Joint NCC & IIS Bull., Comp.Science, 5 (1996), 23–41

Reachability analysis for time Petri nets without overlappings of firing intervals*

E.V. Okunishnikova

This paper discusses a subclass of Merlin's time Petri nets called here time Petri nets without overlappings of firing intervals. In addition to the existing enumerative procedure for time nets, a new technique of reachability analysis for nets in the subclass is presented. Correctness of the presented method is proved. The sufficient condition of the boundedness property is formulated.

Introduction

The basic Petri net model [8] is widely used to specify a large class of concurrent systems which can be represented by abstracting away from time aspects. Several factors contribute to success of Petri nets: a clear pictorial representation of the structure of the designed systems, the possibility of specifying the systems at different levels of abstraction as well as methods and algorithms proposed for analyzing the behavioural properties of designed systems. Nevertheless, the basic Petri net model is not suitable for modelling a lot of systems whose behaviour is based on explicit temporal parameters. Examples of such systems are communication protocols, most of them highly depend on time for reliability or performance aspects.

Several authors have extended the basic Petri net model with timing constraints. These timed models can be conditionally divided in two classes: with fixed and variable delays. Petri nets with fixed delays [15, 16, 17] allow for simple analysis methods but are not very expressive, because durations of most activities are variable in many of the real-time systems. Extensions [1, 5, 9, 10] which model this variability are more interesting. In the paper, we deal with an extension introduced by Merlin [9].

Merlin's time Petri nets (TPN's for short) use delays specified by an interval. The delay of a transition firing is specified by its minimum and maximum value. These nets are very difficult to analyse. In general, there exists an infinite number of possible firing times, and firing of a transition induces some constraints on the transitions that remain enabled. A reach-

*This work is supported by Russian Basic Research Fund (Project 93-01-00986), Intern. Association for the Promotion of Cooperation with Scientists from the Independent States of the Former Soviet Union (INTAS) under contract 1010-CT93-0048 and by grant JCP100 from International Science Foundation and Russian Government.

ability analysis technique for time Petri nets was presented in [2, 3], but it is rather complex.

In the paper, we consider the subclass of time nets introduced in [11]. We call this subclass a TPN's without overlappings of firing intervals. The subclass is specifed by two restrictions on the dynamical behaviour of Merlin's time nets.

The main purpose of the paper is to propose a new reachability analysis technique for time nets in the subclass. Specific features of time nets without overlappings of firing intervals allow us to present a method of reachability analysis which is much simpler than the enumerative procedure for usual time nets.

1. Time Petri Nets

We use the following definitions.

Definition 1. An *ordinary Petri net* is a 4-tuple $N = (P, T, F, M_0)$, where P and T are the node sets of a directed bipartite graph, , F is the arc set, and M_0 assigns a nonnegative integer to each element of P.

The sets P and T are called the set of *places* and the set of *transitions*, respectively. F is called the *flow relation* or the *incidence function*, and M_0 is called the *initial marking* of N.

The predecessors of a place p (transition t) are called its *input transitions (input places)* and are denoted by ${}^\bullet p$ (${}^\bullet t$). Also, the successors of a place p (transition t) are called its *output transitions (output places)* and are denoted by $p^\bullet$ ($t^\bullet$).

Definition 2. A transition t is *enabled* in the net N by a given marking M when all of its input places have at least one token.

An enabled transition t can fire. When this happens, a token is removed from each input place of t and a token is added to each output place. This defines a new marking.

1.1. Definition of time Petri nets

We use the definition of time Petri nets given in [2, 3].

Definition 3. *A Time Petri Net* is a 3-tuple $NT = (N, \mathcal{T}, SI)$, where:

1) N is a ordinary Petri net.
2) $\mathcal{T}$ is the time set totally ordered by the relation $\leq$. $Interv(\mathcal{T})$ represents the set of all closed left bounded intervals.
3) $SI: T \to Interv(\mathcal{T})$ is a *static firing interval function.*

Note that this definition does not restrict the set of times. Time can be either discrete or continuous.

The function SI associates two times with each transition, $SI(t_i) = [\alpha_i^s, \beta_i^s]$ for a transition t_i. We call the interval $[\alpha_i^s, \beta_i^s]$ the *static firing interval* of the transition t_i, the left bound α_i^s the static *earliest firing time* (static EFT for short), and the right bound β_i^s the static *latest firing time* (static LFT for short).

The class of all time Petri nets will be denoted by $\mathcal{N}$.

Definition 4. A transition t is *enabled* in a time net NT by a marking M iff it is enabled by M in the Petri net N (in the net without time restrictions). The set of enabled transitions is denoted by $enabled(M)$.

Some transitions may be enabled by the marking M, but not all of them may be allowed to fire due to the firing constraints of transitions (EFT's and LFT's).

Definition 5. *States* in TPN's will be pairs $S = (M, I)$ in which:

1) M is a marking;
2) I : $enabled(M) \rightarrow Interv(T)$ is a *firing interval function*. I associates the time interval in which a transition is allowed to fire with each enabled transition.

It will appear that for states other than the initial state, firing intervals are in general case different from the static firing intervals. Their lower bounds will be called EFT and their upper bounds LFT, written as α_i and β_i, respectively.

Times α_i^s and β_i^s as well as times α_i and β_i are relative to the moment at which the transition t_i is enabled. If τ is an absolute time at which the transition is enabled, then it can fire in the interval $[\tau + \alpha_i^s, \tau + \beta_i^s]$ ($[\tau + \alpha_i, \tau + \beta_i]$) unless it is disabled before $\tau + \beta_i^s$ ($\tau + \beta_i$) by firing of another transition. In other words, t_i may not fire, while being continuously enabled, before $\tau + \alpha_i^s$ or $\tau + \alpha_i$ and should fire before or at time $\tau + \beta_i^s$ or $\tau + \beta_i$ at the latest.

Firing of a transition is an instantaneous event and "takes no time": firing of a transition at time τ leads to a new state defined at the same time τ.

It should be noted that in the paper we consider only TPN's such that none of their transitions may become enabled more than once "simultaneously". This means that for any marking M and for any enabled transition t the following holds: $\exists p : M(p) < 2 * F(p, t)$, i.e., there is at least one place which prevents t to be firable twice. TPN's with this property are called *T-Safe TPN's*.

Let us denote the smallest of the LFT of all enabled transitions by $deadline(S)$.

Definition 6. A transition t_i is *firable* in the TPN NT at a time τ from a state $S = (M, I)$ iff both of the following conditions hold:

1) $t_i \in enabled(M)$;
2) $\alpha_i \leq \tau \leq deadline(S)$.

Note that 2) holds because the transition which has the minimum LFT should fire at the time $deadline(S)$. Firing of this transition modifies the marking and the state of TPN. We will use $firable(S)$ to denote the set of transitions which are firable from the state S at some time from their firing intervals.

1.2. Firing rule between states

Firing of a transition t_i at a time τ from a state $S_1 = (M_1, I_1)$ leads to a new state $S_2 = (M_2, I_2)$ computed as follows:

1) a new marking M_2 is defined for all places as:
   $M_2(p) = M_1(p) - F(p, t_i) + F(t_i, p)$, as usually in Petri nets;
2) a new value of the firing interval function I_2 is computed as follows:
   a) a new firing interval is empty for all transitions which are not enabled by the marking M_2:
      $I_2(t_j) = \emptyset \quad \forall\ t_j \notin enabled(M_2)$;
   b) if transition is enabled by the marking M_1 and is not in conflict with t_i, then its interval is shifted by the value of τ towards the time origin (restricted to nonnegative values):
      $I_2(t_j) = [max\ (0,\ \alpha_j - \tau),\ \beta_j - \tau]$
      $\forall\ t_j \in enabled(M_1) \cap enabled(M_2) \quad \& \quad {}^\bullet t_i \cap {}^\bullet t_j = \emptyset$;
   c) all other transitions have their intervals set to their static firing intervals:
      $I_2(t_j) = SI(t_j)$.

The firing time τ is relative to the moment at which the state S_1 has been reached. It can be seen as given by a virtual clock, local to transition, which should have the same time value as clocks of the other transitions in the net. The absolute firing time can be defined (when needed) as "τ + the absolute time at which the state S_1 has been reached".

The firing rule above defines the reachability relation among the states of time Petri nets.

Definition 7. The state S_2 is said to be *directly reachable* from S_1 by the firing of transition t_i. This is also denoted by $S_1\ [\ t_i\rangle\ S_2$.

A state S_n is *reachable* from S_1 iff there exists a sequence of transitions $\sigma = t_{i_1} \ldots t_{i_{n-1}}$ such that $S_1[(t_{i_1}, \tau_1)\rangle \ldots [(t_{i_{n-1}}, \tau_{n-1})\rangle S_n$, where τ_j is the firing

time of the transition t_{i_j}. This is denoted by $S_1 [\sigma\rangle S_n$ or by $S_1 [\rangle S_n$ when the sequence is not taken into account.

A sequence of pairs $(t_{i_1}, \tau_1) \ldots (t_{i_n}, \tau_n)$ is called a *firing schedule* (FS) which is firable from the state S_1 and leads to the state S_n.

The sequence of successively firable transitions $t_{i_1} \ldots t_{i_{n-1}}$ is called a *firing sequence.*

Since a transition may fire at any time from its firing interval, the firing sequence $t_{i_1} \ldots t_{i_{n-1}}$ may lead not only to the state S_n. So, several firing schedules may correspond to the same firing sequence.

The behaviour of a TPN is characterized by the set of states reachable from the initial state or by the set of firing schedules feasible from its initial state. Unfortunately, representing the behaviour of a TPN by its reachable states is generally impossible. This is due to the fact that the time may be continuous and then transitions may fire at any time in their allowed intervals. In this case the state has an unbounded number of successors.

1.3. State classes and the enumerative method

A general approach to analysis of the behaviour of TPN's has been presented in [2] and [3].

A state is reached from the initial state by firing of some firing schedule which corresponds to a firing sequence σ. All feasible firing schedules corresponding to the firing sequence σ define a set of states which are reachable by firing of σ. It was proposed in [2, 3] to consider this set of states as the state class associated with the firing sequence σ.

Definition 8. The *state class* associated with a firing sequence σ is a pair $C = (M, D)$, where:

1) M is a marking of the class, all states in the class have the same marking;
2) D is a *firing domain* of the class expressed as a solution set for the following system of linear inequalities:
   $\alpha_i \leq x_i \leq \beta_i \ \forall\, t_i \in enabled(M)$,
   $x_i - x_j \leq \gamma_{ij} \ \forall\, t_i, t_j \in enabled(M)$ with $t_i \neq t_j$.

The initial state class is defined as the class containing the initial state.

Observe that time constraints of a state may be also defined as a solution set of a system of inequalities.

The system with the above form is in a canonical form iff α_i is the smallest possible value of the variable x_i, β_i is the largest possible value of the variable x_i, and γ_{ij} is the largest possible value of the difference $x_i - x_j$.

In practice, it is interesting to compute recursively the set of classes, i.e. to derive the class associated with the sequence $\sigma.t$ from the class associated

with the sequence σ reached by firing of the transition t.

Definition 9. A transition t_i is *firable* from a class $C = (M, D)$ iff both of the following conditions hold:

1) t_i is enabled by the marking M;
2) there is a vector in the domain D whose component corresponding to the transition t_i is not greater than any other component. This is true iff the following system of inequalities is consistent:

$$\begin{aligned} &\alpha_j \le x_j \le \beta_j && \forall\ t_j \in \mathit{enabled}(M), \\ &x_j - x_k \le \gamma_{jk} && \forall\ t_j, t_k \in \mathit{enabled}(M) \text{ with } t_j \ne t_k, \\ &x_i \le x_j && \forall\ t_j \in \mathit{enabled}(M) \text{ with } t_j \ne t_i. \end{aligned}$$

Firing of the transition t_i from the a state class $C_1 = (M_1, D_1)$ associated with the sequence σ leads to a new state class $C_2 = (M_2, D_2)$ associated with the sequence $\sigma.t_i$. The class $C_2 = (M_2, D_2)$ reached from the class $C_1 = (M_1, D_1)$ by firing of the transition t_i is computed as follows:

1) a new marking M_2 is defined as in Petri nets;
2) a new domain D_2 is computed from the domain D_1 by a four-step procedure [2, 3, 11]. Details of the method are not given here.

Definition 10. Two classes are called *equal* iff both of their markings are equal and their firing domains are equal.

Since the domains are computed in the canonical form, comparison for equality can be done efficiently.

The reachability relation defined by the above firing rule allows us to build a tree of state classes: its root is an initial class and there is an arc labelled with a transition t going from a class C_i to a class C_j iff the transition t is firable from C_i and its firing leads to the class C_j. It follows from the definition of the classes that each class can have only a bounded number of successors.

1.4. Some properties of TPN's

We denote $R(M_0)$ the set of markings of a TPN which can be reached from its initial marking M_0.

The Reachability problem is whether or not a given marking belongs to $R(M_0)$.

The Boundedness problem is whether or not all markings in $R(M_0)$ are bounded, i.e. are such that all of their components are smaller than some integer constant K.

The following properties of TPN's are well known:

1. The Reachability and Boundedness problems for time Petri nets are undecidable [7].

2. If static EFT's and LFT's for all transitions are chosen among rational numbers, then the number of the state classes of a TPN is bounded if and only if the net is bounded [3].

The following sufficient condition for the boundedness property provides a sufficient condition for the finiteness of the set of classes [3].

3. A TPN is bounded if there is no a pair of the state classes $C_1 = (M_1, D_1)$ and $C_2 = (M_2, D_2)$ reachable from its initial state class such that:

   1) C_2 is reachable from C_1;
   2) $M_2 \geq M_1$ and $M_2 \neq M_1$;
   3) $D_1 = D_2$.

This condition is not necessary, but can be used to stop enumeration of the classes if the behavior of the net is not the one expected. When TPN is bounded, its graph of state classes allows checking the properties, such as liveness properties, that characterize its correct behavior.

2. Modelling the Time Communication Protocol

The analysis of the Alternating Bit Protocol (ABP for short) is a wellknown example [2, 3] of using TPN's for modelling and verification of the time dependent protocol. The ABP uses timing constraints in its specification: a recovery mechanism for losses of messages is implemented using timeouts.

This protocol transmits messages between two processes, a Sender and a Receiver, allowing only one message in transit at a time. The protocol is a stop-and-wait data transfer protocol. The Sender waits for the acknowledgment of the last message before sending a new message. Hypotheses on the behavior of the environment are that messages or acknowledgments may be lost during the transmission. Recovering from losses is done using a timeout and retransmitting: the Sender records the time at which it sends a message and if its acknowledgment does not return within a given time, the message is retransmitted.

Messages are numbered prior to the transmission with modulo-2 sequence numbers. This allows the Receiver to decide whether the next message it receives is a new message or a duplicate of the last received message when an acknowledgment was lost. The Receiver waits for a message with a particular sequence number. If a message with the correct sequence number arrives, the Receiver returns the acknowledgment with the same sequence number. Thereafter both Sender and Receiver change the sequence number. If a message has the wrong sequence number, the Receiver sends the acknowledgment with the sequence number of the received message.

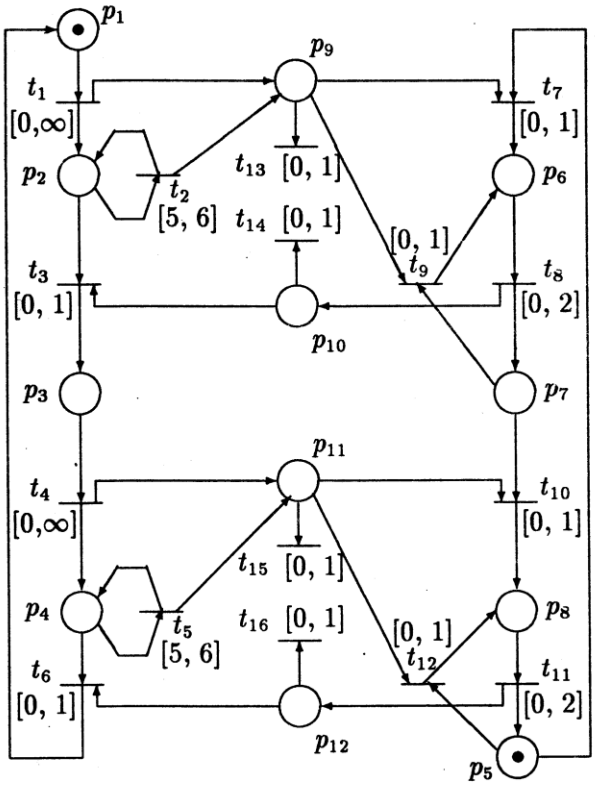

Figure 1

Figure 1 gives a TPN model for the ABP. There are no time constraints, i.e. the intervals $[0, \infty]$, are given for sending the first copies of the numbered messages. Equal estimates (between 0 and 1) are given for losses and reception of the messages and acknowledgments. Retransmission of the message occurs at a time comprised between 5 and 6 units after the last copy of the message has been sent.

The following meanings are given for the transitions:

t_1 (t_4) — Send Packet 0 (1)
t_2 (t_5) — Resend Packet 0 (1)
t_3 (t_6) — Receive Ack 0 (1)
t_{13} (t_{15}) — Lose Packet 0 (1)
t_7 (t_{10}) — Receive and Release Packet 0 (1)
t_9 (t_{12}) — Receive and Reject Packet 0 (1)
t_8 (t_{11}) — Send Ack 0 (1)
t_{14} (t_{16}) — Lose Ack 0 (1)

This net is bounded and live. The reachability graph for this one contains 16 classes [3]. It is clear from these classes that only one message or acknowledgment will be in transmit at a time (all places in the net hold at most one token in any marking). This assures that the retransmission timeout is correctly set. Furthermore, no duplicate message may be released

because the transitions t_7 and t_{10} alternate along all paths of the graph. The transfer of messages actually occurs (the net is live).

3. Time Petri nets without overlappings of firing intervals

It is clear that TPN's are not easy to analyze. This was the reason for introducing the subclass of TPN's [11]. When introducing it, we have a purpose to present a subclass of TPN's whose analysis would be a more simple problem. The subclass has been proposed to model timeout protocols. It has been defined in conformity with a formal model of protocols solving the sequence transmission problem which was proposed by Halpern and Zuck in [6]. Since any structural restrictions are not admissible in modelling of real systems, this subclass is defined by restrictions on the set of enabled transitions which should be fulfilled in any reachable state. The obtained subclass is called time Petri nets without overlappings of firing intervals.

3.1. The definition of TPN's without overlappings of firing intervals

The idea is that a net from the subclass should satisfy the following restrictions:

- firing intervals of two enabled transitions either do not intersect or are equal in any reachable state;
- if two transitions have equal firings intervals in some state, then they should be in a "fair" conflict. The notion of a fair conflict means the following. If both transitions are firable in some reachable state, then in any state when one of them is firable, the other is also firable. The sets of input places may be unequal.

  But if there is no transition in the net whose EFT is equal to its LFT, then it is sufficient to require that simultaneously enabled transitions are in a usual conflict.

The first condition means that if two transitions are simultaneously enabled, then none of them has preference and can fire earlier. The second one tells us that the net is concurrence free: only one of the simultaneously enabled transitions can fire. Unlike untimed concurrence free Petri nets, only simultaneously enabled transitions should be conflicted. The nets which satisfy these conditions behave like state-machine nets with one token. These nets cannot model the systems with concurrent events but can describe timeouts.

A similar approach to the definition of a subclass of Merlin's time nets has been used in [14]. This paper introduces the subclass with the property: if two transitions are enabled in some state, then they are in conflict and have the same EFT. Unlike TPN's without overlappings of firing intervals, in this subclass two transitions cannot be enabled if LFT of one of them is smaller than EFT of the other. Another way to propose a subclass of TPN's has been chosen in [4]. This paper defines simple time Petri nets which are safe (with at most one token per place in any state) Merlin's nets with equal EFT and LFT for any transition. Such a property essentially simplifies the analysis but turns time nets into nets with fixed delays and, as a result, strongly decreases the expressiveness of the model.

The net which models ABP in Fig. 1 is a TPN without overlappings of firing intervals.

In [11], we use a notion of absolute (global) time to more clear define TPN's without overlappings of firing intervals. The time $\tau = 0$ is associated with the marking M_0. The origin of times is not shifted after each firing in the firing time. It should be obvious that the set of firable transitions in any state of the TPN's without overlappings of firing intervals is either one transition or the set of transitions which are in conflict with each other. Hence, EFT of any transition enabled after next firing will be not smaller than the firing time. So, it is unimportant to compute new firing intervals with respect to the origin of the global time scale or with respect to the time of the last firing.

Definition 11. A TPN $NT = (N, T, SI)$ is a *time Petri net without overlappings of firing intervals* iff it satisfies the following requirements:

R1. If transitions t_i, t_j are enabled in some state $S = (M, I)$ of the net NT, then
either $I(t_i) \cap I(t_j) = \emptyset$ or $I(t_i) = I(t_j)$.

R2. If there are simultaneously enabled transitions in some state $S = (M, I)$, then they should be in a fair conflict:
$I(t_i) = I(t_j) \implies {}^\bullet t_i \cap {}^\bullet t_j \neq \emptyset$ &
$\forall S_n = (M_n, I_n)\ S_0[\,\rangle\, S_n \quad t_i \in firable(S_n) \iff t_j \in firable(S_n)$.

The class of TPN's without overlappings of firing intervals is denoted by $\mathcal{N}_{12}$.

3.2. Some properties of TPN's without overlappings of firing intervals

TPN's in the class $\mathcal{N}_{12}$ have the following properties.

1. The Reachability and Boundedness problems for TPN's without overlappings of time intervals are undecidable.

This fact follows straightforward from the proof that reachability and boundedness problems are undecidable for TPN's [7, 11].

Let us first introduce the notion of free languages of TPN's.

A *free language* of a time net is the set $L^f(NT)$ of firing sequences which are feasible from an initial state, i. e., $L^f(NT) = \{ \sigma \in T^* \mid S_0[\sigma, \cdot\rangle \}$.

2. The class of all free languages generated by TPN's without overlappings of firing intervals is strictly contained in the class of all free languages generated by all TPN's.

It is obvious that the class of free languages of all TPN's contains the class of free languages of nets in the class $\mathcal{N}_{12}$. Figure 2 presents the time net whose language cannot be generated by a net in the class $\mathcal{N}_{12}$. It proves that the relation is strict. Details can be found in [13].

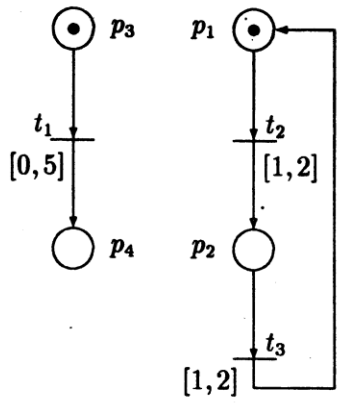

Figure 2

4. Reachability analysis of TPN's without overlappings of firing intervals

In this section, we will consider some special features of the dynamical behaviour of TPN's in the class $\mathcal{N}_{12}$. So, firing of all firing schedules which correspond to the same firing sequence determines the set of states. Here it will be shown that, for any state in the state class, the set of transitions firable from this state is the same. Thus it is not necessary to consider all feasible firing schedules in order to characterize the behaviour of the TPN without overlappings of firing intervals. It is sufficient to consider one of all feasible firing values for any fired transition in order to discover all the sequences of transitions firable in the net.

4.1. The notion of equivalence for states

Let $NT = (N, T, SI)$ be a time net in the class $\mathcal{N}_{12}$. We define the equivalence of two states in the following manner:

Definition 12. Let $S_1 = (M_1, I_1)$ and $S_2 = (M_2, I_2)$ be reachable states of NT.

The sets of enabled transitions *enabled*(M_1) and *enabled*(M_2) are defined as being *equivalent* iff they are equal and satisfy the following requirements:

1) if $I_1(t_i) = I_1(t_j)$, then $I_2(t_i) = I_2(t_j)$ for any enabled transitions t_i, t_j.

2) if transitions t_i, t_j are enabled and LFT of t_i is smaller than EFT of t_j in the state S_1, then this is also true in the state S_2.

The states S_1 and S_2 are *equivalent* iff their markings are equal and their sets of enabled transitions are equivalent.

Observe that the enabledness of transitions is defined in a sense of usual Petri nets and does not depend on the firing constraints of transitions (EFT's and LFT's). Hence, the equality of markings means the equality of the sets of enabled transitions.

Theorem 1. *If static EFT's and LFT's are rational numbers for all transitions, then all the states of TPN NT in the class $\mathcal{N}_{12}$ which are reached by the same firing sequence are equivalent.*

The proof is given in [12].

The net in Fig. 3(a) illustrates the necessity of introducing the notion of a fair conflict. Let S_1 and S_1' be the states which are reached by a firing transition t_1 at the times $\tau = 1$ and $\tau = 3$, respectively. It is obvious that $\beta_3 < \alpha_2$ in the state S_1 and transitions t_2, t_3 are simultaneously enabled in the state S_1'. If the condition R2 of definition 11 requires that simultaneously enabled transitions should be in conflict, then the net in Fig. 3(a) is the net in the class $\mathcal{N}_{12}$. But the states reachable by the same firing sequence are not equivalent in this net. However, if static EFT of at least one transition is not equal to its static LFT, then there exists a firing sequence whose firing leads to the state where the firing intervals of t_2 and t_3 intersect but are not equal.

Hence, if the net does not contain transitions with firing intervals of zero length, then it is sufficient to require in definition 11 that simultaneously enabled transitions should be in the usual conflict.

The net in Fig. 3(b) illustrates the necessity of restricting static EFT's and LFT's of transitions to rational numbers. Let the time set T be the set of natural numbers with an additional element π, the states S_2 and S_2'

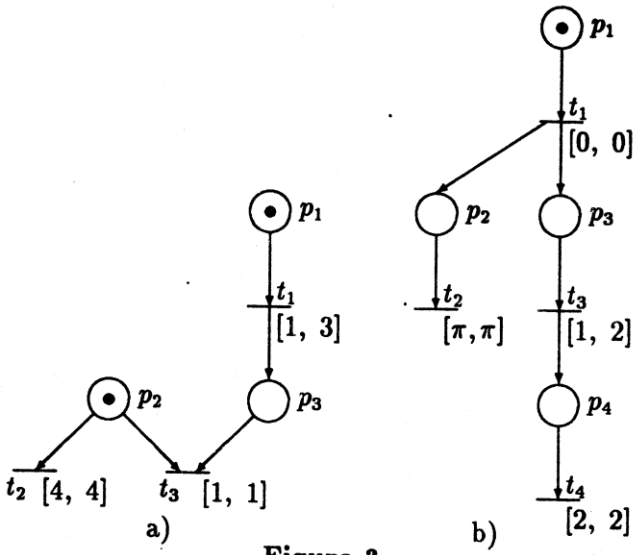

Figure 3

be the states reached by firing of schedules $(t_1, 0)(t_3, 1)$ and $(t_1, 0)(t_3, 2)$, respectively. The first schedule gives $\beta_4 < \alpha_2$ in the state S_2. The second one leads to the state S_2' with $\beta_2 < \alpha_4$. So, the states which are reachable by the same firing sequence are not equivalent.

If we consider the set of real numbers as the set of times, then the net in Fig 3(b) is not the net without overlappings of firing intervals. The firing schedule $(t_1, 0)(t_3, \pi - 2)$ leads to the state when two nonconflicted transitions are simultaneously enabled. It is shown in [12] that if the time set is the set of real numbers then the static EFT's and LFT's may be chosen among real numbers.

Further we suppose that static EFT's and LFT's for all transitions are rational numbers.

Corollary 1. *$firable(S_n) = firable(S_n')$ for any states S_n, S_n' which are reached by the same firing sequence.*

This property is not fulfilled for any TPN. It is not true for the net in Fig. 4. Let S_1 and S_1' be the states which are reached by firing transition t_1 at the times $\tau = 0$ and $\tau = 2$, respectively. It is obvious that $firable(S_1) = \{t_2\}$ and $firable(S_1') = \{t_2, t_3\}$, i. e. $firable(S_1) \neq firable(S_1')$.

Corollary 2. *If some firing schedule $(t_{i_1}, \tau_1) \dots (t_{i_n}, \tau_n)$ is feasible from the initial state of a net NT in the class $\mathcal{N}_{12}$, then the schedules $(t_{i_1}, \alpha_1) \dots (t_{i_n}, \alpha_n)$ (undelayed or immediate firing) and $(t_{i_1}, \beta_1) \dots (t_{i_n}, \beta_n)$ are also feasible from the initial state.*

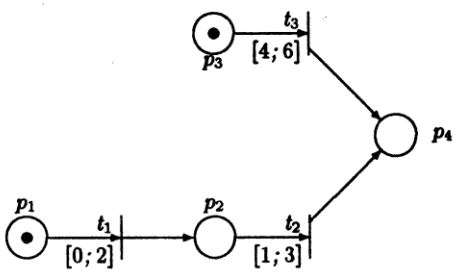

Figure 4

It is obvious that this property is not valid for any time net. In the net in Fig. 5(a), only the sequence $t_1t_2t_3$ can fire immediately. The sequence $t_2t_1t_3$ is also firable in the net but t_2 fires at or after the time $\tau = \alpha_2^s = 1$. Hence, t_1 will fire at the moment at least one time unit later than it becomes firable.

Figure 5(b) presents a TPN in the class $\mathcal{N}_{12}$ where both undelayed firing sequences $t_1t_2t_3$ and $t_2t_1t_3$ are feasible from the initial state.

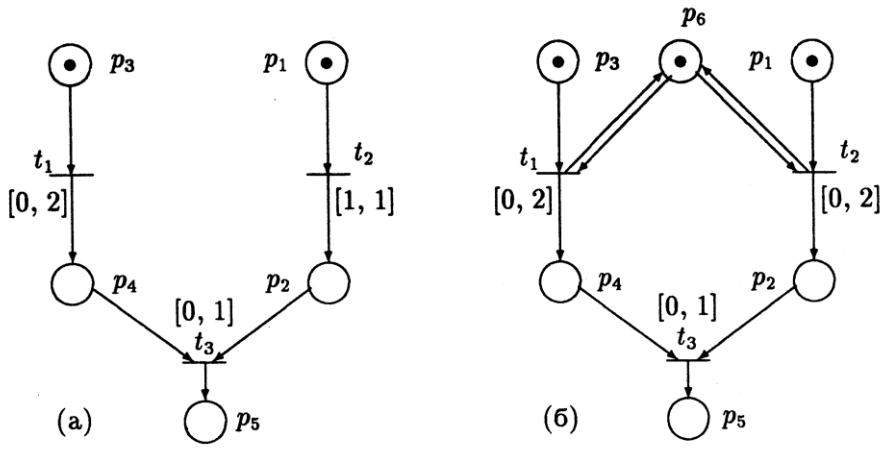

Figure 5

The execution of a time net when all firings occur immediately will be called an *execution in the maximal rate.*

4.2. Construction of a reduced reachability graph

Let us consider the net NT in the class $\mathcal{N}_{12}$. In general, the number of reachable states of NT may be infinite. But all the states reachable by the firing schedules which correspond to the same firing sequence $t_{i_1} \ldots t_{i_n}$ are

equivalent. This allows us to consider one state from an equivalence class in reachability analysis.

While construction of a reduced reachability graph we will consider only the states which are reachable by the execution of a net in the maximal rate. It allows us to avoid a special consideration of transitions which have the LFT's equal to infinity.

The formal procedure for the reachability graph construction is described below.

Procedure RG — Reachability Graph.

Given — a net NT in the class $\mathcal{N}_{12}$.

Compute — the graph $G = (V, E)$ where V is the set of nodes and E is the set of labelled arcs. Each node in V is a reachable state. There is an arc from a state S_i to a state S_j labelled with t, if t is firable from S_i and its firing at EFT leads to S_j.

1. Initially $V = \{S_0\}$, $E = \emptyset$. The initial node is unmarked.
2. **While** there is an unmarked node in V **do**

   a) select any unmarked node $S_i \in V$ and mark it.
   b) compute the set $firable(S_i)$.
   c) **for** all $t \in firable(S_i)$ **do**
      i) Compute the state S_j which is reachable from S_i by firing t at its EFT.
      ii) **if** there is a state S_k in V which is equivalent to S_j **then** the arc from S_i to S_k labelled with t is added to the graph.
      Otherwise, the state S_j is added to V and the arc from S_i to S_j labelled with t is added to E.

It is clear that the construction of a reduced reachability graph is simpler than enumeration of the reachable state classes. The number of computations executed at any step reduces. Moreover, the size of necessary information decreases. Let S be a state of TPN NT. If a transition t_i is enabled in the state S, then its time constraints are EFT α_i and LFT β_i. By definition of the state classes [2, 3], there exists the state class C which contains S and the firing domain of C contains for t_i one inequality of the form $\alpha_i \le x_i \le \beta_i$ and inequalities of the form $x_i - x_j \le \gamma_{ij}$ for any enabled transition t_j with $i \ne j$. However, calculating the reduced reachability graph makes no sense if the graph cannot be used to deduce the properties of the net. Hence, it is necessary to prove that the proposed procedure is correct.

Let a net NT be a time net without overlappings of firing intervals.

Theorem 2. *Any firing sequence which is feasible from the initial state in the net NT is a path in the reduced reachability graph of NT.*

The proof is given in [12, 13].

Observe that the opposite does not hold, since an influence of firing in the net on the firing intervals of transitions which remain enabled is not taken into account. It is obvious that the time net in Fig. 6(a) is in the class $\mathcal{N}_{12}$. The set of enabled transitions contains the transitions t_1, t_2, and their firing intervals do not intersect at any reachable state. The reduced reachability graph is presented in Fig. 6(b). There is a cyclic path $t_1^{k_1}t_2 \ldots t_1^{k_i}t_2 \ldots$ in the graph with arbitrary values k_i. But only the sequence $(t_1^3t_2)^7t_1^4t_2 \ldots$ is firable from the initial state in the net.

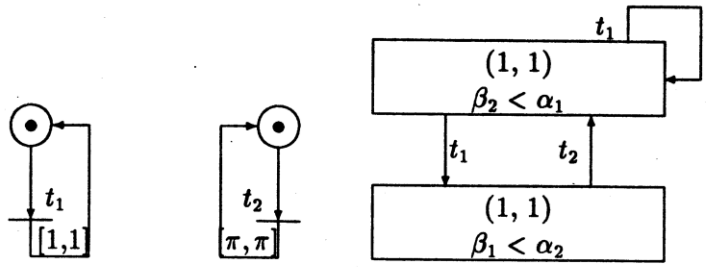

Figure 6

The theorem above tells us that if a marking is reachable in the net NT then it will be a node of a reduced reachability graph. Then, no reachable marking will be lost.

Theorem 3. *The reduced reachability graph is finite if and only if the net NT is bounded.*

The proof can be found in [12, 13].

This theorem guarantees that, when TPN is bounded, the construction of the reduced reachability graph will be finished. Moreover, any sufficient condition for boundedness property provides a sufficient condition for the finiteness of the reachability graph. We define the following sufficient condition of boundedness of a time net without overlappings of firing intervals similary to the sufficient condition of boundedness for the state classes [3].

Corollary 3. *A TPN NT without overlappings of firing intervals is bounded, if no pair of states $S_1 = (M_1, I_1)$ and $S_2 = (M_2, I_2)$, reachable from the initial state, sutisfies the following conditions:*

1) S_2 is reachable from S_1.

2) $M_2 \geq M_1$ and $M_2 \neq M_1$.

3) The sets enabled(M_1) and enabled(M_2) are equivalent.

The proof is given in [12, 13].

Unfortunately, this condition is not necessary. The condition fails for the net represented in Fig. 7(a). However, the net is bounded: the reduced reachability graph shown in Fig. 7(b) contains 5 nodes.

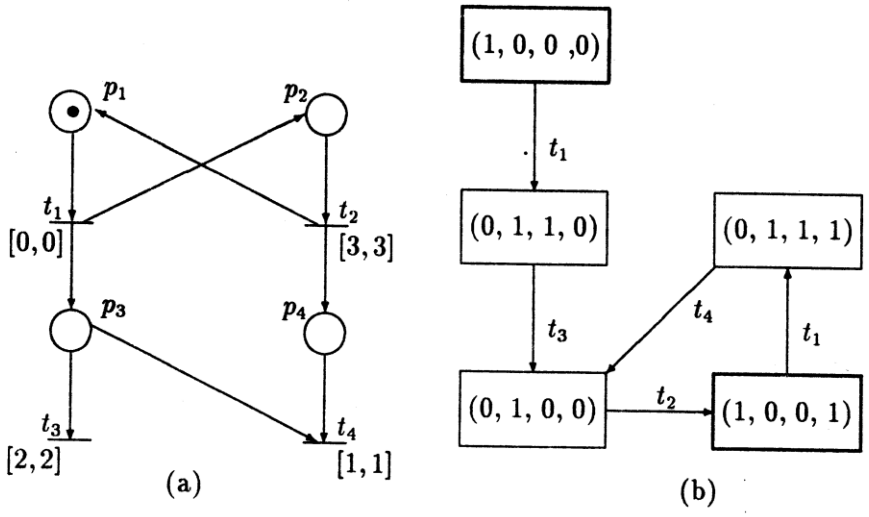

Figure 7

This sufficient condition is useful for stopping costruction of the reachability graph as soon as possible if the behaviour of the net is not the one expected. When a TPN without overlappings of firing intervals is bounded, its reduced reachability graph allows us to prove the properties which characterize its correct behaviour. For example, we can use the graph to prove boundedness, absence of deadlocks, etc. Finally, the reachability graph can be used for performance evaluation of the system modelled by a net. But it requires the information about time distances between enabled transitions, since not any path in the reduced reachability graph is a feasible firing sequence.

Conclusion

In this paper, a technique of reachability analysis for TPN's without overlappings of firing intervals [11] is presented. This technique constructs a reduced reachability graph. As its nodes, the graph contains only the states which are reachable by undelayed firing sequences, i.e. the firing sequences where all transitions fire at their EFT's. This makes the procedure of the reachability graph construction simpler than the analogous procedure for usual time nets [2, 3].

The correctness of the presented procedure is proved.

Since TPN's without overlappings of firing intervals are as expressive as the Turing machines [11], no necessary and sufficient condition can be stated for the boundedness property. A sufficient condition is defined in the paper which allows us to stop the construction of the reachability graph as early as possible, if the behaviour of the net is not the one expected. If the net is bounded, the reduced reachability graph can be completely constructed. The proved correctness of the technique allows us to use the reduced reachability graph in a proof that the net has the specific properties.

Acknowledgments

I would like to thank my scientific supervisor Valery Nepomniaschy for helpful discussions and remarks.

References

[1] W. M. P. van der Aalst, *Interval Timed Coloured Petri Nets and their Analysis*, Lect. Notes Comput. Sci., Springer-Verlag, Berlin, Vol. 691, 1993, 453–472.

[2] B. Berthomiu, M. Menasche, *An enumerative approach for analyzing time Petri nets*, Proc. of IFIP Congress, 1983, 41–46.

[3] B. Berthomieu, M. Diaz, *Modelling and verification of time dependent systems using time Petri nets*, IEEE Trans. on Softw. Eng., Vol. 3, No. 3, 1991, 259–273.

[4] U. Buy, R. H. Sloan, *Analysis of real-time programs with simple time Petri nets*, Softw. Eng. Notes, Vol. 19, 1994, 228–239.

[5] A. Cerone, *A net-based approach for specifying real-time systems*, Ph. D. thesis, TD-16/93, Dipartimento di Informatica, Università di Pisa, Pisa, Italy, 1993.

[6] J. Y. Halpern, L. D. Zuck, *A little knowledge goes a long way: Knowledge-based derivations and correctness proofs for a family of protocols*, J. of the Association for Comp. Machinery, Vol. 39(3), No. 7, 1992, 449–478.

[7] N. D. Jones, L. H. Landweber, Y. E. Lien, *Complexity of some problems in Petri nets*, Theoretical Computer Science, No. 4, 1977, 277–301.

[8] V. E. Kotov, *Petri nets*, Moscow, Nauka, 1984 (in Russian).

[9] P. M. Merlin, D. J. Farber, *Recoverability of communication protocols — implications of a theoretical study*, IEEE Trans. on Comm., No. 9, 1976, 1036–1043.

[10] S. Morasca, M. Pezze, C. Ghezzi, D. Mandrioli, *A Unified High-Level Petri Net Formalism For Time-Critical Systems*, IEEE Trans. on Softw. Eng., Vol. 17, No. 2, 1991, 160–173.

[11] E. V. Okunishnikova, *Time Petri nets without intersections of firing intervals*, Specification, Verification and Net Models of Concurrent Systems, Institute of Informatics Systems, Novosibirsk, 1994, 65–98.

[12] E. V. Okunishnikova, *Reachability analysis for time Petri nets without overlappings of firing intervals*, Problems of Specification and Verification of Concurrent Systems, Institute of Informatics Systems, Novosibirsk, 1995 (in Russian).

[13] E. V. Okunishnikova, *Time Petri nets without overlappings of firing intervals*, Programming, Moscow, Nauka (in Russian, to appear).

[14] L. Popova-Zeugmann, *On time Petri nets*, J. Inform. Process. Cybern. EIK, Vol. 27, No. 4, 1991, 227–244.

[15] C. Ramchandani, *Analysis of asynchronous concurrent systems using Petri nets*, Ph. D. thesis, Project MAC, MAC-TR 120, MIT, 1974.

[16] J. Sifakis, *Performance evaluation of systems using nets*, Lect. Notes Comput. Sci., Springer-Verlag, Berlin, Vol. 84, 1980, 307–319.

[17] P. H. Starke, *Some properties of timed nets under the earliest firing rule*, Lect. Notes Comput. Sci., Springer-Verlag, Berlin, Vol. 424, 1989, 418-432.