Logic Language

1992-22 H. Kuchen, J.J. Moreno Navarro, M.V. Hermenegildo: Independent AND-Parallel Narrowing

1992-23 T. Margaria, B. Steffen: Distinguishing Formulas for Free

1992-24 K. Pohl: The Three Dimensions of Requirements Engineering

1992-25 * R. Stainov: A Dynamic Configuration Facility for Multimedia Communications

1992-26 * Michael von der Beeck: Integration of Structured Analysis and Timed Statecharts for Real-Time and Concurrency Specification

1992-27 W. Hans, St. Winkler: Aliasing and Groundness Analysis of Logic Programs through Abstract Interpretation and its Safety

1992-28 * Gerhard Steinke, Matthias Jarke: Support for Security Modeling in Information Systems Design

1992-29 B. Schinzel: Warum Frauenforschung in Naturwissenschaft und Technik

1992-30 A. Kemper, G. Moerkotte, K. Peithner: Object-Orientation Axiomatised by Dynamic Logic

1992-32 * Bernd Heinrichs, Kai Jakobs: Timer Handling in High-Performance Transport Systems

1992-33 * B. Heinrichs, K. Jakobs, K. Lenßen, W. Reinhardt, A. Spinner: Euro-Bridge: Communication Services for Multimedia Applications

1992-34 C. Gerlhof, A. Kemper, Ch. Kilger, G. Moerkotte: Partition-Based Clustering in Object Bases: From Theory to Practice

1992-35 J. Börstler: Feature-Oriented Classification and Reuse in IPSEN

1992-36 M. Jarke, J. Bubenko, C. Rolland, A. Sutcliffe, Y. Vassiliou: Theories Underlying Requirements Engineering: An Overview of NATURE at Genesis

1992-37 * K. Pohl, M. Jarke: Quality Information Systems: Repository Support for Evolving Process Models

1992-38 A. Zuendorf: Implementation of the imperative / rule based language PROGRES

1992-39 P. Koch: Intelligentes Backtracking bei der Auswertung funktional-logischer Programme

1992-40 * Rudolf Mathar, Jürgen Mattfeldt: Channel Assignment in Cellular Radio Networks

1992-41 * Gerhard Friedrich, Wolfgang Neidl: Constructive Utility in Model-Based Diagnosis Repair Systems

1992-42 * P. S. Chen, R. Hennicker, M. Jarke: On the Retrieval of Reusable Software Components

1992-43 W. Hans, St.Winkler: Abstract Interpretation of Functional Logic Languages

1992-44 N. Kiesel, A. Schuerr, B. Westfechtel: Design and Evaluation of GRAS, a Graph-Oriented Database System for Engineering Applications

1993-01 * Fachgruppe Informatik: Jahresbericht 1992

1993-02 * Patrick Shicheng Chen: On Inference Rules of Logic-Based Information Retrieval Systems

1993-03 G. Hogen, R. Loogen: A New Stack Technique for the Management of Runtime Structures in Distributed Environments

1993-05 A. Zündorf: A Heuristic for the Subgraph Isomorphism Problem in Executing PROGRES

1993-06 A. Kemper, D. Kossmann: Adaptable Pointer Swizzling Strategies in Object Bases: Design, Realization, and Quantitative Analysis

1993-07 * Graduiertenkolleg Informatik und Technik (Hrsg.): Graduiertenkolleg Informatik und Technik

1993-08 * Matthias Berger: k-Coloring Vertices using a Neural Network with Convergence to Valid Solutions

1993-09 M. Buchheit, M. Jeusfeld, W. Nutt, M. Staudt: Subsumption between Queries to Object-Oriented Databases

1993-10 O. Burkart, B. Steffen: Pushdown Processes: Parallel Composition and Model Checking

1993-11 * R. Große-Wienker, O. Hermanns, D. Menzenbach, A. Pollacks, S. Repetzki, J. Schwartz, K. Sonnenschein, B. Westfechtel: Das SUKITS-Projekt: A-posteriori-Integration heterogener CIM-Anwendungssysteme

1993-12 * Rudolf Mathar, Jürgen Mattfeldt: On the Distribution of Cumulated Interference Power in Rayleigh Fading Channels
1993-13 O. Maler, L. Staiger: On Syntactic Congruences for omega-languages
1993-14 M. Jarke, St. Eherer, R. Gallersdoerfer, M. Jeusfeld, M. Staudt: ConceptBase - A Deductive Object Base Manager
1993-15 M. Staudt, H.W. Nissen, M.A. Jeusfeld: Query by Class, Rule and Concept
1993-16 * M. Jarke, K. Pohl, St. Jacobs et al.: Requirements Engineering: An Integrated View of Representation Process and Domain
1993-17 * M. Jarke, K. Pohl: Establishing Vision in Context: Towards a Model of Requirements Processes
1993-18 W. Hans, H. Kuchen, St. Winkler: Full Indexing for Lazy Narrowing
1993-19 W. Hans, J.J. Ruz, F. Saenz, St. Winkler: A VHDL Specification of a Shared Memory Parallel Machine for Babel
1993-20 * K. Finke, M. Jarke, P. Szczurko, R. Soltysiak: Quality Management for Expert Systems in Process Control
1993-21 M. Jarke, M.A. Jeusfeld, P. Szczurko: Three Aspects of Intelligent Cooperation in the Quality Cycle
1994-01 Margit Generet, Sven Martin (eds.), Fachgruppe Informatik: Jahresbericht 1993
1994-02 M. Lefering: Development of Incremental Integration Tools Using Formal Specifications
1994-03 * P. Constantopoulos, M. Jarke, J. Mylopoulos, Y. Vassiliou: The Software Information Base: A Server for Reuse
1994-04 * Rolf Hager, Rudolf Mathar, Jürgen Mattfeldt: Intelligent Cruise Control and Reliable Communication of Mobile Stations
1994-05 * Rolf Hager, Peter Hermesmann, Michael Portz: Feasibility of Authentication Procedures within Advanced Transport Telematics
1994-06 * Claudia Popien, Bernd Meyer, Axel Kuepper: A Formal Approach to Service Import in ODP Trader Federations
1994-07 P. Peters, P. Szczurko: Integrating Models of Quality Management Methods by an Object-Oriented Repository
1994-08 * Manfred Nagl, Bernhard Westfechtel: A Universal Component for the Administration in Distributed and Integrated Development Environments
1994-09 * Patrick Horster, Holger Petersen: Signatur- und Authentifikationsverfahren auf der Basis des diskreten Logarithmusproblems
1994-11 A. Schürr: PROGRES, A Visual Language and Environment for PROgramming with Graph REwrite Systems
1994-12 A. Schürr: Specification of Graph Translators with Triple Graph Grammars
1994-13 A. Schürr: Logic Based Programmed Structure Rewriting Systems
1994-14 L. Staiger: Codes, Simplifying Words, and Open Set Condition
1994-15 * Bernhard Westfechtel: A Graph-Based System for Managing Configurations of Engineering Design Documents
1994-16 P. Klein: Designing Software with Modula-3
1994-17 I. Litovsky, L. Staiger: Finite acceptance of infinite words

1994-18 G. Hogen, R. Loogen: Parallel Functional Implementations: Graphbased vs. Stackbased Reduction

1994-19 M. Jeusfeld, U. Johnen: An Executable Meta Model for Re-Engineering of Database Schemas

1994-20 * R. Gallersdörfer, M. Jarke, K. Klabunde: Intelligent Networks as a Data Intensive Application (INDIA)

1994-21 M. Mohnen: Proving the Correctness of the Static Link Technique Using Evolving Algebras

1994-22 H. Fernau, L. Staiger: Valuations and Unambiguity of Languages, with Applications to Fractal Geometry

1994-24 * M. Jarke, K. Pohl, R. Dömges, St. Jacobs, H. W. Nissen: Requirements Information Management: The NATURE Approach

1994-25 * M. Jarke, K. Pohl, C. Rolland, J.-R. Schmitt: Experience-Based Method Evaluation and Improvement: A Process Modeling Approach

1994-26 * St. Jacobs, St. Kethers: Improving Communication and Decision Making within Quality Function Deployment

1994-27 * M. Jarke, H. W. Nissen, K. Pohl: Tool Integration in Evolving Information Systems Environments

1994-28 O. Burkart, D. Caucal, B. Steffen: An Elementary Bisimulation Decision Procedure for Arbitrary Context-Free Processes

1995-01 * Fachgruppe Informatik: Jahresbericht 1994

1995-02 Andy Schürr, Andreas J. Winter, Albert Zündorf: Graph Grammar Engineering with PROGRES

1995-03 Ludwig Staiger: A Tight Upper Bound on Kolmogorov Complexity by Hausdorff Dimension and Uniformly Optimal Prediction

1995-04 Birgitta König-Ries, Sven Helmer, Guido Moerkotte: An experimental study on the complexity of left-deep join ordering problems for cyclic queries

1995-05 Sophie Cluet, Guido Moerkotte: Efficient Evaluation of Aggregates on Bulk Types

1995-06 Sophie Cluet, Guido Moerkotte: Nested Queries in Object Bases

1995-07 Sophie Cluet, Guido Moerkotte: Query Optimization Techniques Exploiting Class Hierarchies

1995-08 Markus Mohnen: Efficient Compile-Time Garbage Collection for Arbitrary Data Structures

1995-09 Markus Mohnen: Functional Specification of Imperative Programs: An Alternative Point of View of Functional Languages

1995-10 Rainer Gallersdörfer, Matthias Nicola: Improving Performance in Replicated Databases through Relaxed Coherency

1995-11 * M.Staudt, K.von Thadden: Subsumption Checking in Knowledge Bases

1995-12 * G.V.Zemanek, H.W.Nissen, H.Hubert, M.Jarke: Requirements Analysis from Multiple Perspectives: Experiences with Conceptual Modeling Technology

1995-13 * M.Staudt, M.Jarke: Incremental Maintenance of Externally Materialized Views

1995-14 * P.Peters, P.Szczurko, M.Jeusfeld: Oriented Information Management: Conceptual Models at Work

1995-15 * Matthias Jarke, Sudha Ram (Hrsg.): WITS 95 Proceedings of the 5th Annual Workshop on Information Technologies and Systems
1995-16 * W.Hans, St.Winkler, F.Saenz: Distributed Execution in Functional Logic Programming
1996-01 * Jahresbericht 1995
1996-02 Michael Hanus, Christian Prehofer: Higher-Order Narrowing with Definitional Trees
1996-03 * W.Scheufele, G.Moerkotte: Optimal Ordering of Selections and Joins in Acyclic Queries with Expensive Predicates
1996-04 Klaus Pohl: PRO-ART: Enabling Requirements Pre-Traceability
1996-05 Klaus Pohl: Requirements Engineering: An Overview
1996-06 * M.Jarke, W.Marquardt: Design and Evaluation of Computer–Aided Process Modelling Tools
1996-07 Olaf Chitil: The Sigma-Semantics: A Comprehensive Semantics for Functional Programs
1996-08 * S.Sripada: On Entropy and the Limitations of the Second Law of Thermodynamics
1996-09 Michael Hanus (Ed.): Proceedings of the Poster Session of ALP96 - Fifth International Conference on Algebraic and Logic Programming
1996-09-0 Michael Hanus (Ed.): Proceedings of the Poster Session of ALP 96 - Fifth International Conference on Algebraic and Logic Programming: Introduction and table of contents
1996-09-1 Ilies Alouini: An Implementation of Conditional Concurrent Rewriting on Distributed Memory Machines
1996-09-2 Olivier Danvy, Karoline Malmkjær: On the Idempotence of the CPS Transformation
1996-09-3 Víctor M. Gulías, José L. Freire: Concurrent Programming in Haskell
1996-09-4 Sébastien Limet, Pierre Réty: On Decidability of Unifiability Modulo Rewrite Systems
1996-09-5 Alexandre Tessier: Declarative Debugging in Constraint Logic Programming
1996-10 Reidar Conradi, Bernhard Westfechtel: Version Models for Software Configuration Management
1996-11 * C.Weise, D.Lenzkes: A Fast Decision Algorithm for Timed Refinement
1996-12 * R.Dömges, K.Pohl, M.Jarke, B.Lohmann, W.Marquardt: PRO-ART/CE* — An Environment for Managing the Evolution of Chemical Process Simulation Models
1996-13 * K.Pohl, R.Klamma, K.Weidenhaupt, R.Dömges, P.Haumer, M.Jarke: A Framework for Process-Integrated Tools
1996-14 * R.Gallersdörfer, K.Klabunde, A.Stolz, M.Eßmajor: INDIA — Intelligent Networks as a Data Intensive Application, Final Project Report, June 1996
1996-15 * H.Schimpe, M.Staudt: VAREX: An Environment for Validating and Refining Rule Bases
1996-16 * M.Jarke, M.Gebhardt, S.Jacobs, H.Nissen: Conflict Analysis Across Heterogeneous Viewpoints: Formalization and Visualization
1996-17 Manfred A. Jeusfeld, Tung X. Bui: Decision Support Components on the Internet

1996-18 Manfred A. Jeusfeld, Mike Papazoglou: Information Brokering: Design, Search and Transformation

1996-19 * P.Peters, M.Jarke: Simulating the impact of information flows in networked organizations

1996-20 Matthias Jarke, Peter Peters, Manfred A. Jeusfeld: Model-driven planning and design of cooperative information systems

1996-21 * G.de Michelis, E.Dubois, M.Jarke, F.Matthes, J.Mylopoulos, K.Pohl, J.Schmidt, C.Woo, E.Yu: Cooperative information systems: a manifesto

1996-22 * S.Jacobs, M.Gebhardt, S.Kethers, W.Rzasa: Filling HTML forms simultaneously: CoWeb architecture and functionality

1996-23 * M.Gebhardt, S.Jacobs: Conflict Management in Design

1997-01 Michael Hanus, Frank Zartmann (eds.): Jahresbericht 1996

1997-02 Johannes Faassen: Using full parallel Boltzmann Machines for Optimization

1997-03 Andreas Winter, Andy Schürr: Modules and Updatable Graph Views for PROgrammed Graph REwriting Systems

1997-04 Markus Mohnen, Stefan Tobies: Implementing Context Patterns in the Glasgow Haskell Compiler

1997-05 * S.Gruner: Schemakorrespondenzaxiome unterstützen die paargrammatische Spezifikation inkrementeller Integrationswerkzeuge

1997-06 Matthias Nicola, Matthias Jarke: Design and Evaluation of Wireless Health Care Information Systems in Developing Countries

1997-07 Petra Hofstedt: Taskparallele Skelette für irregulär strukturierte Probleme in deklarativen Sprachen

1997-08 Dorothea Blostein, Andy Schürr: Computing with Graphs and Graph Rewriting

1997-09 Carl-Arndt Krapp, Bernhard Westfechtel: Feedback Handling in Dynamic Task Nets

1997-10 Matthias Nicola, Matthias Jarke: Integrating Replication and Communication in Performance Models of Distributed Databases

1997-11 * R. Klamma, P. Peters, M. Jarke: Workflow Support for Failure Management in Federated Organizations

1997-13 Markus Mohnen: Optimising the Memory Management of Higher-Order Functional Programs

1997-14 Roland Baumann: Client/Server Distribution in a Structure-Oriented Database Management System

1997-15 George Botorog: High-Level Parallel Programming and the Efficient Implementation of Numerical Algorithms

1998-01 * Fachgruppe Informatik: Jahresbericht 1997

1998-02 Stefan Gruner, Manfred Nagel, Andy Schürr: Fine-grained and Structure-Oriented Document Integration Tools are Needed for Development Processes

1998-03 Stefan Gruner: Einige Anmerkungen zur graphgrammatischen Spezifikation von Integrationswerkzeugen nach Westfechtel, Janning, Lefering und Schürr

1998-04 * O. Kubitz: Mobile Robots in Dynamic Environments

1998-05 Martin Leucker, Stephan Tobies: Truth - A Verification Platform for Distributed Systems

1998-06 *	Matthias Oliver Berger: DECT in the Factory of the Future
1998-07	M. Arnold, M. Erdmann, M. Glinz, P. Haumer, R. Knoll, B. Paech, K. Pohl, J. Ryser, R. Studer, K. Weidenhaupt: Survey on the Scenario Use in Twelve Selected Industrial Projects
1998-09 *	Th. Lehmann: Geometrische Ausrichtung medizinischer Bilder am Beispiel intraoraler Radiographien
1998-10 *	M. Nicola, M. Jarke: Performance Modeling of Distributed and Replicated Databases
1998-11 *	Ansgar Schleicher, Bernhard Westfechtel, Dirk Jäger: Modeling Dynamic Software Processes in UML
1998-12 *	W. Appelt, M. Jarke: Interoperable Tools for Cooperation Support using the World Wide Web
1998-13	Klaus Indermark: Semantik rekursiver Funktionsdefinitionen mit Striktheitsinformation
1999-01 *	Jahresbericht 1998
1999-02 *	F. Huch: Verifcation of Erlang Programs using Abstract Interpretation and Model Checking — Extended Version
1999-03 *	R. Gallersdörfer, M. Jarke, M. Nicola: The ADR Replication Manager
1999-04	María Alpuente, Michael Hanus, Salvador Lucas, Germán Vidal: Specialization of Functional Logic Programs Based on Needed Narrowing
1999-05 *	W. Thomas (Ed.): DLT 99 - Developments in Language Theory Fourth International Conference
1999-06 *	Kai Jakobs, Klaus-Dieter Kleefeld: Informationssysteme für die angewandte historische Geographie
1999-07	Thomas Wilke: CTL+ is exponentially more succinct than CTL
1999-08	Oliver Matz: Dot-Depth and Monadic Quantifier Alternation over Pictures
2000-01 *	Jahresbericht 1999
2000-02	Jens Vöge, Marcin Jurdzinski A Discrete Strategy Improvement Algorithm for Solving Parity Games
2000-03	D. Jäger, A. Schleicher, B. Westfechtel: UPGRADE: A Framework for Building Graph-Based Software Engineering Tools
2000-04	Andreas Becks, Stefan Sklorz, Matthias Jarke: Exploring the Semantic Structure of Technical Document Collections: A Cooperative Systems Approach
2000-05	Mareike Schoop: Cooperative Document Management
2000-06	Mareike Schoop, Christoph Quix (eds.): Proceedings of the Fifth International Workshop on the Language-Action Perspective on Communication Modelling
2000-07 *	Markus Mohnen, Pieter Koopman (Eds.): Proceedings of the 12th International Workshop of Functional Languages
2000-08	Thomas Arts, Thomas Noll: Verifying Generic Erlang Client-Server Implementations
2001-01 *	Jahresbericht 2000
2001-02	Benedikt Bollig, Martin Leucker: Deciding LTL over Mazurkiewicz Traces
2001-03	Thierry Cachat: The power of one-letter rational languages

2001-04 Benedikt Bollig, Martin Leucker, Michael Weber: Local Parallel Model Checking for the Alternation Free mu-Calculus

2001-05 Benedikt Bollig, Martin Leucker, Thomas Noll: Regular MSC Languages

2001-06 Achim Blumensath: Prefix-Recognisable Graphs and Monadic Second-Order Logic

2001-07 Martin Grohe, Stefan Wöhrle: An Existential Locality Theorem

2001-08 Mareike Schoop, James Taylor (eds.): Proceedings of the Sixth International Workshop on the Language-Action Perspective on Communication Modelling

2001-09 Thomas Arts, Jürgen Giesl: A collection of examples for termination of term rewriting using dependency pairs

2001-10 Achim Blumensath: Axiomatising Tree-interpretable Structures

2001-11 Klaus Indermark, Thomas Noll (eds.): Kolloquium Programmiersprachen und Grundlagen der Programmierung

2002-01 * Jahresbericht 2001

2002-02 Jürgen Giesl, Aart Middeldorp: Transformation Techniques for Context-Sensitive Rewrite Systems

2002-03 Benedikt Bollig, Martin Leucker, Thomas Noll: Generalised Regular MSC Languages

2002-04 Jürgen Giesl, Aart Middeldorp: Innermost Termination of Context-Sensitive Rewriting

2002-05 Horst Lichter, Thomas von der Maßen, Thomas Weiler: Modelling Requirements and Architectures for Software Product Lines

2002-06 Henry N. Adorna: 3-Party Message Complexity is Better than 2-Party Ones for Proving Lower Bounds on the Size of Minimal Nondeterministic Finite Automata

2002-07 Jörg Dahmen: Invariant Image Object Recognition using Gaussian Mixture Densities

2002-08 Markus Mohnen: An Open Framework for Data-Flow Analysis in Java

2002-09 Markus Mohnen: Interfaces with Default Implementations in Java

2002-10 Martin Leucker: Logics for Mazurkiewicz traces

2002-11 Jürgen Giesl, Hans Zantema: Liveness in Rewriting

2003-01 * Jahresbericht 2002

2003-02 Jürgen Giesl, René Thiemann: Size-Change Termination for Term Rewriting

2003-03 Jürgen Giesl, Deepak Kapur: Deciding Inductive Validity of Equations

2003-04 Jürgen Giesl, René Thiemann, Peter Schneider-Kamp, Stephan Falke: Improving Dependency Pairs

2003-05 Christof Löding, Philipp Rohde: Solving the Sabotage Game is PSPACE-hard

2003-06 Franz Josef Och: Statistical Machine Translation: From Single-Word Models to Alignment Templates

2003-07 Horst Lichter, Thomas von der Maßen, Alexander Nyßen, Thomas Weiler: Vergleich von Ansätzen zur Feature Modellierung bei der Softwareproduktlinienentwicklung

2003-08 Jürgen Giesl, René Thiemann, Peter Schneider-Kamp, Stephan Falke: Mechanizing Dependency Pairs

2004-01 * Fachgruppe Informatik: Jahresbericht 2003

2004-02 Benedikt Bollig, Martin Leucker: Message-Passing Automata are expressively equivalent to EMSO logic
2004-03 Delia Kesner, Femke van Raamsdonk, Joe Wells (eds.): HOR 2004 – 2nd International Workshop on Higher-Order Rewriting
2004-04 Slim Abdennadher, Christophe Ringeissen (eds.): RULE 04 – Fifth International Workshop on Rule-Based Programming
2004-05 Herbert Kuchen (ed.): WFLP 04 – 13th International Workshop on Functional and (Constraint) Logic Programming
2004-06 Sergio Antoy, Yoshihito Toyama (eds.): WRS 04 – 4th International Workshop on Reduction Strategies in Rewriting and Programming
2004-07 Michael Codish, Aart Middeldorp (eds.): WST 04 – 7th International Workshop on Termination
2004-08 Klaus Indermark, Thomas Noll: Algebraic Correctness Proofs for Compiling Recursive Function Definitions with Strictness Information
2004-09 Joachim Kneis, Daniel Mölle, Stefan Richter, Peter Rossmanith: Parameterized Power Domination Complexity
2004-10 Zinaida Benenson, Felix C. Gärtner, Dogan Kesdogan: Secure Multi-Party Computation with Security Modules
2005-01 * Fachgruppe Informatik: Jahresbericht 2004
2005-02 Maximillian Dornseif, Felix C. Gärtner, Thorsten Holz, Martin Mink: An Offensive Approach to Teaching Information Security: "Aachen Summer School Applied IT Security"
2005-03 Jürgen Giesl, René Thiemann, Peter Schneider-Kamp: Proving and Disproving Termination of Higher-Order Functions
2005-04 Daniel Mölle, Stefan Richter, Peter Rossmanith: A Faster Algorithm for the Steiner Tree Problem
2005-05 Fabien Pouget, Thorsten Holz: A Pointillist Approach for Comparing Honeypots
2005-06 Simon Fischer, Berthold Vöcking: Adaptive Routing with Stale Information
2005-07 Felix C. Freiling, Thorsten Holz, Georg Wicherski: Botnet Tracking: Exploring a Root-Cause Methodology to Prevent Distributed Denial-of-Service Attacks
2005-08 Joachim Kneis, Peter Rossmanith: A New Satisfiability Algorithm With Applications To Max-Cut
2005-09 Klaus Kursawe, Felix C. Freiling: Byzantine Fault Tolerance on General Hybrid Adversary Structures
2005-10 Benedikt Bollig: Automata and Logics for Message Sequence Charts
2005-11 Simon Fischer, Berthold Vöcking: A Counterexample to the Fully Mixed Nash Equilibrium Conjecture
2005-12 Neeraj Mittal, Felix Freiling, S. Venkatesan, Lucia Draque Penso: Efficient Reductions for Wait-Free Termination Detection in Faulty Distributed Systems
2005-13 Carole Delporte-Gallet, Hugues Fauconnier, Felix C. Freiling: Revisiting Failure Detection and Consensus in Omission Failure Environments
2005-14 Felix C. Freiling, Sukumar Ghosh: Code Stabilization
2005-15 Uwe Naumann: The Complexity of Derivative Computation

2005-16 Uwe Naumann: Syntax-Directed Derivative Code (Part I: Tangent-Linear Code)

2005-17 Uwe Naumann: Syntax-directed Derivative Code (Part II: Intraprocedural Adjoint Code)

2005-18 Thomas von der Maßen, Klaus Müller, John MacGregor, Eva Geisberger, Jörg Dörr, Frank Houdek, Harbhajan Singh, Holger Wußmann, Hans-Veit Bacher, Barbara Paech: Einsatz von Features im Software-Entwicklungsprozess - Abschlußbericht des GI-Arbeitskreises "Features"

2005-19 Uwe Naumann, Andre Vehreschild: Tangent-Linear Code by Augmented LL-Parsers

2005-20 Felix C. Freiling, Martin Mink: Bericht über den Workshop zur Ausbildung im Bereich IT-Sicherheit Hochschulausbildung, berufliche Weiterbildung, Zertifizierung von Ausbildungsangeboten am 11. und 12. August 2005 in Köln organisiert von RWTH Aachen in Kooperation mit BITKOM, BSI, DLR und Gesellschaft fuer Informatik (GI) e.V.

2005-21 Thomas Noll, Stefan Rieger: Optimization of Straight-Line Code Revisited

2005-22 Felix Freiling, Maurice Herlihy, Lucia Draque Penso: Optimal Randomized Fair Exchange with Secret Shared Coins

2005-23 Heiner Ackermann, Alantha Newman, Heiko Röglin, Berthold Vöcking: Decision Making Based on Approximate and Smoothed Pareto Curves

2005-24 Alexander Becher, Zinaida Benenson, Maximillian Dornseif: Tampering with Motes: Real-World Physical Attacks on Wireless Sensor Networks

2006-01 * Fachgruppe Informatik: Jahresbericht 2005

2006-02 Michael Weber: Parallel Algorithms for Verification of Large Systems

2006-03 Michael Maier, Uwe Naumann: Intraprocedural Adjoint Code Generated by the Differentiation-Enabled NAGWare Fortran Compiler

2006-04 Ebadollah Varnik, Uwe Naumann, Andrew Lyons: Toward Low Static Memory Jacobian Accumulation

2006-05 Uwe Naumann, Jean Utke, Patrick Heimbach, Chris Hill, Derya Ozyurt, Carl Wunsch, Mike Fagan, Nathan Tallent, Michelle Strout: Adjoint Code by Source Transformation with OpenAD/F

2006-06 Joachim Kneis, Daniel Mölle, Stefan Richter, Peter Rossmanith: Divide-and-Color

2006-07 Thomas Colcombet, Christof Löding: Transforming structures by set interpretations

2006-08 Uwe Naumann, Yuxiao Hu: Optimal Vertex Elimination in Single-Expression-Use Graphs

2006-09 Tingting Han, Joost-Pieter Katoen: Counterexamples in Probabilistic Model Checking

2006-10 Mesut Günes, Alexander Zimmermann, Martin Wenig, Jan Ritzerfeld, Ulrich Meis: From Simulations to Testbeds - Architecture of the Hybrid MCG-Mesh Testbed

2006-11 Bastian Schlich, Michael Rohrbach, Michael Weber, Stefan Kowalewski: Model Checking Software for Microcontrollers

2006-12 Benedikt Bollig, Joost-Pieter Katoen, Carsten Kern, Martin Leucker: Replaying Play in and Play out: Synthesis of Design Models from Scenarios by Learning

2006-13 Wong Karianto, Christof Löding: Unranked Tree Automata with Sibling Equalities and Disequalities

2006-14 Danilo Beuche, Andreas Birk, Heinrich Dreier, Andreas Fleischmann, Heidi Galle, Gerald Heller, Dirk Janzen, Isabel John, Ramin Tavakoli Kolagari, Thomas von der Maßen, Andreas Wolfram: Report of the GI Work Group “Requirements Management Tools for Product Line Engineering”

2006-15 Sebastian Ullrich, Jakob T. Valvoda, Torsten Kuhlen: Utilizing optical sensors from mice for new input devices

2006-16 Rafael Ballagas, Jan Borchers: Selexels: a Conceptual Framework for Pointing Devices with Low Expressiveness

2006-17 Eric Lee, Henning Kiel, Jan Borchers: Scrolling Through Time: Improving Interfaces for Searching and Navigating Continuous Audio Timelines

2007-01 * Fachgruppe Informatik: Jahresbericht 2006

2007-02 Carsten Fuhs, Jürgen Giesl, Aart Middeldorp, Peter Schneider-Kamp, René Thiemann, and Harald Zankl: SAT Solving for Termination Analysis with Polynomial Interpretations

2007-03 Jürgen Giesl, René Thiemann, Stephan Swiderski, and Peter Schneider-Kamp: Proving Termination by Bounded Increase

2007-04 Jan Buchholz, Eric Lee, Jonathan Klein, and Jan Borchers: coJIVE: A System to Support Collaborative Jazz Improvisation

2007-05 Uwe Naumann: On Optimal DAG Reversal

2007-06 Joost-Pieter Katoen, Thomas Noll, and Stefan Rieger: Verifying Concurrent List-Manipulating Programs by LTL Model Checking

2007-07 Alexander Nyßen, Horst Lichter: MeDUSA - MethoD for UML2-based Design of Embedded Software Applications

2007-08 Falk Salewski and Stefan Kowalewski: Achieving Highly Reliable Embedded Software: An empirical evaluation of different approaches

2007-09 Tina Kraußer, Heiko Mantel, and Henning Sudbrock: A Probabilistic Justification of the Combining Calculus under the Uniform Scheduler Assumption

2007-10 Martin Neuhäußer, Joost-Pieter Katoen: Bisimulation and Logical Preservation for Continuous-Time Markov Decision Processes

2007-11 Klaus Wehrle: 6. Fachgespräch Sensornetzwerke

2007-12 Uwe Naumann: An L-Attributed Grammar for Adjoint Code

2007-13 Uwe Naumann, Michael Maier, Jan Riehme, and Bruce Christianson: Second-Order Adjoints by Source Code Manipulation of Numerical Programs

2007-14 Jean Utke, Uwe Naumann, Mike Fagan, Nathan Tallent, Michelle Strout, Patrick Heimbach, Chris Hill, and Carl Wunsch: OpenAD/F: A Modular, Open-Source Tool for Automatic Differentiation of Fortran Codes

2007-15 Volker Stolz: Temporal assertions for sequential and current programs

2007-16 Sadeq Ali Makram, Mesut Güneç, Martin Wenig, Alexander Zimmermann: Adaptive Channel Assignment to Support QoS and Load Balancing for Wireless Mesh Networks

2007-17 René Thiemann: The DP Framework for Proving Termination of Term Rewriting

2007-20 Joost-Pieter Katoen, Daniel Klink, Martin Leucker, and Verena Wolf: Three-Valued Abstraction for Probabilistic Systems

2007-22 Heiner Ackermann, Paul W. Goldberg, Vahab S. Mirrokni, Heiko Röglin, and Berthold Vöcking: Uncoordinated Two-Sided Markets

* These reports are only available as a printed version.

Please contact `biblio@informatik.rwth-aachen.de` to obtain copies.