Bull. Nov. Comp. Center, Comp. Science, 31 (2010), 179–193
© 2010 NCC Publisher

Real arithmetic based verification of prioritized time Petri nets with parameters

Irina Virbitskaite, Hans Fleischhack, Anna Dedova

Abstract. Time Petri nets with priorities are a widely studied model for real-time systems. The intention of the paper is to develop an algorithm for parametric timing behaviour verification of real-time and concurrent systems represented by prioritized time Petri nets (PrTPNs). To achieve the purpose, we introduce a notion of the parametric PrTPN which is a modification of the PrTPN by using parameter variables in specification of timing constraints on transition firings. System properties are given as formulae of a parametric extension of the real-time branching time temporal logic TCTL, PTCTL. The verification algorithm consists in constructing conditions on timing parameter variables under which the PrTPN with bounded parameters works w.r.t. the checked PTCTL-formula. We have also shown the correctness and evaluated the complexity of the algorithm proposed.

1. Introduction

The idea of adding explicit time to models for concurrency was first introduced in the seventies for Petri nets [15, 16]. Since then, timed models based on Petri nets and finite automata were extensively studied, and various tools were developed for their analysis. Among the models proposed for specification and verification of systems in which time plays an essential role like communication protocols, hardware, or real-time systems, two models are especially prominent: time Petri nets (TPNs) [15] and timed automata (TA) (see, for example, [1]). Several recent papers investigate the relative expressiveness of the models. In particular, in [4], it has been shown that timed automata and time Petri nets are equally expressive in terms of timed language acceptance, but the timed automata are strictly more expressive in terms of weak timed bisimilarity. Though priorities are pervasive in some families of real-time systems, they are not supported by the time Petri net models, and cannot be generally encoded within. Therefore, an extension of TPNs with priorities (PrTPNs for short) has been proposed in [5]. In a PrTPN, a transition is not allowed to fire if some transition with higher priority is firable at the same instant. Also, in [5], it has been proved that priorities strictly extend the expressiveness of time Petri nets, and in particular that bounded PrTPNs can be considered equivalent to timed automata, in terms of weak timed bisimulation.

Within the last two decades, serious attempts have been made to extend the success of model checking to the setting of timed automata and time

Petri nets (see [1, 6] among others). Temporal logics have also been extended to express quantitative real-time properties (see, for example, [14]). One of the major obstacles for real-time model checking is that it usually requires overly detailed specification of timing characteristics of both the model and its properties. In the case when the checked formula is not satisfied by the model, the timing characteristics are changed, and verification algorithm is applied again. It leaves users in repetitive trial-and-error cycles to select proper timings. One of the ways out is parametric reasoning working on a model with parameters — symbolic constants with unknown, fixed values. Using parametric reasoning, one can either verify that the model satisfies some property for all possible values of the parameters, or find constraints on the parameters (i.e. synthesize parameters) defining the set of all possible values for which the model satisfies a property. In [3], Alur et al. have introduced parameters in discrete- and dense-timed automata and have shown that the emptiness problem is decidable when only one clock is compared to parameters. In [12], Hune et al. have studied a subclass of parametric dense-timed automata such that each parameter occurs either as a lower bound or as an upper bound. Wang in [20, 21], Emerson et al. in [10], Alur et al. in [2] and Bruyére et al. in [7], etc. have introduced parameters in temporal logics. There it has been established that the model-checking problem for TCTL extended with parameters (PTCTL) over discrete- and dense-timed automata (without parameters) is decidable. The paper [8] has studied the model-checking and parameter synthesis problems of the language PTCTL over discrete-timed automata with parameters, i.e. parameters are allowed both in the model (timed automaton) and in the property (temporal formula). It has turned out that the model-checking problem of PTCTL is undecidable over discrete-timed automata with only one parametric clock. The model-checking and the parameter synthesis problems become decidable for a fragment of PTCTL where equality is not allowed. The case of dense-timed automata with one parametric clock is not investigated in the literature. The papers [7, 8] have shown that the durations of runs of a timed automata are expressible in the arithmetic of Pressburger (when the time domain is discrete) and the theory of the reals (when the time domain is dense). Other reseachers have also proposed the use of the Pressburger arithmetic and the Real arithmetic in the context of timed automata. In particular, Common et al. [9] have studied the use of the Real arithmetic to express the reachability relation of timed automata. The paper [18] provided a timing behaviour analysis algorithm for one-safe time Petri nets and TCTL-formulae using cactus structures [20] to calculate the durations of runs. More recently, ”on-the-fly” model checking algorithms for parametric time Petri nets with stopwatches w.r.t. a subset of PTCTL-formulae have been put forward in [17].

The intention of the paper is to develop an algorithm for parametric timing behaviour verification of real-time and concurrent systems represented by prioritized time Petri nets (PrTPNs). To fulfill the purpose, we introduce a notion of the parametric PrTPN which is a modification of the PrTPN by using parameter variables in specification of timing constraints on transition firings. Net properties are given as formulae of PTCTL. The durations of computational paths are expressed in formulae of the real arithmetic [7]. A timing behaviour analysis algorithm consists in constructing conditions on timing parameter variables under which the PrTPN with bounded parameters works w.r.t. the checked PTCTL-formula. Thus the approach allows an 'adjustment' of timing specifications of the system w.r.t. the property via a single execution of verification procedure.

The rest of the paper is organized as follows. The next section defines some notations which are needed to introduce parameters into the net model and logic formulae. The basic definitions concerning parametric prioritized time Petri nets are given in the next section. Section 4 recalls the syntax and semantics of PTCTL. Section 5 describes our observations and algorithm for solving the parametric timing behaviour problem. We have also shown the correctness and evaluated the complexity of the algorithm proposed. Section 6 contains conclusion and some remarks on future work.

2. Notations

In this section, we define some notations which are needed to introduce parameters into the net model and logic formulae.

Let $\mathbf{N}$ be the set of natural numbers, $\mathbf{R}$ be the set of nonnegative real numbers, and $\mathbf{R}^+$ be the set of positive real numbers.

Assume a finite set Θ of parameters θ that are shared by the net model and the logical formulae. Let θ with and without subscripts range over Θ. A *parameter valuation* χ for Θ is a mapping from Θ into $\mathbf{N}$ which assigns a natural number to each parameter θ from Θ.

From now on, α, β mean any linear term $\sum\limits_{j \in J} c_j \theta_j + c$, where $c_j, c \in \mathbf{N}$ and $J \subset \{1, ..., n\}$. A parameter valuation χ can be naturally extended to linear terms by defining $\chi(c) = c$ for any $c \in \mathbf{N}$. We shall use $\mathcal{T}$ to denote the set of linear terms.

Let $\mathcal{I}(\mathbf{N}, \mathcal{T})$ be the set of parametric time intervals i such that the left end-point of i, $\downarrow i$, belongs to $\mathbf{N} \cup \mathcal{T}$ and the right end-point of i, $\uparrow i$, belongs to $\mathbf{N} \cup \mathcal{T} \cup \{\infty\}$. Given $i \in \mathcal{I}(\mathbf{N}, \mathcal{T})$ and a parameter valuation χ, i^χ denotes the time interval obtained from i by replacing every occurrence of parameters θ with $\chi(\theta)$.

3. Parametric prioritized time Petri nets

In this section, we define some terminology concerning parametric prioritized time Petri nets.

We start with the notion of Petri nets. A *Petri net* is a tuple $N = (P, T, {}^\bullet(\cdot), (\cdot)^\bullet, m^*)$, where P is a finite set of places, T is a finite set of transitions ($P \cap T = \emptyset$), ${}^\bullet(\cdot) \in (\mathbf{N}^P)^T$ is the forward incidence mapping, $(\cdot)^\bullet \in (\mathbf{N}^P)^T$ is the backward incidence mapping, $m^* \in \mathbf{N}^P$ is the initial marking. A *marking* m of N is a mapping from $\mathbf{N}^P$. A transition t is *enabled* in a marking m if $m \geq^\bullet t$, otherwise it is *disabled*. Let $enable(m)$ be the set of transitions enabled in m. A transition t' is called *newly enabled* after firing a transition t in a marking m, if it is enabled in the marking $m -{}^\bullet t + t^\bullet$ and it is either disabled in the intermediate marking $m -{}^\bullet t$ or $t = t'$. Formally, define a predicate $\uparrow enabled(t', m, t) \in \{true, false\}$ which is true, if a transition t' is newly enabled after firing a transition t in a marking m, and false, otherwise: $\uparrow enabled(t', m, t) = [t' \in enabled(m-{}^\bullet t+t^\bullet)] \wedge [t' \notin enabled(m-{}^\bullet t) \vee (t = t')]$.

Time Petri Nets introduced in [15] extend Petri Nets with timing constraints on the firings of transitions. In a time Petri net, a time interval is associated with each transition. Also, an implicit clock is associated with each transition and gives the elapsed time since it was last enabled. An enabled transition can be fired if its clock value belongs to the interval of the transition. Furthermore, time cannot progress if time elapsing would result in leaving the interval of an enabled transition. Firing a transition takes no time. An extension of time Petri nets with priorities (PrTPNs for short) has been proposed in [5]. In a PrTPN, a transition is not allowed to fire if some transition with higher priority is fireable at the same instant. We introduce an extension of PrTPNs — parametric PrTPNs whose transitions are associated with time predicates representing unspecified timing constraints on transition firings. The following definitions formalize these principles.

Let $\mathcal{V} = [T \to \mathbf{R}]$ be the set of *time assignments* for transitions from T. Given $\nu \in \mathcal{V}$ and $\delta \in \mathbf{R}$, we let $\nu + \delta$ denote the time assignment of the value $\nu(t) + \delta$ to each t from T.

Definition 1. A *parametric prioritized time Petri net (PPrTPN)* is a tuple $\mathcal{N} = (P, T, {}^\bullet(\cdot), (\cdot)^\bullet, m^*, \succ, \Theta, I, \nu^*)$, where

- $(P, T, {}^\bullet(\cdot), (\cdot)^\bullet, m^*)$ is a Petri net,
- $\succ \in T \times T$ is a transitive, asymmetric, irreflexive binary *priority* relation,
- Θ is a finite set of parameters ($\Theta \cap (P \cup T) = \emptyset$),
- $I : T \to \mathcal{I}(\mathbf{N}, \mathcal{T})$ is a function that associates each transition t with a parametric time interval $I(t) \in \mathcal{I}(\mathbf{N}, \mathcal{T})$,

- $\nu^* \in \mathcal{V}$ is the *initial time assignment*.

Let $c_{\mathcal{N}}$ mean the biggest constant from $\mathbf{N}$ appearing as the endpoint of a time interval and $\Theta_{\mathcal{N}}$ denote the set of parameters appearing in linear terms in a specification of $\mathcal{N}$.

Example 1.

$\mathcal{N}_1$:

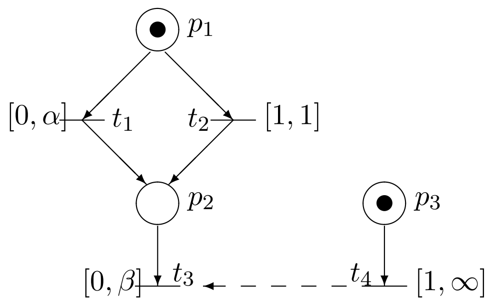


Figure 1

A simple example of PPrTPN with $\Theta = \{\theta_1, \theta_2\}$ and two linear terms $\alpha = \theta_1 + 1$, $\beta = \theta_2$ is shown in Fig. 1.

The semantics of a PPrTPN $\mathcal{N}$ is defined at a parameter valuation χ. From now on, $\mathcal{N}^\chi$ means a PrTPN obtained from PPrTPN $\mathcal{N}$ by replacing every occurrence of a parameter θ with $\chi(\theta)$ for all $\theta \in \Theta_{\mathcal{N}}$.

A *state* q of $\mathcal{N}^\chi$ is a pair $\langle m, \nu \rangle$, where m is a marking of $\mathcal{N}^\chi$ and $\nu \in \mathcal{V}$. The *initial state* of $\mathcal{N}^\chi$ is the pair $q^* = \langle m^*, \nu^* \rangle$. The states of $\mathcal{N}^\chi$ change, if time passes or if a transition fires. In a state $q = \langle m, \nu \rangle$, time $\delta \in \mathbf{R}^+$ *can pass*, if for all $t \in \mathit{enable}(m)$ there exists $\delta' \geq \delta$ such that $\nu(t) + \delta' \in (I(t))^\chi$. In this case, the state $q' = \langle m', \nu' \rangle$ is *obtained by passing* δ from q (written $q \overset{\delta}{\Rightarrow} q'$), if $m' = m$ and $\nu' = \nu + \delta$. In a state $q = \langle m, \nu \rangle$, a transition $t \in T$ is *fireable*, if $t \in \mathit{enable}(m)$, $\nu(t) \in (I(t))^\chi$, and for all $t' \in \mathit{enabled}(m)$ if $t' \succ t$ then $\nu(t') \notin (I(t))^\chi$. In this case, the state $q' = \langle m', \nu' \rangle$ is *obtained by firing* t from q (written $q \overset{0}{\Rightarrow} q'$), if $m' = m - {}^\bullet t + t^\bullet$, and $\forall t' \in T \diamond \nu'(t') = \begin{cases} 0, & \text{if } \uparrow \mathit{enabled}(t', m, t), \\ \nu(t'), & \text{otherwise.} \end{cases}$ In the case when it is essential that q' is obtained from q by firing a concrete transition t, we shall also write $q \overset{t}{\Rightarrow} q'$.

A *q-run* (*run*) r of $\mathcal{N}^\chi$ is a finite (infinite) sequence $r = (q_i)_{0 \leq i \leq j}$ ($r = (q_i)_{i \geq 0}$) of states and real numbers $\delta_i \in \mathbf{R}$ of the form: $q = q_0 \overset{\delta_0}{\Rightarrow} q_1 \ldots\ q_{j-1} \overset{\delta_{j-1}}{\Rightarrow} q_j$ ($q = q_0 \overset{\delta_0}{\Rightarrow} q_1 \ldots\ q_{n-1} \overset{\delta_{n-1}}{\Rightarrow} q_n \overset{\delta_n}{\Rightarrow} q_{n+1} \ldots$). A *position* $\mathbf{p}$ in r is a state $q_i + \delta$, where $i \geq 0$ and either $\delta = 0$ or $0 < \delta < \delta_i$. The *duration* $D(r, \mathbf{p})$ of a run r in a position $\mathbf{p} = q_j + \delta$ is equial to $\sum_{0 \leq j < i} \delta_i + \delta$. Since we allow consecutive firings of several instantaneous transitions in a run, different positions can have the same duration. The set of positions in a run r can be totally ordered as follows. Let $\mathbf{p} = q_i + \delta$ and $\mathbf{p}' = q_{i'} + \delta'$ be two pisitions in r. Then $\mathbf{p} < \mathbf{p}'$ iff either $i < i'$ or $i = i'$ and $\delta < \delta'$. We shall write $\mathbf{p} \leq \mathbf{p}'$ if $\mathbf{p} < \mathbf{p}'$ or $\mathbf{p} = \mathbf{p}'$. A state q is *reachable* in $\mathcal{N}^\chi$ if it appears in a q^*-run of $\mathcal{N}^\chi$. Let $RS(\mathcal{N}^\chi)$ denote the set of all reachable states of $\mathcal{N}^\chi$.

To guarantee that in any run of $\mathcal{N}^\chi$ time is increasing beyond any bound, we need the following *progress condition*: for every set of transitions $\{t_1, t_2, \ldots, t_n\}$ s.t. $\forall\, 1 \leq i < n \diamond t_i^\bullet \cap {}^\bullet t_{i+1} \neq \emptyset$ and $t_n^\bullet \cap {}^\bullet t_1 \neq \emptyset$ it holds $\sum_{1 \leq i \leq n} \downarrow (I(t_i))^\chi > 0$. We call $\mathcal{N}^\chi$ *bounded*, if there is $K \in \mathbf{N}$ such that for any $\langle m, \nu \rangle \in RS(\mathcal{N}^\chi)$ and any $p \in P$ it holds that $m(p) \leq K$. In the sequel, $\mathcal{N}^\chi$ will always denote a bounded PrTPN satisfying the progress condition.

4. PTCTL: syntax and semantics

In this section, we review the syntax and semantics of PTCTL (Parametric Timed Computation Tree Logic) proposed in [20].

Definition 2. The *PTCTL-formula* φ is inductively defined by the following grammar: $\phi ::= \mathcal{P} \mid \neg\phi \mid \phi \vee \phi \mid \alpha \sim \beta \mid \phi\ Q\ U_{\sim\alpha}\phi$, where $\sim \in \{<, \leq, =, \geq, >\}$, $Q \in \{\exists, \forall\}$, $\mathcal{P} \in PR$ and $PR = \{\mathcal{P} \mid \mathcal{P}\ :\ m \to \{true, false\}\}$ is a set of propositions on the net marking. The set of free parameters of φ is denoted by Θ_φ.

Given a PTCTL-formula φ and a parameter valuation χ, we let φ^χ be the PTCTL-formula obtained from φ by replacing every occurrence of θ with $\chi(\theta)$ for all $\theta \in \Theta_\varphi$. PTCTL-formulae φ^χ are interpreted on the states of a model $\mathcal{M} = (RS(\mathcal{N}^\chi), \mathcal{W})$, where $\mathcal{W}\ :\ RS(\mathcal{N}^\chi) \to 2^{PR}$ is a function such that $\mathcal{W}(q = \langle m, \nu \rangle) = \{\mathcal{P} \in PR \mid \mathcal{P}(m) = true\}$. Given a state $q \in RS(\mathcal{N}^\chi)$ and a PTCTL-formula φ^χ, the *satisfaction* relation $\mathcal{N}^\chi, q \models \varphi^\chi$ is defined inductively as follows:

$\mathcal{N}^\chi, q \models \mathcal{P}^\chi$	$\iff$	$\mathcal{P} \in \mathcal{W}(q)$
$\mathcal{N}^\chi, q \models (\neg\phi)^\chi$	$\iff$	$\mathcal{N}^\chi, q \not\models \phi^\chi$
$\mathcal{N}^\chi, q \models (\phi \vee \psi)^\chi$	$\iff$	$\mathcal{N}^\chi, q \models \phi^\chi$ or $\mathcal{N}^\chi, q \models \psi^\chi$
$\mathcal{N}^\chi, q \models (\alpha \sim \beta)^\chi$	$\iff$	$\chi(\alpha) \sim \chi(\beta)$
$\mathcal{N}^\chi, q \models (\phi\ Q\ U_{\sim\alpha}\psi)^\chi$	$\iff$	for any/some (depending on Q) q-run $r = (q_i)_{i \geq 0}$in $\mathcal{N}^\chi$, there exists a position $\mathbf{p}$ in r such that $D(r, \mathbf{p}) \sim \chi(\alpha)$, $\mathcal{N}^\chi$, $\mathbf{p} \models \psi^\chi$ and $\mathcal{N}^\chi, \mathbf{p}' \models \phi^\chi$ for all positions $\mathbf{p}'$ in r such that $\mathbf{p}' < \mathbf{p}$

We say that $\mathcal{N}^\chi$ satisfies φ^χ (written $\mathcal{N}^\chi \models \varphi^\chi$) iff $\mathcal{N}^\chi, q^* \models \varphi^\chi$.

The parametric timing behaviour analysis problem $\mathcal{PTBA}(\mathcal{N}, \varphi)$ is formulated as follows: compute a symbolic representation of the set of parameter valuations χ on Θ_φ such that $\mathcal{N}^\chi \models \varphi^\chi$. The structural translation preserving timed language acceptance proposed in [4] from a TA into a bounded TPN can straightforwardly be extended to parametric TA. As the emptiness problem (and then, the reachability problem) is undecidable for parametric TA [3], it is also undecidable for parametric (bounded) TPNs. Since the emptiness problem is a particular case of the model checking problem, the latter is undecidable for parametric (bounded) TPNs and hence for parametric (bounded) PrTPNs. Clearly, the same holds for $\mathcal{PTBA}(\mathcal{N}, \varphi)$.

5. Parametric timing behaviour analysis

5.1. Region graphs

In this subsection, we recall the definition of regions (equivalence classes of states) and region graphs [1] in order to get a finite representation of the state-space of the PrTPN $\mathcal{N}^\chi$.

Before introducing the notion of a region, we have to give the following auxiliary definitions. Let $c_{\mathcal{N}^\chi}$ mean the biggest constant from $\mathbf{N}$ appearing as the endpoint of a time interval in $\mathcal{N}^\chi$. For any $\delta \in \mathbf{R}$, $\{\delta\}$ denotes the fractional part of δ, and $\lfloor \delta \rfloor$ denotes the integral part of δ. Given $\nu, \nu' \in \mathcal{V}$, $\nu \simeq \nu'$ iff the following conditions are met: (i) for each $t \in T$: either $\lfloor \nu(t) \rfloor = \lfloor \nu'(t) \rfloor$ or $\nu(t), \nu'(t) > c_{\mathcal{N}^\chi}$, (ii) for each $t, t' \in T$ such that $\nu(t) \leq c_{\mathcal{N}^\chi}$ and $\nu'(t) \leq c_{\mathcal{N}^\chi}$: (a) $\{\nu(t)\} \leq \{\nu(t')\} \Leftrightarrow \{\nu'(t)\} \leq \{\nu'(t')\}$; (b) $\{\nu(t)\} = 0 \Leftrightarrow \{\nu'(t)\} = 0$. Given $\nu \in \mathcal{V}$, we use $[\nu]$ to denote the equivalence class of ν w.r.t. $\simeq$.

Lemma 1. *Let $q = \langle m, \nu \rangle$, $q' = \langle m, \nu' \rangle \in RS(\mathcal{N}^\chi)$ such that $\nu \simeq \nu'$. Then, for any PTCTL-formula φ, $\mathcal{N}^\chi, \langle m, \nu \rangle \models \varphi^\chi \iff \mathcal{N}^\chi, \langle m, \nu' \rangle \models \varphi^\chi$.*

A *region* of $\mathcal{N}^\chi$ is called to be a set $[q] = \langle m, [\nu] \rangle = \{\langle m', \nu' \rangle \in RS(\mathcal{N}^\chi) \mid m = m' \wedge \nu' \simeq \nu\}$. From now on, v^* denotes the initial region $[q^*]$ of $G(\mathcal{N}^\chi)$. A region $\langle m, [\nu] \rangle$ is called *boundary*, if $\nu \not\simeq \nu + \delta$ for any $\delta > 0$; *unbounded*, if $\nu(t) > c_{\mathcal{N}^\chi}$ for any $t \in T$. Let $\langle m, [\nu] \rangle$ and $\langle m', [\nu'] \rangle$ be two distinct regions. Then $\langle m', [\nu'] \rangle$ is said to be a *successor* of $\langle m, [\nu] \rangle$ (written $\langle m', [\nu'] \rangle = succ(\langle m, [\nu] \rangle)$), if $m = m'$, $\nu' = \nu + \delta$ for some positive $\delta \in \mathbf{R}^+$ and $\nu + \delta' \in [\nu] \cup [\nu']$ for all $\delta' < \delta$. The *region graph* of $\mathcal{N}^\chi$ is defined to be the labelled directed graph $G(\mathcal{N}^\chi) = (V, E, l)$. The vertex set V is the set of all regions of $\mathcal{N}^\chi$. The edge set E consists of two types of edges: (i) the edge $(\langle m, [\nu] \rangle, \langle m', [\nu'] \rangle)$ may represent firing a transition if $\langle m', \nu' \rangle$ is obtained from $\langle m, \nu \rangle$ by firing some $t \in T$; (ii) the edge $(\langle m, [\nu] \rangle, \langle m', [\nu'] \rangle)$ may represent the passage of time if either $\langle m', [\nu'] \rangle = succ(\langle m, [\nu] \rangle)$ or $\langle m, [\nu] \rangle = \langle m', [\nu'] \rangle$ is an unbounded region. The function l labels an edge either with the symbol $'t'$ (if the edge represents firing t) or with the symbol $'\delta'$ (if the edge represents the passage of time). It is well-known that the size of the region graph $G(\mathcal{N}^\chi)$ is bounded by $2^{|\mathcal{N}^\chi|}$.

Example 2. Contemplate the PPrTPN $\mathcal{N}_1$ in Fig. 1 and a parameter valuation $\chi_1(\theta_1) = 0$, $\chi_1(\theta_2) = 0$. The region graph $G(\mathcal{N}_1^{\chi_1})$ is shown in Fig. 2.

There is a correspondence between the runs r in $\mathcal{N}^\chi$ and the paths ρ in $G(\mathcal{N}^\chi)$. Let $r = (q_i)_{i\geq 0}$. Consider $q_i \overset{\delta_i}{\Rightarrow} q_{i+1}$. If $\delta_i = 0$ or $[q_i] = [q_{i+1}]$ is an unbounded region, then $([q_i], [q_{i+1}])$ is an edge in $G(\mathcal{N}^\chi)$, according to the definition of $G(\mathcal{N}^\chi)$. If $\delta_i > 0$, then there are positions $\mathbf{p}_j$ $(0 \leq j \leq n_i+1)$ in r such that $q_i = \mathbf{p}_0$, $q_{i+1} = \mathbf{p}_{n_i+1}$, and $[\mathbf{p}_{j+1}] = succ([\mathbf{p}_j])$ for all $0 \leq j \leq n_i$. In this case, the obtained path $\pi(r)$ in $G(\mathcal{N}^\chi)$ corresponds to the run r in $\mathcal{N}^\chi$, and we say that $\pi(r)$ is the path *associated* with r. On the other hand, for any path ρ in $G(\mathcal{N}^\chi)$, we can find a corresponding run in $\mathcal{N}^\chi$ which may be not unique.

5.2. Paths durations

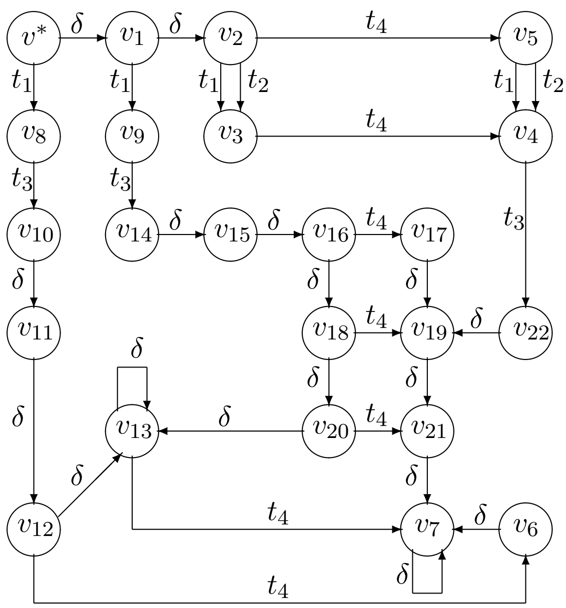
Real Arithmetic (RA) is the set of first-order formulae of $\langle \mathbf{R}, +, <, \mathbb{N}, 0, 1\rangle$, where $\mathbb{N}$ is a unary predicate. The interpretation of $\mathbb{N}$ is defined so that $\mathbb{N}(x)$ holds iff x is a natural number. The RA-formulae are interpreted over the real numbers. The theory of RA is the set of RA-sentences, i.e. formulae without free variables. RA has a decidable theory with complexity in 3ExpTime in the size of the sentence [22].

Consider the definitions of auxiliary sets.

Definition 3. Given a region graph $G(\mathcal{N}^\chi) = (V, E, l)$ with $v, v' \in V$ and $S \subseteq V$, we define:

- $\lambda^\chi_{S,v,v'}$ as the set of $x \in \mathbf{R}$ such that
  - there exists a finite run $r = (q_i)_{0\leq i\leq j}$ in $\mathcal{N}^\chi$ with duration $x = D(r, q_j)$,
  - $v = v_0, v' = v_k$ and $v_l \in S$ $(0 \leq l < k)$ for the path $\pi(r) = (v_l)_{0\leq l\leq k}$ in $G(\mathcal{N}^\chi)$ associated with r,
- $\mu^\chi_{S,v,v'}$ as the set of $\lambda^\chi_{S,v,v'}$ such that $\lambda^\chi_{S,v,v'} \subset \mathbf{N}$ and $D(r, \mathbf{p}) < x$ for any position $q_0 \leq \mathbf{p} < q_j$.

Let $\phi(y)$ be a formula with a single free variable y. A set $Y \subseteq \mathbf{R}$ is *definable* by an RA-formula if Y is the set of all assignments of the variable y making the formula $\phi(y)$ true.



v	m	$[\nu]$				order
		t_1	t_2	t_3	t_4	
v^*	101	0	0	0	0	
v_1	101	$(0,1)$	$(0,1)$	$(0,1)$	$(0,1)$	
v_2	101	1	1	1	1	
v_3	011	1	1	0	1	
v_4	010	1	1	0	1	
v_5	100	1	1	1	1	
v_6	000	1	1	1	1	
v_7	000	>1	>1	>1	>1	
v_8	011	0	0	0	0	
v_9	011	$(0,1)$	$(0,1)$	0	$(0,1)$	
v_{10}	001	0	0	0	0	
v_{11}	001	$(0,1)$	$(0,1)$	$(0,1)$	$(0,1)$	
v_{12}	001	1	1	1	1	
v_{13}	001	>1	>1	>1	>1	
v_{14}	001	$(0,1)$	$(0,1)$	0	$(0,1)$	
v_{15}	001	$(0,1)$	$(0,1)$	$(0,1)$	$(0,1)$	$\{\nu_{15}(t_3)\} <$ $\{\nu_{15}(t_{1,2,4})\}$
v_{16}	001	1	1	$(0,1)$	1	
v_{17}	000	1	1	$(0,1)$	1	
v_{18}	001	>1	>1	$(0,1)$	>1	
v_{19}	000	>1	>1	$(0,1)$	>1	
v_{20}	001	>1	>1	1	>1	
v_{21}	000	>1	>1	1	>1	
v_{22}	000	1	1	0	1	

Figure 2

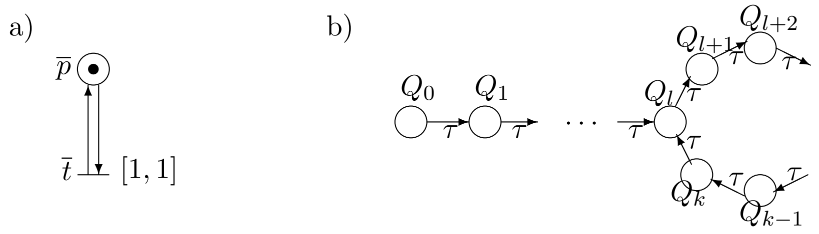


Figure 3

Proposition 1. *Given a region graph $G(\mathcal{N}^\chi) = (V, E, l)$ with $v, v' \in V$ and $S \subseteq V$, the set $\lambda^\chi_{S,v,v'}$ ($\mu^\chi_{S,v,v'}$) is definable by an RA-formula; the construction of the formula is effective.*

Proof. We consider the proof of the case with $\lambda^\chi_{S,v,v'}$ (the proof of the case with $\mu^\chi_{S,v,v'}$ is similar).

In order to prove that the set $\lambda^\chi_{S,v,v'}$ is definable by an RA-formula, we add to $\mathcal{N}$ the net fragment shown in Fig. 3a), where the transition $\overline{t}$ is such that $\overline{t} \succ t$ for all $t \in T$. W.l.o.g. assume $v = \langle \overline{m}, [\overline{\nu}] \rangle$. The new net is denoted by $\overline{\mathcal{N}} = (\overline{P}, \overline{T}, \overline{^\bullet(\cdot)}, \overline{(\cdot)^\bullet}, \overline{m}, \overline{\succ}, \Theta, \overline{I}, \overline{\nu})$. For any $\langle m, \nu \rangle$-run in $\mathcal{N}^\chi$, there is the corresponding $\langle \overline{m}, \overline{\nu} \rangle$-run in $\overline{\mathcal{N}}^\chi$ by adding $\overline{t}$ ($\overline{\nu}(\overline{t}) = 0$). And conversely, for any $\langle \overline{m}, \overline{\nu} \rangle$-run in $\overline{\mathcal{N}}^\chi$, there is the corresponding $\langle m, \nu \rangle$-run in $\mathcal{N}^\chi$ by erasing $\overline{t}$.

Claim A. Let $x \in \lambda^\chi_{S,v,v'}$. If $x \in (c, c+1)$ for some $c \in \mathbf{N}$, then $(c, c+1) \subseteq \lambda^\chi_{S,v,v'}$.
Proof. It is routine to show that the result holds for $\overline{\mathcal{N}}$.

Let $\overline{G}^\chi = (\overline{V}, \overline{E}, \overline{l})$ be a region graph of $\overline{\mathcal{N}}^\chi$. Construct a classical automaton $\widetilde{C} = (\widetilde{V}, \widetilde{E}, \widetilde{l})$ as a subraph of $\overline{G}^\chi$ such that $\widetilde{V} = \{\overline{v}_1 \in \overline{V} \mid v_1 \in S \cup \{v'\}\}$ and $\widetilde{E} = \{(\overline{v}_1, \overline{v}_2) \in \overline{E} \mid (v_1, v_2) \in E \cap (S \times (S \cup \{v'\}))\}$. Any edge from $\widetilde{E}$ is labelled by τ, if it corresponds to a firing of $\overline{t}$, and by ϵ otherwise. The standard subset construction is then applied to $\widetilde{C}$ to get a deterministic automaton $\widetilde{C}'$ without ϵ-edges, which has the structure shown in Fig. 3b).

Suppose $x = \lfloor x \rfloor + y \in \lambda^\chi_{S,v,v'}$ with $y \in [0, 1)$. According to the definition of $\lambda^\chi_{S,v,v'}$, there exists a finite run $r = (q_i)_{0 \le i \le j}$ in $\mathcal{N}^\chi$ such that duration $x = D(r, q_j)$ and $v = v_0$, $v' = v_k$ and $v_l \in S$ $(0 \le l < k)$ for the path $\pi(r) = (v_l)_{0 \le l \le k}$ in $G(\mathcal{N}^\chi)$. Due to the construction of $\widetilde{C}$, the associated path $\pi(r)$ may be represented as a path $\widetilde{\pi}$ of $\widetilde{C}$ starting at its initial state, say σ_i, and ending at its final state, say σ_f. In $\widetilde{C}'$, the path $\widetilde{\pi}$ is represented by the path $\widetilde{\pi}'$ starting at its initial state Q_0 and ending at its state, say Q_m. Clearly, the length of the path is equal to $\lfloor x \rfloor$. Furthermore, if the value of a time assignment for $\overline{t}$ is different from 0 at σ_f, then for any $y' \in (0, 1)$, $x' = \lfloor x \rfloor + y' \in \lambda^\chi_{S,v,v'}$ with the same path $\widetilde{\pi}'$ in $\widetilde{C}'$, due to

Claim A. Therefore, the set $\lambda^{\chi}_{S,v,v'}$ is definable by an RA-formula given by a disjunction of terms like $(x = m)$, $(\exists z(\mathbb{N}(z)) \wedge (x = m + cz))$ or $(m < x < m + 1)$, $(\exists z(\mathbb{N}(z)) \wedge (m + cz < x < m + cz + 1))$, where m, c are natural constants. $\diamondsuit$

5.3. Principles of parametric timing behaviour analysis

In this subsection, we formulate and solve a restricted variant of $\mathcal{PTBA}(\mathcal{N}, \varphi)$. Let $\Omega \subseteq \mathbf{R}^{\Theta_{\mathcal{N}}}$ be a convex polyhedron that is the domain of the parameters from $\Theta_{\mathcal{N}}$, and $\Omega_{\mathbf{N}}$ be the set of the natural valued points of Ω, that is finite and can be defined by using standard techniques. We restrict ourselves to constructing a symbolic representation of parameter valuations on Θ_φ, which belong to $\Omega_{\mathbf{N}}$ on $\Theta_{\mathcal{N}}$, and denote the restricted problem as $\mathcal{PTBA}(\mathcal{N}_\Omega, \varphi)$. Define an equivalence relation $\approx$ on the set $\Upsilon = \{\chi \mid \chi|_{\Theta_{\mathcal{N}}} \in \Omega_{\mathbf{N}}\}$ as follows: $\chi_1 \approx \chi_2$ iff $\chi_1(\theta) = \chi_2(\theta)$, for all $\theta \in \Theta_{\mathcal{N}}$. Let $\Upsilon_\approx$ denote the set of $\approx$-equivalent classes of Υ, and $\gamma \in \Upsilon_\approx$. Clearly, for each $\chi \in \gamma$ we have the same $\mathcal{N}^\chi$ (resp. $\lambda^{\chi}_{S,v,v'}$), so we can denote it as $\mathcal{N}^\gamma$ (resp. $\lambda^{\gamma}_{S,v,v'}$). To symbolically represent parameter valuations on Θ_φ, we construct for each $\gamma \in \Upsilon_\approx$ an RA-formula $\Delta(\varphi, v^*, \gamma)$ with free variables $\theta_1, \ldots, \theta_k \in \Theta_\varphi$, such that $\mathcal{N}^\chi, v^* \models \varphi^\chi$ for some valuation $\chi \in \gamma$ iff the sentence $\exists\theta_1 \ \ldots \ \exists\theta_k \Delta(\varphi, v^*, \chi)$ is true. The approach is correct because RA has a decidable theory and $\Upsilon_\approx$ is a finite set. The main instrument of the approach is to describe by an RA-formula, for two given regions $v = [q]$ and $v' = [q']$ in $G(\mathcal{N}^\gamma)$, all the possible values of duration $D(r, q_j)$ for finite runs r from q to q' in $\mathcal{N}^\gamma$. For a region v of $G(\mathcal{N}^\gamma)$ and a PTCTL-formula φ, the construction of $\Delta(\varphi, v, \gamma)$ is easily performed by induction on the length of φ.

Theorem 1. *Given $\gamma \in \Upsilon_\approx$, a region v of $\mathcal{N}^\gamma$, a PTCTL-formula φ with $\Theta_\varphi = \{\theta_1, ..., \theta_k\}$, there exists an RA-formula $\Delta(\varphi, v, \gamma)$ such that $\mathcal{N}^\chi, v \models \varphi^\chi$ for some valuation $\chi \in \gamma$ iff the sentence $\exists\theta_1 ... \exists\theta_k \Delta(\varphi, v, \gamma)$ is true. The construction of $\Delta(\varphi, v, \gamma)$ is effective.*

Proof. The structure of $\Delta(\varphi, v, \gamma)$ is defined by induction on the length of the formula φ. The set of free variables of $\Delta(\varphi, v, \gamma)$ is equal to Θ_φ.

As shown in [7], we can work with the simpler grammar $\phi ::= \mathcal{P} \mid \neg\phi \mid \phi \vee \phi \mid \phi \exists U_{\sim\alpha} \phi \mid \exists\Box_{\sim\alpha}\phi$. (The construction for the case with $\varphi = Q\ \theta\ \phi$ is immediate.)

Construct $\Delta(\varphi, v, \gamma)$ for the first three cases:

$$\begin{array}{ll}
\varphi = \mathcal{P}: & \Delta(\varphi, v, \gamma) = \mathit{true} \iff \mathcal{P} \in \mathcal{W}(q) \\
\varphi = \neg\psi: & \Delta(\varphi, v, \gamma) = \neg\Delta(\psi, v, \gamma) \\
\varphi = \psi \vee \phi: & \Delta(\varphi, v, \gamma) = \Delta(\psi, v, \gamma) \vee \Delta(\phi, v, \gamma)
\end{array}$$

Consider the case $\varphi = \psi \exists U_{\sim\alpha} \phi$. Suppose $\mathcal{N}^\chi, q \models \varphi^\chi$. According to the definition, for some q-run $r = (q_i)_{i \geq 0}$ in $\mathcal{N}^\chi$ with $q = q_0$, there exists a position $\mathbf{p}$ in r s.t. $D(r, \mathbf{p}) \sim \chi(\alpha)$, $\mathcal{N}^\chi, \mathbf{p} \models \phi^\chi$ and $\mathcal{N}^\chi, \mathbf{p}' \models \psi^\chi$ for all positions $\mathbf{p}'$ in r s.t. $\mathbf{p}' < \mathbf{p}$. If $\mathbf{p} = q$, then we have $\mathcal{N}^\chi, q \models \phi^\chi$. Otherwise, assume $x = D(r, \mathbf{p})$. Consider the path $\pi(r) = (v_k)_{k \geq 0}$ in $G(\mathcal{N}^\chi) = (V, E, l)$. Then, we get $v_0 = v$ and $v_l = [\mathbf{p}]$ for some $l > 0$. Let $v' = v_l$ and $S = \{v_k \mid 0 \leq k < l\}$. Thus, it holds: $x \in \lambda^\chi_{S,v,v'}$, $\mathcal{N}^\chi, v' \models \phi^\chi$ and $\mathcal{N}^\chi, s \models \psi^\chi$ for any $s \in S$. The formula $\neg B(v') \to \Delta(\psi, v', \gamma)$ means that $B(v')$ is true iff v' is a boundary region. Note, if v' is not a boundary region, then $\mathcal{N}^\chi, v' \models \psi^\chi$. Thus, for $\Delta(\varphi, v, \gamma)$, we get the following:

$$\Delta(\psi \exists U_{\sim\alpha} \phi, v, \gamma) = (\Delta(\phi, v, \gamma) \wedge (0 \sim \alpha)) \vee$$

$$\bigvee_{v' \in V} \bigvee_{S \subseteq V} (\exists x \sim \alpha \ \lambda^\chi_{S,v,v'}(x) \wedge \Delta(\phi, v', \gamma) \wedge \bigwedge_{s \in S} \Delta(\psi, s, \gamma) \wedge$$

$$\wedge (\neg B(v') \to \Delta(\psi, v', \gamma))).$$

Here, $\lambda^\chi_{S,v,v'}(x)$ denotes an RA-formula defining the set $\lambda^\chi_{S,v,v'}$. Application of χ to the set Θ_φ of free variables of $\Delta(\varphi, v, \gamma)$ gives a sentence which is true in RA. Conversely, it is easy to see that if $\exists \theta_1 ... \exists \theta_k \ \Delta(\psi \exists U_{\sim\alpha} \phi, v, \gamma)$ is true, then it holds $\mathcal{N}^\chi, v \models \varphi^\chi$, due to Lemma 1.

Reasoning analogously for the remaining cases, we get the below table as a result:

$\exists \Box_{\geq\alpha} \psi$:	$\bigvee_{v' \in V} \bigvee_{S \subseteq V} (\mu^\chi_{V,v,v'}(\alpha) \wedge Path_S(v') \wedge \bigwedge_{s \in S} \Delta(\psi, s, \gamma))$
$\exists \Box_{<\alpha} \psi$:	$\bigvee_{v' \in V} \bigvee_{S \subseteq V} (\mu^\chi_{S,v,v'}(\alpha) \wedge \bigwedge_{s \in S} \Delta(\psi, s, \gamma) \wedge$ $\wedge (\neg B(v') \to \Delta(\psi, v', \gamma)))$
$\exists \Box_{\leq\alpha} \psi$:	$\bigvee_{S \subseteq V} \bigvee_{v', v'' \in V} (\lambda^\chi_{S,v,v'}(\alpha) \wedge \bigwedge_{s \in S} \Delta(\psi, s, \gamma) \wedge E(v', v''))$
$\exists \Box_{>\alpha} \psi$:	$\bigvee_{S \subseteq V} \bigvee_{v', v'' \in V} (\lambda^\chi_{V,v,v'}(\alpha) \wedge E(v', v'') \wedge \bigwedge_{s \in S} \Delta(\psi, s, \gamma) \wedge$ $\wedge \ Path_S(v'') \wedge (\neg B(v') \to \Delta(\psi, v', \gamma)))$
$\exists \Box_{=\alpha} \psi$:	$\bigvee_{S \subseteq V} \bigvee_{v', v'', v''' \in V} (\mu^\chi_{V,v,v'}(\alpha) \wedge \lambda^\chi_{S,v',v''}(0) \wedge$ $\wedge \bigwedge_{s \in S \cup \{v''\}} \Delta(\psi, s, \gamma) \ E(v'', v''')$

Here, the predicate $Path_S(v)$ is true iff v belongs to a path in $G(\mathcal{N}^\chi)$ with all its vertices in S and its first vertex equal to v. Also, the predicate $E(v', v')$ is true iff (v', v'') is an edge labelled with $\delta > 0$ in $G(\mathcal{N}^\chi)$. $\diamondsuit$

Theorem 2. *There exists a procedure for solving $\mathcal{PTBA}(\mathcal{N}_\Omega, \varphi)$ which is in 2ExpTime in the product of the sizes of $\mathcal{N}$ and φ.*

Proof. First, RA-formulae $\lambda^\chi_{S,v,v'}(x)$ and $\mu^\chi_{S,v,v'}(x)$ have a size and can be constructed in time bounded by $O(2^{2 \cdot |G(\mathcal{N}^\chi)|})$, due to Proposition 24 [7]. Second, $\Delta(f, g, \gamma)$ has a size and can be constructed in time bounded by

$O(2^{7\cdot|G(\mathcal{N}^\chi)|\cdot|\varphi|})$, by proposition 25 [7]. Third, the size of $G(\mathcal{N}^\chi)$ is bounded by $O(|T!| \cdot 2^{2|T|+|P|})$ as shown in [19]. Fourth, the size of of the set Υ_c is bounded by $O((c+1)^{|\Theta_\mathcal{N}|})$. Thus, there exists a procedure for solving $\mathcal{PTBA}(\mathcal{N}_c, \varphi)$ which is in 2ExpTime in the product of the sizes of $\mathcal{N}$ and φ. $\diamondsuit$

Example 3. Consider the PPrTPN $\mathcal{N}_1$ in Fig. 1. We assume $\Omega : \theta_1 = 0$, $0 \leq \theta_2 \leq 1$ and $\gamma = \{\chi \mid \chi(\theta_1) = 0, \chi(\theta_2) = 0\} \in \Upsilon_\approx$. Also, contemplate the PTCTL-formula $\varphi = \forall\ \Box_{>\theta}\ (m(p_2) = 0\ \vee\ m(p_3) = 0)$. Applying standard transformations, we get $\varphi = \neg(true \exists\ U_{>\theta}(m(p_2) > 0 \wedge m(p_3) > 0))$. Using the reasonings in the proof of Theorem 1, $\Delta(true\exists U_{>\theta}(m(p_2) > 0 \wedge m(p_3) > 0), v^*, \gamma) = [\Delta(m(p_2) > 0 \wedge m(p_3) > 0, v^*, \gamma) \wedge (0 > \theta)] \vee \bigvee_{v' \in V} \bigvee_{S \subseteq V} [\exists x > \theta \lambda^\gamma_{S,v^*,v'}(x) \wedge \Delta(m(p_2) > 0 \wedge m(p_3) > 0, v', \gamma) \wedge \bigwedge_{s \in S} \Delta(true, s, \gamma) \wedge (\neg B(v') \rightarrow \Delta(true, v', \gamma))]$. One can see that $\Delta(m(p_2) > 0 \wedge m(p_3) > 0, v', \gamma)$ is true only for $v' = v_3$, $v' = v_8$ and $v' = v_9$ (see Fig. 2). Then, $\lambda^\gamma_{S,v^*,v_3}(x) =$ "$x = 1$", $\lambda^\gamma_{S,v^*,v_8}(x) =$ "$x = 0$", $\lambda^\gamma_{S,v^*,v_9}(x) =$ "$0 < x < 1$", for all $S \subseteq V$ such that $\lambda^\gamma_{S,v^*,v'} \neq \emptyset$. Thus, we have $\Delta(true\exists U_{>\theta}(m(p_2) > 0 \wedge m(p_3) > 0), v^*, \gamma) = \exists x > \theta\ 0 \leq x \leq 1$. So, $\Delta(\forall \Box_{>\theta}(m(p_2) = 0\ \vee\ m(p_3) = 0), v^*, \gamma) = \neg(\exists x > \theta\ \ 0 \leq x \leq 1)$, i.e. $\theta \geq 1$. For the other possible $\gamma \in \Upsilon_\approx$, the results are obtained analogously.

6. Conclusion

In this paper, we have extended the model checking algorithm for TA w.r.t. PTCTL-formulae from the paper [7] to the setting of PrTPNs with parameters. To fulfill the purpose, we have introduced a notion of the parametric PrTPN (PPrTPN) which is a modification of the PrTPN by using parameter variables in specification of timing constraints on transition firings and have developed a timing behaviour analysis algorithm which consists in constructing conditions on free parameters of the checked PTCTL-formula under which the PrTPN with bounded parameters works w.r.t. the formula. Formulae of the real arithmetic have been used to express the durations of paths between vertices of the region graphs of the PPrTPN at some parameter valuation. The real arithmetic based technique has turned out to be much simpler and cleaner than the cactus technique used for the same purpose in the papers [20, 18]. It is worth noticing that the version of PPrTPNs introduced in this paper makes use of static priorities. We see nothing to prevent replacing them by more flexible dynamic priorities depending on net markings.

We conclude the paper by pointing out some possible research directions for the future. First, we plan to investigate the applicability of the available state space abstractions taking advantage of the paper [6] in order to reduce the state space of PPrTPNs. Second, the adoption of PPrTPNs with unbounded parameters is also desirable. Third, we intend to exploit the approach from [11] as part of our future work to develop parametric model checking for hybrid Petri nets.

References

[1] Alur R., DillD. The theory of timed automata // Theor. Comput. Sci. – 1994. – Vol. 126. – P. 183–235.

[2] Alur R., Etessami K., La Torre S., Peled D. Parametric temporal logic for "model measuring" // Proc. ICALP'99. – Lecture Notes in Computer Science. – 1999. – Vol. 1644. – P. 159–168.

[3] Alur R., Henzinger T.A., Vardi M.Y. Parametric real-time reasoning // Proc. STOC'93. – ACM Press, 1993. – P. 592–601.

[4] Bérard B., Cassez F., Haddad S., Lime D., Roux O. H. Comparison of the expressiveness of timed automata and time Petri nets // Lect. Notes Comput. Sci. – 2005. – Vol. 3829. – P. 211–225.

[5] Berthomieu B., Peres F., Vernadat F. Bridging the gap between timed automata and bounded time petri nets // Lect. Notes Comput. Sci. – 2006. – Vol. 4202. – P. 82–97.

[6] Berthomieu B., Peres F., Vernadat F. Model checking bounded prioritized time Petri nets // Lect. Notes Comput. Sci. – 2007. – Vol. 4762. – P. 523–532.

[7] Bruyère V., Dall'olio E., Raskin J.-F. Durations, parametric model-checking in timed automata with Pressburger arithmetic // Lect. Notes Comput. Sci. – 2003. – Vol. 2607. – P. 687–698.

[8] Bruyère V., Raskin, J.-F. Real-time model-checking: Parameters everywhere // Logical Methods in Computer Science. – 2007. – Vol. 3 (1:7). – P. 1–30.

[9] Comon H., Jurski Y. Timed automata and the thory of real numbers // Lect. Notes Comput. Sci. – 1999. – Vol. 1664. – P. 242–257.

[10] Emerson E., Trefler R.J. Parametric quantitative temporal reasoning // Proc. LICS'99. – 1999. – P. 336–343.

[11] Henzinger T.A., Ho P.-H., Wong-Toi H. HyTech: a model checker for hybrid systems // Int. J. on Software Tools for Technology Transfer. – 1997. – Vol 1(1-2). – P. 110-122.

[12] Hune T., Romijn J., Stoelinga M., Vaandrager F. Linear parametric model checking of timed automata // J. of Logic and Algebraic Programming. – 2002. – Vol. 52–53. – P. 183–220.

[13] Bruyere V., Dall'Olio E., Raskin J.-F. Durations, parametric model-checking in timed Automata with Pressburger Arithmetic // Lect. Notes Comput. Sci. – 2003. – Vol. 2604. – P. 687–698.

[14] Henzinger T.A. It's about time: real-time logics reviwed // Lect. Notes Comput. Sci. – 1998. – Vol. 1466. – P. 439–454.

[15] Merlin P., Faber D.J. Recoverability of communication protocols // IEEE Trans. of Communication. – 1996. – Vol. COM-24(9).

[16] Ramchandani C. Analysis of asynchronous concurrent systems by timed Petri nets / PhD Thesis. – Cambridge, Mass.: MIT, Dept. Electronical Engineering, 1974.

[17] Traonouez L.-M., Lime D., Roux O.H. Parametric model-checking of time Petri nets with stopwatches using the state-class graph // Lect. Notes Comput. Sci. – 2008. – Vol. 5215. – P. 280–294.

[18] Virbitskaite I.B., Pokozy E.A. Parametric behaviour analysis for time Petri nets // Lect. Notes Comput. Sci. – 1999. – Vol. 1662. – P. 134–140.

[19] Virbitskaite I.B., Pokozy E.A. A Partial Order Method for the Verification of Time Petri Nets // Proc. FCT'99. – Lect. Notes Comput. Sci. – 1999. – Vol. 1684. – P. 547–558.

[20] Wang F. Parametric timing analysis for real-time systems // Information and Computation. – 1996. – Vol. 130. – P. 131–150.

[21] Wang F., Hsiung P.-A. Parametric analysis of computer systems // Proc. AMAST'97. – 1997. – P. 539–553.

[22] Weispfenning Mixed real-integer linear quantifier elimination // Proc. of ACM SIGSAM International Symposium on Symbolic and Algebraic Computation, 1999.