RWTH Aachen

Department of Computer Science
Technical Report

Quantitative Model Checking of Continuous-Time Markov Chains Against Timed Automata Specifications

Taolue Chen, Tingting Han, Joost-Pieter Katoen, and Alexandru Mereacre

ISSN 0935–3232 · Aachener Informatik Berichte · AIB-2009-02

RWTH Aachen · Department of Computer Science · August 2009

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/

Quantitative Model Checking of Continuous-Time Markov Chains Against Timed Automata Specifications

Taolue Chen[1], Tingting Han[2,3], Joost-Pieter Katoen[2,3], and Alexandru Mereacre[2]

[1] Design and Analysis of Communication Systems, University of Twente, The Netherlands
[2] Software Modelling and Verification, RWTH Aachen University, Germany
[3] Formal Methods and Tools, University of Twente, The Netherlands

Abstract. We study the following problem: given a continuous-time Markov chain (CTMC) $\mathcal{C}$, and a linear real-time property provided as a deterministic timed automaton (DTA) $\mathcal{A}$, what is the probability of the set of paths of $\mathcal{C}$ that are accepted by $\mathcal{A}$ ($\mathcal{C}$ satisfies $\mathcal{A}$)? It is shown that this set of paths is measurable and computing its probability can be reduced to computing the reachability probability in a piecewise deterministic Markov process (PDP). The reachability probability is characterized as the least solution of a system of integral equations and is shown to be approximated by solving a system of partial differential equations. For the special case of single-clock DTA, the system of integral equations can be transformed into a system of linear equations where the coefficients are solutions of ordinary differential equations.

1 Introduction

Continuous-time Markov chains (CTMCs) are one of the most important models in performance and dependability analysis. They are exploited in a broad range of applications, and constitute the underlying semantical model of a plethora of modeling formalisms for real-time probabilistic systems such as Markovian queueing networks, stochastic Petri nets, stochastic variants of process algebras, and, more recently, calculi for system biology. CTMC model checking has been focused on the temporal logic CSL (Continuous Stochastic Logic [ASSB00,BHHK03]), a variant of timed CTL where the CTL path quantifiers are replaced by a probabilistic operator. CSL model checking proceeds — like CTL model checking — by a recursive descent over the parse tree of the formula. One of the key ingredients is that reachability probabilities for time-bounded until-formulae can be approximated arbitrarily closely by a reduction to transient analysis in CTMCs. This results in a polynomial-time algorithm that has been realized in model checkers such as PRISM and MRMC.

This paper concerns the problem of verifying CTMCs versus *linear* real-time specifications, which are based on timed automata. Concretely speaking, we explore the following problem: given a CTMC $\mathcal{C}$, and a linear real-time property provided as a *deterministic timed automaton* [AD94] (DTA) $\mathcal{A}$, what is the probability of the set of paths of $\mathcal{C}$ which are accepted by $\mathcal{A}$ ($\mathcal{C} \models \mathcal{A}$)? We consider two kinds of acceptance conditions: the reachability condition and the Muller acceptance condition. The former accepts (finite) paths which reach some final state and the latter accepts (infinite) paths that infinitely often visit some set of final states. We set off to show that this problem is well-defined in the sense that the path set is *measurable*. Computing its probability can then be reduced to computing the reachability probability in a piecewise deterministic Markov process (PDP) [Dav93], a model that is used in, e.g., stochastic control theory and financial mathematics. This result relies on a product construction of CTMC $\mathcal{C}$ and DTA $\mathcal{A}$, denoted $\mathcal{C} \otimes \mathcal{A}$, yielding *deterministic Markov timed automata* (DMTA), a variant of DTA in which, besides the usual ingredients of timed automata, like guards and clock resets, the location residence time is exponentially distributed. We show that the probability of $\mathcal{C} \models \mathcal{A}$ coincides with the reachability probability of accepting paths in $\mathcal{C} \otimes \mathcal{A}$. The underlying PDP of a DMTA is obtained by a slight adaptation of the standard region construction. The desired reachability probability is characterized as the least solution of a system of *integral equations* that is obtained from the PDP. Finally, this probability is shown to be approximated by solving a system of *partial differential equations* (PDEs). For single-clock DTA, we show that the system of integral equations can be transformed into a system of *linear equations*, where the coefficients are solutions of some *ordinary differential equations* (ODEs), which can either have an analytical solution (for small state space) or an arbitrarily closely approximated solution efficiently.

Related work is model checking of asCSL [BCH+07] and CSL^{TA} [DHS09]. asCSL allows to impose a time constraint on action sequences described by regular expressions; its model-checking algorithm is based on a deterministic Rabin automaton construction. In CSL^{TA}, time constraints (of until modalities) are specified by *single-clock* DTA. In [DHS09], $\mathcal{C} \otimes \mathcal{A}$ is interpreted as a Markov renewal processes and model checking CSL^{TA} is reduced to computing reachability probabilities in a DTMC whose transition probabilities are given by subordinate CTMCs. This technique cannot be generalized to multiple clocks. Our approach does not restrict the number of clocks and supports more specifications than CSL^{TA}. For the single-clock case, our approach produces the same result as [DHS09], but yields a conceptually simpler formulation whose correctness can be derived from the simplification of the system of integral equations obtained in the general case. Moreover, measurability has not been addressed in [DHS09]. Other related work [BBB+07,BBB+08,BBBM08] provides a quantitative interpretation to timed automata where delays and discrete choices are interpreted probabilistically. In this approach, delays of unbounded clocks are governed by exponential distributions like in CTMCs. Decidability results have been obtained for almost-sure properties [BBB+08] and quantitative verification [BBBM08] for (a subclass of) single-clock timed automata.

2 Preliminaries

Given a set H, let $\Pr : \mathcal{F}(H) \to [0,1]$ be a probability measure on the measurable space $(H, \mathcal{F}(H))$, where $\mathcal{F}(H)$ is a σ-algebra over H. Let $\mathit{Distr}(H)$ denote the set of probability measures on this measurable space.

2.1 Continuous-time Markov chains

Definition 1 (CTMC). *A (labeled)* continuous-time Markov chain *(*CTMC*) is a tuple* $\mathcal{C} = (S, \text{AP}, L, \alpha, \mathbf{P}, E)$ *where* S *is a* finite *set of* states*;* AP *is a finite set of* atomic propositions*;* $L : S \to 2^{\text{AP}}$ *is the* labeling function*;* $\alpha \in \mathit{Distr}(S)$ *is the* initial distribution*;* $\mathbf{P} : S \times S \to [0,1]$ *is a stochastic* transition probability matrix*; and* $E : S \to \mathbb{R}_{\geqslant 0}$ *is the* exit rate function.

The probability to exit state s as well as to take the transition $s \to s'$ in t time units is $\int_0^t E(s) \cdot e^{-E(s)\tau} d\tau$ and $\mathbf{P}(s,s') \cdot \int_0^t E(s) \cdot e^{-E(s)\tau} d\tau$, respectively. A state s is *absorbing* if $\mathbf{P}(s,s) = 1$. The *embedded discrete-time Markov chain* (DTMC) of CTMC $\mathcal{C}$ is obtained by deleting the exit rate function E, i.e., $\mathit{emb}(\mathcal{C}) = (S, \text{AP}, L, \alpha, \mathbf{P})$.

Definition 2 (Timed paths). *Let* $\mathcal{C}$ *be a* CTMC. $\mathit{Paths}_n^{\mathcal{C}} := S \times (\mathbb{R}_{>0} \times S)^n$ *is the set of paths of length* n *in* $\mathcal{C}$*; the set of finite paths in* $\mathcal{C}$ *is defined by* $\mathit{Paths}_\star^{\mathcal{C}} = \bigcup_{n \in \mathbb{N}} \mathit{Paths}_n^{\mathcal{C}}$ *and* $\mathit{Paths}_\omega^{\mathcal{C}} := (S \times \mathbb{R}_{>0})^\omega$ *is the set of infinite paths in* $\mathcal{C}$. $\mathit{Paths}^{\mathcal{C}} = \mathit{Paths}_\star^{\mathcal{C}} \cup \mathit{Paths}_\omega^{\mathcal{C}}$ *denotes the set of all paths in* $\mathcal{C}$.

We denote a path $\rho \in \mathit{Paths}^{\mathcal{C}}(s_0)$ ($\rho \in \mathit{Paths}(s_0)$ for short) as the sequence $\rho = s_0 \xrightarrow{t_0} s_1 \xrightarrow{t_1} s_2 \cdots$ starting in state s_0 such that for $n \leqslant |\rho|$ ($|\rho|$ is the number of transitions in ρ if ρ is finite); $\rho[n] := s_n$ is the n-th state of ρ and $\rho\langle n \rangle := t_n$ is the time spent in state s_n. Let $\rho@t$ be the state occupied in ρ at time $t \in \mathbb{R}_{\geqslant 0}$, i.e. $\rho@t := \rho[n]$

where n is the smallest index such that $\sum_{i=0}^{n} \rho\langle i\rangle > t$. We assume w.l.o.g. that the time to stay in any state is strictly greater than 0.

The definition of a Borel space on paths through CTMCs follows [Var85,BHHK03]. A CTMC $\mathcal{C}$ with initial state s_0 yields a probability measure $\Pr^{\mathcal{C}}$ on paths as follows: Let $s_0, \cdots, s_k \in S$ with $\mathbf{P}(s_i, s_{i+1}) > 0$ for $0 \leqslant i < k$ and $I_0, \cdots, I_{k-1}$ nonempty intervals in $\mathbb{R}_{\geqslant 0}$, $C(s_0, I_0, \cdots, I_{k-1}, s_k)$ denotes the *cylinder set* consisting of all paths $\rho \in \mathit{Paths}(s_0)$ such that $\rho[i] = s_i$ ($i \leqslant k$), and $\rho\langle i\rangle \in I_i$ ($i < k$). $\mathcal{F}(\mathit{Paths}(s_0))$ is the smallest σ-algebra on $\mathit{Paths}(s_0)$ which contains all sets $C(s_0, I_0, \cdots, I_{k-1}, s_k)$ for all state sequences $(s_0, \cdots, s_k) \in S^{k+1}$ with $\mathbf{P}(s_i, s_{i+1}) > 0$ ($0 \leqslant i < k$) and $I_0, \cdots, I_{k-1}$ range over all sequences of nonempty intervals in $\mathbb{R}_{\geqslant 0}$. The probability measure $\Pr^{\mathcal{C}}$ on $\mathcal{F}(\mathit{Paths}(s_0))$ is the unique measure defined by induction on k by $\Pr^{\mathcal{C}}(C(s_0)) = \alpha(s_0)$ and for $k > 0$:

$$\Pr^{\mathcal{C}}\big(C(s_0, I_0, \cdots, I_{k-1}, s_k)\big) = \Pr^{\mathcal{C}}\big(C(s_0, I_0, \cdots, I_{k-2}, s_{k-1})\big) \cdot \int_{I_{k-1}} \mathbf{P}(s_{k-1}, s_k) E(s_{k-1}) \cdot e^{-E(s_{k-1})\tau} d\tau. \tag{1}$$

Example 1. An example CTMC is illustrated in Fig. 4(b) (page 13), where AP $= \{a, b, c\}$ and s_0 is the initial state, i.e., $\alpha(s_0) = 1$ and $\alpha(s) = 0$ for any $s \neq s_0$. The exit rates and transition probabilities are as shown.

2.2 Deterministic timed automata

Let $\mathcal{X} = \{x_1, \ldots, x_n\}$ be a set of variables in $\mathbb{R}$. An $\mathcal{X}$-valuation is a function $\eta : \mathcal{X} \to \mathbb{R}$ assigning to each variable x a value $\eta(x)$. Let $\mathcal{V}(\mathcal{X})$ denote the set of all valuations over $\mathcal{X}$. A *constraint* over $\mathcal{X}$, denoted by g, is a subset of $\mathbb{R}^n$. Let $\mathcal{B}(\mathcal{X})$ denote the set of constraints over $\mathcal{X}$. An $\mathcal{X}$-valuation η *satisfies* constraint g, denoted as $\eta \models g$ if $(\eta(x_1), \ldots, \eta(x_n)) \in g$.

Occasionally we use a special case of *nonnegative* variables, called *clocks*. We write $\vec{0}$ for the valuation that assigns 0 to all clocks. For a subset $X \subseteq \mathcal{X}$, the reset of X, denoted $\eta[X := 0]$, is the valuation η' such that $\forall x \in X.\ \eta'(x) := 0$ and $\forall x \notin X.\ \eta'(x) := \eta(x)$. For $\delta \in \mathbb{R}_{\geqslant 0}$, $\eta + \delta$ is the valuation η'' such that $\forall x \in \mathcal{X}.\ \eta''(x) := \eta(x) + \delta$, which implies that all clocks proceed at the same speed, or equivalently, $\forall x_i \in \mathcal{X}.\ \dot{x}_i = 1$. A *clock constraint* on $\mathcal{X}$ is an expression of the form $x \bowtie c$, or $x - y \bowtie c$, or the conjunction of any clock constraints, where $x, y \in \mathcal{X}$, $\bowtie \in \{<, \leqslant, >, \geqslant\}$ and $c \in \mathbb{N}$.

Definition 3 (DTA). *A* deterministic timed automaton *is a tuple* $\mathcal{A} = (\Sigma, \mathcal{X}, Q, q_0, Q_{\mathbf{F}}, \to)$ *where*

- Σ *is a finite* alphabet*;*
- $\mathcal{X}$ *is a finite set of* clocks*;*
- Q *is a nonempty finite set of* locations*;*
- $q_0 \in Q$ *is the* initial location*;*
- $\to \in Q \times \Sigma \times \mathcal{B}(\mathcal{X}) \times 2^{\mathcal{X}} \times Q$ *is an* edge relation *satisfying:* $q \xrightarrow{a,g,X} q'$ *and* $q \xrightarrow{a,g',X'} q''$ *with* $g \neq g'$ *implies* $g \cap g' = \varnothing$*; and*
- $Q_{\mathbf{F}}$ *is the* Y acceptance condition*, where*
  - ▶ *if* $Y=$reachability*, then* $Q_{\mathbf{F}} := Q_F \subseteq Q$ *is a set of* accepting locations*;*
  - ▶ *if* $Y=$Muller*, then* $Q_{\mathbf{F}} := Q_{\mathcal{F}} \subseteq 2^Q$ *is the* acceptance family*.*

We refer to $q \xrightarrow{a,g,X} q'$ as an *edge*, where $a \in \Sigma$ is the input symbol, the *guard* g is a clock constraint on the clocks of $\mathcal{A}$, $X \subseteq \mathcal{X}$ is a set of clocks to be reset and q' is the successor location. The intuition is that the DTA $\mathcal{A}$ can move from location q to location q' when the input symbol is a and the guard g holds, while the clocks in X should be reset when entering q'. Note that we don't consider diagonal constraints like $x - y \bowtie c$ in DTA. However, it is known that this does not harm the expressiveness of a TA [BPDG98].

We will denote $\mathrm{DTA}^{\diamond}$ and DTA^{ω} for the DTA with reachability and Muller acceptance conditions, respectively; while with DTA we denote the general case covering both $\mathrm{DTA}^{\diamond}$ and DTA^{ω}. As a convention, we assume each location $q \in Q_F$ in $\mathrm{DTA}^{\diamond}$ is a sink.

An (infinite) *timed path* in $\mathcal{A}$ is of the form $\theta = q_0 \xrightarrow{a_0, t_0} q_1 \xrightarrow{a_1, t_1} \cdots$, satisfying that $\eta_0 = \vec{0}$, and for all $j \geqslant 0$, it holds that $t_j > 0$, $\eta_j + t_j \models g_j$ and $\eta_{j+1} = (\eta_j + t_j)[X_j := 0]$, where η_j is the clock evaluation on *entering* q_j. Let $\mathit{inf}(\theta)$ denote the set of states $q \in Q$ such that $q = q_i$ for infinitely many $i \geqslant 0$. Furthermore, all the definitions on paths in CTMCs can be adapted.

Definition 4 (DTA accepting paths). *An infinite path θ is* accepted *by a* $\mathrm{DTA}^{\diamond}$ *if there exists some $i \geqslant 0$ such that $\theta[i] \in Q_F$; θ is accepted by a* DTA^{ω} *if $\mathit{inf}(\theta) \in Q_{\mathcal{F}}$.*

Example 2 ($\mathrm{DTA}^{\diamond}$ *and* DTA^{ω}*).* An example $\mathrm{DTA}^{\diamond}$ is shown in Fig. 4(c) (page 13) over the alphabet $\{a, b\}$. The reachability acceptance condition is characterized by the accepting location set $Q_F = \{q_1\}$. The unique initial location is q_0 and the guards $x < 1$ and $1 < x < 2$ are disjoint on the edges labeled with a and emanating from q_0. This guarantees the determinism.

We then consider the DTA^{ω} in Fig. 1 over $\Sigma = \{a, b, c\}$. The unique initial location is q_0 and the Muller acceptance family is $Q_{\mathcal{F}} = \big\{\{q_0, q_2\}\big\}$. Since $Q_{\mathcal{F}}$ is a singleton, we can indicate it in the figure by the double-lined states. Any accepting path should cycle between the states q_0 and q_1 for *finitely* many times, and between states q_0 and q_2 for *infinitely* many times. The determinism is guaranteed of the similar reason.

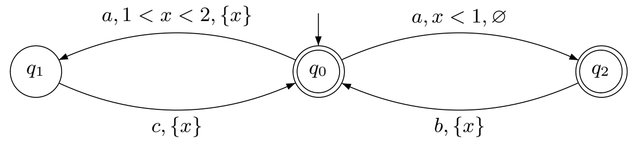


Fig. 1. DTA with Muller acceptance conditions (DTA^{ω})

Remark 1 (Muller not Büchi). According to *[AD94]*, the expressive power of *(deterministic) timed Muller automata (D)TMA* and *(deterministic) timed Büchi automata (D)TBA* has the following relation:

$$TMA = TBA > DTMA > DTBA.$$

Also notice that *DTMA* are closed under all Boolean operators (union, intersection and complement), while *DTBA* are *not* closed under complement. These two points justify our choice of *DTMA* (or DTA^{ω}) instead of *DTBA*.

Remark 2 (Successor location). Due to the determinism, we can replace the transition relation $\rightarrow\, \in Q \times \Sigma \times \mathcal{B}(\mathcal{X}) \times 2^{\mathcal{X}} \times Q$ by a function $\mathit{succ} : Q \times \Sigma \times \mathcal{B}(\mathcal{X}) \mapsto 2^{\mathcal{X}} \times Q$. Namely, given a location q, an action a and a guard g, there is at most one successor location q'. Note that the set of reset clocks is irrelevant to the successor location. Therefore, if only the successor location is of interest, then we can thus simplify the function succ to $\widetilde{\mathit{succ}} : Q \times \Sigma \times \mathcal{B}(\mathcal{X}) \mapsto Q$, i.e., $q' = \widetilde{\mathit{succ}}(q, a, g)$.

2.3 Piecewise-Deterministic Markov Processes

The model PDP was introduced by Davis in 1984 [Dav84]. We abbreviate it as PDP instead of literally PDMP, following the convention by Davis [Dav93]. A PDP constitutes a general framework that can model virtually any stochastic system without diffusions [Dav93] and for which powerful analysis and control techniques exist [LL85,LY91,CD88]. A PDP is a stochastic process of hybrid type, i.e., the stochastic process concerns both a discrete location and a continuous variable. The class of PDPs was recognized as a very wide class holding many types of stochastic hybrid system. This makes PDP a useful model for an enormous variety of applied problems in engineering, operations research, management science and economics; examples include queueing systems, stochastic scheduling, fault detection in systems engineering, etc.

Given a set H, let $\Pr : \mathcal{F}(H) \to [0,1]$ be a probability measure on the measurable space $(H, \mathcal{F}(H))$, where $\mathcal{F}(H)$ is a σ-algebra over H. Let $Distr(H)$ denote the set of probability measures on this measurable space.

Definition 5 (PDP [Dav93]). *A* piecewise-deterministic (Markov) process *is a tuple* $\mathcal{Z} = (Z, \mathcal{X}, Inv, \phi, \Lambda, \mu)$ *with:*

- Z *is a finite set of* locations*;*
- $\mathcal{X}$ *is a finite set of* variables*;*
- $Inv : Z \to \mathcal{B}(\mathcal{X})$ *is an* invariant function*;*
- $\phi : Z \times \mathcal{V}(\mathcal{X}) \times \mathbb{R} \to \mathcal{V}(\mathcal{X})$ *is a* flow function[1]*;*
- $\Lambda : \mathbb{S} \to \mathbb{R}_{\geqslant 0}$ *is an* exit rate function*;*
- $\mu : \mathring{\mathbb{S}} \cup \partial\mathbb{S} \to Distr(\mathbb{S})$ *is the* transition probability function*, where:*

$\mathbb{S} := \{\xi := (z, \eta) \mid z \in Z, \eta \models Inv(z)\}$ *is the state space of the* PDP $\mathcal{Z}$*,* $\mathring{\mathbb{S}}$ *is the interior of* $\mathbb{S}$ *and* $\partial\mathbb{S} = \bigcup_{z \in Z}\{z\} \times \partial Inv(z)$ *is the boundary of* $\mathbb{S}$ *with* $\partial Inv(z) = \overline{Inv(z)} \setminus \mathring{Inv}(z)$ *as the boundary of* $Inv(z)$*,* $\mathring{Inv}(z)$ *the interior of* $Inv(z)$ *and* $\overline{Inv(z)}$ *the closure of* $Inv(z)$*. Functions* Λ *and* μ *satisfy the following conditions:*

- $\forall \xi \in \mathbb{S}.\ \exists \epsilon(\xi) > 0.$ *function* $t \mapsto \Lambda(\xi \oplus t)$ *is integrable on* $[0, \epsilon(\xi)[$*, where* $\xi \oplus t = \big(z, \phi(z, \eta, t)\big)$*, for* $\xi = (z, \eta)$*;*
- *Function* $\xi \mapsto \mu(\xi, A)$[2] *is measurable for any* $A \in \mathcal{F}(\mathbb{S})$*, where* $\mathcal{F}(\mathbb{S})$ *is a* σ*-algebra generated by the countable union* $\bigcup_{z \in Z}\{z\} \times A_z$ *with* A_z *being a subset of* $\mathcal{F}(Inv(z))$ *and* $\mu(\xi, \{\xi\}) = 0$*.*

We will explain the behavior of a PDP by the aid of Fig. 2. A PDP consists of a finite set of *locations* each with a *location invariant* over a set of *variables*. A *state* consists of a location and a valuation of the variables. A PDP is only allowed to stay in location z when the constraint $Inv(z)$ is satisfied. If e.g., $Inv(z)$ is $x_1^2 - 2x_2 \leqslant 1.5 \wedge x_3 > 2$, then its interior $\mathring{Inv}(z)$ is $x_1^2 - 2x_2 < 1.5 \wedge x_3 > 2$ and its closure $\overline{Inv(z)}$ is $x_1^2 - 2x_2 \leqslant 1.5 \wedge x_3 \geqslant 2$, and the boundary $\partial Inv(z)$ is $x_1^2 - 2x_2 = 1.5 \wedge x_3 = 2$. In Fig. 2, there are three locations z_0, z_1, z_2 and the gray zones are the valid valuations for respective locations. A state is a black dot. A boundary state is a white dot. When a new state $\xi = (z, \eta)$ is entered and $Inv(z)$ is valid, i.e., $\xi \in \mathbb{S}$, the PDP can (i) either *delay* to state $\xi' = (z, \eta') \in \mathbb{S} \cup \partial\mathbb{S}$ according to both the flow function ϕ and the time delay t (in this case $\xi' = \xi \oplus t$); (ii) or take a *Markovian jump* to state $\xi'' = (z'', \eta'') \in \mathbb{S}$ with probability $\mu(\xi, \{\xi''\})$. Note that the residence time of a location is exponentially distributed. When the variable valuation satisfies the boundary (i.e., $\xi \in \partial\mathbb{S}$), the PDP is *forced to take a boundary jump* and leave the current location z with probability $\mu(\xi, \{\xi''\})$ to state ξ''.

The flow function ϕ defines the time-dependent behavior in a single location, in particular, how the variable valuations change when time elapses. State $\xi \oplus t$ is the

[1] The flow function is the solution of a system of ODEs with a Lipschitz continuous vector field.

[2] $\mu(\xi, A)$ is a shorthand for $(\mu(\xi))(A)$.

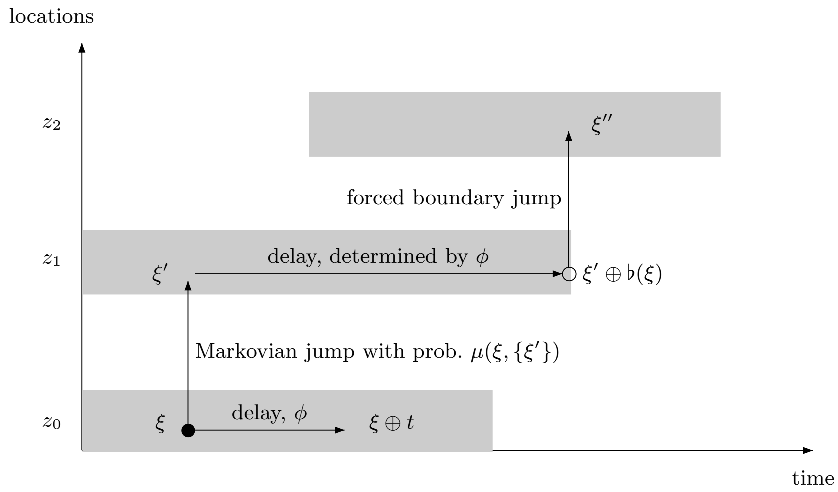


Fig. 2. The behavior of a PDP

timed successor of state ξ (on the same location) given that t time units have passed. The PDP is piecewise-deterministic because in each location (one piece) the behavior is deterministically determined by ϕ. The process is *Markovian* as the current state contains all the information to predict the future progress of the process.

The embedded *discrete-time Markov process* (DTMP) $emb(\mathcal{Z})$ of the PDP $\mathcal{Z}$ has the same state space $\mathbb{S}$ as $\mathcal{Z}$. The (one-jump) *transition probability* from a state ξ to a set $A \subseteq \mathbb{S}$ of states (on different locations as ξ), denoted $\hat{\mu}(\xi, A)$, is given by [Dav93]:

$$\hat{\mu}(\xi, A) = \int_0^{\flat(\xi)} (\mathcal{Q}\mathbf{1}_A)(\xi \oplus t) \cdot \Lambda(\xi \oplus t)\, e^{-\int_0^t \Lambda(\xi \oplus \tau) d\tau}\, dt \tag{2}$$

$$+ (\mathcal{Q}\mathbf{1}_A)(\xi \oplus \flat(\xi)) \cdot e^{-\int_0^{\flat(\xi)} \Lambda(\xi \oplus \tau) d\tau}, \tag{3}$$

where $\flat(\xi) = \inf\{t > 0 \mid \xi \oplus t \in \partial\mathbb{S}\}$ is the minimal time to hit the boundary if such time exists; $\flat(\xi) = \infty$ otherwise. $(\mathcal{Q}\mathbf{1}_A)(\xi) = \int_{\mathbb{S}} \mathbf{1}_A(\xi')\mu(\xi, d\xi')$ is the accumulative (one-jump) transition probability from ξ to A and $\mathbf{1}_A(\xi)$ is the characteristic function such that $\mathbf{1}_A(\xi) = 1$ when $\xi \in A$ and $\mathbf{1}_A(\xi) = 0$ otherwise. Term (2) specifies the probability to delay to state $\xi \oplus t$ (on the same location) and take a Markovian jump from $\xi \oplus t$ to A. Note the delay t can take a value from $[0, \flat(\xi))$. Term (3) is the probability to stay in the same location for $\flat(\xi)$ time units and then it is forced to take a boundary jump from $\xi \oplus \flat(\xi)$ to A since $Inv(z)$ is invalid.

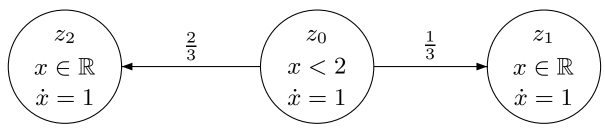


Fig. 3. An example PDP $\mathcal{Z}$

Example 3. Fig. 3 depicts a 3-location PDP $\mathcal{Z}$ with one variable x, where $Inv(z_0)$ is $x < 2$ and $Inv(z_1)$, $Inv(z_2)$ are both $x \in [0, \infty)$. Solving $\dot{x} = 1$ gives the flow function $\phi(z_i, \eta(x), t) = \eta(x) + t$ for $i = 0, 1, 2$. The state space of $\mathcal{Z}$ is $\{(z_0, \eta) \mid$

$0 < \eta(x) < 2\} \cup \{(z_1, \mathbb{R})\} \cup \{(z_2, \mathbb{R})\}$. Let exit rate $\Lambda(\xi) = 5$ for any $\xi \in \mathbb{S}$. For $\eta \models \mathit{Inv}(z_0)$, let $\mu\big((z_0, \eta), \{(z_1, \eta)\}\big) := \frac{1}{3}$, $\mu\big((z_0, \eta), \{(z_2, \eta)\}\big) := \frac{2}{3}$ and the boundary measure $\mu\big((z_0, 2), \{(z_1, 2)\}\big) := 1$. Given state $\xi_0 = (z_0, 0)$ and the set of states $A = (z_1, \mathbb{R})$, the time for ξ_0 to hit the boundary is $\flat(\xi_0) = 2$. Then $(\mathcal{Q}\mathbf{1}_A)(\xi_0 \oplus t) = \frac{1}{3}$ if $t < 2$, and $(\mathcal{Q}\mathbf{1}_A)(\xi_0 \oplus t) = 1$ if $t = 2$. In $\mathit{emb}(\mathcal{Z})$, the transition probability from state ξ_0 to A is:

$$\hat{\mu}(\xi_0, A) = \int_0^2 \frac{1}{3} \cdot 5 \cdot e^{-\int_0^t 5\, d\tau}\, dt + 1 \cdot e^{-\int_0^2 5\, d\tau} = \frac{1}{3} + \frac{2}{3} e^{-10}. \qquad \blacklozenge$$

3 Model checking DTA specifications

In this section, we deal with model checking linear real-time properties specified by DTA. The aim of model checking is to compute the probability of the set of paths in CTMC $\mathcal{C}$ accepted by a DTA $\mathcal{A}$. We prove that this can be reduced to computing the reachability probability in the product of $\mathcal{C}$ and $\mathcal{A}$ (Theorem 2). To simplify the notations, we assume w.l.o.g. that a CTMC has only one initial state s_0, i.e., $\alpha(s_0) = 1$, and $\alpha(s) = 0$ for $s \neq s_0$.

Definition 6 (CTMC paths accepted by a DTA). *Given a* CTMC $\mathcal{C} = (S, \mathrm{AP}, L, s_0, \mathbf{P}, E)$ *and a* DTA $\mathcal{A} = (2^{\mathrm{AP}}, \mathcal{X}, Q, q_0, Q_{\mathbf{F}}, \rightarrow)$*, a* CTMC *path* $\sigma = s_0 \xrightarrow{t_0} s_1 \xrightarrow{t_1} s_2 \cdots$ *is* accepted by $\mathcal{A}$ *if the* DTA *path*

$$q_0 \xrightarrow{L(s_0), t_0} \underbrace{\widetilde{succ}\big(q_0, L(s_0), g_0\big)}_{q_1} \xrightarrow{L(s_1), t_1} \underbrace{\widetilde{succ}\big(q_1, L(s_1), g_1\big)}_{q_2} \cdots$$

is accepted by $\mathcal{A}$*, where* $\eta_0 = \vec{0}$*,* g_i *is the unique guard (if it exists) such that* $\eta_i + t_i \models g_i$ *and* $\eta_{i+1} = (\eta_i + t_i)[X_i := 0]$*.*

The model checking problem on CTMC $\mathcal{C}$ against DTA $\mathcal{A}$ is to compute the probability of the set of paths in $\mathcal{C}$ that can be accepted by $\mathcal{A}$. Formally, let

$$\mathit{Paths}^{\mathcal{C}}(\mathcal{A}) := \{\ \rho \in \mathit{Paths}_{\mathcal{C}} \mid \rho \text{ is accepted by DTA } \mathcal{A}\ \}.$$

Prior to computing the probability of this set, we first prove its measurability:

Theorem 1. *For any* CTMC $\mathcal{C}$ *and* DTA $\mathcal{A}$*,* $\mathit{Paths}^{\mathcal{C}}(\mathcal{A})$ *is measurable.*

Proof. We first deal with the case that $\mathcal{A}$ only contains *strict* inequality. Since $\mathit{Paths}^{\mathcal{C}}(\mathcal{A})$ is a set of *finite* paths, $\mathit{Paths}^{\mathcal{C}}(\mathcal{A}) = \bigcup_{n \in \mathbb{N}} \mathit{Paths}^{\mathcal{C}}_n(\mathcal{A})$, where $\mathit{Paths}^{\mathcal{C}}_n(\mathcal{A})$ is the set of accepting paths by $\mathcal{A}$ of length n. For any path $\rho := s_0 \xrightarrow{t_0} s_1 \cdots s_{n-1} \xrightarrow{t_{n-1}} s_n \in \mathit{Paths}^{\mathcal{C}}_n(\mathcal{A})$, we can associate ρ with a path $\theta := q_0 \xrightarrow{L(s_0), t_0} q_1 \cdots q_{n-1} \xrightarrow{L(s_{n-1}), t_{n-1}} q_n$ of $\mathcal{A}$ induced by the location sequence:

$$q_0 \xrightarrow{L(s_0), g_0, X_0} q_1 \cdots q_{n-1} \xrightarrow{L(s_{n-1}), g_{n-1}, X_{n-1}} q_n,$$

such that $q_n \in Q_F$ and there exist $\{\eta_i\}_{1 \leqslant i < n}$ with 1) $\eta_0 = \vec{0}$; 2) $(\eta_i + t_i) \models g_i$; and 3) $\eta_{i+1} = (\eta_i + t_i)[X_i := 0]$, where η_i is the clock valuation on entering q_i.

To prove the measurability of $\mathit{Paths}^{\mathcal{C}}_n(\mathcal{A})$, it suffices to show that for each path $\rho := s_0 \xrightarrow{t_0} \cdots \xrightarrow{t_{n-1}} s_n \in \mathit{Paths}^{\mathcal{C}}_n(\mathcal{A})$, there exists a cylinder set $C(s_0, I_0, \ldots, I_{n-1}, s_n)$ (C_ρ for short) that contains ρ and that each path in C_ρ is accepted by $\mathcal{A}$. The interval I_i is constructed according to t_i as $I_i = [t_i^-, t_i^+]$ such that

– If $t_i \in \mathbb{Q}$, then $t_i^- = t_i^+ := t_i$;

– else if $t_i \in \mathbb{R} \setminus \mathbb{Q}$, then let $t_i^-, t_i^+ \in \mathbb{Q}$ such that
  – $t_i^- \leqslant t_i \leqslant t_i^+$ and $\lfloor t_i^- \rfloor = \lfloor t_i \rfloor$ and $\lceil t_i^+ \rceil = \lceil t_i \rceil$;
  – $t_i^+ - t_i^- < \dfrac{\Delta}{2 \cdot n}$, where (with $\{\cdot\}$ denoting the fractional part) $\Delta = \min_{0 \leqslant j < n,\ x \in \mathcal{X}} \Big\{ \{\eta_j(x) + t_j\}, 1 - \{\eta_j(x) + t_j\} \ \Big|\ \{\eta_j(x) + t_j\} \neq 0 \Big\}$[3].

To show that $\rho' := s_0 \xrightarrow{t'_0} \cdots \xrightarrow{t'_{n-1}} s_n \in C_\rho$ is accepted by $\mathcal{A}$, let $\eta'_0 := \vec{0}$ and $\eta'_{i+1} := (\eta'_i + t'_i)[X_i := 0]$. We will show that $\eta'_i + t'_i \models g_i$. To this end, it suffices to observe that $\eta'_0 = \eta_0$, and for any $i > 0$ and any clock variable x,

$$\big|\eta'_i(x) - \eta_i(x)\big| \leqslant \sum_{j=0}^{i-1} \big|t'_j - t_j\big| \leqslant \sum_{j=0}^{i-1} t_j^+ - t_j^- \leqslant n \cdot (t_j^+ - t_j^-) \leqslant \frac{\Delta}{2}.$$

We claim that since DTA $\mathcal{A}$ is open, it must be the case that $\eta'_i + t'_i \models g_i$. To see this, suppose g_i is of the form $x > K$ for some integer K. We have that $|\eta'_i(x) - \eta_i(x)| \leqslant \frac{\Delta}{2}$ and $|t'_i - t_i| < \frac{\Delta}{2}$, therefore $|(\eta'_i(x) + t'_i) - (\eta_i(x) + t_i)| < \Delta$. Note that $\eta_i(x) + t_i > K$, and thus $\eta_i(x) + t_i - \{\eta_i(x) + t_i\} = \lceil \eta_i(x) + t_i \rceil \geq K$. Hence $\eta_i(x) + t_i - \Delta \geq K$ since $\Delta \leqslant \{\eta_i(x) + t_i\}$. It follows that $\eta'_i(x) + t'_i > K$. A similar argument applies to the case $x < K$ and can be extended to any constraint g_i. Thus, $\eta'_i + t'_i \models g_i$.

It follows that C_ρ is a cylinder set of $\mathcal{C}$ and each path in this cylinder set is accepted by $\mathcal{A}$, namely, $\rho \in C_\rho$ and $C_\rho \subseteq Paths_n^{\mathcal{C}}(\mathcal{A})$ with $|\rho| = n$. Together with the fact that $Paths_n^{\mathcal{C}}(\mathcal{A}) \subseteq \bigcup_{\rho \in Paths_n^{\mathcal{C}}(\mathcal{A})} C_\rho$, we have:

$$Paths_n^{\mathcal{C}}(\mathcal{A}) = \bigcup_{\rho \in Paths_n^{\mathcal{C}}(\mathcal{A})} C_\rho \quad \text{and} \quad Paths^{\mathcal{C}}(\mathcal{A}) = \bigcup_{n \in \mathbb{N}} \bigcup_{\rho \in Paths_n^{\mathcal{C}}(\mathcal{A})} C_\rho.$$

We note that each interval in the cylinder set C_ρ has rational bounds, thus C_ρ is measurable. It follows that $Paths^{\mathcal{C}}(\mathcal{A})$ is a union of *countably many* cylinder sets, and hence is measurable.

We then deal with $\mathcal{A}$ with equalities of the form $x = n$ for $n \in \mathbb{N}$. We show the measurability by induction on the number of equalities appearing in $\mathcal{A}$. We have shown the base case (DTA with only strict inequalities). Now suppose there exists a transition $\iota = q \xrightarrow{a,g,X} q'$ where g contains $x = n$. We first consider a DTA $\mathcal{A}_\iota$ obtained from $\mathcal{A}$ by deleting the transitions from q other than ι. We then consider three DTA $\bar{\mathcal{A}}_\iota$, $\mathcal{A}_\iota^>$ and $\mathcal{A}_\iota^<$ where $\bar{\mathcal{A}}_\iota$ is obtained from $\mathcal{A}_\iota$ by replacing $x = n$ by *true*; $\mathcal{A}_\iota^>$ is obtained from $\mathcal{A}_\iota$ by replacing $x = n$ by $x > n$ and $\mathcal{A}_\iota^<$ is obtained from $\mathcal{A}_\iota$ by replacing $x = n$ by $x < n$. It is not difficult to see that

$$Paths^{\mathcal{C}}(\mathcal{A}_\iota) = Paths^{\mathcal{C}}(\bar{\mathcal{A}}_\iota) \setminus (Paths^{\mathcal{C}}(\mathcal{A}_\iota^>) \cup Paths^{\mathcal{C}}(\mathcal{A}_\iota^<)).$$

Note that this holds since $\mathcal{A}$ is deterministic. By induction hypothesis, $Paths^{\mathcal{C}}(\bar{\mathcal{A}}_\iota)$, $Paths^{\mathcal{C}}(\mathcal{A}_\iota^>)$ and $Paths^{\mathcal{C}}(\mathcal{A}_\iota^<)$ are measurable. Hence $Paths^{\mathcal{C}}(\mathcal{A}_\iota)$ is measurable. Furthermore, we note that

$$Paths^{\mathcal{C}}(\mathcal{A}) = \bigcup_{\iota = q \xrightarrow{a,g,X} q'} Paths^{\mathcal{C}}(\mathcal{A}_\iota),$$

therefore $Paths^{\mathcal{C}}(\mathcal{A})$ is measurable as well.

For arbitrary $\mathcal{A}$ with time constraints of the form $x \bowtie n$ where $\bowtie \in \{\geq, \leq\}$, we consider two DTA $\mathcal{A}_=$ and $\mathcal{A}_{\bowtie}$. Clearly $Paths^{\mathcal{C}}(\mathcal{A}) = Paths^{\mathcal{C}}(\mathcal{A}_=) \cup Paths^{\mathcal{C}}(\mathcal{A}_{\bar{\bowtie}})$, where $\bar{\bowtie} => $ if $\bowtie = \geq$; $<$ otherwise. Clearly $Paths^{\mathcal{C}}(\mathcal{A})$ is measurable. ■

[3] Note that we are considering open timed automata. Hence for any i with $\eta_i + t_i \models g_i$, it must be the case that $\{\eta_i(x) + t_i\} \neq 0$.

3.1 Product of CTMC and DTA

As the traditional way of verifying the automata specifications, a product between the model and the automaton is built first and the (adapted) property can then be checked on the product model. Our approach is carried out in the same fashion. In this section, we focus on building the product (and some more transformations on the product) and in Section 4 and 5, the probability measure $Prob^{\mathcal{C}}(\mathcal{A})$ will be computed.

We will first exploit the product of a CTMC and a DTA, which is what we call a *deterministic Markovian timed automaton*:

Definition 7 (DMTA). *A* deterministic Markovian timed automaton *is a tuple* $\mathcal{M} = (Loc, \mathcal{X}, \ell_0, Loc_{\mathbf{F}}, E, \rightsquigarrow)$*, where*

- Loc *is a finite set of* locations*;*
- $\mathcal{X}$ *is a finite set of* clocks*;*
- $\ell_0 \in Loc$ *is the* initial location*;*
- $Loc_{\mathbf{F}}$ *is the* acceptance condition *with* $Loc_{\mathbf{F}} := Loc_F \subseteq Loc$ *the reachability condition and* $Loc_{\mathbf{F}} := Loc_{\mathcal{F}} \subseteq 2^{Loc}$ *the Muller condition;*
- $E : Loc \to \mathbb{R}_{\geqslant 0}$ *is the* exit rate function*; and*
- $\rightsquigarrow \subseteq Loc \times \mathcal{B}(\mathcal{X}) \times 2^{\mathcal{X}} \times Distr(Loc)$ *is an* edge relation *satisfying* $(\ell, g, X, \zeta), (\ell, g', X', \zeta') \in \rightsquigarrow$ *with* $g \neq g'$ *implies* $g \cap g' = \varnothing$.

The set of clocks $\mathcal{X}$ and the related concepts, e.g., clock valuation, clock constraints are defined as for DTA. We refer to $\ell \overset{g,X}{\rightsquigarrow} \zeta$ for distribution $\zeta \in Distr(Loc)$ as an *edge* and refer to $\ell \xmapsto[\zeta(\ell')]{g,X} \ell'$ as a *transition* of this edge. The intuition is that when entering location ℓ, the DMTA chooses a residence time which is governed by the exponential distribution, i.e., the probability to leave ℓ within t time units is $1 - e^{-E(\ell)t}$. When it decides to jump, at most one edge, say $\ell \overset{g,X}{\rightsquigarrow} \zeta$, due to the determinism, is enabled and the probability to jump to ℓ' is given by $\zeta(\ell')$. The DMTA is *deterministic* as it has a unique initial location and disjoint guards for all edges emanating from any location. Similar as in DTAs, DMTA$^{\Diamond}$ and DMTA$^{\omega}$ are defined in an obvious way and DMTA refers to both cases.

Definition 8 (Paths in DMTAs). *Given a* DMTA $\mathcal{M}$*, an* (*infinite*) symbolic path *is of the form:*

$$\ell_0 \xmapsto[p_0]{g_0, X_0} \ell_1 \xmapsto[p_1]{g_1, X_1} \ell_2 \cdots$$

where $p_i = \zeta_i(\ell_{i+1})$ *is the transition probability of* $\ell_i \xmapsto[\zeta_i(\ell_{i+1})]{g_i, X_i} \ell_{i+1}$.

An infinite path *in* $\mathcal{M}$ (*induced from the symbolic path*) *is of the form* $\tau = \ell_0 \xrightarrow{t_0} \ell_1 \xrightarrow{t_1} \ell_2 \cdots$ *and has the property that* $\eta_0 = \vec{0}$, $(\eta_i + t_i) \models g_i$*, and* $\eta_{i+1} = (\eta_i + t_i)[X_i := 0]$ *where* $i \geqslant 0$ *and* η_i *is the clock valuation of* $\mathcal{X}$ *in* $\mathcal{M}$ *on* entering *location* ℓ_i.

The path τ *is* accepted *by a* DMTA$^{\Diamond}$ *if there exists* $n \geqslant 0$*, such that* $\tau[n] \in Loc_F$. *It is* accepted *by* DMTA$^{\omega}$ *iff* $inf(\tau) \in Loc_{\mathcal{F}}$*, i.e.,* $\exists L_F \in Loc_{\mathcal{F}}$ *such that* $inf(\tau) = L_F$.

All definitions on paths in CTMC*s can be carried over to* DMTA *paths.*

DMTA *Semantics.* First we characterize the *one-jump* probability $\ell \xmapsto[\mathbf{P}(\ell,\ell')]{g,X} \ell'$ within time interval I starting at clock valuation η, denoted $p_\eta(\ell, \ell', I)$, as follows:

$$p_\eta(\ell, \ell', I) = \int_I \underbrace{E(\ell) \cdot e^{-E(\ell)\tau}}_{\text{(i) density to leave } \ell \text{ at } \tau} \cdot \underbrace{\mathbf{1}_g(\eta + \tau)}_{\text{(ii) } \eta+\tau \models g?} \cdot \underbrace{\mathbf{P}(\ell, \ell')}_{\text{(iii) probabilistic jump}} d\tau \quad (4)$$

Actually, (i) characterizes the delay τ at location ℓ which is exponentially distributed with rate $E(\ell)$; (ii) is the *characteristic function*, where $\mathbf{1}_g(\eta+\tau)=1$, if $\eta+\tau \models g$; 0, otherwise. It compares the current valuation $\eta+\tau$ with g and rules out the paths that cannot lead to ℓ'; and (iii) indicates the probabilistic jump. Note that (i) and (iii) are features from CTMCs while (ii) is from DTA. The characteristic function is Riemann integrable as it is bounded and its support is an interval, therefore $p_\eta(\ell,\ell',I)$ is well-defined.

Based on the one-jump probability, we can now consider the probability of a set of paths. Given DMTA $\mathcal{M}$, $C(\ell_0, I_0, \ldots, I_{n-1}, \ell_n)$ is the cylinder set where $(\ell_0,\ldots,\ell_n) \in Loc^{n+1}$ and $I_i \subseteq \mathbb{R}_{\geqslant 0}$. It denotes a set of paths τ in $\mathcal{M}$ such that $\tau[i]=\ell_i$ and $\tau\langle i\rangle \in I_i$. Let $\Pr^{\mathcal{M}}_{\eta_0}(C(\ell_0, I_0, \ldots, I_{n-1}, \ell_n))$ denote the probability of $C(\ell_0, I_0, \ldots, I_{n-1}, \ell_n)$ such that the initial clock valuation in location ℓ_0 is η_0. We define
$\Pr^{\mathcal{M}}_{\eta_0}(C(\ell_0, I_0, \ldots, I_{n-1}, \ell_n)) := \mathbb{P}^{\mathcal{M}}_0(\eta_0)$, where $\mathbb{P}^{\mathcal{M}}_i(\eta)$ for $0 \leqslant i \leqslant n$ is defined as: $\mathbb{P}^{\mathcal{M}}_n(\eta)=1$ and for $0 \leqslant i < n$, we note that there exists a transition from ℓ_i to ℓ_{i+1} with $\ell_i \xmapsto[p_i]{g_i, X_i} \ell_{i+1}$ $(0 \leqslant i < n)$ and thus we define

$$\mathbb{P}^{\mathcal{M}}_i(\eta) = \int_{I_i} \underbrace{E(\ell_i)\cdot e^{-E(\ell_i)\tau}\cdot \mathbf{1}_{g_i}(\eta+\tau)\cdot p_i}_{(\star)} \cdot \underbrace{\mathbb{P}^{\mathcal{M}}_{i+1}(\eta')}_{(\star\star)} \, d\tau, \tag{5}$$

where $\eta' := (\eta+\tau)[X_i := 0]$. Intuitively, $\mathbb{P}^{\mathcal{M}}_i(\eta_i)$ is the probability of the suffix cylinder set starting from ℓ_i and η_i to ℓ_n. It is recursively computed by the product of the probability of taking a transition from ℓ_i to ℓ_{i+1} within time interval I_i (cf. $(\star)$ and (4)) and the probability of the suffix cylinder set from ℓ_{i+1} and η_{i+1} on (cf. $(\star\star)$). For the same reason as $p_\eta(\ell,\ell',I)$ was well-defined, $\mathbb{P}^{\mathcal{M}}_i(\eta)$ is well-defined.

*Example 4 (*DMTA$^\diamond$ *and* DMTA$^\omega$*).* The DMTA$^\diamond$ in Fig. 4(a) has initial location ℓ_0 with two edges, with guards $x<1$ and $1<x<2$. We use the small black dots to indicate distributions. Assume t time units elapsed. If $t<1$, then the upper edge is enabled and the probability to go to ℓ_1 within time t is $p_{\vec{0}}(\ell_0,\ell_1,t) = (1-e^{-r_0 t})\cdot 1$, where $E(\ell_0)=r_0$; no clock is reset. It is similar for $1<t<2$, except that x will be reset. $Loc_F=\{q_3\}$. It is obvious to see the determinism in this automaton. The DMTA$^\omega$ in Fig. 5(c) has Muller acceptance family $Loc_{\mathcal{F}} = \big\{\{\ell_1,\ell_2,\ell_3\},\{\ell_4,\ell_5,\ell_6\}\big\}$.

3.2 Product DMTAs

Given a CTMC $\mathcal{C}$ and a DTA $\mathcal{A}$, the product $\mathcal{C}\otimes\mathcal{A}$ is a DMTA defined by:

Definition 9 (Product of CTMC and DTA). *Let* $\mathcal{C}=(S,\mathrm{AP},L,s_0,\mathbf{P},E)$ *be a* CTMC *and* $\mathcal{A}=(2^{\mathrm{AP}},\mathcal{X},Q,q_0,Q_{\mathbf{F}},\rightarrow)$ *be a* DTA*. We define*
$\mathcal{C}\otimes\mathcal{A}=(Loc,\mathcal{X},\ell_0,Loc_{\mathbf{F}},E,\rightsquigarrow)$ *as the product* DMTA*, where*

- $Loc := S\times Q$; $\ell_0 := \langle s_0,q_0\rangle$; $E(\langle s,q\rangle) := E(s)$;
- $Loc_{\mathbf{F}} = Loc_F := S\times Q_F$, *if* $Q_{\mathbf{F}}=Q_F$; (*reachability condition*)
  $Loc_{\mathbf{F}} = Loc_{\mathcal{F}} := \bigcup_{F\in Q_{\mathcal{F}}} S\times F$, *if* $Q_{\mathbf{F}}=Q_{\mathcal{F}}$; (*Muller condition*)
- $\rightsquigarrow$ *is defined as the smallest relation defined by the rule:*

$$\frac{\mathbf{P}(s,s')>0 \;\wedge\; q \xrightarrow{L(s),g,X} q'}{\langle s,q\rangle \overset{g,X}{\rightsquigarrow} \zeta}, \text{ such that } \zeta(\langle s',q'\rangle)=\mathbf{P}(s,s').$$

Example 5 (Product DMTA$^\diamond$*).* Let CTMC $\mathcal{C}$ and DTA$^\diamond$ $\mathcal{A}$ be in Fig. 4(b) and 4(c), the product DMTA$^\diamond$ $\mathcal{C}\otimes\mathcal{A}$ is as in Fig. 4(a). Since $Q_F=\{q_1\}$ in $\mathcal{A}$, the set of accepting locations in DMTA$^\diamond$ is $Loc_F=\{\langle s_2,q_1\rangle\}=\{\ell_3\}$.

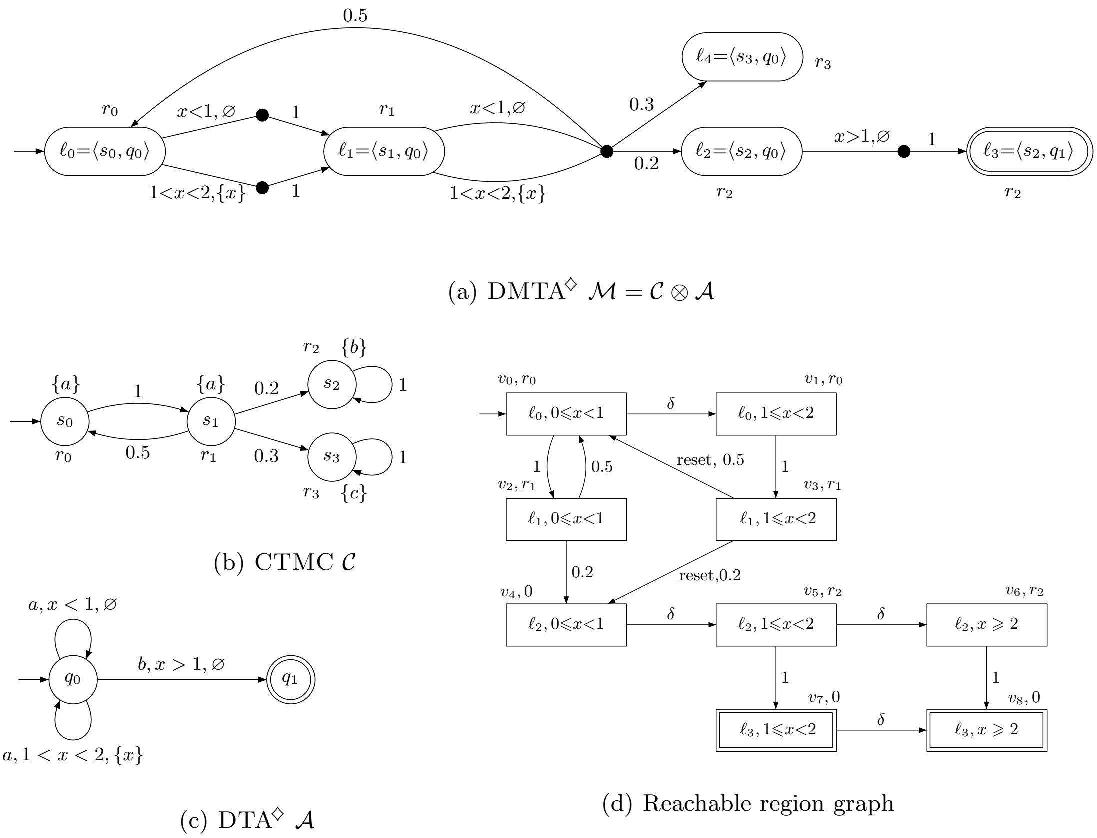


Fig. 4. Example product DMTA$^{\diamondsuit}$ of CTMC $\mathcal{C}$ and DTA$^{\diamondsuit}$ $\mathcal{A}$

Example 6 (Product DMTA$^{\omega}$*).* For the CTMC $\mathcal{C}$ in Fig. 5(a) and the DTA$^{\omega}$ $\mathcal{A}^{\omega}$ in Fig. 5(b) with acceptance family $Q_{\mathcal{F}} = \big\{\{q_1, q_2\}, \{q_3, q_4\}\big\}$, the product DMTA$^{\omega}$ $\mathcal{C} \otimes \mathcal{A}^{\omega}$ is shown in Fig. 5(c). $Loc_{\mathcal{F}} = \big\{\{\langle s_i, q_1\rangle, \langle s_j, q_2\rangle\}, \{\langle s'_i, q_3\rangle, \langle s'_j, q_4\rangle\}\big\}$, for any $s_i, s'_i, s_j, s'_j \in S$, in particular, $Loc_{\mathcal{F}} = \big\{\{\ell_1, \ell_2, \ell_3\}, \{\ell_4, \ell_5, \ell_6\}\big\}$.

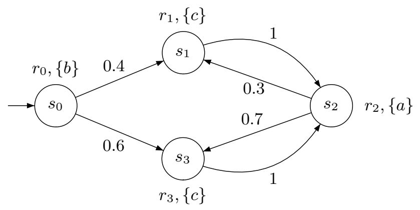
Remark 3. It is easy to see from the construction that $\mathcal{C} \otimes \mathcal{A}$ is indeed a DMTA. The determinism of the DTA $\mathcal{A}$ guarantees that the induced product is also deterministic. In $\mathcal{C} \otimes \mathcal{A}$, from each location there is at most one "action" possible, viz. $L(s)$. We can thus omit actions from the product DMTA.

For DTA$^{\diamondsuit}$ $\mathcal{A}$ with the set of accepting locations Loc_F, we denote $Paths^{\mathcal{C}\otimes\mathcal{A}}(\diamondsuit\, Loc_F) := \{\, \tau \in Paths^{\mathcal{C}\otimes\mathcal{A}} \mid \tau \text{ is accepted by } \mathcal{C}\otimes\mathcal{A} \,\}$ as the set of accepted paths in $\mathcal{C}\otimes\mathcal{A}$. Recall that $Paths^{\mathcal{C}}(\mathcal{A})$ is the set of paths in CTMC $\mathcal{C}$ that are accepted by DTA $\mathcal{A}$. For any n-ary tuple J, let $J{\downharpoonleft_i}$ denote the i-th entry in J, for $1 \leqslant i \leqslant n$. For a $(\mathcal{C}\otimes\mathcal{A})$-path $\tau = \langle s_0, q_0\rangle \xrightarrow{t_0} \langle s_1, q_1\rangle \xrightarrow{t_1} \cdots$, let $\tau{\downharpoonleft_1} := s_0 \xrightarrow{t_0} s_1 \xrightarrow{t_1} \cdots$, and for any set Π of $(\mathcal{C}\otimes\mathcal{A})$-paths, let $\Pi{\downharpoonleft_1} = \bigcup_{\tau\in\Pi} \tau{\downharpoonleft_1}$.

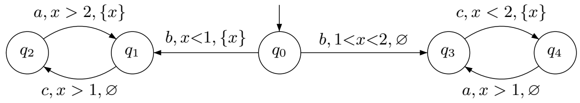
Lemma 1. *For any* CTMC $\mathcal{C}$ *and* DTA$^{\diamondsuit}$ $\mathcal{A}$, $Paths^{\mathcal{C}}(\mathcal{A}) = Paths^{\mathcal{C}\otimes\mathcal{A}}(\diamondsuit\, Loc_F){\downharpoonleft_1}$.

Proof. ($\Longrightarrow$) It is to prove that for any path $\rho \in Paths^{\mathcal{C}}(\mathcal{A})$, there exists a path $\tau \in Paths^{\mathcal{C}\otimes\mathcal{A}}(\diamondsuit\, Loc_F)$ such that $\tau{\downharpoonleft_1} = \rho$.

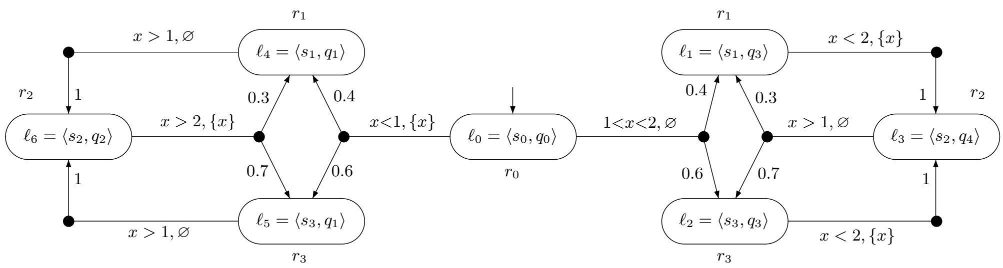
We assume w.l.o.g. that $\rho = s_0 \xrightarrow{t_0} s_1 \cdots s_{n-1} \xrightarrow{t_{n-1}} s_n \in Paths^{\mathcal{C}}$ is accepted by $\mathcal{A}$, i.e., $s_n \in Q_F$ and for $0 \leqslant i < n$, $\eta_0 \models \vec{0}$ and $\eta_i + t_i \models g_i$ and $\eta_{i+1} = (\eta_i + t_i)[X_i := 0]$,



(a) CTMC $\mathcal{C}$



(b) DTA$^\omega$ $\mathcal{A}^\omega$



(c) DMTA$^\omega$ $\mathcal{C} \otimes \mathcal{A}^\omega$

Fig. 5. Example product DMTA$^\omega$ of CTMC $\mathcal{C}$ and DTA$^\omega$ $\mathcal{A}^\omega$

where η_i is the time valuation on entering state s_i. We can then construct a path $\theta \in$ *Paths*$^{\mathcal{A}}$ from ρ such that $\theta = q_0 \xrightarrow{L(s_0),t_0} q_1 \cdots q_{n-1} \xrightarrow{L(s_{n-1}),t_{n-1}} q_n$, where s_i and q_i have the same entering clock valuation. From ρ and θ, we can construct the path

$$\tau = \langle s_0, q_0\rangle \xrightarrow{t_0} \langle s_1, q_1\rangle \cdots \langle s_{n-1}, q_{n-1}\rangle \xrightarrow{t_{n-1}} \langle s_n, q_n\rangle,$$

where $\langle s_n, q_n\rangle \in Loc_F$. It follows from the definition of an accepting path in a DTA$^\omega$ that $\tau \in Paths^{\mathcal{C}\otimes\mathcal{A}}(\Diamond\, Loc_F)$ and $\tau{\downharpoonleft_1} = \rho$.

($\Longleftarrow$) It is to prove that for any path $\tau \in Paths^{\mathcal{C}\otimes\mathcal{A}}(\Diamond\, Loc_F)$, $\tau{\downharpoonleft_1} \in Paths^{\mathcal{C}}(\mathcal{A})$.

We assume w.l.o.g. that path

$$\tau = \langle s_0, q_0\rangle \xrightarrow{t_0} \cdots \xrightarrow{t_{n-1}} \langle s_n, q_n\rangle \in Paths^{\mathcal{C}\otimes\mathcal{A}}(\Diamond\, Loc_F),$$

it holds that $\langle s_n, q_n\rangle \in Loc_F$ and for $0 \leqslant i < n$, $\eta_0 \models \vec{0}$ and $\eta_i + t_i \models g_i$ and $\eta_{i+1} = (\eta_i + t_i)[X_i := 0]$, where η_i is the time valuation on entering state $\langle s_i, q_i\rangle$. It then directly follows that $q_n \in Q_F$ and $\tau{\downharpoonleft_1} \in Paths^{\mathcal{C}}(\mathcal{A})$, given η_i the entering clock valuation of state s_i. ■

The following theorem establishes the link between CTMC $\mathcal{C}$ and DMTA$^\Diamond$ $\mathcal{C} \otimes \mathcal{A}$.

Theorem 2. *For any* CTMC $\mathcal{C}$ *and* DTA$^\Diamond$ $\mathcal{A}$*,*

$$\Pr{}^{\mathcal{C}}\left(Paths^{\mathcal{C}}(\mathcal{A})\right) = \Pr{}^{\mathcal{C}\otimes\mathcal{A}}_{\vec{0}}\left(Paths^{\mathcal{C}\otimes\mathcal{A}}(\Diamond Loc_F)\right).$$

Proof. According to Theorem 1, $Paths^{\mathcal{C}}(\mathcal{A})$ can be rewritten as the combination of cylinder sets of the form $C(s_0, I_0, \ldots, I_{n-1}, s_n)$ which are all accepted by DTA $\mathcal{A}$[4]. By Lemma 1, namely by path lifting, we can establish exactly the same combination of cylinder sets $C(\ell_0, I_0, \ldots, I_{n-1}, \ell_0)$ for $Paths^{\mathcal{C}\otimes\mathcal{A}}(\Diamond Loc_F)$, where $s_i = \ell_i{\downharpoonleft_1}$. It then

[4] Note that this means each path in the cylinder set is accepted by $\mathcal{A}$.

suffices to show that for each cylinder set $C(s_0, I_0, \ldots, I_{n-1}, s_n)$ which is accepted by $\mathcal{A}$, $\Pr^{\mathcal{C}}$ and $\Pr^{\mathcal{C}\otimes\mathcal{A}}$ yield the same probabilities. Note that a cylinder set C is accepted by a DTA $\mathcal{A}$, if each path that C generates can be accepted by $\mathcal{A}$.

For the measure $\Pr^{\mathcal{C}}$, according to Eq. 1 (page 5),

$$\Pr^{\mathcal{C}}\big(C(s_0, I_0, \ldots, I_{n-1}, s_n)\big) = \prod_{0\leqslant i<n} \int_{I_i} \mathbf{P}(s_i, s_{i+1}) \cdot E(s_i) \cdot e^{-E(s_i)\tau} d\tau.$$

For the measure $\Pr^{\mathcal{C}\otimes\mathcal{A}}_{\vec{0}}$, according to Section 3.1, it is given by $\mathbb{P}^{\mathcal{C}\otimes\mathcal{A}}_0(\vec{0})$ where $\mathbb{P}^{\mathcal{C}\otimes\mathcal{A}}_n(\eta) = 1$ for any clock valuation η and

$$\mathbb{P}^{\mathcal{C}\otimes\mathcal{A}}_i(\eta_i) = \int_{I_i} \mathbf{1}_{g_i}(\eta_i + \tau_i)\cdot p_i \cdot E(\ell_i)\cdot e^{-E(\ell_i)\tau_i} \cdot \mathbb{P}^{\mathcal{C}\otimes\mathcal{A}}_{i+1}(\eta_{i+1})\ d\tau_i,$$

where $\eta_{i+1} = (\eta_i + \tau_i)[X_i := 0]$ and $\mathbf{1}_{g_i}(\eta_i + \tau_i) = 1$, if $\eta_i + \tau_i \models g_i$; 0, otherwise.

We will show, by induction, that $\mathbb{P}^{\mathcal{C}\otimes\mathcal{A}}_i(\eta_i)$ is a constant, i.e., is independent of η_i, if the cylinder set $C(\ell_0, I_0, \ldots, I_{n-1}, \ell_n)$ is accepted by $\mathcal{C}\otimes\mathcal{A}$. Firstly let us note that for $C(\ell_0, I_0, \ldots, I_{n-1}, \ell_n)$, there must exist some sequence of transitions

$$\ell_0 \xmapsto[p_0]{g_0, X_0} \ell_1 \cdots \ell_{n-1} \xmapsto[p_{n-1}]{g_{n-1}, X_{n-1}} \ell_n$$

with $\eta_0 = \vec{0}$ and $\forall t_i \in I_i$ with $0 \leqslant i < n$, $\eta_i + t_i \models g_i$ and $\eta_{i+1} := (\eta_i + t_i)[X_i := 0]$. Moreover, according to Def. 9, we have:

$$p_i = \mathbf{P}(s_i, s_{i+1}) \qquad \text{and} \qquad E(\ell_i) = E(s_i). \tag{6}$$

We apply a backward induction on n down to 0. The base case is trivial since $\mathbb{P}^{\mathcal{C}\otimes\mathcal{A}}_n(\eta) = 1$. By I.H., $\mathbb{P}^{\mathcal{C}\otimes\mathcal{A}}_{i+1}(\eta)$ is a constant. For the induction step, consider $i < n$. For any $\tau_i \in I_i$, since $\eta_i + \tau_i \models g_i$, $\mathbf{1}_{g_i}(\eta_i + \tau_i) = 1$, it follows that

$$\begin{aligned}
\mathbb{P}^{\mathcal{C}\otimes\mathcal{A}}_i(\eta_i) &= \int_{I_i} \mathbf{1}_{g_i}(\eta_i + \tau_i)\cdot p_i \cdot E(\ell_i)\cdot e^{-E(\ell_i)\tau_i} \cdot \mathbb{P}^{\mathcal{C}\otimes\mathcal{A}}_{i+1}(\eta_{i+1})\ d\tau_i \\
&\overset{\text{I.H.}}{=} \int_{I_i} p_i \cdot E(\ell_i)\cdot e^{-E(\ell_i)\tau_i} d\tau_i \cdot \mathbb{P}^{\mathcal{C}\otimes\mathcal{A}}_{i+1}(\eta_{i+1}) \\
&\overset{\text{Eq.(6)}}{=} \int_{I_i} \mathbf{P}(s_i, s_{i+1})\cdot E(s_i)\cdot e^{-E(s_i)\tau_i} d\tau_i \cdot \mathbb{P}^{\mathcal{C}\otimes\mathcal{A}}_{i+1}(\eta_{i+1}).
\end{aligned}$$

Clearly, this is a constant. It is thus easy to see that

$$\Pr^{\mathcal{C}\otimes\mathcal{A}}_{\vec{0}}\big(C(\ell_0, I_0, \ldots, I_{n-1}, \ell_n)\big) \ := \ \mathbb{P}^{\mathcal{C}\otimes\mathcal{A}}_0(\vec{0}) = \prod_{0\leqslant i<n} \int_{I_i} \mathbf{P}(s_i, s_{i+1})\cdot E(s_i)\cdot e^{-E(s_i)\tau} d\tau,$$

which completes the proof. ■

3.3 Region Construction for DMTA

In the remainder of this section, we focus on how to compute the probability measure $\Pr^{\mathcal{C}\otimes\mathcal{A}}_{\vec{0}}\Big(Paths^{\mathcal{C}\otimes\mathcal{A}}(\Diamond\, Loc_F)\Big)$ in an effective way. Since the state space $\big\{(\ell, \eta) \mid \ell \in Loc, \eta \in \mathcal{V}(\mathcal{X})\big\}$ of $\mathcal{C}\otimes\mathcal{A}$ is uncountable, we start with adopting the standard *region construction* [AD94] to DMTA$^{\Diamond}$ to discretize the state space into a finite one. As we will see in Section 3.4, this allows us to obtain a piecewise-deterministic Markov process from a DMTA$^{\Diamond}$ in a natural way.

As usual, a region is a constraint. For regions $\Theta, \Theta' \in \mathcal{B}(\mathcal{X})$, Θ' is the *successor region* of Θ if for all $\eta \models \Theta$ there exists $\delta \in \mathbb{R}_{>0}$ such that $\eta + \delta \models \Theta'$ and for all $\delta' < \delta$, $\eta + \delta' \models \Theta \vee \Theta'$. A region Θ *satisfies* a guard g (denoted $\Theta \models g$) iff $\forall \eta \models \Theta.\ \eta \models g$. A *reset operation* on region Θ is defined as $\Theta[X := 0] := \big\{\eta[X := 0] \mid \eta \models \Theta\big\}$.

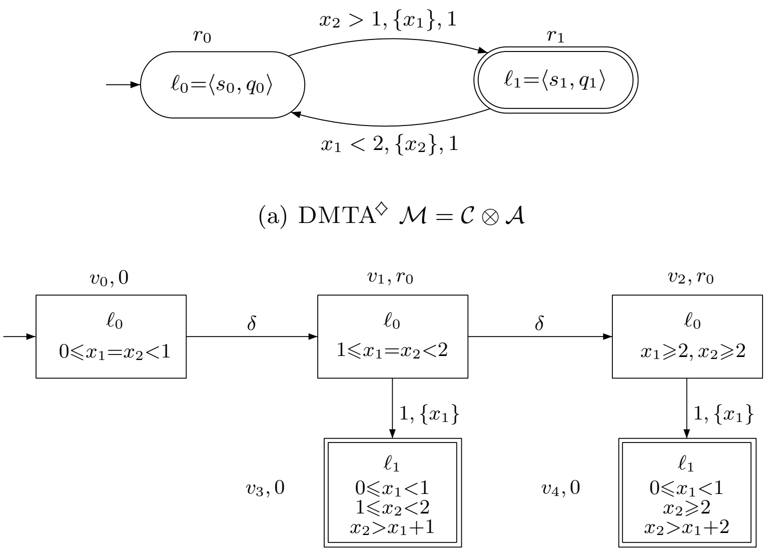
Definition 10 (Region graph of DMTA$^\diamond$). *Given* DMTA$^\diamond$ $\mathcal{M} = (Loc, \mathcal{X}, \ell_0, Loc_F, E, \rightsquigarrow)$, *the region graph is* $\mathcal{G}(\mathcal{M}) = (V, v_0, V_F, \Lambda, \hookrightarrow)$, *where*

- $V := Loc \times \mathcal{B}(\mathcal{X})$ *is a finite set of* vertices, *consisting of a location* ℓ *in* $\mathcal{M}$ *and a region* Θ*;*
- $v_0 \in V$ *is the* initial vertex *if* $(\ell_0, \vec{0}) \in v_0$*;*
- $V_F := \{v \mid v{\downharpoonleft_1} \in Loc_F\}$ *is the set of* accepting vertices*;*
- $\hookrightarrow \subseteq V \times \big(([0,1] \times 2^{\mathcal{X}}) \cup \{\delta\}\big) \times V$ *is the* transition (edge) relation*, such that:*
  - ▶ $v \overset{\delta}{\hookrightarrow} v'$ *is a* delay transition *if* $v{\downharpoonleft_1} = v'{\downharpoonleft_1}$ *and* $v'{\downharpoonleft_2}$ *is a successor region of* $v{\downharpoonleft_2}$*;*
  - ▶ $v \overset{p,X}{\hookrightarrow} v'$ *is a* Markovian transition *if there exists some transition* $v{\downharpoonleft_1} \overset{g,X}{\underset{p}{\longmapsto}} v'{\downharpoonleft_1}$ *in* $\mathcal{M}$ *such that* $v{\downharpoonleft_2} \models g$ *and* $v{\downharpoonleft_2}[X := 0] \models v'{\downharpoonleft_2}$*; and*
- $\Lambda : V \to \mathbb{R}_{\geqslant 0}$ *is the* exit rate function *where* $\Lambda(v) := E(v{\downharpoonleft_1})$ *if there exists a Markovian transition from* v*,* $\Lambda(v) := 0$ *otherwise.*

Note that in the obtained region graph, Markovian transitions emanating from any boundary region do *not* contribute to the reachability probability as the time to hit the boundary is always zero (i.e., $\flat(v, \eta) = 0$ in (8), page 18). Therefore, we can remove all the Markovian transitions emanating from boundary regions and then collapse each of them with its unique *non-boundary* (direct) successor. In the sequel, by slightly abusing the notation we still denote this *collapsed* region graph as $\mathcal{G}(\mathcal{M})$.

Remark 4 (Exit rates). The exit rate $\Lambda(v)$ is set to 0 if there is only delay transition from v. The probability to take the delay transition within time t is $e^{-\Lambda(v)t} = 1$ and the probability to take Markovian transitions is 0.

Example 7. For the DMTA$^\diamond$ $\mathcal{C}\otimes\mathcal{A}$ in Fig. 6(a), the reachable part (forward reachable from the initial vertex and backward reachable from the accepting vertices) of the collapsed region graph $\mathcal{G}(\mathcal{C}\otimes\mathcal{A})$ is shown in Fig. 6(b). The accepting vertices are sinks.



(a) DMTA$^\diamond$ $\mathcal{M} = \mathcal{C} \otimes \mathcal{A}$

(b) Reachable region graph $\mathcal{G}(\mathcal{C} \otimes \mathcal{A})$

Fig. 6. Example of a region graph

Notice that DMTA$^\diamond$ and DMTA$^\omega$ have the same locations and edge relations. The only difference is their acceptance condition. This guarantees that their obtained region graphs are the same except for the definition and interpretation of the final set V_F. We will present how V_F is derived in the region graph for DMTA$^\omega$ in Section 5.

3.4 From Region Graph to PDP

We can now define the underlying PDP of a DMTA$^\Diamond$ by using the region graph $\mathcal{G}(\mathcal{M})$. Actually, a region graph is a PDP.

Definition 11 (PDP for DMTA$^\Diamond$). *For* DMTA$^\Diamond$ $\mathcal{M} = (Loc, \mathcal{X}, \ell_0, Loc_F, E, \rightsquigarrow)$ *and region graph* $\mathcal{G}(\mathcal{M}) = (V, v_0, V_F, \Lambda, \hookrightarrow)$*, let* PDP $\mathcal{Z}(\mathcal{M}) = (V, \mathcal{X}, Inv, \phi, \Lambda, \mu)$ *where for any* $v \in V$*,*

- $Inv(v) := v\lfloor_2$ *and the state space* $\mathbb{S} := \big\{(v, \eta) \mid v \in V, \eta \in Inv(v)\big\}$*;*
- $\phi(v, \eta, t) := \eta + t$ *for* $\eta \models Inv(v)$*;*
- $\Lambda(v, \eta) := \Lambda(v)$ *is the exit rate of state* (v, η)*;*
- [boundary jump] *for each delay transition* $v \overset{\delta}{\hookrightarrow} v'$ *in* $\mathcal{G}(\mathcal{M})$ *we have* $\mu(\xi, \{\xi'\}) := 1$*, where* $\xi = (v, \eta)$*,* $\xi' = (v', \eta)$ *and* $\eta \models \partial Inv(v)$*;*
- [Markovian jump] *for each Markovian transition* $v \overset{p,X}{\hookrightarrow} v'$ *in* $\mathcal{G}(\mathcal{M})$ *we have* $\mu(\xi, \{\xi'\}) := p$*, where* $\xi = (v, \eta)$*,* $\eta \models Inv(v)$ *and* $\xi' = (v', \eta[X := 0])$*.*

From now on we write $\Lambda(v)$ instead of $\Lambda(v, \eta)$ as they coincide.

4 Model Checking DTA$^\Diamond$ Specifications

With the model and problem transformation presented in the last section, we are now ready to model check CTMC against DTA$^\Diamond$ specifications. We first consider the general case, i.e., DTA$^\Diamond$ with arbitrary number of clocks and then the special case of single clock DTA$^\Diamond$ specifications is investigated.

4.1 General DTA$^\Diamond$ Specifications

Recall that the aim of model checking is to compute the probability of the set of paths in CTMC $\mathcal{C}$ accepted by a DTA$^\Diamond$ $\mathcal{A}$. For the general case, we have proven that this is reducible to computing the reachability probability in the product $\mathcal{C} \otimes \mathcal{A}$ (Theorem 2, page 14), which can be further reduced to computing the reachability probability in a corresponding PDP (Theorem 3 below), which will be established in Section 4.1. The characterization by a system of integral equations is usually difficult to solve. Therefore we propose an approach to approximate the reachability probabilities in Section 4.1.

Characterizing Reachability Probabilities. Computing $\Pr_{\vec{0}}^{\mathcal{C}\otimes\mathcal{A}}\Big(Paths^{\mathcal{C}\otimes\mathcal{A}}(\Diamond\, Loc_F)\Big)$ is now reduced to computing the (time-unbounded) reachability probability in the PDP $\mathcal{Z}(\mathcal{C} \otimes \mathcal{A})$ — basically the region graph of $\mathcal{C} \otimes \mathcal{A}$ — given the initial state $(v_0, \vec{0})$ and the set of goal states $\{(v, \eta) \mid v \in V_F, \eta \in Inv(v)\}$ $\big((V_F, \cdot)$ for short$\big)$. Reachability probabilities of untimed events in a PDP $\mathcal{Z}$ can be computed in the embedded DTMP $emb(\mathcal{Z})$. Note that the set of locations of $\mathcal{Z}$ and $emb(\mathcal{Z})$ are equal. In the sequel, let $\mathcal{D}$ denote $emb(\mathcal{Z})$.

For each vertex $v \in V$, we define recursively $Prob^{\mathcal{D}}\big((v, \eta), (V_F, \cdot)\big)$ $\big($or shortly $Prob_v^{\mathcal{D}}(\eta)\big)$ as the probability to reach the goal states $(V_F, \cdot)$ in $\mathcal{D}$ from state (v, η).

- for the delay transition $v \overset{\delta}{\hookrightarrow} v'$,

$$Prob_{v,\delta}^{\mathcal{D}}(\eta) = e^{-\Lambda(v)\flat(v,\eta)} \cdot Prob_{v'}^{\mathcal{D}}\big(\eta + \flat(v, \eta)\big). \tag{7}$$

Recall that $\flat(v, \eta)$ is the minimal time for (v, η) to hit the boundary $\partial Inv(v)$.

– for the Markovian transition $v \overset{p,X}{\hookrightarrow} v'$,

$$Prob^{\mathcal{D}}_{v,v'}(\eta) = \int_0^{\flat(v,\eta)} p{\cdot}\Lambda(v){\cdot}e^{-\Lambda(v)\tau}{\cdot}Prob^{\mathcal{D}}_{v'}\big((\eta+\tau)[X := 0]\big)\ d\tau. \tag{8}$$

Overall, for each vertex $v \in V$, we obtain:

$$Prob^{\mathcal{D}}_{v}(\eta) = \begin{cases} Prob^{\mathcal{D}}_{v,\delta}(\eta) + \sum_{v \overset{p,X}{\hookrightarrow} v'} Prob^{\mathcal{D}}_{v,v'}(\eta), & \text{if } v \notin V_F \\ 1, & \text{otherwise} \end{cases}. \tag{9}$$

Note that here the notation η is slightly abused. It represents a vector of clock variables (see Example 8). Eq. (7) and (8) are derived based on (3) and (2), respectively. In particular, the multi-step reachability probability is computed using a sequence of one-step transition probabilities.

Hence we obtain a system of *integral equations* (9). One can read (9) either in the form $f(\xi) = \int_{Dom(\xi)} K(\xi,\xi')f(d\xi')$, where K is the kernel and $Dom(\xi)$ is the domain of integration depending on the continuous state space $\mathbb{S}$; or in the operator form $f(\xi) = (\mathcal{J}f)(\xi)$, where $\mathcal{J}$ is the integration operator. Generally, (9) does *not* necessarily have a unique solution. It turns out that the reachability probability $Prob^{\mathcal{D}}_{v_0}(\vec{0})$ coincides with the least fixpoint of the operator $\mathcal{J}$ (denoted by lfp$\mathcal{J}$) i.e., $Prob^{\mathcal{D}}_{v_0}(\vec{0}) = (\text{lfp}\mathcal{J})(v_0,\vec{0})$. Formally, we have:

Theorem 3. *For any* CTMC $\mathcal{C}$ *and* DTA$^\diamondsuit$ $\mathcal{A}$, $\Pr^{\mathcal{C}\otimes\mathcal{A}}_{\vec{0}}\big(Paths^{\mathcal{C}\otimes\mathcal{A}}(\diamondsuit\, Loc_F)\big)$ *is the least solution of* $Prob^{\mathcal{D}}_{v_0}(\cdot)$, *where* $\mathcal{D}$ *is the embedded* DTMP *of* $\mathcal{C}\otimes\mathcal{A}$.

Proof. We can express the set of all finite paths in $\mathcal{C}\otimes\mathcal{A}$ ending in some accepting location $\ell_n \in Loc_F$ for $n \in \mathbb{N}$ as the union over all location sequences i.e.,

$$\begin{aligned} \Pi^{\mathcal{C}\otimes\mathcal{A}} &= \bigcup_{n\in\mathbb{N}} \bigcup_{(\ell_0,\ldots,\ell_n)\in Loc^{n+1}} C(\ell_0, I_0, \ldots, I_{n-1}, \ell_n) \\ &= Paths^{\mathcal{C}\otimes\mathcal{A}}(\diamondsuit\, Loc_F) \cup \overline{Paths^{\mathcal{C}\otimes\mathcal{A}}(\diamondsuit\, Loc_F)}. \end{aligned}$$

where $C(\ell_0, I_0, \ldots, I_{n-1}, \ell_n)$ is a cylinder set, $I_i = [0,\infty[$ and $\overline{Paths^{\mathcal{C}\otimes\mathcal{A}}(\diamondsuit\, Loc_F)}\downharpoonright_1$ are the set of paths which are not accepted by the DTA $\mathcal{A}$. Notice that we can easily extend the measure $\Pr^{\mathcal{C}\otimes\mathcal{A}}_{\vec{0}}$ to $\Pi^{\mathcal{C}\otimes\mathcal{A}}$ such that

$$\Pr^{\mathcal{C}\otimes\mathcal{A}}_{\vec{0}}\left(\Pi^{\mathcal{C}\otimes\mathcal{A}}\right) = \Pr^{\mathcal{C}\otimes\mathcal{A}}_{\vec{0}}\big(Paths^{\mathcal{C}\otimes\mathcal{A}}(\diamondsuit\, Loc_F)\big).$$

This means that in order to prove the theorem we need to show that

$$\Pr^{\mathcal{C}\otimes\mathcal{A}}_{\vec{0}}\left(\Pi^{\mathcal{C}\otimes\mathcal{A}}\right) = Prob^{\mathcal{D}}_{v_0}(\hat{\vec{0}}), \tag{10}$$

where $Prob^{\mathcal{D}}_{v_0}(\hat{\vec{0}})$ is the short form of $Prob^{\mathcal{D}}\big((v_0,\hat{\vec{0}}),(V_F,\cdot)\big)$, i.e., the reachability probability from state $(v_0,\hat{\vec{0}})$ to $(V_F,\cdot)$. Note that for better readability, we indicate clock valuations in $\mathcal{D}$ by adding a "^".

Eq. (10) is to be shown on cylinder sets. Note that each cylinder set $C(\ell_0, I_0, \ldots, I_{n-1}, \ell_n) \subseteq \Pi^{\mathcal{C}\otimes\mathcal{A}}$ (C_n for short) induces a region graph $\mathcal{G}(C_n) = (V, v_0, V_F, \Lambda, \hookrightarrow)$, where its underlying PDP and embedded DTMP is $\mathcal{Z}(C_n)$ and $\mathcal{D}(C_n)$, respectively. To prove Eq. (10), it suffices to show that for each C_n,

$$\Pr^{\mathcal{C}\otimes\mathcal{A}}_{\vec{0}}(C_n) = Prob^{\mathcal{D}(C_n)}_{v_0}(\hat{\vec{0}}),$$

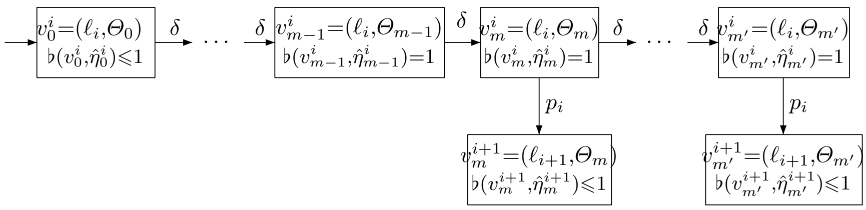
since $\Pi^{\mathcal{C}\otimes\mathcal{A}} = \bigcup_{n\in\mathbb{N}} \bigcup_{(\ell_0,\ldots,\ell_n)\in Loc^{n+1}} C_n$ and $\mathcal{D} = \bigcup_{n\in\mathbb{N}} \bigcup_{(\ell_0,\ldots,\ell_n)\in Loc^{n+1}} \mathcal{D}(C_n)$.

We will prove it by induction on the length n of the cylinder set $C_n \subseteq \Pi^{\mathcal{C}\otimes\mathcal{A}}$.

– By B.C. of $n = 0$, i.e. $C_0 = C(\ell_i)$ and $\ell_i \in Loc_F$, it holds that $\Pr^{\mathcal{C}\otimes\mathcal{A}}_{\eta_i}(C_0) = 1$; while in the embedded DTMP $\mathcal{D}(C_0)$, since the initial vertex of $\mathcal{G}(C_0)$ is $v_0 = (\ell_i, \Theta_0)$, where $\eta_i \in \Theta_0$ and v_0 is consequently the initial location of $\mathcal{Z}(C_0)$ as well as $\mathcal{D}(C_0)$ which is accepting, $Prob^{\mathcal{D}(C_0)}_{v_0}(\hat{\eta}_i) = 1$. Note $\ell_i \in Loc$ is not necessarily the initial location ℓ_0.
– By I.H., we have that for $n = k-1$, $\Pr^{\mathcal{C}\otimes\mathcal{A}}_{\eta_{i+1}}(C_{k-1}) = Prob^{\mathcal{D}(C_{k-1})}_{v_{i+1}}(\hat{\eta}_{i+1})$, where $C_{k-1} = C(\ell_{i+1}, I_{i+1}, \ldots, I_{i+k-1}, \ell_{i+k})$ and $\ell_{i+k} \in Loc_F$. Note $\ell_{i+1} \in Loc$ is not necessarily the initial location ℓ_0.
– For $n = k$, let $C_k = C(\ell_i, I_i, \ell_{i+1}, I_{i+1}, \ldots, I_{i+k-1}, \ell_{i+k})$. As a result, there exists a transition $\ell_i \xmapsto[p_i]{g_i, X_i} \ell_{i+1}$ where $\eta_i + \tau_i \models g_i$ for every $\tau_i \in]t_1, t_2[$. $t_1, t_2 \in \mathbb{Q}_{\geqslant 0} \cup \{\infty\}$ can be obtained from g_i, such that $\tau_j \in]t_1, t_2[$ iff $\eta_i + \tau_j \models g_i$. According to the semantics of MTA we have

$$\Pr^{\mathcal{C}\otimes\mathcal{A}}_{\eta_i}(C_k) = \int_{t_1}^{t_2} p_i \cdot E(\ell_i) \cdot e^{-E(\ell_i)\tau_i} \cdot \Pr^{\mathcal{C}\otimes\mathcal{A}}_{\eta_{i+1}}(C_{k-1})\, d\tau_i, \tag{11}$$

where $\eta_{i+1} = (\eta_i + \tau_i)[X_i := 0]$.



Now we deal with the inductive step for $\mathcal{D}(C_k)$. Let us assume that C_k induces the region graph $\mathcal{G}(C_k)$ whose subgraph corresponding to transition $\ell_i \xmapsto[p_i]{g_i, X_i} \ell_{i+1}$ is depicted in the figure above. For simplicity we consider that location ℓ_i induces the vertices $\{v^i_j = (\ell_i, \Theta_j) \mid 0 \leqslant j \leqslant m'\}$ and location ℓ_{i+1} induces the vertices $\{v^{i+1}_j = (\ell_{i+1}, \Theta_j) \mid m \leqslant j \leqslant m'\}$, respectively. Note that for Markovian transitions, the regions stay the same. We denote $\hat{\eta}^i_j$ (resp. $\hat{\eta}^{i+1}_j$) as the entering clock valuation on vertex v^i_j (resp. $\hat{\eta}^{i+1}_j$), for j the indices of the regions. For any $\hat{\eta} \in \bigcup_{j=0}^{m-1} \Theta_j \cup \bigcup_{j>m'} \Theta_j$, $\hat{\eta} \not\models g_i$; or more specifically,

$$t_1 = \sum_{j=0}^{m-1} \flat(v^i_j, \hat{\eta}^i_j) \qquad \text{and} \qquad t_2 = \sum_{j=0}^{m'} \flat(v^i_j, \hat{\eta}^i_j).$$

Recall that $\hat{\eta}_i$ (in the I.H.) is the clock valuation to first hit a region with ℓ_i and $\hat{\eta}_i$. Given the fact that from v^i_0 the process can only execute a delay transition before time t_1, it holds that

$$Prob^{\mathcal{D}(C_k)}_{v^i_0}(\hat{\eta}_i) = e^{-t_1 \Lambda(v_i)} \cdot Prob^{\mathcal{D}(C_k)}_{v^i_m}(\hat{\eta}^i_m)$$
$$Prob^{\mathcal{D}(C_k)}_{v^i_m}(\hat{\eta}^i_m) = Prob^{\mathcal{D}(C_k)}_{v^i_m,\delta}(\hat{\eta}^i_m) + Prob^{\mathcal{D}(C_k)}_{v^i_m, v^{i+1}_m}(\hat{\eta}_{i+1}).$$

Therefore, we get by substitution of variables:

$$\begin{aligned}
Prob^{\mathcal{D}}_{v^i_0}(\hat{\eta}_i) &= e^{-t_1\Lambda(v_i)} \cdot Prob^{\mathcal{D}(C_k)}_{v^i_m,\delta}(\hat{\eta}^i_m) + e^{-t_1\Lambda(v_i)} \cdot Prob^{\mathcal{D}(C_k)}_{v^i_m, v^{i+1}_m}(\hat{\eta}_{i+1}) \\
&= e^{-t_1\Lambda(v_i)} \cdot Prob^{\mathcal{D}(C_k)}_{v^i_m,\delta}(\hat{\eta}^i_m) \\
&\quad + e^{-t_1\Lambda(v_i)} \cdot \int_0^{\flat(v^i_m, \hat{\eta}^i_m)} p_i \Lambda(v_i) e^{-\Lambda(v_i)\tau} \cdot Prob^{\mathcal{D}(C_{k-1})}_{v^{i+1}_m}\big((\hat{\eta}^i_m + \tau)[X_i := 0]\big) d\tau \\
&= e^{-t_1\Lambda(v_i)} \cdot Prob^{\mathcal{D}(C_k)}_{v^i_m,\delta}(\hat{\eta}^i_m) \\
&\quad + \int_{t_1}^{t_1 + \flat(v^i_m, \hat{\eta}^i_m)} p_i \Lambda(v_i) e^{-\Lambda(v_i)\tau} \cdot Prob^{\mathcal{D}(C_{k-1})}_{v^{i+1}_m}\big((\hat{\eta}^i_m + \tau - t_1)[X_i := 0]\big) d\tau.
\end{aligned}$$

Evaluating each term $Prob^{\mathcal{D}(C_k)}_{v^i_m,\delta}(\hat{\eta}^i_m)$ we get the following sum of integrals:

$$Prob^{\mathcal{D}(C_k)}_{v^i_0}(\hat{\eta}_i) = \sum_{j=0}^{m'-m} \int_{t_1+\sum_{h=0}^{j-1}\flat(v^i_{m+h},\hat{\eta}^i_{m+h})}^{t_1+\sum_{h=0}^{j}\flat(v^i_{m+h},\hat{\eta}^i_{m+h})} p_i \Lambda(v_i) e^{-\Lambda(v_i)\tau}$$

$$\cdot\; Prob^{\mathcal{D}(C_{k-1})}_{v^{i+1}_{m+j}}\big((\hat{\eta}^i_{m+j}+\tau-t_1-\sum_{h=0}^{j-1}\flat(v^i_{m+h},\hat{\eta}^i_{m+h}))[X_i{:=}0]\big)d\tau.$$

Now we define the function $F^{\mathcal{D}(C_{k-1})}(t) : [t_1, t_2] \to [0,1]$, such that when $t \in [t_1 + \sum_{h=0}^{j-1}\flat(v^i_{m+h},\hat{\eta}^i_{m+h}), t_1 + \sum_{h=0}^{j}\flat(v^i_{m+h},\hat{\eta}^i_{m+h})]$ for $j \leqslant m'-m$ then $F^{\mathcal{D}(C_{k-1})}(t) = Prob^{\mathcal{D}(C_{k-1})}_{v^{i+1}_{m+j}}\big((\hat{\eta}^i_{m+j}+t-t_1-\sum_{h=0}^{j-1}\flat(v^i_{m+h},\hat{\eta}^i_{m+h}))[X_i := 0]\big)$. Using $F^{\mathcal{D}(C_{k-1})}(t)$ we can rewrite $Prob^{\mathcal{D}(C_k)}_{v^i_0}(\hat{\eta}_i)$ to an equivalent form as:

$$\begin{aligned} Prob^{\mathcal{D}(C_k)}_{v^i_0}(\hat{\eta}_i) &= \sum_{j=0}^{m'-m} \int_{t_1+\sum_{h=0}^{j-1}\flat(v^i_{m+h},\hat{\eta}^i_{m+h})}^{t_1+\sum_{h=0}^{j}\flat(v^i_{m+h},\hat{\eta}^i_{m+h})} p_i \Lambda(v_i) e^{-\Lambda(v_i)\tau} F^{\mathcal{D}(C_{k-1})}(\tau)d\tau \\ &= \int_{t_1}^{t_2} p_i \Lambda(v_i) e^{-\Lambda(v_i)\tau} F^{\mathcal{D}(C_{k-1})}(\tau)d\tau. \end{aligned}$$

By the I.H. we now have that for every $t \in [t_1 + \sum_{h=0}^{j-1}\flat(v^i_{m+h},\hat{\eta}^i_{m+h}), t_1 + \sum_{h=0}^{j}\flat(v^i_{m+h},\hat{\eta}^i_{m+h})]$ for $j \leqslant m'-m$ we have that:

$$\Pr^{\mathcal{C}\otimes\mathcal{A}}_{\eta_{i+1}}(C_{k-1}) = Prob^{\mathcal{D}(C_{k-1})}_{v^{i+1}_{m+j}}\big((\hat{\eta}^i_{m+j}+t-t_1-\sum_{h=0}^{j-1}\flat(v^i_{m+h},\hat{\eta}^i_{m+h}))[X_i{:=}0]\big) = F^{\mathcal{D}(C_{k-1})}(t),$$

where $\eta_{i+1} = (\eta_i + t)[X_i := 0]$ and $\hat{\eta}^i_{m+j} = \hat{\eta}_i + t_1 + \sum_{h=0}^{j-1}\flat(v^i_{m+h},\hat{\eta}^i_{m+h})$. This shows that $\Pr^{\mathcal{C}\otimes\mathcal{A}}_{\eta_i}(C_k) = Prob^{\mathcal{D}(C_k)}_{v_i}(\hat{\eta}_i)$ which proves the theorem. ■

Remark 5. Clock valuations η and η' in region Θ may induce different reachability probabilities. The reason is that η and η' may have different periods of time to hit the boundary, thus the probability for η and η' to either delay or take a Markovian transition may differ. This is in contrast with the traditional timed automata theory as well as probabilistic timed automata *[KNSS02]*, where η and η' are not distinguished.

Example 8. For the region graph in Fig. 6(b), the system of integral equations for v_1 in location ℓ_0 is as follows for $1 \leqslant x_1 = x_2 < 2$:

$$Prob^{\mathcal{D}}_{v_1}(x_1,x_2) = Prob^{\mathcal{D}}_{v_1,\delta}(x_1,x_2) + Prob^{\mathcal{D}}_{v_1,v_3}(x_1,x_2),$$

where

$$Prob^{\mathcal{D}}_{v_1,\delta}(x_1,x_2) = e^{-(2-x_1)r_0}\cdot Prob^{\mathcal{D}}_{v_2}(2,2)$$

and

$$Prob^{\mathcal{D}}_{v_1,v_3}(x_1,x_2) = \int_0^{2-x_1} r_0\cdot e^{-r_0\tau}\cdot Prob^{\mathcal{D}}_{v_3}(0,x_2+\tau)\ d\tau$$

where $Prob^{\mathcal{D}}_{v_3}(0,x_2+\tau) = 1$. The integral equations for v_2 can be derived similarly.

Approximating Reachability Probabilities. Finally, we discuss how to obtain a solution of (9). The integral equations (9) are *Volterra equations of the second type* [AW95]. For a general reference on solutions to Volterra equations, cf., e.g. [Cor91]. As an alternative option to solve (9), we proceed to give a general formulation of $\Pr^{\mathcal{C}}\big(Paths^{\mathcal{C}}(\mathcal{A})\big)$ using a system of *partial differential equations* (PDEs). Let the *augmented* DTA$^\diamond$ $\mathcal{A}[t_f]$ be obtained from $\mathcal{A}$ by adding a new clock variable y which is never reset and a clock constraint $y < t_f$ on all edges entering the accepting locations in Loc_F, where t_f is a finite (and usually very large) integer. The purpose of this augmentation is to ensure that the value of all clocks reaching Loc_F is bounded. It is clear that $Paths^{\mathcal{C}}(\mathcal{A}[t_f]) \subseteq Paths^{\mathcal{C}}(\mathcal{A})$. More precisely, $Paths^{\mathcal{C}}(\mathcal{A}[t_f])$ coincides with those paths which can reach the accepting states of $\mathcal{A}$ within the time bound t_f. Note that $\lim_{t_f \to \infty} \Pr^{\mathcal{C}}(Paths^{\mathcal{C}}(\mathcal{A}[t_f])) = \Pr^{\mathcal{C}}(Paths^{\mathcal{C}}(\mathcal{A}))$. We can approximate $\Pr^{\mathcal{C}}(Paths^{\mathcal{C}}(\mathcal{A}))$ by solving the PDEs with a large t_f as follows:

Proposition 1. *Given a* CTMC $\mathcal{C}$*, an augmented* DTA$^{\diamondsuit}$ $\mathcal{A}[t_f]$ *and the underlying* PDP $\mathcal{Z}(\mathcal{C} \otimes \mathcal{A}[t_f]) = (V, \mathcal{X}, Inv, \phi, \Lambda, \mu)$, $\Pr^{\mathcal{C}}\left(Paths^{\mathcal{C}}(\mathcal{A}[t_f])\right) = \hbar_{v_0}(0, \vec{0})$ *(which is the probability to reach the final states in* $\mathcal{Z}$ *starting from initial state* $(v_0, \vec{0}_{\mathcal{X}\cup\{y\}}$[5]$)$ *) is the unique solution of the following system of* PDE*s:*

$$\frac{\partial \hbar_v(y,\eta)}{\partial y} + \sum_{i=1}^{|\mathcal{X}|} \frac{\partial \hbar_v(y,\eta)}{\partial \eta^{(i)}} + \Lambda(v) \cdot \sum_{v \overset{p,X}{\hookrightarrow} v'} p \cdot (\hbar_{v'}(y, \eta[X := 0]) - \hbar_v(y,\eta)) = 0,$$

where $v \in V \setminus V_F$, $\eta \models Inv(v)$, $\eta^{(i)}$ *is the i'th clock variable and* $y \in [0, t_f)$*. For every* $\eta \models \partial Inv(v)$ *and transition* $v \overset{\delta}{\hookrightarrow} v'$*, the boundary conditions take the form:* $\hbar_v(y,\eta) = \hbar_{v'}(y,\eta)$*. For every vertex* $v \in V_F$, $\eta \models Inv(v)$ *and* $y \in [0, t_f)$*, we have the following* PDE*:*

$$\frac{\partial \hbar_v(y,\eta)}{\partial y} + \sum_{i=1}^{|\mathcal{X}|} \frac{\partial \hbar_v(y,\eta)}{\partial \eta^{(i)}} + 1 = 0.$$

The final boundary conditions are that for every vertex $v \in V$ *and* $\eta \models Inv(v) \cup \partial Inv(v)$, $\hbar_v(t_f, \eta) = 0$*.*

Proof. For any set of clocks $\mathcal{X}$ (n clocks) of the PDP $\mathcal{Z} = (Z, \mathcal{X}, Inv, \phi, \Lambda, \mu)$ we define a system of ODEs:

$$\frac{d\eta(y)}{dy} = \vec{1}, \eta(y_0) = \eta_0 \in \mathbb{R}^n_{\geqslant 0}, \tag{12}$$

which describe the evolution of clock values $\eta(y)$ at time y given the initial value η_0 of all clocks at time y_0. Notice that contrary to our DTA notation, Eq. (12) describes a system of ODEs where $\eta(y)$ is a vector of clock valuations at time y and $\frac{d\eta^{(i)}(y)}{dy}$ gives the timed evolution of clock $\eta^{(i)}$. Given a continuous differentiable functional $f : Z \times \mathbb{R}^n_{\geqslant 0} \to \mathbb{R}_{\geqslant 0}$, for every $z \in Z$ let:

$$\frac{df(z,\eta(y))}{dy} = \sum_{i=1}^{n} \frac{\partial f(z,\eta(y))}{\partial \eta^{(i)}} \cdot \frac{d\eta^{(i)}(y)}{dy} \overset{\text{Eq.(12)}}{=} \sum_{i=1}^{n} \frac{\partial f(z,\eta(y))}{\partial \eta^{(i)}}.$$

For notation simplicity we define the vector field from Eq. (13) as the operator Ξ which acts on functional $f(z,\eta(y))$ i.e., $\Xi f(z,\eta(y)) = \sum_{i=1}^{n} \frac{\partial f(z,\eta(y))}{\partial \eta^{(i)}}$. We also define the equivalent notation $\Xi f(\xi)$ for the state $\xi = (z, \eta(y))$ and any $y \in \mathbb{R}_{\geqslant 0}$.

We define the value of $\Pr^{\mathcal{C}}\left(Paths^{\mathcal{C}}(\mathcal{A})\right)$ as the expectation $\hbar(0, \xi_0)$ on PDP $\mathcal{Z}$ as follows:

$$\hbar(0,\xi_0) = \mathbb{E}\left[\int_0^{t_f} \mathbf{1}_{\mathcal{Z}}(X_\tau) d\tau \mid X_0 = \xi_0\right] = \mathbb{E}_{(0,\xi_0)}\left[\int_0^{t_f} \mathbf{1}_{\mathcal{Z}}(X_\tau) d\tau\right],$$

where the initial starting time is 0 the starting state is $\xi_0 = (z_0, \vec{0})$, X_τ is the underlying stochastic process of $\mathcal{Z}$ defined on the state space $\mathbb{S}$ and $\mathbf{1}_{\mathcal{Z}}(X_\tau) = 1$ when

[5] denoting the valuation η with $\eta(x) = 0$ for $x \in \mathcal{X} \cup \{y\}$.

$X_\tau \in \{(z, \eta(\tau)) \mid z \in V_F, \eta(\tau) \in Inv(z)\}$, $\mathbf{1}_{\mathcal{Z}}(X_\tau) = 0$, otherwise. Notice that we can also define the expectation in Eq. (13) for any starting time $y < t_f$ and state ξ as $\mathbb{E}_{(y,\xi)}\left[\int_y^{t_f} \mathbf{1}_{\mathcal{Z}}(X_\tau) d\tau\right]$.

We can obtain the expectation $\hbar(0, \xi_0)$ by following the construction in [Dav93]. For this we form the new state space $\hat{\mathbb{S}} = ([0, t_f] \times \mathbb{S}) \cup \{\Delta\}$ where Δ is the sink state and the boundary is $\partial\hat{\mathbb{S}} := ([0, t_f] \times \partial\mathbb{S}) \cup (\{t_f\} \times \mathbb{S})$. We define the following functions: $\hat{\Lambda}(y, \xi) = \Lambda(\xi)$, $\hat{\mu}((y, \xi), \{y\} \times A) = \mu(\xi, A)$ and $\hat{\mu}((t_f, \xi), \{\Delta\}) = 1$ for $y \in [0, t_f[$, $A \subseteq \mathbb{S}$ and $\xi \in \mathbb{S}$.

Given the construction we obtain an equivalent form for the expectation (13) i.e.,:

$$\hbar(0, \xi_0) = \mathbb{E}_{(0,\xi_0)}\left[\int_0^\infty \vec{\hat{1}}_{\mathcal{Z}}(\tau, X_\tau) d\tau\right], \tag{13}$$

where $\vec{\hat{1}}_{\mathcal{Z}} : \hat{\mathbb{S}} \to \{0, 1\}$, $\vec{\hat{1}}_{\mathcal{Z}}(\tau, X_\tau) = 1$ when $X_\tau \in \{(z, \eta(\tau)) \mid z \in V_F, \eta(\tau) \in Inv(z)\}$ and $\tau \in [0, t_f[$, $\vec{\hat{1}}_{\mathcal{Z}}(\tau, X_\tau) = 0$, otherwise. We also define $\vec{\hat{1}}_{\mathcal{Z}}(\Delta)$ to be zero. Notice that we introduce the sink state Δ in order to ensure that $\lim_{y\to\infty} \mathbb{E}_{(0,\xi)}\hbar(y, X_y) = 0$, which is a crucial condition in order to obtain a unique value for the expectation $\hbar(0, \xi_0)$.

For the expectation (13) [Dav93] defines the following integro-differential equations (for any $y \in [0, t_f[$):

$$\mathcal{U}\hbar(y, \xi) = \Xi\hbar(y, \xi) + \hat{\Lambda}(y, \xi) \cdot \int_{\mathbb{S}} (\hbar(y, \xi') - \hbar(y, \xi))\, \hat{\mu}((y, \xi), (y, d\xi')), \xi \in \mathbb{S} \tag{14}$$

$$\hbar(y, \xi) = \int_{\mathbb{S}} \hbar(y, \xi')\hat{\mu}((y, \xi), (y, d\xi')), \xi \in \partial\mathbb{S} \tag{15}$$

$$\mathcal{U}\hbar(y, \xi) + \vec{\hat{1}}_{\mathcal{Z}}(y, \xi) = 0, \xi \in \mathbb{S} \tag{16}$$

Equation (14) denotes the generator of the stochastic process X_y and Eq. (15) states the boundary conditions for Eq. (16). We can rewrite the integro-differential equations (14),(15) and (16) into a system of PDEs with boundary conditions given the fact that the measure $\hat{\mu}$ is not uniform. For each vertex $v \notin V_F$, $\eta \in Inv(v)$ and $y \in [0, t_f[$ of the region graph $\mathcal{G}$ we write the PDE as follows (here we define $\hbar_v(y, \eta) := \hbar(y, \xi)$ for $\xi = (v, \eta)$):

$$\frac{\partial \hbar_v(y, \eta)}{\partial y} + \sum_i \frac{\partial \hbar_v(y, \eta)}{\partial \eta^{(i)}} + \Lambda(v) \sum_{v \overset{p,X}{\hookrightarrow} v'} p \cdot (\hbar_{v'}(y, \eta[X := 0]) - \hbar_v(y, \eta)) = 0,$$

Notice that for any edge $v \overset{p,X}{\hookrightarrow} v'$ in the region graph $\mathcal{G}$, $\hat{\mu}((y, (v, \eta)), (y, (v', \eta'))) = p$. For every $\eta \in \partial Inv(v)$ and transition $v \overset{\delta}{\hookrightarrow} v'$ the boundary conditions take the form: $\hbar_v(y, \eta) = \hbar_{v'}(y, \eta)$. For every vertex $v \in V_F$, $\eta \in Inv(v)$ and $y \in [0, t_f[$ we get:

$$\frac{\partial \hbar_v(y, \eta)}{\partial y} + \sum_i \frac{\partial \hbar_v(y, \eta)}{\partial \eta^{(i)}} + 1 = 0$$

Notice that all final states are made absorbing. The final boundary conditions are that for every vertex $v \in Z$ and $\eta \in Inv(v) \cup \partial Inv(v)$, $\hbar_v(t_f, \eta)$=0. ■

4.2 Single-Clock DTA$^\diamond$ Specifications

For single-clock DTA$^\diamond$ specifications, we can simplify the system of integral equations obtained in the previous section to a system of *linear* equations where the coefficients are a solution of a system of ODEs that can be calculated efficiently.

Given a DMTA$^\diamond$ $\mathcal{M}$, we denote the set of constants appearing in the clock constraints of $\mathcal{M}$ as $\{c_0, \ldots, c_m\}$ with $c_0 = 0$. We assume the following order: $0 = c_0 < c_1 < \cdots < c_m$.

Let $\Delta c_i = c_{i+1} - c_i$ for $0 \leqslant i < m$. Note that for one clock DMTA$^\diamond$, the regions in the region graph $\mathcal{G}(\mathcal{M})$ can be represented by the following intervals: $[c_0, c_1), \ldots, [c_m, \infty)$. We partition the region graph $\mathcal{G}(\mathcal{M}) = (V, v_0, V_F, \Lambda, \hookrightarrow)$, or $\mathcal{G}$ for short, into a set of subgraphs $\mathcal{G}_i = (V_i, V_{Fi}, \Lambda_i, \{M_i, F_i, B_i\})$, where $0 \leqslant i \leqslant m$ and $\Lambda_i(v) = \Lambda(v)$, if $v \in V_i$, 0 otherwise. These subgraphs are obtained by partitioning V, V_F and $\hookrightarrow$ as follows:

- $V = \bigcup_{0 \leqslant i \leqslant m} \{V_i\}$, where $V_i = \{(\ell, \Theta) \in V \mid \Theta \subseteq [c_i, c_{i+1})\}$;
- $V_F = \bigcup_{0 \leqslant i \leqslant m} \{V_{Fi}\}$, where $v \in V_{Fi}$ iff $v \in V_i \cap V_F$;
- $\hookrightarrow = \bigcup_{0 \leqslant i \leqslant m} \{M_i \cup F_i \cup B_i\}$, where
  - M_i is the set of *Markovian transitions* (*without reset*) between vertices inside $\mathcal{G}_i$;
  - F_i is the set of *delay transitions* from the vertices in $\mathcal{G}_i$ to that in $\mathcal{G}_{i+1}$ (Forward);
  - B_i is the set of *Markovian transitions* (*with reset*) from $\mathcal{G}_i$ to $\mathcal{G}_0$ (Backward).

  It is easy to see that M_i, F_i, and B_i are pairwise disjoint.

Since the initial vertex of $\mathcal{G}_0$ is v_0 and the initial vertices of $\mathcal{G}_i$ for $0 < i \leqslant m$ are implicitly given by F_{i-1}, we omit them in the definition.

Example 9. Given the region graph in Fig. 7, the vertices are partitioned as indicated by the ovals. The M_i edges are unlabeled while the F_i and B_i edges are labeled with δ and "reset", respectively. The V_F vertices (double circles) may appear in any $\mathcal{G}_i$. Actually, if $v = (\ell, [c_i, c_{i+1})) \in V_F$, then $v' = (\ell, [c_j, c_{j+1})) \in V_F$ for $i < j \leqslant m$. This is true because $V_F = \{(\ell, \mathsf{true}) \mid \ell \in Loc_F\}$. It implies that for each final vertex not in the last region, there is a delay transition from it to the next region, see e.g. the final vertex in $\mathcal{G}_{i+1}$ in Fig. 7. The exit rate functions and the probabilities on Markovian edges are omitted in the graph.

Given a subgraph $\mathcal{G}_i$ ($0 \leqslant i \leqslant m$) of $\mathcal{G}$ with k_i states, let the probability vector $\vec{U}_i(x) = [u_i^1(x), \ldots, u_i^{k_i}(x)]^\top \in \mathbb{R}^{k_i \times 1}$ where $u_i^j(x)$ is the probability to go from vertex $v_i^j \in V_i$ to some vertex in V_F (in $\mathcal{G}$) at time x. Starting from (7)-(9), we provide a set of integral equations for $\vec{U}_i(x)$ which we later on reduce to a system of linear equations. Distinguish two cases:

Case $0 \leqslant i < m$**:** $\vec{U}_i(x)$ is given by:

$$\vec{U}_i(x) = \int_0^{\Delta c_i - x} \mathbf{M}_i(\tau) \vec{U}_i(x+\tau) d\tau + \int_0^{\Delta c_i - x} \mathbf{B}_i(\tau) d\tau \cdot \vec{U}_0(0) + \mathbf{D}_i(\Delta c_i - x) \cdot \mathbf{F}_i \vec{U}_{i+1}(0), \tag{17}$$

where $x \in [0, \Delta c_i]$ and

- $\mathbf{D}_i(x) \in \mathbb{R}^{k_i \times k_i}$ is the delay probability matrix, where for any $0 \leqslant j \leqslant k_i$, $\mathbf{D}_i(x)[j,j] = e^{-E(v_i^j)x}$ (the off-diagonal elements are zero);
- $\mathbf{M}_i(x) = \mathbf{D}_i(x) \cdot \mathbf{E}_i \cdot \mathbf{P}_i \in \mathbb{R}^{k_i \times k_i}$ is the probability density matrix for the Markovian transitions inside $\mathcal{G}_i$, where $\mathbf{P}_i$ and $\mathbf{E}_i$ are the transition probability matrix and exit rate matrix for vertices inside $\mathcal{G}_i$, respectively;
- $\mathbf{B}_i(x) \in \mathbb{R}^{k_i \times k_0}$ is the probability density matrix for the reset edges B_i, where $\mathbf{B}_i(x)[j, j']$ indicates the probability density function to take the Markovian jump with reset from the j-th vertex in $\mathcal{G}_i$ to the j'-th vertex in $\mathcal{G}_0$; and
- $\mathbf{F}_i \in \mathbb{R}^{k_i \times k_{i+1}}$ is the incidence matrix for delay edges F_i. More specifically, $\mathbf{F}_i[j, j'] = 1$ indicates that there is a delay transition from the j-th vertex in $\mathcal{G}_i$ to the j'-th vertex in $\mathcal{G}_{i+1}$; 0 otherwise.

Let us explain these equations. The third summand of (17) is obtained from (7) where $\mathbf{D}_i(\Delta c_i - x)$ indicates the probability to delay until the "end" of region i, and $\mathbf{F}_i \vec{U}_{i+1}(0)$ denotes the probability to continue in $\mathcal{G}_{i+1}$ (at relative time 0). Similarly, the first and second summands are obtained from (8); the former reflects the case where clock x is not reset, while the latter considers the reset of x (thus, implying a return to $\mathcal{G}_0$).

Case $i = m$**:** $\vec{U}_m(x)$ is simplified as follows:

$$\vec{U}_m(x) = \int_0^{\infty} \hat{\mathbf{M}}_m(\tau)\vec{U}_m(x+\tau)d\tau + \vec{1}_F + \int_0^{\infty} \mathbf{B}_m(\tau)d\tau \cdot \vec{U}_0(0) \tag{18}$$

where $\hat{\mathbf{M}}_m(\tau)[v,\cdot] = \mathbf{M}_m(\tau)[v,\cdot]$ for $v \notin V_F$, 0 otherwise. $\vec{1}_F$ is a vector such that $\vec{1}_F[v] = 1$ if $v \in V_F$, 0 otherwise. We note that $\vec{1}_F$ stems from the second clause of (9), and $\hat{\mathbf{M}}_m$ is obtained by setting the corresponding elements of $\mathbf{M}_m$ to 0. Also note that as the last subgraph $\mathcal{G}_m$ involves infinite regions, it has no delay transitions.

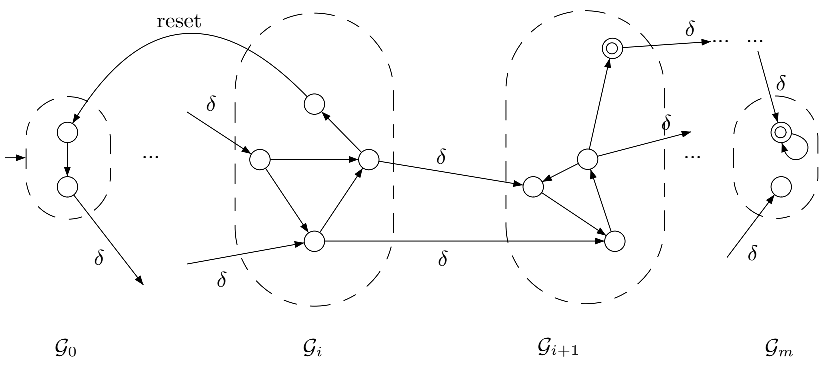


Fig. 7. Partitioning the region graph

Before solving the system of integral equations (17)-(18), we first make the following observations:

(i) Due to the fact that inside $\mathcal{G}_i$ there are only Markovian jumps with neither resets nor delay transitions, $\mathcal{G}_i$ with (V_i, Λ_i, M_i) forms a CTMC $\mathcal{C}_i$, say. For each $\mathcal{G}_i$ we define an *augmented* CTMC $\mathcal{C}_i^a$ with state space $V_i \cup V_0$, such that all V_0-vertices are made absorbing in $\mathcal{C}_i^a$. The edges connecting V_i to V_0 are kept and all the edges inside $\mathcal{C}_0$ are removed. The augmented CTMC is used to calculate the probability to start from a vertex in $\mathcal{G}_i$ and take a reset edge within a certain period of time.

(ii) Given any CTMC $\mathcal{C}$ with k states and rate matrix $\mathbf{P} \cdot \mathbf{E}$, the matrix $\mathbf{\Pi}(x)$ is given by:

$$\mathbf{\Pi}(x) = \int_0^x \mathbf{M}(\tau)\mathbf{\Pi}(x-\tau)d\tau + \mathbf{D}(x). \tag{19}$$

Intuitively, $\mathbf{\Pi}(t)[j,j']$ indicates the probability to start from vertex j and reach j' at time t.

The following proposition states the close relationship between $\mathbf{\Pi}(x)$ and the transient probability vector:

Proposition 2. *Given a* CTMC $\mathcal{C}$ *with initial distribution* α*, rate matrix* $\mathbf{P}{\cdot}\mathbf{E}$ *and* $\mathbf{\Pi}(t)$*,* $\vec{\wp}(t)$ *satisfies the following two equations:*

$$\vec{\wp}(t) = \alpha \cdot \mathbf{\Pi}(t), \tag{20}$$

$$\frac{d\vec{\wp}(t)}{dt} = \vec{\wp}(t) \cdot \mathbf{Q}, \tag{21}$$

where $\mathbf{Q} = \mathbf{P}{\cdot}\mathbf{E} - \mathbf{E}$ *is the infinitesimal generator.*

Proof. The transition probability matrix $\mathbf{\Pi}(t)$ for a CTMC $\mathcal{C}$ with state space S is denoted by the following system of integral equations:

$$\mathbf{\Pi}(t) = \int_0^t \mathbf{M}(\tau)\mathbf{\Pi}(t-\tau)d\tau + \mathbf{D}(t), \tag{22}$$

where $\mathbf{M}(\tau) = \mathbf{P}{\cdot}\mathbf{E}{\cdot}\mathbf{D}(\tau)$. Now we define for the CTMC $\mathcal{C}$ a stochastic process $X(t)$. The probability $\Pr(X(t+\Delta t) = s_j)$ to be in state s_j at time $t + \Delta t$ can be defined as:

$$\Pr(X(t+\Delta t) = s_j) = \sum_{s_i \in S} \Pr(X(t) = s_i) \cdot \Pr(X(t+\Delta t) = s_j | X(t) = s_i)$$

We can define $\Pr(X(t+\Delta t) = s_j)$ in the vector form as follows:

$$\vec{\wp}(t+\Delta t) = \vec{\wp}(t)\mathbf{\Phi}(t, t+\Delta t),$$

where $\vec{\wp}(t) = [\Pr(X(t) = s_1), \ldots, \Pr(X(t) = s_n)]$ and $\mathbf{\Phi}(t, t+\Delta t)[i,j] = \Pr(X(t+\Delta t) = s_j | X(t) = s_i)$.

As the stochastic process $X(t)$ is time-homogeneous we have that

$$\Pr(X(t+\Delta t) = s_j | X(t) = s_i) = \Pr(X(\Delta t) = s_j | X(0) = s_i),$$

which means that $\mathbf{\Phi}(t, t+\Delta t) = \mathbf{\Phi}(0, \Delta t)$. As $\Pr(X(\Delta t) = s_j | X(0) = s_i)$ denotes the transition probability to go from state s_i to state s_j at time Δt we have that $\mathbf{\Phi}(0, \Delta t) = \mathbf{\Pi}(\Delta t)$, which results in the equation:

$$\vec{\wp}(t+\Delta t) = \vec{\wp}(t)\mathbf{\Pi}(\Delta t). \tag{23}$$

Now we transform Eq. (23) as follows:

$$\begin{aligned}
& \vec{\wp}(t+\Delta t) = \vec{\wp}(t)\mathbf{\Pi}(\Delta t) \\
\Longrightarrow\ & \vec{\wp}(t+\Delta t) - \vec{\wp}(t) = \vec{\wp}(t)\mathbf{\Pi}(\Delta t) - \vec{\wp}(t) \\
\Longrightarrow\ & \vec{\wp}(t+\Delta t) - \vec{\wp}(t) = \vec{\wp}(t)(\mathbf{\Pi}(\Delta t) - \mathbf{I}) \\
\Longrightarrow\ & \frac{d\vec{\wp}(t)}{dt} = \lim_{\Delta t \to 0} \frac{\vec{\wp}(t+\Delta t) - \vec{\wp}(t)}{\Delta t} = \vec{\wp}(t) \lim_{\Delta t \to 0} \frac{\mathbf{\Pi}(\Delta t) - \mathbf{I}}{\Delta t}.
\end{aligned}$$

Now it is to compute $\lim_{\Delta t \to 0} \frac{\mathbf{\Pi}(\Delta t) - \mathbf{I}}{\Delta t}$. For this we rewrite the right hand limit as:

$$\lim_{\Delta t \to 0} \frac{1}{\Delta t} \int_0^{\Delta t} \mathbf{M}(\tau)\mathbf{\Pi}(\Delta t - \tau)d\tau + \lim_{\Delta t \to 0} \frac{1}{\Delta t} \left(\mathbf{D}(\Delta t) - \mathbf{I}\right).$$

The $\lim_{\Delta t \to 0} \frac{1}{\Delta t} \int_0^{\Delta t} \mathbf{M}(\tau)\mathbf{\Pi}(\Delta t - \tau)d\tau$ is of the type $\frac{0}{0}$, which means we have to use l'Hospital rule:

$$\begin{aligned}
\frac{d(\Delta t)}{d\Delta t} &= 1, \\
\frac{d}{d\Delta t}\left(\int_0^{\Delta t} \mathbf{M}(\tau)\mathbf{\Pi}(\Delta t - \tau)d\tau\right) &= \mathbf{M}(\Delta t)\mathbf{\Pi}(0) + \int_0^{\Delta t} \mathbf{M}(\tau)\frac{\partial}{\partial \Delta t}\mathbf{\Pi}(\Delta t - \tau)d\tau.
\end{aligned}$$

Notice that $\mathbf{\Pi}(0) = \mathbf{I}$ and we obtain:

$$\begin{aligned}
& \lim_{\Delta t \to 0} \frac{1}{\Delta t} \int_0^{\Delta t} \mathbf{M}(\tau)\mathbf{\Pi}(\Delta t - \tau)d\tau \\
= & \lim_{\Delta t \to 0} \left(\mathbf{M}(\Delta t)\mathbf{\Pi}(0) + \int_0^{\Delta t} \mathbf{M}(\tau)\frac{\partial}{\partial \Delta t}\mathbf{\Pi}(\Delta t - \tau)d\tau\right) = \mathbf{M}(0)\mathbf{\Pi}(0) = \mathbf{P}{\cdot}\mathbf{E}.
\end{aligned}$$

The $\lim_{\Delta t \to 0} \frac{1}{\Delta t} (\mathbf{D}(\Delta t) - \mathbf{I})$ is of the type $\frac{0}{0}$, which means the use of l'Hospital rule:

$$\frac{d(\Delta t)}{d\Delta t} = 1$$
$$\frac{d}{d\Delta t} (\mathbf{D}(\Delta t) - \mathbf{I}) = -\mathbf{E}\mathbf{D}(\Delta t)$$

Therefore, we obtain $\lim_{\Delta t \to 0} \frac{1}{\Delta t} (\mathbf{D}(\Delta t) - \mathbf{I}) = -\mathbf{E}$ and

$$\lim_{\Delta t \to 0} \frac{\mathbf{\Pi}(\Delta t) - \mathbf{I}}{\Delta t} = \mathbf{P}{\cdot}\mathbf{E} - \mathbf{E} = \mathbf{Q},$$

where $\mathbf{Q}$ is the infinitesimal generator of the CTMC $\mathcal{C}$. As a result we obtain:

$$\frac{d\vec{\wp}(t)}{dt} = \vec{\wp}(t) \lim_{\Delta t \to 0} \frac{\mathbf{\Pi}(\Delta t) - \mathbf{I}}{\Delta t} = \vec{\wp}(t)\mathbf{Q}.$$

Combining with Eq. (23) we get:

$$\begin{aligned} \vec{\wp}(t) =& \alpha \cdot \mathbf{\Pi}(t), \\ \frac{d\vec{\wp}(t)}{dt} =& \vec{\wp}(t) \cdot \mathbf{Q}. \end{aligned}$$

■

$\vec{\wp}(t)$ is the *transient probability vector* with $\wp_s(t)$ indicating the probability to be in state s at time t given the initial probability distribution α. Eq. (21) is the celebrated forward Chapman-Kolmogorov equations. According to this proposition, solving the integral equation $\mathbf{\Pi}(t)$ boils down to selecting the appropriate initial distribution vector α and solving the system of ODEs (21), which can be done very efficiently using *uniformization.*

Prior to exposing how to solve the system of integral equations by solving a system of *linear* equations, we define $\bar{\mathbf{\Pi}}_i^a \in \mathbb{R}^{k_i \times k_0}$ for an augmented CTMC $\mathcal{C}_i^a$ to be part of $\mathbf{\Pi}_i^a$, where $\bar{\mathbf{\Pi}}_i^a$ only keeps the probabilities starting from V_i and ending in V_0. Actually,

$$\mathbf{\Pi}_i^a(x) = \left(\begin{array}{c|c} \mathbf{\Pi}_i(x) & \bar{\mathbf{\Pi}}_i^a(x) \\ \mathbf{0} & \mathbf{I} \end{array} \right),$$

where $\mathbf{0} \in \mathbb{R}^{k_0 \times k_i}$ is the zero matrix and $\mathbf{I} \in \mathbb{R}^{k_0 \times k_0}$ is the identity matrix.

Theorem 4. *For subgraph $\mathcal{G}_i$ of $\mathcal{G}$ with k_i states, it holds for $0 \leqslant i < m$ that:*

$$\vec{U}_i(0) = \mathbf{\Pi}_i(\Delta c_i) \cdot \mathbf{F}_i \vec{U}_{i+1}(0) + \bar{\mathbf{\Pi}}_i^a(\Delta c_i) \cdot \vec{U}_0(0), \quad (24)$$

where $\mathbf{\Pi}_i(\Delta c_i)$ and $\bar{\mathbf{\Pi}}_i^a(\Delta c_i)$ are for CTMC *$\mathcal{C}_i$ and the augmented* CTMC *$\mathcal{C}_i^a$, respectively. For case $i = m$,*

$$\vec{U}_m(0) = \hat{\mathbf{P}}_i \cdot \vec{U}_m(0) + \vec{1}_F + \hat{\mathbf{B}}_m \cdot \vec{U}_0(0), \quad (25)$$

where $\hat{\mathbf{P}}_i(v, v') = \mathbf{P}_i(v, v')$ if $v \notin V_F$; 0 otherwise and $\hat{\mathbf{B}}_m = \int_0^\infty \mathbf{B}_m(\tau) d\tau$.

Proof. We first deal with the case $i < m$. If in $\mathcal{G}_i$, there exists some backward edge, namely, for some j, j', $\mathbf{B}_i(x)[j, j'] \neq 0$, then we shall consider the *augmented* CTMC $\mathcal{C}_i^a$ with $k_i^a = k_i + k_0$ states. In view of this, the augmented integral equation $\vec{U}_i^a(x)$ is defined as:

$$\vec{U}_i^a(x) = \int_0^{\Delta c_i - x} \mathbf{M}_i^a(\tau) \vec{U}_i^a(x + \tau) d\tau + \mathbf{D}_i^a(\Delta c_i - x) \cdot \mathbf{F}_i^a \hat{\vec{U}}_i(0)$$

where $\vec{U}_i^a(x) = \begin{pmatrix} \vec{U}_i(x) \\ \hline \vec{U}_i'(x) \end{pmatrix} \in \mathbb{R}^{k_i^a \times 1}$, $\vec{U}_i'(x) \in \mathbb{R}^{k_0 \times 1}$ is the vector representing reachability probability for the augmented states in $\mathcal{G}_i$, $\mathbf{F}_i^a = \left(\mathbf{F}_i' \middle| \mathbf{B}_i' \right) \in \mathbb{R}^{k_i^a \times (k_{i+1} + k_0)}$ such that

$\mathbf{F}'_i = \begin{pmatrix} \mathbf{F}_i \\ \hline \mathbf{0} \end{pmatrix} \in \mathbb{R}^{k_i^a \times k_{i+1}}$ is the incidence matrix for delay edges and $\mathbf{B}'_i = \begin{pmatrix} \mathbf{0} \\ \hline \mathbf{I} \end{pmatrix} \in \mathbb{R}^{k_i^a \times k_0}$, $\vec{\hat{U}}_i(0) = \begin{pmatrix} \vec{U}_{i+1}(0) \\ \hline \vec{U}_0(0) \end{pmatrix} \in \mathbb{R}^{(k_{i+1}+k_0)\times 1}$.

First, we prove the following equation:

$$\vec{U}_i^a(x) = \mathbf{\Pi}_i^a(\mathit{\Delta} c_i - x) \cdot \mathbf{F}_i^a \vec{\hat{U}}_i(0),$$

where

$$\mathbf{\Pi}_i^a(x) = \int_0^x \mathbf{M}_i^a(\tau)\mathbf{\Pi}_i^a(x-\tau)d\tau + \mathbf{D}_i^a(x). \tag{26}$$

We consider the iterations of the solution of the following system of integral equations: set $c_{i,x} = \mathit{\Delta} c_i - x$.

$$\begin{aligned} \vec{U}_i^{a,(0)}(x) &= \vec{0} \\ \vec{U}_i^{a,(j+1)}(x) &= \int_0^{c_{i,x}} \mathbf{M}_i^a(\tau)\vec{U}_i^{a,(j)}(x+\tau)d\tau + \mathbf{D}_i^a(c_{i,x}) \cdot \mathbf{F}_i^a \vec{\hat{U}}_i(0). \end{aligned}$$

and

$$\begin{aligned} \mathbf{\Pi}_i^{a,(0)}(c_{i,x}) &= \mathbf{0} \\ \mathbf{\Pi}_i^{a,(j+1)}(c_{i,x}) &= \int_0^{c_{i,x}} \mathbf{M}_i^a(\tau)\mathbf{\Pi}_i^{a,(j)}(c_{i,x}-\tau)d\tau + \mathbf{D}_i^a(c_{i,x}). \end{aligned}$$

By induction on j, we prove the following relation:

$$\vec{U}_i^{a,(j)}(x) = \mathbf{\Pi}_i^{a,(j)}(c_{i,x}) \cdot \mathbf{F}_i^a \vec{\hat{U}}_i(0).$$

- Base case: $\vec{U}_i^{a,(0)}(x) = \vec{0}$ and $\mathbf{\Pi}_i^{a,(0)}(c_{i,x}) = \mathbf{0}$.
- Induction hypothesis: $\vec{U}_i^{a,(j)}(x) = \mathbf{\Pi}_i^{a,(j)}(c_{i,x}) \cdot \mathbf{F}_i^a \vec{\hat{U}}_i(0)$.
- Induction step $j \to j+1$:

$$\vec{U}_i^{a,(j+1)}(x) = \int_0^{c_{i,x}} \mathbf{M}_i^a(\tau)\vec{U}_i^{a,(j)}(x+\tau)d\tau + \mathbf{D}_i^a(c_{i,x}) \cdot \mathbf{F}_i^a \vec{\hat{U}}_i(0).$$

By induction hypothesis we have

$$\begin{aligned} \vec{U}_i^{a,(j+1)}(x) &= \int_0^{c_{i,x}} \mathbf{M}_i^a(\tau)\vec{U}_i^{a,(j)}(x+\tau)d\tau + \mathbf{D}_i^a(c_{i,x}) \cdot \mathbf{F}_i^a \vec{\hat{U}}_i(0) \\ &= \int_0^{c_{i,x}} \mathbf{M}_i^a(\tau)\mathbf{\Pi}_i^{a,(j)}(c_{i,x}-\tau) \cdot \mathbf{F}_i^a \vec{\hat{U}}_i(0)d\tau + \mathbf{D}_i^a(c_{i,x}) \cdot \mathbf{F}_i^a \vec{\hat{U}}_i(0) \\ &= \left(\int_0^{c_{i,x}} \mathbf{M}_i^a(\tau)\mathbf{\Pi}_i^{a,(j)}(c_{i,x}-\tau)d\tau + \mathbf{D}_i^a(c_{i,x}) \right) \cdot \mathbf{F}_i^a \vec{\hat{U}}_i(0) \\ &= \mathbf{\Pi}_i^{a,(j+1)}(c_{i,x}) \cdot \mathbf{F}_i \vec{\hat{U}}_i(0). \end{aligned}$$

Clearly, $\mathbf{\Pi}_i^a(c_{i,x}) = \lim_{j\to\infty} \mathbf{\Pi}_i^{a,(j+1)}(c_{i,x})$ and $\vec{U}_i^a(x) = \lim_{j\to\infty} \vec{U}_i^{a,(j+1)}(x)$.

Let $x = 0$ and we obtain

$$\vec{U}_i^a(0) = \mathbf{\Pi}_i^a(c_{i,0}) \cdot \mathbf{F}_i^a \vec{\hat{U}}_i(0).$$

We can also write the above relation for $x = 0$ as:

$$\begin{aligned}
\left(\frac{\vec{U}_i(0)}{\vec{U}'_i(0)}\right) &= \mathbf{\Pi}^a_i(\varDelta c_i)\left(\mathbf{F}'_i \middle| \mathbf{B}'_i\right)\left(\frac{\vec{U}_{i+1}(0)}{\vec{U}_0(0)}\right) \\
&= \left(\begin{array}{c|c} \mathbf{\Pi}_i(\varDelta c_i) & \bar{\mathbf{\Pi}}^a_i(\varDelta c_i) \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right)\left(\begin{array}{c|c} \mathbf{F}_i & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right)\left(\frac{\vec{U}_{i+1}(0)}{\vec{U}_0(0)}\right) \\
&= \left(\begin{array}{c|c} \mathbf{\Pi}_i(\varDelta c_i)\mathbf{F}_i & \bar{\mathbf{\Pi}}^a_i(\varDelta c_i) \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right)\left(\frac{\vec{U}_{i+1}(0)}{\vec{U}_0(0)}\right) \\
&= \left(\frac{\mathbf{\Pi}_i(\varDelta c_i)\mathbf{F}_i\vec{U}_{i+1}(0) + \bar{\mathbf{\Pi}}^a_i(\varDelta c_i)\vec{U}_0(0)}{\vec{U}_0(0)}\right).
\end{aligned}$$

As a result we can represent $\vec{U}_i(0)$ in the following matrix form

$$\vec{U}_i(0) = \mathbf{\Pi}_i(\varDelta c_i)\mathbf{F}_i\vec{U}_{i+1}(0) + \bar{\mathbf{\Pi}}^a_i(\varDelta c_i)\vec{U}_0(0)$$

by noting that $\mathbf{\Pi}_i$ is formed by the first k_i rows and columns of matrix $\mathbf{\Pi}^a_i$ and $\bar{\mathbf{\Pi}}^a_i$ is formed by the first k_i rows and the last $k^a_i - k_i$ columns of $\mathbf{\Pi}^a_i$.

For $i = m$, i.e., the last graph $\mathcal{G}_m$, the region size is infinite, therefore delay transitions do not exist. The vector $\vec{U}_m(x+\tau)$ in $\int_0^\infty \hat{\mathbf{M}}_m(\tau)\vec{U}_m(x+\tau)d\tau$ does not depend on entering time x, therefore we can take it out of the integral. As a result we obtain $\int_0^\infty \hat{\mathbf{M}}_m(\tau)d\tau \cdot \vec{U}_m(0)$. More than that $\int_0^\infty \hat{\mathbf{M}}_m(\tau)d\tau$ boils down to $\hat{\mathbf{P}}_m$ and $\int_0^\infty \mathbf{B}_m(\tau)d\tau$ to $\hat{\mathbf{B}}_m$. Also we add the vector $\vec{1}_F$ to ensure that the probability to start from a state in V_F is one (see (9)). ■

Since the coefficients of the linear equations are all known, solving the system of linear equations yields $\vec{U}_0(0)$, which contains the probability $Prob_{v_0}(0)$ of reaching V_F from initial vertex v_0.

Now we explain how (24) is derived from (17). The term $\mathbf{\Pi}_i(\varDelta c_i)\cdot\mathbf{F}_i\vec{U}_{i+1}(0)$ is for the delay transitions, where $\mathbf{F}_i$ specifies how the delay transitions are connected between $\mathcal{G}_i$ and $\mathcal{G}_{i+1}$. The term $\bar{\mathbf{\Pi}}^a_i(\varDelta c_i)\cdot\vec{U}_0(0)$ is for Markovian transitions with reset. $\bar{\mathbf{\Pi}}^a_i(\varDelta c_i)$ in the augmented CTMC $\mathcal{C}^a_i$ specifies the probabilities to take first transitions inside $\mathcal{G}_i$ and then a one-step Markovian transition back to $\mathcal{G}_0$. Eq. (25) is derived from (18). Since it is the last region and time goes to infinity, the time to enter the region is irrelevant (thus set to 0). Thus $\int_0^\infty \hat{\mathbf{M}}_i(\tau)d\tau$ boils down to $\hat{\mathbf{P}}_i$. In fact, the Markovian jump probability inside $\mathcal{G}_m$ can be taken from the embedded DTMC of $\mathcal{C}_m$, which is $\hat{\mathbf{P}}_i$.

Example 10. For the single-clock DMTA$^\Diamond$ in Fig. 4(a) (page 13), we show how to compute the reachability probability $Prob((v_0, 0),(v_5,\cdot))$ on the region graph $\mathcal{G}$ (cf. Fig. 4(d)), which has been partitioned into subgraphs $\mathcal{G}_0$, $\mathcal{G}_1$ and $\mathcal{G}_2$ as in Fig. 8.

The matrices for $\mathcal{G}_0$ are given as

$$\mathbf{M}_0(x) = \begin{pmatrix} 0 & 1{\cdot}r_0{\cdot}e^{-r_0x} & 0 \\ 0.5{\cdot}r_1{\cdot}e^{-r_1x} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad \mathbf{F}_0 = \begin{pmatrix} 1&0&0&0 \\ 0&0&0&0 \\ 0&0&1&0 \end{pmatrix}$$

The matrices for $\mathcal{G}_1$ are given as

$$\mathbf{M}_1(x) = \begin{pmatrix} 0 & r_0{\cdot}e^{-r_0x} & 0 & 0 \\ 0&0&0&0 \\ 0&0&0& r_2{\cdot}e^{-r_2x} \\ 0&0&0&0 \end{pmatrix} \quad \mathbf{F}_1 = \begin{pmatrix} 0&0 \\ 0&0 \\ 1&0 \\ 0&1 \end{pmatrix} \quad \mathbf{B}_1 = \begin{pmatrix} 0&0&0 \\ 1&0&1 \\ 0&0&0 \\ 0&0&0 \end{pmatrix}$$

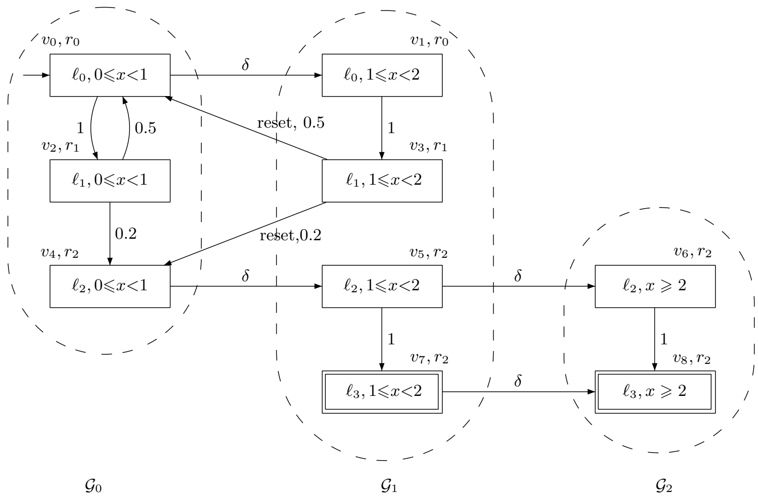


Fig. 8. Partition the region graph in Fig. 4(d)

$$\mathbf{M}_1^a(x) = \begin{pmatrix} 0 & r_0{\cdot}e^{-r_0 x} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5{\cdot}r_1{\cdot}e^{-r_1 x} & 0 & 0.2{\cdot}r_1{\cdot}e^{-r_1 x} \\ 0 & 0 & 0 & r_2{\cdot}e^{-r_2 x} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

The matrices for $\mathcal{G}_2$ are given as

$$\hat{\mathbf{M}}_2(x) = \begin{pmatrix} 0 & r_2{\cdot}e^{-r_2 x} \\ 0 & 0 \end{pmatrix} \quad \hat{\mathbf{P}}_2 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

To obtain the system of linear equations, we need:

$$\mathbf{\Pi}_0(1) = \begin{pmatrix} p_{00} & p_{02} & p_{04} \\ p_{20} & p_{22} & p_{24} \\ p_{40} & p_{42} & p_{44} \end{pmatrix} \quad \mathbf{\Pi}_1(1) = \begin{pmatrix} p_{11} & p_{13} & p_{15} & p_{17} \\ p_{31} & p_{33} & p_{35} & p_{37} \\ p_{51} & p_{53} & p_{55} & p_{57} \\ p_{71} & p_{73} & p_{75} & p_{77} \end{pmatrix} \quad \bar{\mathbf{\Pi}}_1^a(1) = \begin{pmatrix} \bar{p}_{10} & \bar{p}_{12} & \bar{p}_{14} \\ \bar{p}_{30} & \bar{p}_{32} & \bar{p}_{32} \\ \bar{p}_{50} & \bar{p}_{52} & \bar{p}_{54} \\ \bar{p}_{70} & \bar{p}_{72} & \bar{p}_{74} \end{pmatrix}$$

All elements in these $\mathbf{\Pi}$-matrices can be computed by the transient probability in the corresponding CTMCs $\mathcal{C}_0$, $\mathcal{C}_1$ and $\mathcal{C}_1^a$ (cf. Fig. 9).

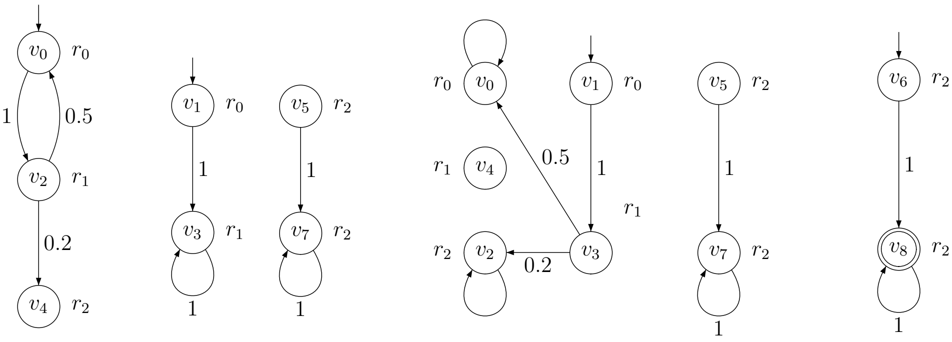
The obtained system of linear equations by applying Theorem 4 is:

$$\begin{pmatrix} u_0 \\ u_2 \\ u_4 \end{pmatrix} = \begin{pmatrix} p_{00} & p_{02} & p_{04} \\ p_{20} & p_{22} & p_{24} \\ p_{40} & p_{42} & p_{44} \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_3 \\ u_5 \\ u_7 \end{pmatrix}$$

$$\begin{pmatrix} u_1 \\ u_3 \\ u_5 \\ u_7 \end{pmatrix} = \begin{pmatrix} p_{11} & p_{13} & p_{15} & p_{17} \\ p_{31} & p_{33} & p_{35} & p_{37} \\ p_{51} & p_{53} & p_{55} & p_{57} \\ p_{71} & p_{73} & p_{75} & p_{77} \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} u_6 \\ u_8 \end{pmatrix} + \begin{pmatrix} \bar{p}_{10} & \bar{p}_{12} & \bar{p}_{14} \\ \bar{p}_{30} & \bar{p}_{32} & \bar{p}_{32} \\ \bar{p}_{50} & \bar{p}_{52} & \bar{p}_{54} \\ \bar{p}_{70} & \bar{p}_{72} & \bar{p}_{74} \end{pmatrix} \cdot \begin{pmatrix} u_0 \\ u_1 \\ u_3 \end{pmatrix}$$

$$\begin{pmatrix} u_6 \\ u_8 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} u_6 \\ u_8 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

This can be solved easily.



(a) $\mathcal{C}_0$ (b) $\mathcal{C}_1$ (c) $\mathcal{C}_1^a$ (d) $\mathcal{C}_2$

Fig. 9. Derived CTMCs

Remark 6. We note that for two-clock DTA$^\diamond$ which yield two-clock DMTA$^\diamond$, the approach given in this section fails in general. In the single-clock case, the reset guarantees to jump to $\mathcal{G}_0(0)$ and delay to $\mathcal{G}_{i+1}(0)$ when it is in $\mathcal{G}_i$. However, in the two-clock case, after a delay or reset generally only one clock has a fixed value while the value of the other one is not determined.

The time-complexity of computing the reachability probability in the single-clock DTA$^\diamond$ case is $\mathcal{O}(m \cdot |S|^2 \cdot |Loc|^2 \cdot \lambda \cdot \Delta c + m^3 \cdot |S|^3 \cdot |Loc|^3)$, where m is the number of constants appearing in the DTA$^\diamond$, $|S|$ is the number of states in the CTMC, $|Loc|$ is the number of locations in the DTA$^\diamond$, λ is the maximal exit rate in the CTMC and $\Delta c = \max_{0 \le i < m}\{c_{i+1} - c_i\}$. The first term $m \cdot |S|^2 \cdot |Loc|^2 \cdot \lambda \cdot \Delta c$ is due to the uniformization technique for computing transient distribution; and the second term $m^3 \cdot |S|^3 \cdot |Loc|^3$ is the time complexity for solving a system of linear equations with $\mathcal{O}(m \cdot |S| \cdot |Loc|)$ variables.

5 Model Checking DTA$^\omega$ Specifications

We now deal with DTA$^\omega$ specifications. Given the product $\mathcal{M}^\omega = (Loc, \mathcal{X}, \ell_0, Loc_\mathcal{F}, E, \rightsquigarrow)$, we first define the region graph $\mathcal{G}^\omega(\mathcal{M}^\omega)$ (or simply $\mathcal{G}^\omega$) as $(V, v_0, V_F^\omega, \Lambda, \hookrightarrow)$ without specifying how the accepting set V_F^ω is defined. This will become clear later. The elements V, v_0, Λ and $\hookrightarrow$ are defined in the same way as in Def. 10 (page 16).

The Muller acceptance conditions $Q_\mathcal{F}$ in the DTA$^\omega$ consider the infinite paths that visit the locations in $F \in Q_\mathcal{F}$ infinitely often. For this sake, BSCCs in the *region graph* $\mathcal{G}^\omega$ that consist of set of vertices corresponding to $L_F \in Loc_\mathcal{F}$ are of most importance. Note that it is not sufficient to consider the BSCCs in the DMTA$^\omega$. The reason will become clear in Remark 7. Let $v \in B$ denote that vertex v is in the BSCC B. We define *accepting* BSCC*s* as follows:

Definition 12 (*a*BSCC). *Given a product* $\mathcal{C} \otimes \mathcal{A}^\omega = (Loc, \mathcal{X}, \ell_0, Loc_\mathcal{F}, E, \rightsquigarrow)$ *and its region graph* $\mathcal{G}^\omega$*, a* BSCC B *in* $\mathcal{G}^\omega$ *is* accepting *if there exists* $L_F \in Loc_\mathcal{F}$ *such that for any* $v \in B$, $v{\downharpoonright}_1 \in L_F$. *Let* $a\mathcal{B}$ *denote the set of accepting* BSCC*s in* $\mathcal{G}^\omega$.

Based on $a\mathcal{B}$, we can now define the set of accepting vertices of $\mathcal{G}^\omega$ as $V_F^\omega = \{v \in B \mid B \in a\mathcal{B}\}$. Note that it is *not* an acceptance family but a set of accepting vertices.

Example 11. For the DMTA$^\omega$ in Fig. 5(c) with $Loc_\mathcal{F} = \big\{\{\ell_1, \ell_2, \ell_3\}, \{\ell_4, \ell_5, \ell_6\}\big\}$, the region graph is as in Fig. 10. There is one accepting BSCC, which has been labeled with gray. This BSCC corresponds to the set $\{\ell_4, \ell_5, \ell_6\} \in Loc_\mathcal{F}$ in the DMTA$^\omega$. There is no BSCC corresponding to the set $\{\ell_1, \ell_2, \ell_3\}$ because in the region graph v_{12} and v_{14} are sink vertices connecting to the SCC. In other words, the probabilities will leak when $x > 2$ on either ℓ_1 or ℓ_2. This is determined by the guards on the DTA$^\omega$.

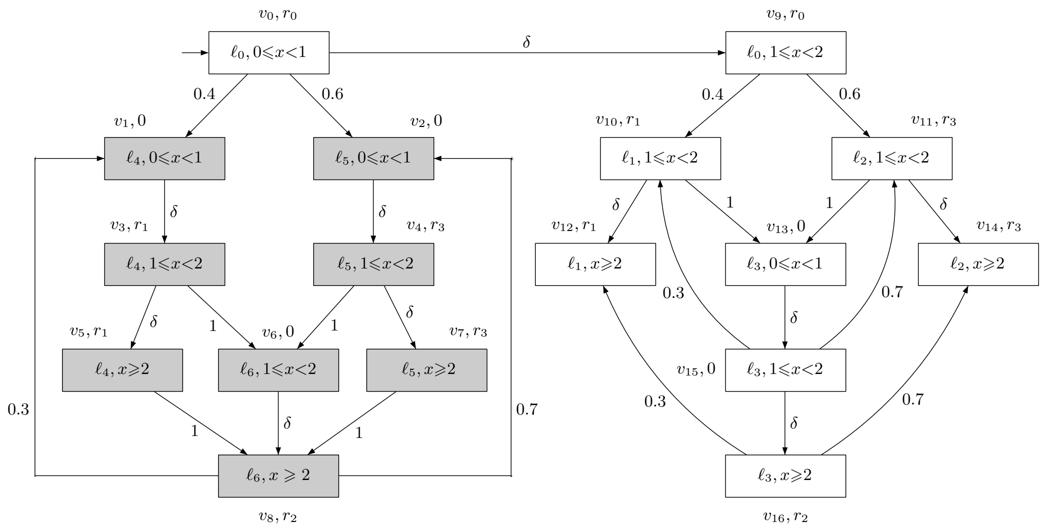


Fig. 10. Region graph of the product DMTA$^\omega$ in Fig. 5(c)

We remark on two points: 1) the probability of staying in an aBSCC is 1, considering both the delay and Markovian transitions. That is to say, there are no outgoing transitions from which probabilities can "leak"; 2) any two aBSCCs are disjoint, such that the probabilities to reach two BSCCs can be added. These two points are later important for the computation of the reachability probability.

Let $Prob^{\mathcal{C}}(\mathcal{A}^\omega)$ be the probability of the set of infinite paths in $\mathcal{C}$ that can be accepted by $\mathcal{A}^\omega$. The following theorem computes $Prob^{\mathcal{C}}(\mathcal{A}^\omega)$ on the region graph:

Theorem 5. *For any* CTMC $\mathcal{C}$*,* DTA$^\omega$ $\mathcal{A}^\omega$*, and the region graph* $\mathcal{G}^\omega = (V, v_0, V_F^\omega, \Lambda, \hookrightarrow)$ *of the product, it holds that:*

$$Prob^{\mathcal{C}}(\mathcal{A}^\omega) = Prob^{\mathcal{G}^\omega}(v_0, \Diamond V_F^\omega).$$

Proof. We show the theorem by the following three steps:

1. $Prob^{\mathcal{C}}(\mathcal{A}^\omega) = Prob^{\mathcal{C}\otimes\mathcal{A}}(Loc_{\mathcal{F}})$, where $Prob^{\mathcal{C}\otimes\mathcal{A}}(Loc_{\mathcal{F}})$ denotes the probability of accepting paths of DMTA $\mathcal{C} \otimes \mathcal{A}$ w.r.t. Muller accepting conditions;
2. $Prob^{\mathcal{C}\otimes\mathcal{A}}(Loc_{\mathcal{F}}) = Prob^{\mathcal{G}^\omega}(v_0, Loc_{\mathcal{F}})$;
3. $Prob^{\mathcal{G}^\omega}(v_0, Loc_{\mathcal{F}}) = Prob^{\mathcal{G}^\omega}(v_0, \Diamond\, V_F^\omega)$.

For the first step, we note that $Paths^{\mathcal{C}}(\mathcal{A}^\omega) = \bigcap_{1\leqslant i\leqslant k} Paths^i$ where

$$Paths^i = \bigcap_{n\geqslant 0} \bigcup_{m\geqslant n} \bigcup_{s_0,\ldots,s_n,s_{n+1}\cdots,s_m} C(s_0, I_0, \ldots, I_{n-1}, s_n, \ldots, I_{m-1}, s_m), \qquad \text{where}$$

- $\{s_{n+1}, \ldots, s_m\} = L_{F_i}$;
- $C(s_0, I_0, \ldots, I_{n-1}, s_n, \ldots, I_{m-1}, s_m)$ is the cylinder set such that each timed path of the cylinder set of the form $s_0 \xrightarrow{t_0} \cdots \xrightarrow{t_{n-1}} s_n \cdots \xrightarrow{t_{m-1}} s_m$ is a prefix of an accepting path of $\mathcal{A}$.

Similar to Lemma 1, one can easily see that each path of CTMC $\mathcal{C}$ can be lifted to a unique path of DMTA$^\omega$ $\mathcal{C} \otimes \mathcal{A}^\omega$. Following the same argument as in Theorem 2, one can obtain that for each cylinder set of the form $C(s_0, I_0, \ldots, I_{n-1}, s_n, \ldots, I_{m-1}, s_m)$, $\mathcal{C}$ and $\mathcal{C} \otimes \mathcal{A}^\omega$ give rise to the same probability. Hence $\mathit{Prob}^{\mathcal{C}}(\mathcal{A}^\omega) = \mathit{Prob}^{\mathcal{C}\otimes\mathcal{A}^\omega}(\mathit{Loc}_{\mathcal{F}})$.

For the second step, we need to define a timed path of $\mathcal{G}^\omega$, which is of the form $v_0 \xrightarrow{t_0} v_1 \xrightarrow{t_2} \cdots$ such that given the initial valuation η_0, one can construct a sequence $\{\eta_i\}$ such that

- $\eta_{i+1} = (\eta_i + t_i)[X_i := 0]$ if $\eta_i + t_i \models \mathit{Inv}(v_i)$ (namely, the transition from v_i to v_{i+1} is via a Markovian transition); and
- $\eta_{i+1} = \eta_i + t_i$ if $\eta_i + t_i \in \partial \mathit{Inv}(v_i)$ (namely, the transition from v_i to v_{i+1} is via a forced boundary jump).

A path of $\mathcal{G}^\omega$ is accepted if the discrete part of the path, namely $v_0 v_1 \cdots$ meets the Muller condition.

Following the standard region construction, one can lift a timed path of DMTA$^\omega$ $\mathcal{C}\otimes\mathcal{A}^\omega$ to a unique timed path of the corresponding region graph $\mathcal{G}^\omega$. Moreover, following the same argument of Theorem 3, one can show that $\mathcal{C} \otimes \mathcal{A}^\omega$ and $\mathcal{G}^\omega$ give rise to the same probability to the accepted paths.

For the third step, we note that according to the ergodicity of PDP (region graph), for each path of $\mathcal{G}^\omega$, with probability 1 the states visited infinitely often constitute a BSCC. It follows that

$$\mathit{Prob}^{\mathcal{G}^\omega}(v_0, \mathit{Loc}_{\mathcal{F}}) = \sum_{B \in a\mathcal{B}} \mathit{Prob}\{\rho \mid \mathit{inf}(\rho) = B\}.$$

We note that for each note v in an accepting BSCC, $\mathit{Prob}\{\mathit{Paths}^{\mathcal{G}^\omega}(v)\} = 1$. Hence

$$\mathit{Prob}^{\mathcal{G}^\omega}(v_0, \mathit{Loc}_{\mathcal{F}}) = \mathit{Prob}^{\mathcal{G}^\omega}(v_0, \Diamond V_F^\omega).$$

■

Actually, the region graph $\mathcal{G}^\omega$ can be simplified to $\mathcal{G}^\omega_{abs}$ to compute $\mathit{Prob}^{\mathcal{C}}(\mathcal{A}^\omega)$. $\mathcal{G}^\omega_{abs}$ is obtained by making (i) all vertices in V_F^ω and (ii) all vertices that *cannot* reach V_F^ω absorbing. (i) is justified by the fact that for these $v \in V_F^\omega$, $\mathit{Prob}^{\mathcal{G}}(v, \Diamond V_F^\omega) = 1$; while (ii) is because $\mathit{Prob}^{\mathcal{G}}(v', \Diamond V_F^\omega) = 0$, for v' cannot reach V_F^ω. It is obvious to see that

$$\mathit{Prob}^{\mathcal{G}^\omega}(v_0, \Diamond V_F^\omega) = \mathit{Prob}^{\mathcal{G}^\omega_{abs}}(v_0, \Diamond V_F^\omega).$$

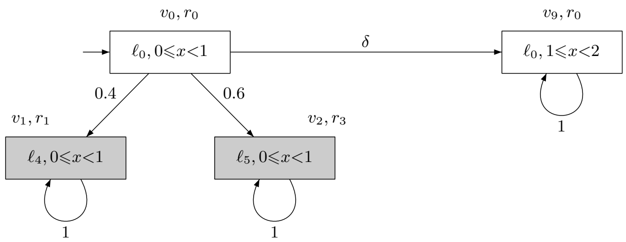


Fig. 11. The transformed region graph $\mathcal{G}^\omega_{abs}$

Example 12. The transformed region graph $\mathcal{G}^\omega_{abs}$ of that in Fig. 10 is shown in Fig. 11. We omit all the vertices that cannot be reached from v_0 in $\mathcal{G}^\omega_{abs}$. In this new model, $V_F^\omega = \{v_1, v_2\}$. We now can perform the approach for computing timed-unbounded reachability probabilities in Section 4 such that Eq. (7)-(9) can be applied. We have:

$Prob^{\mathcal{G}^{\omega}_{abs}}(v_0, \Diamond V^{\omega}_F) = Prob^{\mathcal{G}^{\omega}_{abs}}(v_0, \Diamond at_{v_1}) + Prob^{\mathcal{G}^{\omega}_{abs}}(v_0, \Diamond at_{v_2})$. Note that $Prob^{\mathcal{G}^{\omega}_{abs}}(v_i, \Diamond V^{\omega}_F) = 1$ for $i = 1, 2$ and 0 for $i = 9$. For the delay transition $v_0 \overset{\delta}{\hookrightarrow} v_9$,

$$Prob_{v_0,\delta}(0) = e^{-r_0 \cdot 1} \cdot Prob_{v_9}(1) = e^{-r_0 \cdot 1} \cdot 0 = 0.$$

For the Markovian transition $v_0 \overset{0.4,\{x\}}{\hookrightarrow} v_1$,

$$Prob_{v_0,v_1}(0) = \int_0^1 0.4 \cdot r_0 \cdot e^{-r_0 \cdot \tau} \cdot Prob_{v_1}(\tau) d\tau = \int_0^1 0.4 \cdot r_0 \cdot e^{-r_0 \cdot \tau} d\tau.$$

A similar reasoning applies to $v_0 \overset{0.6,\{x\}}{\hookrightarrow} v_2$. In the end, we have

$$Prob^{\mathcal{C}}(\mathcal{A}^{\omega}) = \int_0^1 (0.4 + 0.6) \cdot r_0 \cdot e^{-r_0 \cdot \tau} d\tau = \int_0^1 r_0 \cdot e^{-r_0 \cdot \tau} d\tau = 1 - e^{-r_0}.$$

Remark 7 (Why not BSCC*s in the product?).* There are two BSCCs in the product DMTA$^{\omega}$: one formed by $\{\ell_1, \ell_2, \ell_3\}$ and the other by $\{\ell_4, \ell_5, \ell_6\}$. As turned out in the example that only the latter forms a BSCC in the region graph while the former does not. This is because the guards on the transitions also play a role on whether a path can be accepted. The impact of guards, however, is not immediately clear in the product DMTA$^{\omega}$, but is implicitly consumed in the region graph. This justifies finding BSCCs in the region graph instead of in the product.

Theorem 5 implies that computing the probability of a set of infinite paths (LHS) can be reduced to computing the probability of a set of finite paths (RHS). The latter has been solved in Section 4 with the characterization of a system of integral equations and also the approximation by a system of PDEs. The case of a single clock DTA$^{\omega}$, due to this reduction, can also be solved as a system of ODEs (as in Section 4.2).

6 Conclusion

We addressed the quantitative verification of a CTMC $\mathcal{C}$ against a DTA$^{\Diamond}$ $\mathcal{A}$ (DTA$^{\omega}$ $\mathcal{A}^{\omega}$). As a key result, we showed that the set of the accepting paths in $\mathcal{C}$ by DTA is measurable and the probability of $\mathcal{C} \models \mathcal{A}$ can be reduced to computing reachability probabilities in the embedded DTMP of a PDP. The probabilities can be characterized by a system of Volterra integral equations of the second type and can be approximated by a system of PDEs. For single-clock DTA$^{\Diamond}$, this reduces to solving a system of linear equations whose coefficients are a system of ODEs. The probability of $\mathcal{C} \models \mathcal{A}^{\omega}$ is reducible to computing the reachability probabilities to the accepting BSCCs in the region graph and the thus obtained PDP.

Acknowledgement

We thank Jeremy Sproston for the fruitful and insightful discussions. This research is funded by the DFG research training group 1295 AlgoSyn, the Dutch Bsik project BRICK, the NWO project QUPES and the EU FP7 project QUASIMODO.

References

[AD94] Rajeev Alur and David L. Dill. A theory of timed automata. *Theor. Comput. Sci.*, 126(2):183–235, 1994.

[ASSB00] Adnan Aziz, Kumud Sanwal, Vigyan Singhal, and Robert K. Brayton. Model-checking continous-time Markov chains. *ACM Trans. Comput. Log.*, 1(1):162–170, 2000.

[AW95] George B. Arfken and Hans J. Weber. *Mathematical Methods for Physicists (4th ed.)*. Academic Press, 1995.

[BBB+07] Christel Baier, Nathalie Bertrand, Patricia Bouyer, Thomas Brihaye, and Marcus Größer. Probabilistic and topological semantics for timed automata. In *FSTTCS*, pages 179–191, 2007.

[BBB+08] Christel Baier, Nathalie Bertrand, Patricia Bouyer, Thomas Brihaye, and Marcus Grösser. Almost-sure model checking of infinite paths in one-clock timed automata. In *LICS*, pages 217–226, 2008.

[BBBM08] Nathalie Bertrand, Patricia Bouyer, Thomas Brihaye, and Nicolas Markey. Quantitative model-checking of one-clock timed automata under probabilistic semantics. In *QEST*, pages 55–64, 2008.

[BCH+07] Christel Baier, Lucia Cloth, Boudewijn R. Haverkort, Matthias Kuntz, and Markus Siegle. Model checking Markov chains with actions and state labels. *IEEE Trans. Software Eng.*, 33(4):209–224, 2007.

[BHHK03] Christel Baier, Boudewijn R. Haverkort, Holger Hermanns, and Joost-Pieter Katoen. Model-checking algorithms for continuous-time Markov chains. *IEEE Trans. Software Eng.*, 29(6):524–541, 2003.

[BPDG98] Béatrice Bérard, Antoine Petit, Volker Diekert, and Paul Gastin. Characterization of the expressive power of silent transitions in timed automata. *Fundam. Inform.*, 36(2-3):145–182, 1998.

[CD88] Oswaldo L.V. Costa and Mark H.A. Davis. Approximations for optimal stopping of a piecewise-deterministic process. *Math. Control Signals Systems*, 1(2):123–146, 1988.

[Cor91] C. Corduneanu. *Integral Equations and Applications*. Cambridge University Press, 1991.

[Dav84] Mark H. A. Davis. Piecewise-deterministic Markov processes: A general class of non-diffusion stochastic models. *Journal of the Royal Statistical Society (B)*, 46(3):353–388, 1984.

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

[DHS09] Susanna Donatelli, Serge Haddad, and Jeremy Sproston. Model checking timed and stochastic properties with $\mathrm{CSL}^{\mathrm{TA}}$. *IEEE Trans. Software Eng.*, 35(2):224–240, 2009.

[KNSS02] Marta Z. Kwiatkowska, Gethin Norman, Roberto Segala, and Jeremy Sproston. Automatic verification of real-time systems with discrete probability distributions. *Theor. Comput. Sci.*, 282(1):101–150, 2002.

[LL85] Suzanne M. Lenhart and Yu-Chung Liao. Integro-differential equations associated with optimal stopping time of a piecewise-deterministic process. *Stochastics*, 15(3):183–207, 1985.

[LY91] Suzanne M. Lenhart and Naoki Yamada. Perron's method for viscosity solutions associated with piecewise-deterministic processes. *Funkcialaj Ekvacioj*, 34:173–186, 1991.

[Var85] Moshe Y. Vardi. Automatic verification of probabilistic concurrent finite-state programs. In *FOCS*, pages 327–338, 1985.

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 (editor): 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 concurrent 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

2007-18 Uwe Naumann: Call Tree Reversal is NP-Complete

2007-19 Jan Riehme, Andrea Walther, Jörg Stiller, Uwe Naumann: Adjoints for Time-Dependent Optimal Control

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

2007-21 Tingting Han, Joost-Pieter Katoen, and Alexandru Mereacre: Compositional Modeling and Minimization of Time-Inhomogeneous Markov Chains

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

2008-01 * Fachgruppe Informatik: Jahresbericht 2007

2008-02 Henrik Bohnenkamp, Marielle Stoelinga: Quantitative Testing

2008-03 Carsten Fuhs, Jürgen Giesl, Aart Middeldorp, Peter Schneider-Kamp, René Thiemann, Harald Zankl: Maximal Termination

2008-04 Uwe Naumann, Jan Riehme: Sensitivity Analysis in Sisyphe with the AD-Enabled NAGWare Fortran Compiler

2008-05 Frank G. Radmacher: An Automata Theoretic Approach to the Theory of Rational Tree Relations

2008-06 Uwe Naumann, Laurent Hascoet, Chris Hill, Paul Hovland, Jan Riehme, Jean Utke: A Framework for Proving Correctness of Adjoint Message Passing Programs

2008-07 Alexander Nyßen, Horst Lichter: The MeDUSA Reference Manual, Second Edition

2008-08 George B. Mertzios, Stavros D. Nikolopoulos: The λ-cluster Problem on Parameterized Interval Graphs

2008-09 George B. Mertzios, Walter Unger: An optimal algorithm for the k-fixed-endpoint path cover on proper interval graphs

2008-10 George B. Mertzios, Walter Unger: Preemptive Scheduling of Equal-Length Jobs in Polynomial Time

2008-11 George B. Mertzios: Fast Convergence of Routing Games with Splittable Flows

2008-12 Joost-Pieter Katoen, Daniel Klink, Martin Leucker, Verena Wolf: Abstraction for stochastic systems by Erlang's method of stages

2008-13 Beatriz Alarcón, Fabian Emmes, Carsten Fuhs, Jürgen Giesl, Raúl Gutiérrez, Salvador Lucas, Peter Schneider-Kamp, René Thiemann: Improving Context-Sensitive Dependency Pairs

2008-14 Bastian Schlich: Model Checking of Software for Microcontrollers

2008-15 Joachim Kneis, Alexander Langer, Peter Rossmanith: A New Algorithm for Finding Trees with Many Leaves

2008-16 Hendrik vom Lehn, Elias Weingärtner and Klaus Wehrle: Comparing recent network simulators: A performance evaluation study

2008-17 Peter Schneider-Kamp: Static Termination Analysis for Prolog using Term Rewriting and SAT Solving

2008-18 Falk Salewski: Empirical Evaluations of Safety-Critical Embedded Systems

2008-19 Dirk Wilking: Empirical Studies for the Application of Agile Methods to Embedded Systems

2009-02 Taolue Chen, Tingting Han, Joost-Pieter Katoen, Alexandru Mereacre: Quantitative Model Checking of Continuous-Time Markov Chains Against Timed Automata Specifications

2009-03 Alexander Nyßen: Model-Based Construction of Embedded Real-Time Software - A Methodology for Small Devices

2009-04 Daniel Klünder: Entwurf eingebetteter Software mit abstrakten Zustandsmaschinen und Business Object Notation

2009-05 George B. Mertzios, Ignasi Sau, Shmuel Zaks: A New Intersection Model and Improved Algorithms for Tolerance Graphs

2009-06 George B. Mertzios, Ignasi Sau, Shmuel Zaks: The Recognition of Tolerance and Bounded Tolerance Graphs is NP-complete

2009-07 Joachim Kneis, Alexander Langer, Peter Rossmanith: Derandomizing Non-uniform Color-Coding I
2009-08 Joachim Kneis, Alexander Langer: Satellites and Mirrors for Solving Independent Set on Sparse Graphs
2009-09 Michael Nett: Implementation of an Automated Proof for an Algorithm Solving the Maximum Independent Set Problem
2009-10 Felix Reidl, Fernando Sánchez Villaamil: Automatic Verification of the Correctness of the Upper Bound of a Maximum Independent Set Algorithm
2009-11 Kyriaki Ioannidou, George B. Mertzios, Stavros D. Nikolopoulos: The Longest Path Problem is Polynomial on Interval Graphs
2009-12 Martin Neuhäußer, Lijun Zhang: Time-Bounded Reachability in Continuous-Time Markov Decision Processes
2009-13 Martin Zimmermann: Time-optimal Winning Strategies for Poset Games
2009-14 Ralf Huuck, Gerwin Klein, Bastian Schlich (eds.): Doctoral Symposium on Systems Software Verification (DS SSV'09)
2009-15 Joost-Pieter Katoen, Daniel Klink, Martin Neuhäußer: Compositional Abstraction for Stochastic Systems

* These reports are only available as a printed version.
Please contact ttbiblio@informatik.rwth-aachen.de to obtain copies.