Fundamenta Informaticae XXV (2004) 1001–1018
IOS Press

A Method to Prove Non-Reachability in Priority Duration Petri Nets*

Matthias Werner
Communication and Operating Systems Group
Department for Electrical Engineering
and Computer Science
TU Berlin
`mwerner@cs.tu-berlin.de`

Jan Richling
Computer Architecture and Communication Group
Department of Computer Science
Humboldt University of Berlin
`richling@informatik.hu-berlin.de`

Louchka Popova-Zeugmann
Automata and System Theory Group
Department of Computer Science
Humboldt University of Berlin
`popova@informatik.hu-berlin.de`

Abstract. Times and priorities are important concepts that are frequently used to model real-world systems. Thus, there exist extensions for Petri nets which allow to model times and priorities. In contrast, many proof techniques are based on classical (time-less and priority-less) Petri nets. However, this approach fails frequently for timed and prioritized Petri nets.

In this paper, we present an approach to prove non-reachability in a Priority Duration Petri net. We use for this proving technique a state equation as well as conditions for firing that include a priority rule and a maximal step rule. Our approach leads to a system of equations and inequalities, which provide us with a sufficient condition of non-reachability. We demonstrate the application of our approach with an example.

Keywords: Priority Duration Petri net, state equation, non-reachability

*This is an extended version of the paper "Non-Reachability in Priority Duration" presented at the CS&P 2003.

Address for correspondence: Matthias Werner, Kommunikations- und Betriebssysteme, Sekr. FR 6-3, Franklinstr. 28/29, TU Berlin, 10587 Berlin, Germany

1. Introduction

1.1. Motivation

Times and priorities are important concepts which are frequently used in real-world systems. Such systems can be built, e.g, with the *Message Scheduled System* (MSS) architecture [5] that we have developed. MSS supports composability of real-time systems. It ensures that every component meets its contracted specification on its real-time behavior. The MSS architecture relies on standard priority-based scheduling methods, e.g., [1]. In order to prove the formal correctness of the MSS architecture we need a way to deal with priorities and times.

The correctness of an MSS system can be mapped to the non-reachability of certain states. In [7] and [6] we presented a formal specification of MSS' behavior based on prioritized Duration Petri nets (DPN), and a tool-supported way to generate the specification for a desired system instance, respectively. In [3], we showed how a non-reachability proof can be conducted in a DPN. As a next step, we present in this paper the extension of our method to systems with priorities.

Frequently, non-reachability proofs in extended Petri nets are done by proving the non-reachability in the underlying classic Petri net. However, this approach fails in cases like MSS where the system's correctness relies on the restriction introduced by time and priorities.

In this paper we present an approach that allows to prove non-reachability in the Priority Duration Petri net (PDPN) itself. For that purpose, we give a state equation and a number of firing conditions that lead to a system of equations and inequalities. This system provides a sufficient condition to prove non-reachability.

The remainder of this paper is organized as follows: The rest of this section names the state of art. Section 2 provides the definitions needed. In Section 3, we derive from the definitions a state equation, priority conditions and conditions for maximal steps that support an algebraic reasoning about PDPN. We show in Section 4 how to apply our methods and prove a non-reachability in an example net. The paper concludes with some final remarks.

1.2. Related Work

The question of reachability and non-reachability raises for several problems which can be described by Petri nets. In general, the state space of a Petri net is infinite. Different methods have been developed in order to prove certain properties without computing resp. knowing the whole state space of the net.

One approach in classical Petri nets is the computing of invariants. There, it is relatively easy to obtain place or transition invariants using the state equation of the net (cf. [8]).

But, if the Petri net is a time dependent one or uses priorities, the state space becomes smaller. In this case, the use of the state equation of the skeleton is not very reasonable, because this state equation does not take into account the additional constraints of the net.

There exists quite a range of possibilities to represent times in Petri nets. Times can be assigned to transitions, places and tokens (cf. [9]); for each case several semantics exist. In this work we consider an extension to Duration Petri nets (also called Timed Petri nets). Duration Petri nets were defined by Ramchandani [4]. Here, times are assigned to transitions and describe a delay in the process of firing.

Also, priorities are an often used concept in Petri nets. Usually, priorities will be assigned to transitions or to tokens. We use the first alternative.

There exist approaches to analytically deal with time and priorities in stochastic Petri nets, cf. e.g., [2]. The reachability graph of stochastic Petri net is the same as the graph of the underlying classical Petri net and therefore reachability analysis for this kind of Petri nets is purchased on the reachability graph of the classical reachability graph preferably. In the non-stochastic area, there exist quite a few simulation tools that consider time and priorities.

However, to the best of our knowledge, there exists no analytical approach to evaluate non-reachability in non-stochastic prioritized time-depended Petri nets.

2. Definitions

2.1. Notation

This subsection introduces the basic notations we use in our paper. $\mathbb{N}$ and $\mathbb{R}$ denote the set of natural numbers and rational numbers, respectively. $\mathbb{N}^+ = \mathbb{N}\backslash\{0\}$ denotes the set of natural numbers without 0, and $\mathbb{Q}_0^+$ denotes the set of nonnegative rational numbers.

$\mathcal{M}(m, n)$ is the set of all matrices with m rows and n columns. A superscript in parentheses distinguishes different matrices. For the arbitrary matrix $A^{(k)} \in \mathcal{M}(m, n)$, $A_i^{(k)}$ is the i-th row and $A_{\cdot j}^{(k)}$ is the j-th column of the matrix.

Let M be a finite set. $|M|$ is the number of elements of M. Let Q be a finite multiset (bag) in M. $\kappa_Q(m)$ with $m \in M$ denotes the multiplicity of m, i.e., how many instances of m are in Q.

$E_n = (e_{i,j}) \in \mathcal{M}(n, n)$ denotes the (unit-)matrix with $e_{i,j} = \begin{cases} 1 & i = j \\ 0 & \text{otherwise} \end{cases}$ and $\mathcal{O}_n = (o_{i,j}) \in \mathcal{M}(n, n)$ the (zero-)matrix with $o_{i,j} = 0 \; \forall i, j$.

The relation $r^{(1)} \ngeq r^{(2)}$ of the two vectors $r^{(1)}, r^{(2)} \in \mathcal{M}(m, 1)$ means, that there exists at least one $i,\ i \in \{1, \ldots, m\}$ with $r_i^{(1)} < r_i^{(2)}$.

2.2. Structural Descriptions

We begin with the usual definition of a Petri net:

Definition 2.1. (Petri net)
The structure $N = (P, T, F, V, m_o)$ is called a Petri net (PN) iff

1. P, T, F are finite sets with $P \cap T = \emptyset,\ P \cup T \neq \emptyset,\ F \subseteq (P \times T) \cup (T \times P)$ and $dom(F) \cup cod(F) = P \cup T$
2. $V : F \longrightarrow \mathbb{N}^+$ (weight of the arcs)
3. $m_o : P \longrightarrow \mathbb{N}$ (initial marking)

A marking of a PN is a function $m : P \longrightarrow \mathbb{N}$, such that $m(p)$ denotes the number of tokens at the place p. The pre-sets and post-sets of a transition t are given by $Ft = \{p \mid p \in P \wedge (p, t) \in F\}$ and $tF = \{p \mid p \in P \wedge (t, p) \in F\}$, respectively. Each transition $t \in T$ induces the markings t^- and t^+, defined as follows:

$$t^-(p) = \begin{cases} V(p,t) & \text{iff} \quad (p,t) \in F \\ 0 & \text{iff} \quad (p,t) \notin F \end{cases} \qquad t^+(p) = \begin{cases} V(t,p) & \text{iff} \quad (t,p) \in F \\ 0 & \text{iff} \quad (t,p) \notin F \end{cases}$$

A transition $t \in T$ is enabled (may fire) at a marking m iff $t^- \leq m$ (i.e., $t^-(p) \leq m(p)$ for every place $p \in P$). When an enabled transition t at a marking m fires, this yields a new marking m' given by $m'(p) := m(p) + t^+(p) - t^-(p)$. The firing is denoted by $m \xrightarrow{t} m'$.

Definition 2.2. (Duration Petri net (DPN))
The structure $Z = (N, D)$ is called a Duration Petri net[1] (DPN) iff:

1. $S(Z) = N$ is a PN called the *skeleton* of Z.
2. $D : T \longrightarrow \mathbb{Q}_0^+$ (duration function).

$d_i := D(t_i)$ is the duration of transition's t_i firing. It is easy to see, that without loss of generality we may consider DPNs with $D : T \longrightarrow \mathbb{N}$. Therefore, only such time functions D will be considered subsequently.

A DPN behaves similar to a PN with a maximal step semantic. However, the token(s) will reach the post-set of a transition only after the delay of this transition is elapsed.

Definition 2.3. (time dimension)
$d := \max\limits_{t \in T}\{D(t)\} + 1$ is called the time dimension of the DPN.

Definition 2.4. (Priority Duration Petri nets, (PDPN))
The structure $Z = (N, \Theta)$ is called a Priority Duration Petri net (PDPN) iff:

1. $S(Z) = N$ a DPN called the skeleton of Z
2. $\Theta : T \longrightarrow \mathbb{N}$ (priority function).

$\theta_i = \Theta(t_i)$ denotes the priority of transition t_i. Without loss of generality we assume that a higher priority value means that the transition is preferred to a transition with a lower priority value.

Example 2.1. The PDPN Z_1 which is used for illustration is shown in Figure 1.
The time dimension of $Z_1$is $d = 3$.

In order to describe the relation between tokens and time, we use (as for DPN in [3]) the notion of a *time marking*.

Definition 2.5. (time marking)
Let Z be a PDPN. A matrix m with $m \in \mathcal{M}(|P|, d)$ is a time marking in Z.

Definition 2.6. (initial time marking)
The time marking $m^{(0)}$ is an initial time marking, iff $m_{\cdot 1}^{(0)} = m_0$ and $m_{i,j}^{(0)} = 0$ for $i = 1 \dots |P|$ and $j = 2 \dots d$.

Each column of the time marking matrix represents a (partial) marking of a place for different delays. The first column represents the present, and thus the marking of the underlying PN skeleton. The second column represents tokens which are on their way to the place and will arrive in one time unit. The same is true for the third column with two time units, etc.

[1]Also called Timed Petri net.

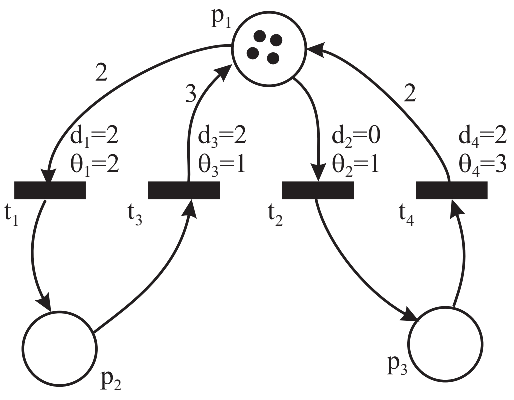

Figure 1. Net from Example 2.1

Example 2.2. The net from Example 2.1 has the following initial time marking:

$$m^{(0)} = \begin{pmatrix} 4 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Let $m^{(c)}$ the marking of a PDPN in the classic notation, and $m^{(t)}$ the time marking variant. Then the following is true:

$$\forall i, i = 1 \ldots |P|, m_i^{(c)} = \sum_{j=1}^{d} m_{i,j}^{(t)}$$

In the remainder of this paper, m denotes always a time marking.

2.3. Dynamics

A Priority Duration Petri net has a dynamical behavior that leads to a change of the marking. The notation for a change from marking $m^{(1)}$ to marking $m^{(2)}$ is $m^{(1)} \to m^{(2)}$. In general, two kinds of changing actions have to be distinguished:

- *Firing of transitions.* Firing transitions is similar to classical Petri nets, however, we allow a firing of maximal sets of transitions only.

  We denote firing of the set $\{t_1, \ldots, t_k\}$ by writing the set of transitions in action atop the arrow: $m^{(1)} \xrightarrow{\{t_1,\ldots,t_k\}} m^{(2)}$. If there is a need for distinction, we denote a marking that is reached by firing with a hat: $m^{(1)} \xrightarrow{\{t_1,\ldots,t_k\}} \hat{m}^{(2)}$

- *Elapsing of time.* A DPN (and thus a PDPN) may change its marking by elapsing of time. The time elapsing happens synchronously for all transitions in the net. It reduces the time a token has to wait till its delivery. In our notation, time elapsing affects the row m_i of the time marking m corresponding to the place p_i. An interaction between different places does not take place.

We denote elapsing of τ time units by writing the number of time units beneath the arrow: $m^{(1)} \underset{\tau}{\rightarrow} m^{(2)}$. If there is a need for distinction, we denote a marking that is reached be time elapsing by a tilde: $m^{(1)} \underset{\tau}{\rightarrow} \tilde{m}^{(2)}$.

In the following, we define the change actions in detail. We start with the elapsing of time. It is equivalent to DPN, cf. [3]. To allow consideration about time, we define a transition clock vector:

Definition 2.7. (transition clock vector)
Let Z be a PDPN. Then, the vector $h \in \mathcal{M}(|T|, 1)$ is called transition clock vector of Z, iff
$\forall i \, (1 \leq i \leq |T| \rightarrow h_i \leq D(t_i))$

An element h_i of h is non-zero if the related transition is firing, and zero otherwise. A $h_i \neq 0$ shows remaining time until the firing is finished. The pair (m, h) describes the state of the PDPN.

Now we are prepared to define time elapsing:

Definition 2.8. (time elapsing)
Let Z be a PDPN and $m^{(1)}$ and $m^{(2)}$ time markings in Z. The time marking $m^{(2)}$ is yielded from $m^{(1)}$ by time elapsing, iff

$$m^{(2)}_{i,j} := \begin{cases} m^{(1)}_{i,1} + m^{(1)}_{i,2} & , j = 1 \\ m^{(1)}_{i,j+1} & , 2 \leq j \leq d-1 \\ 0 & , j = d \end{cases} \quad \text{and } h^{(2)}_i := \max(0, h^{(1)}_i - 1)$$

In other words, all tokens in the time marking move one column to the left, except the first column, that accumulates the token from the first and second column, and the last column, which is filled with zero. In addition, all non-zero clocks are decreased.

Firing is equivalent to the firing in DPN too, as long as the selection of transitions to fire is not considered:

Definition 2.9. (firing)
Let Z be a PDPN and $m^{(1)}$ and $m^{(2)}$ time markings in Z. The time marking $m^{(2)}$ is yielded from $m^{(1)}$ by firing the set of transitions $\mathcal{B}$, iff

$$m^{(2)}_{i,j} := \begin{cases} m^{(1)}_{i,j} - \sum\limits_{t_s \in \mathcal{B}} V(p_i, t_s) + \sum\limits_{\substack{t_s \in \mathcal{B}, \\ d_s = 0}} V(t_s, p_i) & , j = 1 \\ m^{(1)}_{i,j} + \sum\limits_{\substack{t_s \in \mathcal{B}, \\ d_s = j-1}} V(t_s, p_i) & , j > 1 \end{cases}$$

and

$$h^{(2)}_i := \begin{cases} D(t_i) & , t_i \in \mathcal{B} \\ h^{(1)}_i & \text{otherwise} \end{cases}$$

In other words, firing a transition removes a number of tokens from the first column of all places in the firing transition's pre-set and adds a number of tokens to a certain column of all places in the transition's post-set. The column to add corresponds to the delay of the transition (0: first column, 1: second column, etc.) and the numbers of tokens correspond to the weight of arcs form and to the transition, respectively.

h_i is working like an egg-timer: If a transition t_i starts to fire, then its clock is set to d_i, i.e., $h_i := d_i$. After then, h_i is decreased by each time elapsing.

Until here, the definition of a PDPN does not differ from the definition of a DPN.[2] The impact of the priorities is in the notion of a step, i.e., the set of all token moves that happen before time elapses. For PDPN, we use a modified maximal step, that considers priorities:

Definition 2.10. (prioritized maximal step)
Let Z be an PDPN. $\mathcal{B} \subseteq T$ is called a prioritized maximal step on the time marking m with the transition clock vector h iff

1. $\mathcal{B} \subseteq T$
2. $\sum\limits_{t \in \mathcal{B}} t^- \leq m_{\cdot 1}$
3. $\forall t (t \in \mathcal{B} \rightarrow h(t) = 0)$
4. $\forall t \forall t_1 \Bigg(\big(t \in \mathcal{B} \wedge t_1 \notin \mathcal{B} \ \wedge \ Ft \cap Ft_1 \neq \emptyset \ \wedge \ t_1^- \leq m_{\cdot 1} \ \wedge \ h(t_1) = 0\big) \rightarrow$
$$\Big(\Theta(t) \geq \Theta(t_1) \vee m_{\cdot 1} - \sum_{\substack{t \in \mathcal{B} \\ \Theta(t) \geq \Theta(t_1)}} t^- \not\geq t_1^- \Big) \Bigg)$$
5. $\neg \exists \mathcal{B}^* \left((\mathcal{B}^* \supset \mathcal{B}) \ \wedge \ (\mathcal{B}^* \text{ satisfies 1. – 4.}) \right)$

In other words, Definition 2.10 describes a maximal (5) set of transitions (1) which are enabled (2) and not in the process of firing (3), that contains no lower prioritized transition as long as it could contain a higher prioritized transition instead (4).

Example 2.3. Consider the nets in Figure 2. Assumed, no firing is in progress, the net in Figure 2(a) allows the prioritized maximal step $\mathcal{B} = \{t_1, t_2\}$ only. In Figure 2(b), the prioritized maximal step is $\mathcal{B} = t_4, t_6$. t_5 is not in $\mathcal{B}$, since it is in conflict with the higher prioritized t_4, cf. Definition 2.10, item 4.

Please note, that in general more than one prioritized maximal step may exist.

Since we allow zero-time transitions (i.e. $d_i = 0$), it is possible that more than one prioritized maximal step takes place before a time unit may elapse. Thus, we define a global step that includes all firing actions that take place before time elapses.

Definition 2.11. (global step)
Let Z be an PDPN with the time marking m. $\mathcal{G}$ is a multiset (bag) that is constructed in the following way:

[2]However, a DPN as defined in [3] does not allow zero-time delays. But—as also discussed there—it is easy to extend the definition in the way used in the current paper.

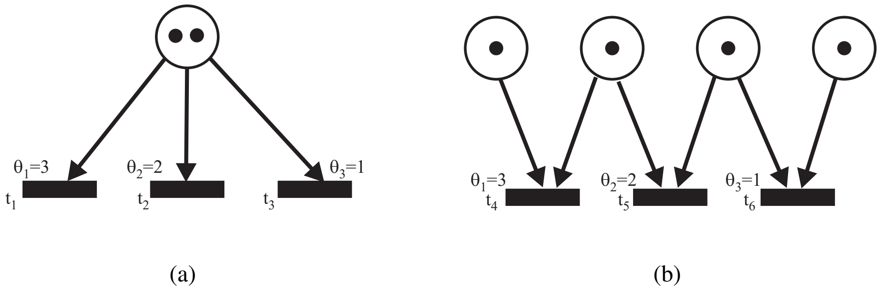

Figure 2. Prioritized maximal step

1. $\mathcal{G} = \emptyset$
2. If there exists a prioritized maximal step $\mathcal{B} \neq \emptyset$, then $\mathcal{G} := \mathcal{G} + \mathcal{B}$; else the construction is ready.
3. m and h are changed according to Definition 2.9 by firing the set $\mathcal{B}$.
   Continue with 2

A global step $\mathcal{G}$ is a multiset yielded by addition of the prioritized maximal steps $\mathcal{B}_1, \ldots, \mathcal{B}_r$ where $m^{(*)} \xrightarrow{\mathcal{B}_1} m^{(1)} \xrightarrow{\mathcal{B}_2} \cdots \xrightarrow{\mathcal{B}_r} m^{(r)}$. Since $\mathcal{G}$ is a multiset, it may include more than one instances of a certain transition. However, this is only true, if this transition has a zero-time delay. Obviously, a global step may be an empty set.

The dynamical behavior of a PDPN is marked by a strict alternation of firing with global steps and time elapsing:

$$m^{(0)} \xrightarrow{\mathcal{G}_1} \hat{m}^{(1)} \underset{1}{\rightarrow} \tilde{m}^{(1)} \xrightarrow{\mathcal{G}_2} \cdots \xrightarrow{\mathcal{G}_n} \hat{m}^{(n)} \underset{1}{\rightarrow} \tilde{m}^{(n)}$$

Without loss of generality, we use one time unit for the time elapsing. However, each common factor of all transition delays would be possible.

Since the number of elapsed time units after a firing is constant for a PDPN and in our considerations always 1, we skip in the following the notion of the elapsed time. I.e., a firing sequence $(\mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3)$ means the alternating sequence of firing and time elapsing $(\mathcal{G}_1, 1, \mathcal{G}_2, 1, \mathcal{G}_3)$ or $(\mathcal{G}_1, 1, \mathcal{G}_2, 1, \mathcal{G}_3, 1)$.

Definition 2.12. (reachability)
Let Z be a PDPN. A time marking m' is reachable in Z iff there exists a sequence $\sigma = (\mathcal{G}_1, \ldots, \mathcal{G}_\alpha)$ of global steps such that

$$m^{(0)} \xrightarrow{\mathcal{G}_1} \hat{m}^{(1)} \underset{1}{\rightarrow} \tilde{m}^{(1)} \xrightarrow{\mathcal{G}_2} \hat{m}^{(2)} \underset{1}{\rightarrow} \tilde{m}^{(2)} \ldots \rightarrow m'$$

σ may end either with firing, or with time elapsing.

3. State Equation and Firing Conditions

In this section we introduce some further concepts that support an algebraic reasoning about PDPN. Especially, we present a state equation and few firing invariants that are valid in every PDPN.

The structure of a PDPN as given by Definition 2.4 may be described by an incidence matrix.

Definition 3.1. (incidence matrix)
Let Z be a PDPN. The matrix $C \in \mathcal{M}(|P|, d \cdot |T|)$ is called the incidence matrix of Z, iff $C := (C^{(1)}, C^{(2)}, \ldots, C^{(|T|)})$ with $C^{(k)} \in \mathcal{M}(|P|, d)$,$k \in \{1, \ldots, |T|\}$ and $C^{(k)} = (c^{(k)}_{i,j})$ where

$$c^{(k)}_{i,j} := \begin{cases} V(t_k, p_i) - V(p_i, t_k) & , d_k = 0, j = 1 \\ -V(p_i, t_k) & , d_k > 0, j = 1 \\ V(t_k, p_i) & , (d_k > 0), 0 < j - 1 = d_k \\ 0 & , \text{otherwise} \end{cases}$$

I.e., a submatrix $C^{(i)}$ describes how the transition t_i is connected with places. If the transition has a zero-delay ($d_i = 0$), only the first column $C^{(i)}_{.1}$ is used. It includes differences of the weights of all arcs *to* the places and these *from* the places. If the transition has a delay ($0 < d_i < d$), the first column includes the weights *from* the places and the $d_i + 1$-th column the weights *to* the places.

Example 3.1. The incidence matrix of Example 2.1 in Figure 1 is:

$$C = \begin{pmatrix} -2 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \end{pmatrix}$$

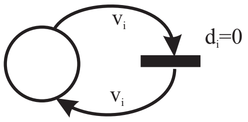

Figure 3. A loop in a Petri net

Please note, that—similar to a classic Petri net—a loop (i.e., two arcs (p, t) and (t, p) with the same weight, cf. Figure 3) will not appear in the incidence matrix, if the transition has a zero-time delay. However, if the transition in the loop has a non-zero delay ($d_i > 0$) both arcs will be represented in the incidence matrix.

Next, we define a bag matrix to represent global steps.

Definition 3.2. (bag matrix)
Let $\mathcal{G}$ be a (maximal or global) step and $\kappa_{\mathcal{G}}(t_i) = \kappa_{\mathcal{G}_i}$ the multiplicity of transition t_i in $\mathcal{G}$. The matrix $G \in \mathcal{M}(d \cdot |T|, d)$ is called the bag matrix of $\mathcal{G}$ iff

$$G = (g_{i,j})_{\substack{i=1\ldots d\cdot|T| \\ j=1\ldots d}} = \begin{pmatrix} G^{(1)} \\ G^{(2)} \\ \vdots \\ G^{(|T|)} \end{pmatrix} \text{ and } G^{(s)} = \kappa_{\mathcal{G}_s} \cdot E_d$$

Obviously, the submatrix $G^{(s)}$ is the zero-matrix if the transition t_s does not belong to the global step $\mathcal{G}$.

Example 3.2. Let us consider the step $\mathcal{G} = \{t_1, t_2, t_2\}$ in Z_1 (Figure 1):

$$G = \begin{pmatrix} G^{(1)} \\ G^{(2)} \\ G^{(3)} \\ G^{(4)} \end{pmatrix} \text{ with } \begin{array}{cc} G^{(1)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} & G^{(2)} = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix} \\ \\ G^{(3)} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & G^{(4)} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \end{array} \text{ i.e.: } G = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ \\ 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \\ \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Next, we define a progress matrix that allows us to describe the impact of time elapsing to the time marking.

Definition 3.3. (progress matrix)
Let Z be a PDPN. The matrix $R \in \mathcal{M}(d, d)$ is called the progress matrix of Z, iff

$$r_{i,j} := \begin{cases} 1 & \text{if } (i = j = 1) \text{ or } (i = j + 1) \\ 0 & \text{otherwise} \end{cases}$$

Example 3.3. In Example 2.1 the progress matrix R of the net Z_1 is $R = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$.

Finally, we define a Parikh matrix that represents a sequence of steps and time units.

Definition 3.4. (Parikh matrix)
The matrix $\Psi \in \mathcal{M}(|P|, d)$ is called Parikh matrix of the sequence $\sigma = (\mathcal{G}_1, \ldots, \mathcal{G}_n)$, iff

$$\Psi_\sigma := \sum_{i=1}^{n} G^{(i)} \cdot R^{n-i}$$

Using the introduced elements, we can formulate a state equation for PDPNs:

Theorem 3.1. (state equation)
Let Z be a PDPN, $\sigma = (\mathcal{G}_1, \dots, \mathcal{G}_n)$ a firing sequence in Z, $m^{(0)}$ the initial time marking of Z and
$m^{(0)} \xrightarrow{\mathcal{G}_1} \hat{m}^{(1)} \underset{1}{\longrightarrow} \tilde{m}^{(1)} \xrightarrow{\mathcal{G}_2} \hat{m}^{(2)} \underset{1}{\longrightarrow} \dots \xrightarrow{\mathcal{G}_n} m^{(n)}$. Then it holds:

$$m^{(n)} = m^{(0)} \cdot R^{n-1} + C \cdot \Psi_\sigma \tag{1}$$

To prove Theorem 3.1 let us consider following lemmata:

Lemma 3.1. Let Z be a PDPN, $m^{(1)}$ a reachable time marking in Z and $m^{(1)} \underset{1}{\longrightarrow} m^{(2)}$. Then it holds $m^{(2)} = m^{(1)} \cdot R$.

Proof:
$m^{(2)}_{i,j} = m^{(1)}_{i,j} \cdot R$

$$\begin{array}{llcl}
\textit{Case 1:} & j = 1 & \rightsquigarrow & \sum_{s=1}^{d} m^{(1)}_{i,s} \cdot r_{s,1} = m^{(1)}_{i,1} \cdot 1 + m^{(1)}_{i,2} \cdot 1 \underset{acc.def}{=} m^{(2)}_{i,1} \\
\textit{Case 2:} & d > j \geq 2 & \rightsquigarrow & \sum_{s=1}^{d} m^{(1)}_{i,s} \cdot r_{s,j} = m^{(1)}_{i,j+1} \cdot 1 \underset{acc.def}{=} m^{(2)}_{i,j} \\
\textit{Case 3:} & j = d & \rightsquigarrow & \sum_{s=1}^{d} m^{(1)}_{i,s} \cdot r_{s,j} = 0 = m^{(2)}_{i,j}
\end{array}$$

□

Lemma 3.2. Let Z be a PDPN and $m^{(1)} \xrightarrow{\mathcal{G}} m^{(2)}$, $\mathcal{G} = \{t_{i_1}, \dots, t_{i_q}\}$. Then holds: $m^{(2)} = m^{(1)} + C \cdot G$.

Proof:
We consider $m^{(2)}_{i,j}$.
Case 1: $j = 1$
According to Definition 2.9 and Definition 2.11 the following is true:

$$m^{(2)}_{i,1} = m^{(1)}_{i,1} - \kappa_{\mathcal{G}_s} \sum_{t_s \in \mathcal{G}} V(p_i, t_s) + \kappa_{\mathcal{G}_s} \sum_{\substack{d_s = j-1 = 0 \\ t_s \in \mathcal{G}}} V(t_s, p_i) \ .$$

Therefore, it is sufficient to show that

$$\sum_{l=1}^{d \cdot |T|} c_{i,l} g_{j,l} = -\kappa_{\mathcal{G}_s} \sum_{t_s \in \mathcal{G}} V(p_i, t_s) + \kappa_{\mathcal{G}_s} \sum_{\substack{d_s = 0 \\ t_s \in \mathcal{G}}} V(t_s, p_i) \ .$$

Let us consider the bag matrix G. The first column $G_{.1}$ of G is $G_{.1} = (g^{(1)}, g^{(2)}, \ldots, g^{(|T|)})^T$, where $g^{(k)}$ is a d-dimensional vector with

$$g^{(k)} = (g_1^{(k)}, \ldots, g_d^{(k)}) \quad k = 1, \ldots, |T| \text{ and } g_j^{(k)} = \begin{cases} \kappa_{\mathcal{G}_k} & \text{if } j = 1 \\ 0 & \text{otherwise} \end{cases}$$

That means, $g^{(k)}$ is the d-dimensional vector $(\kappa_{\mathcal{G}_k}, 0, \ldots, 0)^T$ if $t_k \in \mathcal{G}$ and the d-dimensional zero-vector otherwise.

Hence,

$$\sum_{l=1}^{d \cdot |T|} c_{i,l} \cdot g_{l,1} = \sum_{k=1}^{|T|} \sum_{l=1}^{d} c_{i,l}^{(k)} \cdot g_l^{(k)} = \kappa_{\mathcal{G}_k} \sum_{k=1}^{|T|} c_{i,1}^{(k)} \underset{\substack{acc.def\ 3.1 \\ \text{for } j=1}}{=} -\kappa_{\mathcal{G}_k} \sum_{t_k \in \mathcal{G}} V(p_i, t_k) + \kappa_{\mathcal{G}_k} \sum_{\substack{d_k=0 \\ t_k \in \mathcal{G}}} V(t_k, p_i)$$

Case 2: $j \geq 2$

Obviously, now we have to show that $\sum_{s=1}^{d \cdot |T|} c_{i,s} \cdot g_{s,j} = \kappa_{\mathcal{G}_k} \sum_{d_k = j-1} V(t_s, p_i)$. We consider the j-th column of the bag matrix G, the vector $G_{.j}$:

$$G_{.j} = (g^{(1)}, g^{(2)}, \ldots, g^{(|T|)}) \text{ with } g^{(k)} = (g_1^{(k)}, \ldots, g_d^{(k)})^T \quad \overset{|T|}{\underset{1}{\forall}} k \text{ and } g_s^{(k)} = \begin{cases} \kappa_{\mathcal{G}_k} & , s = k \\ 0 & , \text{otherwise} \end{cases}$$

i.e. $g^{(k)} = \underbrace{(0, \ldots, 0)}_{d}$ if $t_k \notin \mathcal{G}$ and $g^{(k)} = \underbrace{(0, \ldots, \underset{\uparrow k.}{\kappa_{\mathcal{G}_k}}, 0, \ldots, 0)^T}_{d}$, if $t_k \in \mathcal{G}$.

Now we can compute

$$\sum_{l=1}^{d \cdot |T|} c_{i,l} \cdot g_{l,j} = \kappa_{\mathcal{G}_k} \sum_{k=1}^{|T|} c_{i,j}^{(k)} \underset{\substack{acc.def\ 3.1 \\ \text{for } j \geq 2}}{=} \kappa_{\mathcal{G}_k} \sum_{\substack{d_k = j-1 \\ t_k \in \mathcal{G}}} V(t_k, p_i)$$

□

Now we can prove Theorem 3.1 by induction:

Proof:
Proof by induction on n.

Basis: $n = 1$.

We have to show that for $m^{(0)} \xrightarrow{\mathcal{G}_1} m^{(1)}$ holds:

$$m^{(1)} = m^{(0)} \cdot R^0 + C \cdot \Psi_\sigma = m^{(0)} + C \cdot \Psi_\sigma \text{ with } \Psi_\sigma = \sum_{i=1}^{1} G^{(i)} \cdot R^{n-i} = G^{(1)} \cdot R^0 = G^{(1)}$$

Following, we have to proof that $m^{(1)} = m^{(0)} + C \cdot G^{(1)}$. That is true because of Lemma 3.2.

Step: We consider the firing sequence $\sigma' = (\sigma, \mathcal{G}_{n+1})$ with $m^{(0)} \xrightarrow{\sigma} \hat{m}^{(n)} \underset{1}{\longrightarrow} \tilde{m}^{(n)} \xrightarrow{\mathcal{G}_{n+1}} \hat{m}^{(n+1)}$. We have to show that

$$\hat{m}^{(n+1)} = m^{(0)} \cdot R^n + C \cdot \Psi_{\sigma'} \text{ and } \Psi_{\sigma'} = \sum_{i=1}^{n+1} G^{(i)} \cdot R^{n+1-i}$$

Because of the induction hypothesis it holds:

$$m^{(n)} = m^{(0)} \cdot R^{n-1} + C \cdot \Psi_\sigma = m^{(0)} \cdot R^{n-1} + C \cdot \left(\sum_{i=1}^{n} G^{(i)} \cdot R^{n-1} \right) \quad (2)$$

Because of Lemma 3.1 and resp. Lemma 3.2 it holds too:

$$\tilde{m}^{(n)} = \hat{m}^{(n)} \cdot R \quad (3)$$

resp.

$$\hat{m}^{(n+1)} = \tilde{m}^{(n)} + C \cdot G^{(n+1)} \quad (4)$$

From (2), (3) and (4) it follows:

$$\begin{aligned}
\hat{m}^{(n+1)} &= \tilde{m}^{(n)} + C \cdot G^{(n+1)} \underset{(3)}{=} \hat{m}^{(n)} \cdot R + C \cdot G^{(n+1)} \\
&\underset{(2)}{=} \left(m^{(0)} \cdot R^{n-1} + C \cdot \left(\sum_{i=1}^{n} G^{(i)} R^{n-1} \right) \right) \cdot R + C \cdot G^{(n+1)} \\
&= m^{(0)} \cdot R^n + C \cdot \left(\sum_{i=1}^{n} G^{(i)} R^{n+1-i} \right) + C \cdot G^{(n+1)} \cdot \underbrace{E_d}_{=R^0} \\
&= m^{(0)} \cdot R^n + C \cdot \left(\sum_{i=1}^{n} G^{(i)} R^{n+1-i} + G^{(n+1)} \cdot R^{(n+1)-(n+1)} \right) \\
&= m^{(0)} \cdot R^n + C \cdot \left(\sum_{i=1}^{n+1} G^{(i)} R^{n+1-i} \right) \\
&= m^{(0)} R^n + C \cdot \Psi_{\sigma'}
\end{aligned}$$

□

Corollary 3.1. (from Theorem 3.1)
Let Z be a PDPN, $\sigma = (\mathcal{G}_1, \ldots, \mathcal{G}_n)$ a firing sequence in Z, $m^{(0)}$ the initial time marking of Z and $m^{(0)} \xrightarrow{\mathcal{G}_1} \hat{m}^{(1)} \underset{1}{\longrightarrow} \tilde{m}^{(1)} \xrightarrow{\mathcal{G}_2} \hat{m}^{(2)} \underset{1}{\longrightarrow} \ldots \underset{1}{\longrightarrow} m^{(n)}$. Then it holds:

$$m^{(n)} = m^{(0)} \cdot R^n + C \cdot \Psi_\sigma R \quad (5)$$

Corollary 3.2. Let $m^{(1)} \underset{\tau}{\rightarrow} m^{(2)}$, $\tau \in \mathbb{N}^+$. The following holds:

$$\forall j \left((1 \leq j \leq |P|) \rightarrow \sum_{i=1}^{d} m_{j,i}^{(1)} = \sum_{i=1}^{d} m_{j,i}^{(2)} \right) \quad (6)$$

The following three remarks conclude directly from the Definitions 2.10 and 2.11:

Remark 3.1. (maximum condition for prioritized maximal steps)
Let $m^{(1)} \xrightarrow{\mathcal{B}} m^{(2)}$. Then it is true, that

$$\forall t \left((t \in T \wedge h(t) = 0) \rightarrow m^{(1)}_{\cdot 1} - \sum_{\hat{t} \in \mathcal{B}} \hat{t}^- \not\geq t^- \right) \tag{7}$$

Remark 3.2. (maximum condition for global steps)
Let $m^{(1)} \xrightarrow{\mathcal{G}} m^{(2)}$. Then the following holds:

$$\forall t \left((t \in T, h(t) = 0) \rightarrow m^{(2)}_{\cdot 1} \not\geq t^- \right) \tag{8}$$

Remark 3.3. (priority rule)
Let Z be an PDPN with $t_1, t_2 \in T$, $t_1^- \leq t_2^-$ $\Theta(t_1) \geq \Theta(t_2)$ and $D(t_1) \leq D(t_2)$. Then the following holds:

$$\forall \sigma \left((\sigma = \{\mathcal{G}_1, \ldots, \mathcal{G}_n\}) \rightarrow \sum_{i=1}^{n} \kappa_{\mathcal{G}_i}(t_1) \geq \sum_{i=1}^{n} \kappa_{\mathcal{G}_i}(t_2) \right) \tag{9}$$

Together with the state equation (Theorem 3.1), the three conditions above may be used to test non-reachability in a PDPN, as demonstrated in the next section.

4. An Example

In this section, we show the application of our approach. Please, consider the PDPN from Figure 4. We want to examine the following proposition:

Proposition 4.1. The PDPN Z in Figure 4 with the initial time marking $m^{(0)} = \left(\begin{smallmatrix} 2 & 0 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right)$ will never reach the time marking $m^* = \left(\begin{smallmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right)$.

For a DPN (i.e., in case the priorities would be ignored) the marking m^* is reachable, i.e., the Proposition 4.1 does not hold: Then, the sequence $\sigma = \{\mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3\}$ with $\mathcal{G}_1 = \{t_1, t_2, t_3, t_4, t_4\}$, $\mathcal{G}_2 = \emptyset$, $\mathcal{G}_3 = \{t_2\}$, i.e.,

$$\left(\begin{smallmatrix} 2 & 0 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right) \xrightarrow{\{t_1,t_2\},\{t_3,t_4\},\{t_4\}} \left(\begin{smallmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \end{smallmatrix}\right) \underset{1}{\rightarrow} \left(\begin{smallmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right) \xrightarrow{\emptyset} \left(\begin{smallmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right) \underset{1}{\rightarrow} \left(\begin{smallmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right) \xrightarrow{\{t_2\}} \left(\begin{smallmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right)$$

leads to m^*. Please note, that the first global step consists of a number of maximal steps:

$$\left(\begin{smallmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right) \xrightarrow{\{t_1,t_2\}} \left(\begin{smallmatrix} 0 & 0 & 0 \\ 3 & 0 & 0 \end{smallmatrix}\right) \xrightarrow{\{t_3,t_4\}} \left(\begin{smallmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \end{smallmatrix}\right) \xrightarrow{\{t_4\}} \left(\begin{smallmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right).$$

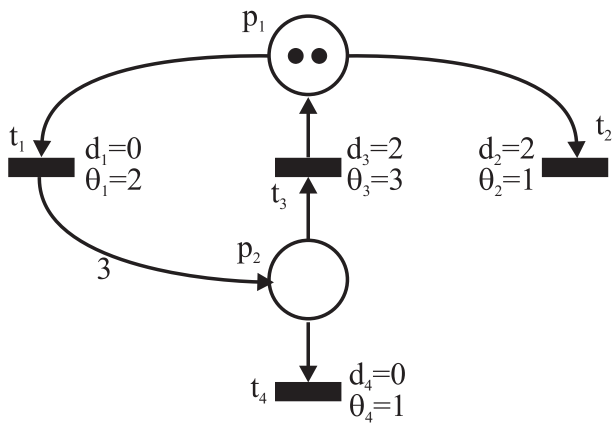

Figure 4. A example PDPN

We prove Proposition 4.1 by contradiction.

Proof:
Assume Proposition 4.1 is wrong. Then, there exists a sequence $\sigma = \{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_n\}$ with $m^{(0)} \xrightarrow{\sigma} m^*$. However, we will show that each sequence that leads to $\left(\begin{smallmatrix}0&0&0\\0&0&0\end{smallmatrix}\right)$ has to start from $\left(\begin{smallmatrix}0&0&0\\0&0&0\end{smallmatrix}\right)$.

It is sufficient to consider the last action in the sequence only. We have to consider two cases.

- Case A: the last action of the sequence is an elapsing of time, and
- Case B: the last action of the sequence is a firing.

In case A, i.e., $\hat{m}^{(n)} \underset{1}{\rightarrow} m^*$, it is easy to see that $\hat{m}^{(n)}$ has to be $\left(\begin{smallmatrix}0&0&0\\0&0&0\end{smallmatrix}\right)$, because of Corollary 3.2 and because a time marking can not contain any element smaller than zero.
To consider case B, we define:

- $x_i = \sum\limits_{j=1}^{n-1} \kappa_{\mathcal{G}_j}(t_i)$, i.e., x_i is the number of firings of the transition t_i within the sequence $\{\mathcal{G}_1, \ldots, \mathcal{G}_{n-1}\}$.

- $\alpha_i = \kappa_{\mathcal{G}_n}(t_i)$, i.e., α_i is the number of firing of the transition t_i within $\mathcal{G}_n$.

- $y_i = \sum\limits_{i=j}^{n-2} \kappa_{\mathcal{G}_j}(t_i)$, i.e., y_i is the number of firing of the transition t_i within the sequence $\{\mathcal{G}_1, \ldots, \mathcal{G}_{n-2}\}$.

- $\beta_i = \kappa_{\mathcal{G}_{n-1}}(t_i)$, i.e., β_i is the number of firing of the transition t_i within $\mathcal{G}_{n-1}$.

And of course, the following is true:

$$x_i = y_i + \beta_i \tag{10}$$

From Remark 3.3 we know:

$$x_1 \geq x_2, y_1 \geq y_2, \alpha_1 \geq \alpha_2, \text{ and } \beta_1 \geq \beta_2 \tag{11}$$

We want to apply the state equation. In our example,

$$C = \begin{pmatrix} -1 & 0 & 0 & & -1 & 0 & 0 & & 0 & 0 & 1 & & 0 & 0 & 0 \\ 3 & 0 & 0 & & 0 & 0 & 0 & & -1 & 0 & 0 & & -1 & 0 & 0 \end{pmatrix}, R = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \text{ and } R^i = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

for all $i > 1$.
Inserted in (1), we get:

$$\begin{aligned}
m^* = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} &= m^{(0)} R^{n-1} + C \cdot \Psi \\
&= m^{(0)} R^{n-1} + C \left(\sum_{i=1}^{n} G^{(i)} R^{n-i} \right) \\
&= \left(\begin{smallmatrix} 2 & 0 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right) \left(\begin{smallmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{smallmatrix}\right) + C \left(\sum_{i=1}^{n-2} G^{(i)} \left(\begin{smallmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{smallmatrix}\right) + G^{(n-1)} \left(\begin{smallmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{smallmatrix}\right) + G^{(n)} \right) \\
&= \begin{pmatrix} 2 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} + C \left(\begin{pmatrix} y_1 & 0 & 0 \\ y_1 & 0 & 0 \\ y_1 & 0 & 0 \\ y_2 & 0 & 0 \\ y_2 & 0 & 0 \\ y_2 & 0 & 0 \\ y_3 & 0 & 0 \\ y_3 & 0 & 0 \\ y_3 & 0 & 0 \\ y_4 & 0 & 0 \\ y_4 & 0 & 0 \\ y_4 & 0 & 0 \end{pmatrix} + \begin{pmatrix} \beta_1 & 0 & 0 \\ \beta_1 & 0 & 0 \\ 0 & \beta_1 & 0 \\ \beta_2 & 0 & 0 \\ \beta_2 & 0 & 0 \\ 0 & \beta_2 & 0 \\ \beta_3 & 0 & 0 \\ \beta_3 & 0 & 0 \\ 0 & \beta_3 & 0 \\ \beta_4 & 0 & 0 \\ \beta_4 & 0 & 0 \\ 0 & \beta_4 & 0 \end{pmatrix} + \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_1 & 0 \\ 0 & 0 & \alpha_1 \\ \alpha_2 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & \alpha_2 \\ \alpha_3 & 0 & 0 \\ 0 & \alpha_3 & 0 \\ 0 & 0 & \alpha_3 \\ \alpha_4 & 0 & 0 \\ 0 & \alpha_4 & 0 \\ 0 & 0 & \alpha_4 \end{pmatrix} \right) \\
&= \begin{pmatrix} 2 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} + C \begin{pmatrix} y_1 + \beta_1 + \alpha_1 & 0 & 0 \\ y_1 + \beta_1 & \alpha_1 & 0 \\ y_1 & \beta_1 & \alpha_1 \\ & \vdots & \\ y_4 + \beta_4 + \alpha_4 & 0 & 0 \\ y_4 + \beta_4 & \alpha_4 & 0 \\ y_4 & \beta_4 & \alpha_4 \end{pmatrix} \\
&= \begin{pmatrix} 2 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} -y_1 - \beta_1 - \alpha_1 - y_2 - \beta_2 - \alpha_2 + y_3 & \beta_3 & \alpha_3 \\ 3y_1 + 3\beta_1 + 3\alpha_1 - y_3 - \beta_3 - \alpha_3 - y_4 - \beta_4 - \alpha_4 & 0 & 0 \end{pmatrix}
\end{aligned}$$

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 2 - y_1 - \beta_1 - \alpha_1 - y_2 - \beta_2 - \alpha_2 + y_3 & \beta_3 & \alpha_3 \\ 3y_1 + 3\beta_1 + 3\alpha_1 - y_3 - \beta_3 - \alpha_3 - y_4 - \beta_4 - \alpha_4 & 0 & 0 \end{pmatrix}$$

We get $\beta_3 = 0$ and therefore by (10): $x_3 = y_3$. In addition, we get $\alpha_3 = 0$. $\alpha_3 = \beta_3 = 0$ means that t_3 does not fire during the last step $\mathcal{G}_n$ or during the step before, $\mathcal{G}_{n-1}$. Thus, after $\mathcal{G}_n$

$$h(t_3) = 0 \tag{12}$$

Also t_4 does not fire during $\mathcal{G}_n$, since t_4 has the same pre-condition as t_3, but a lower priority, i.e., $\alpha_4 = 0$

Assume, t_1 does fire during $\mathcal{G}_n$. Since $d_1 = 0$, $\alpha_3 = 0$, and (12), maximum condition (8) is not met. Therefore, t_1 does not fire during $\mathcal{G}_n$, thus $\alpha_1 = 0$.

Following, because of (11), $\alpha_2 = 0$.

We have shown, that none of the transitions $t_1, \ldots, t_4$ belong to $\mathcal{G}_n$, i.e., $\mathcal{G}_n = \emptyset$. Thus, regarding the state equation (1) for $\tilde{m}^{(n-1)} \xrightarrow{\mathcal{G}_n} m^{(*)}$

$$\begin{aligned} m^* = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} &= \tilde{m}^{(n-1)} \cdot R^0 + C \cdot \begin{pmatrix} \mathcal{O}_d \\ \mathcal{O}_d \\ \mathcal{O}_d \\ \mathcal{O}_d \end{pmatrix} \\ &= \tilde{m}^{(n-1)} \cdot E_d + \mathcal{O}_d \\ &= \tilde{m}^{(n-1)} \end{aligned}$$

i.e.

$$\tilde{m}^{(n-1)} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

That concludes the proof. □

5. Conclusions and Future Work

Within this paper, we have provided an approach that allows for reasoning about Priority Duration Petri nets. We have given sufficient conditions to prove non-reachability, and presented an example.

Currently, we are working on a big-scale application of our approach: the correctness proof of the Message Scheduled System architecture.

In order to reach this goal it is not sufficient to apply our technique just to one specific net, we have to apply it to a class of nets describing generic MSS instances. These nets can be automatically composed out of basic building blocks using the technology presented in [6].

In general, the number of places, transitions and arcs of the PDPN, modeling the MSS architecture, is defined parametrically. This parametrically defined net represents a class of PDPNs.

To proof correctness of the MSS architecture we must show that in this class of nets a specific error state is not reachable if the MSS configuration specified by the net is valid according to the rules of MSS.

6. Acknowledgement

We like to thank the unknown reviewers of this paper for their comments.

References

[1] Lui, C. L., Layland, J. W.: Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Enviroment, *JACM*, **20**(1), January 1973, 46–61.

[2] Marsan, M. A., Balbo, G., Conte, G., Donatelli, S., Franceschinis, G.: *Modelling with Generalized Stochastic Petri Nets*, John Wiley and Sons, 1996.

[3] Popova-Zeugmann, L., Werner, M., Richling, J.: Using state-equation to prove non-reachability in Timed Petrinets, *Fundamenta Informaticae*, 2003.

[4] Ramchandani, C.: Analysis of Asynchronous Concurrent Systems by Timed Petri Nets, *Project MAC-TR 120, MIT*, February 1974.

[5] Richling, J.: Message Scheduled System - A Composable Architecture for Embedded Real-Time-Systems, *Prooceedings of 2000 Int. Conference on Parallel and Distributed Processing techniques and Applications (PDPTA 2000)*, vol. 4, Jun 2000, 2143–2150.

[6] Richling, J., M.Werner, Popova-Zeugmann, L.: Automatic Composition of Timed Petrinet Specifications for a Real-Time Architecture, *Proceedings of 2002 IEEE International Conference on Robotics and Automation*, Washington DC, May 2002.

[7] Richling, J., Popova-Zeugmann, L., Werner, M.: Verification of Non-functional Properties of a Composable Architecture with Petrinets, *Fundamenta Informaticae*, **51**, 2002, 185–200.

[8] Starke, P. H.: *Analyse von Petri-Netz-Modellen*, B.G. Teubner, Stuttgart, 1990.

[9] Starke, P. H.: *A Memo on Time Constraints in Petri Nets*, Informatik-Bericht 46, Humboldt University of Berlin, 1995.