Results on Equivalence, Boundedness, Liveness, and Covering Problems of BPP-Petri Nets

Ernst W. Mayr Jeremias Weihmann

March 29, 2013

Abstract

Yen proposed a construction for a semilinear representation of the reachability set of BPP-Petri nets which can be used to decide the equivalence problem of two BPP-PNs in doubly exponential time. We first address a gap in this construction which therefore does not always represent the reachability set. We propose a solution which is formulated in such a way that a large portion of Yen's construction and proof can be retained, preserving the size of the semilinear representation and the doubly exponential time bound (except for possibly larger values of some constants). In the second part of the paper, we propose very efficient algorithms for several variations of the boundedness and liveness problems of BPP-PNs. For several more complex notions of boundedness, as well as for the covering problem, we show NP-completeness. To demonstrate the contrast between BPP-PNs and a slight generalization regarding edge multiplicities, we show that the complexity of the classical boundedness problem increases from linear time to coNP-hardness. Our results also imply corresponding complexity bounds for related problems for process algebras and (commutative) context-free grammars.

1 Introduction

There is a long tradition of investigating nontrivial subclasses of Petri nets. The reason is not only that many problems of Petri nets have a lower complexity for a number of restricted classes than for general Petri nets but also that the behavior of general Petri nets is still not very well understood. A non primitive recursive algorithm for the reachability problem was given by Mayr [13]. Since then, not much progress was made on the journey of finding a primitive recursive algorithm. In this paper, we investigate a subclass of Petri nets calles Basic Parallel Processes Petri nets (BPP-PNs, also known as communication-free Petri nets). The Petri nets of this class are characterized by the simple topological constraint that each transition has exactly one input place (connected by an edge with multiplicity 1). This class is closely related to both Basic Parallel Processes, a subclass of Milner's Calculus of Communicating Systems (CCS, see, e.g., [1, 2]), as well as (commutative) context-free grammars (see, e.g., [8, 3]).

The strong topological constraint on BPP-PNs limits the computational power of these nets in the sense that they are unable to model synchronizing actions since the fireability of a transition only depends on exactly one place. Esparza [3] showed that the reachability problem of BPP-PNs is, nevertheless, NP-hard. Furthermore, he showed that the problem is in NP. Both results together yield an alternative proof for the NP-completeness of the uniform word problem for commutative context-free grammars (as shown earlier by Huynh [8]).

Another proof for NP-membership, based on canonical firing sequences, was given by Yen [20]. In addition, he proposed an exponential time construction for a semilinear representation of the reachability set of BPP-PNs. He then used this semilinear representation to argue that the equivalence problem for BPP-PNs has a doubly exponential time bound.

In section 3, we address a gap in this construction. We show that, in general, the construction actually computes a proper superset of the reachability set. We then show how to fix the construction in such a way that most parts of Yen's argumentation can be retained while maintaining the size and running time bounds of the original construction (in the sense that all specified constants stay the same).

For some notions of boundedness and liveness of BPPs/BPP-PNs ([11, 14, 15], also see [16]) and for finiteness of context-free grammars ([4]), polynomial time algorithms are already known. In addition to these, we also investigate a number of other variations of the boundedness, the covering, and the liveness problem for BPP-PNs in sections 4 and 5. For two variants of the boundedness problem, and for the covering problem, we show NP-completeness. Using, among other things, results from section 3, we can decide most of the remaining problems very efficiently in linear time. These algorithms are also applicable to related problems of BPPs and (commutative) context-free grammars.

Linear time algorithms not only make these problems tractable in practice but also show that BPP-PNs are too restricted if we are searching for classes of Petri nets where these problems are hard. Further variations and generalizations of BPP-PNs need to be investigated in order to mark the boundary where these problems cease to be easy. As a first example, we show that the classical boundedness problem becomes coNP-hard if we slightly weaken the restriction on the multiplicities of edges from places to transitions in BPP-PNs.

2 Preliminaries

$\mathbb{Z}$, $\mathbb{N}_0$, and $\mathbb{N}$ denote the sets of all integers, all nonnegative integers, and all positive integers, respectively, while $[a,b] = \{a, a+1, \ldots, b\} \subsetneqq \mathbb{Z}$, and $[k] = [1,k] \subsetneqq \mathbb{N}$. For two vectors $u, v \in \mathbb{Z}^k$, we write $u \geq v$ if $u(i) \geq v(i)$ for all $i \in [k]$, and $u > v$ if $u \geq v$ and $u(i) > v(i)$ for some $i \in [k]$. When k is understood, $\vec{a}$ denotes, for a number $a \in \mathbb{Z}$, the k-dimensional vector with $\vec{a}_i = a$ for all $i \in [k]$.

A Petri net N is a 3-tuple (P, T, F) where P is a finite set of n places, T is a finite set of m transitions with $S \cap T = \emptyset$, and $F : P \times T \cup T \times P \to \mathbb{N}_0$ is a flow function. A marking μ (of N) is a function $P \to \mathbb{N}_0$. A pair (N, μ_0) such that μ_0 is a marking of N is called a marked Petri net, and μ_0 is called its initial marking. We will omit the term "marked" if the presence of a certain initial marking is clear from the context.

Throughout this paper, n and m will always refer to the number of places and transitions of the Petri net under consideration. For a transition $t \in T$, ${}^\bullet t$ ($t^\bullet$, resp.) is the preset (postset, resp.) of t and denotes the set of all places p such that $F(p,t) > 0$ ($F(t,p) > 0$, resp.). Analogously, ${}^\bullet p$ and $p^\bullet$ are defined for the places $p \in P$.

A Petri net naturally corresponds to a directed bipartite graph with edges from P to T and vice versa such that there is an edge from $p \in P$ to $t \in T$ (from t to p, resp.) labelled with w if $0 < F(p,t) = w$ (if $0 < F(t,p) = w$, resp.). The label of an edge is called multiplicity. If a Petri net is visualized, places are usually drawn as circles and transitions as bars. If the Petri net is marked by μ, then for each place p the circle corresponding to p contains $\mu(p)$ so called tokens.

For a Petri net $N = (P, T, F)$ and a marking μ of N, a transition $t \in T$ can be applied at μ producing a vector $\mu' \in \mathbb{Z}^n$ with $\mu'(p) = \mu(p) - F(p,t) + F(t,p)$ for all $p \in P$. The transition t is enabled at μ or in (N, μ) if $\mu(p) \geq F(p,t)$ for all $p \in P$. We say that t is fired at marking μ if t is enabled and applied at μ. If t is fired at μ, then the produced vector μ' is a marking, and we write $\mu \xrightarrow{t} \mu'$.

Intuitively, if a transition is fired, it first removes $F(p,t)$ tokens from p and then adds $F(t,p)$ tokens to p. An element of T^* is called a transition sequence, an element of T^∞ is called an ∞-transition sequence. For the empty sequence $\sigma = ()$ of transitions, we define $\mu \xrightarrow{\sigma} \mu$. For a nonempty transition sequence $\sigma = (t_1, \ldots, t_k)$, we write $\mu_0 \xrightarrow{\sigma} \mu_k$ if there are markings $\mu_1, \ldots, \mu_{k-1}$ such that $\mu_0 \xrightarrow{t_1} \mu_1 \xrightarrow{t_2} \mu_2 \ldots \xrightarrow{t_k} \mu_k$.

The Parikh map $\Psi : T^* \to \mathbb{N}_0^m$ maps a transition sequence σ to its Parikh image $\Psi[\sigma]$ where $\Psi[\sigma](t) = k$ for a transition t if t appears exactly k times in σ. A Parikh vector is simply an element of $\mathbb{N}_0^m$ (hence each Parikh vector is the Parikh image of some transition sequence). For a Parikh vector Φ we write $t \in \Phi$ if $\Phi(t) > 0$, and $t \in \sigma$ if $t \in \Psi[\sigma]$.

If there is a marking μ' such that $\mu \xrightarrow{\sigma} \mu'$, then we say that σ (the Parikh vector $\Psi[\sigma]$, resp.) is enabled at μ and leads from μ to μ'. For a marked Petri net (N, μ_0), we call a transition sequence that is enabled at μ_0 a firing sequence, and we say that a marking μ is reachable if there is a firing sequence leading to μ. Analogously, an ∞-transition sequence σ is enabled at μ if each finite prefix of σ is enabled at μ. If σ is enabled at μ_0, we call σ an ∞-firing sequence. The reachability set $\mathcal{R}(N, \mu_0)$ of (N, μ_0) consists of all markings μ of N for which there is a firing sequence σ such that $\mu_0 \xrightarrow{\sigma} \mu$. We say that a marking μ can be covered or μ is coverable if there is a reachable marking $\mu' \geq \mu$.

The displacement $\Delta : \mathbb{N}_0^m \to \mathbb{Z}_0^n$ maps Parikh vectors $\Phi \in \mathbb{N}_0^m$ onto the change of tokens at the places $p_1, \ldots, p_n$ when applying transition sequences with Parikh image Φ. That is, we have $\Delta[\Phi](p) = \sum_{t \in T} \Phi(t) \cdot (F(t,p) - F(p,t))$ for all places p. Accordingly, we define the displacement $\Delta[\sigma]$ of a transition sequence σ by $\Delta[\sigma] := \Delta[\Psi[\sigma]]$. A Parikh vector or a transition sequence having nonnegative displacement at all places is called a nonnegative loop since, immediately after being fired, the loop is enabled again. A nonnegative loop having positive displacement at place p is a positive loop (for p).

Sometimes it is convenient to only consider those places and transitions that are relevant w.r.t. a given set of transitions or a Parikh vector. For a Petri net $\mathcal{P} = (P, T, F, \mu_0)$, and a set D of transitions, the Petri net $\mathcal{P}_D$ consists of all transitions $t \in D$, all places $p \in \bigcup_{t \in D} ({}^\bullet t \cup t^\bullet)$, and the flow function F and initial marking μ_0 restricted to these subsets of transitions and places. For a Parikh vector Φ we define $\mathcal{P}_\Phi := \mathcal{P}_D$ where $D = \{t \mid t \in \Phi\}$.

In the case of BPP-PNs the strongly connected components (SCCs) are also of major interest. The directed acyclic graph obtained by shrinking all SCCs to super nodes while maintaining the edges between distinct SCC as edges between the corresponding super nodes is called the condensation (of the graph). We call an SCC C a top component (bottom component, resp.) if it has no incoming (no outgoing, resp.) edges in the condensation. For two not necessarily distinct SCCs C_1, C_2, we write $C_1 \geq C_2$ if there is a path from C_1 to C_2 in the condensation.

An important concept in the analysis of Petri nets are traps. A subset $T \subseteq P$ of places is a trap if ${}^\bullet t \cap T \neq \emptyset$ implies $t^\bullet \cap T \neq \emptyset$, i.e., every transition that removes a token from T also adds a token to T. Once a trap is marked, it cannot be unmarked by firing a transition.

Some marked Petri nets have reachability sets that are semilinear. A set $S \subseteq \mathbb{N}_0^n$ is semilinear if there is a finite number of linear sets $L_1, \ldots, L_k \subseteq \mathbb{N}_0^n$ such that $S = \bigcup_{i \in [k]} L_i$. A set $L \subseteq \mathbb{N}_0^n$ is linear if there is a finite number of vectors $b, p_1, \ldots, p_\ell \in \mathbb{N}_0^n$ such that $L = \mathcal{L}(b, \{p_1, \ldots, p_\ell\})$ where $\mathcal{L}(b, \{p_1, \ldots, p_\ell\}) := \{b + \sum_{i \in [\ell]} a_i p_i \mid a_i \in \mathbb{N}_0, i \in [\ell]\}$. The vector b is the constant vector of L while the vectors p_i are the periods of L. A semilinear representation of a semilinear set S is a set consisting of k pairs $(b_i, \{p_{i,1}, \ldots, p_{i,\ell_i}\})$, $i \in [k]$, such that $S = \bigcup_{i \in [k]} \mathcal{L}(b_i, \{p_{i,1}, \ldots, p_{i,\ell_i}\})$.

If two Petri nets allow the construction of semilinear representations of the respective reachability sets within a certain time bound, then not only many problems that are in general undecidable are decidable for this class but time bounds can be given as well. For example, the equivalence problem of (conveniently encoded representations of) semilinear sets is in Π_2^P [7, 9]. This implies that the equivalence problem for that class, i.e., the question if two Petri nets of the class have the same set of reachable markings, is decidable in exponential time w.r.t. the combined time to construct the semilinear sets.

When we talk about the input size, a "reasonable" encoding/description is assumed. We specifically assume that every number is encoded in binary representation. Furthermore, we assume for convenience that a Petri net is encoded as an enumeration of places $p_1, \ldots, p_n$ and transitions $t_1 \ldots, t_m$ followed by an enumeration of the edges with their respective edge multiplicities. A vector of $\mathbb{N}_0^k$ is encoded as a k-tuple. If we regard a tuple as an input (e.g. a marked Petri net), then it is encoded as a tuple of the encodings of the particular components. All running times given in later sections assume the random-access machine (RAM) as the model of computation.

3 The Equivalence Problem of BPP-Petri Nets Revisited

In this section we consider the equivalence problem of BPP-PNs.

Definition 3.1 (Equivalence problem of BPP-PNs)**.** Given two BPP-PNs $\mathcal{P}$ and $\mathcal{P}'$, are $\mathcal{R}(\mathcal{P})$ and $\mathcal{R}(\mathcal{P}')$ equal?

In [20], Yen proposed a construction for a semilinear representation of the reachability set of BPP-PNs. The obtained representation has exponential size in the size of the BPP-PN. The author used the fact that the equivalence problem of semilinear sets is in Π_2^P (see [7, 9]) to show a double exponential time bound for this problem. The construction of the semilinear representation is contained in the proposed proof of the following theorem.

Theorem 3.2 ([20], Theorem 5)**.** *Let $\mathcal{P} = (P, T, F, \mu_0)$ be a BPP-Petri net of size s. For some fixed constants c_1, c_2, d_1, d_2, d_3 independent of s, we can construct in DTIME$(2^{c_2 s^3})$ a semilinear reachability set $\mathcal{R}(\mathcal{P}) = \bigcup_{\nu \in B} \mathcal{L}(\nu, \rho_\nu)$ whose size is bounded by $\mathcal{O}(2^{c_1 s^3})$, where*

1. *B is the set of reachable markings with no component larger than $2^{d_1 s^2}$, and*
2. *ρ_ν is the set of all $\vartheta \in \mathbb{N}^k$ such that*
   (a) *ϑ has no component larger than $2^{d_2 s^2}$, and*
   (b) *$\exists\ \sigma, \sigma_1, \sigma_2 \in T^*$, $\exists$ marking μ_1,*
      (i) $\mu_0 \xrightarrow{\sigma_1} \mu_1 \xrightarrow{\sigma_2} \nu$,
      (ii) $\mu_1 \xrightarrow{\sigma} \mu_1 + \vartheta$,
      (iii) $|\sigma|, |\sigma_1\sigma_2| \leq 2^{d_3 s^2}$.

We show that there are BPP-PNs such that the constructed semilinear set contains markings that are not reachable. Consider the BPP-PN $\mathcal{P}$ with initial marking $\mu_0 = (1, 0, 0, 0)$ of Figure 1. The marking $\nu = (0, 0, 0, 1)$ is reachable.
In particular we have $\mu_0 \xrightarrow{t_1} \mu_1 = (0, 1, 0, 0) \xrightarrow{t_2} \nu$ as well as $\mu_0 \xrightarrow{t_3} \mu_1' = (0, 0, 1, 0) \xrightarrow{t_4} \nu$. Notice that we can safely and w.l.o.g. assume $\nu \in B$ since we can blow up the size of the net by adding unrelated places. Now observe that $\mu_1 \xrightarrow{t_5} \mu_1 + \vartheta$ where $\vartheta = (0, 1, 0, 0)$, as well as $\mu_1' \xrightarrow{t_6} \mu_1' + \vartheta'$ where $\vartheta' = (0, 0, 1, 0)$. As before, we can safely assume $|t_1 t_2|, |t_3 t_4|, |t_5|, |t_6| \leq 2^{d_3 s^2}$. Therefore, we find $\vartheta, \vartheta' \in \rho_\nu$. But then, the unreachable marking $(0, 1, 1, 1)$ is in $\mathcal{L}(\nu, \rho_\nu)$. Hence, the constructed semilinear set $S := \bigcup_{\nu \in B} \mathcal{L}(\nu, \rho_\nu)$ cannot equal $\mathcal{R}(\mathcal{P})$.

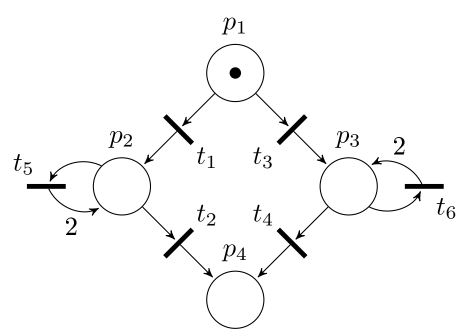

Figure 1: A counter example for the construction proposed in Theorem 3.2.

The inclusion $\mathcal{R}(\mathcal{P}) \subseteq S$ is proven correctly in [20]. Our goal is to repair the construction in such a way that we can more or less completely reuse the proof given for this direction. Our first step is to show that there is a certain subclass of BPP-PNs for which the other direction $S \subseteq \mathcal{R}(\mathcal{P})$ is also true. To this end, observe that the crucial property of the net of Figure 1 that makes this net a counter example is that ν is reachable by the two firing sequences $t_1 t_2$ and $t_3 t_4$ which have different Parikh images. We will later show that the restriction to those BPP-PNs having the nice property that any two firing sequences leading to the same marking have the same Parikh image yields a variation of this theorem which is correct.

Before we can prove such a theorem, we first need some observations about enabled Parikh vectors and nonnegative loops in BPP-PNs.

Lemma 3.3. *Let $\mathcal{P} = (N, \mu_0)$ be a BPP-PN. A Parikh vector Φ is enabled in $\mathcal{P}$ if and only if*

(a) *$\mu_0 + \Delta[\Phi] \geq \vec{0}$, and*

(b) *each top component of $\mathcal{P}_\Phi$ has a marked place.*

Proof. This Lemma is a variation of Theorem 3.1 of [3] which is better suited for our purposes. The theorem states that Φ is enabled if and only if (a) holds and if, within $\mathcal{P}_\Phi$, each place can be reached from a marked place. $\square$

Lemma 3.4. *Let $\mathcal{P} = (P, T, F, \mu_0)$ be a Petri net, $\sigma = \sigma_1 \cdots \sigma_k \in T^k$ a firing sequence in $\mathcal{P}$, and let μ_i, $i \in [k]$, be defined by $\mu_0 \xrightarrow{\sigma_1} \mu_1 \xrightarrow{\sigma_2} \ldots \xrightarrow{\sigma_k} \mu_k$. Then, for each place p of $\mathcal{P}_{\Psi[\sigma]}$, there is an $i \in [0, k]$ such that p is marked in μ_i.*

Proof. Each place p of $\mathcal{P}_{\Psi[\sigma]}$ is in the pre- or postset of some transition σ_i. If $p \in {}^\bullet\sigma_i$, then p must be marked in μ_{i-1}. If $p \in \sigma_i{}^\bullet$, then p is marked in μ_i. $\square$

Lemma 3.5. *Let Φ be a nonnegative loop of a BPP-PN $\mathcal{P} = (P, T, F)$, and let $C_1, \ldots, C_k$, $k \geq 1$, denote the top components of $\mathcal{P}_\Phi$. Then Φ can be split into nonnegative loops $\Phi_1, \ldots, \Phi_k$, $k \leq n$, such that*

(a) $\Phi = \sum_{i=1}^{k} \Phi_i$, *and*

(b) *the only top component of $\mathcal{P}_{\Phi_i}$ is C_i.*

Proof. For $\Phi = \vec{0}$, the lemma is obviously true, hence we assume $\Phi > \vec{0}$. We show that we can extract Φ_1 from Φ. By iteratively applying this procedure, we obtain the nonnegative loops of interest. $\mathcal{P}_{\Phi_1}$ will contain C_1 and all nodes that are reachable from C_1. Since it is possible that firing a transition t of an SCC C with $C \leq C_1, C_2$ and $C \neq C_1, C_2$ exactly $\Phi(t)$ times requires tokens coming from C_1 and C_2, $\mathcal{P}_{\Phi_1}$ and $\mathcal{P}_{\Phi-\Phi_1}$ will in general not be disjoint.

We first note that a top component of $\mathcal{P}_\Phi$ always contains a transition since otherwise there would be a transition of Φ that removes tokens from the component but no transition that adds tokens to it, a contradiction to Φ being a nonnegative loop. Let $\Phi' = \Phi$ and $\Phi_1 = \vec{0}$. By moving a transition t from Φ' to Φ_1 we mean setting $\Phi'(t) := \Phi'(t) - 1$ and $\Phi_1(t) := \Phi_1(t) + 1$.

We start with a top component C_1 of $\mathcal{P}_\Phi$, and move each transition $t \in C_1$ exactly $\Phi'(t)$ times from Φ' to Φ_1. Then we iterate the following process:

(i) If $\Delta[\Phi_1](p) > 0$ for a place p and $p = {}^\bullet t$ for a transition $t \in \Phi'$, then we move t from Φ' to Φ_1.

(ii) Otherwise, if there is a top component of $\mathcal{P}_{\Phi'}$ that is not a top component of $\mathcal{P}_\Phi$, we move each transition t of this component exactly $\Phi'(t)$ times from Φ' to Φ_1.

The procedure ends when none of these two cases is applicable.

We first prove that at each step of the procedure the only top component of $\mathcal{P}_{\Phi_1}$ is C_1. After the first step this is obviously true since we completely move C_1. Assume this holds for $\ell - 1$ steps. If the ℓ-th step is of case (i), then it holds after ℓ steps since the only nodes that are possibly added to the induced graph $\mathcal{P}_{\Phi_1}$ by moving t are t and $t^\bullet$ which can be reached by the place ${}^\bullet t$ which is already part of $\mathcal{P}_{\Phi_1}$ before moving t. If the ℓ-th step is of case (ii), then the moved component C was originally created by moving a transition t such that $t^\bullet$ and C share a place. This shows that $\mathcal{P}_{\Phi_1}$ has only one top component after the last step of the procedure.

Observe that the top components of $\mathcal{P}_{\Phi'}$ are exactly $C_2, \ldots, C_k$. The reason is that C_1 is moved, $C_2, \ldots, C_k$ remain untouched, and (ii) ensures the moving of all newly created top components.

Now, we show that $\Delta[\Phi_1] \geq \vec{0}$ holds *at each step of the procedure*. After the first step, i.e., after moving the top component, this holds since otherwise Φ wouldn't be a nonnegative loop. Suppose this holds after $\ell - 1$ steps. If the ℓ-th step is of case (i), then it obviously still holds after that.

Suppose, the ℓ-th step is of case (ii), where C is the new top component that is moved during this step. Consider the situation immediately *before the ℓ-th step*. Let Φ_C be defined by $\Phi_C(t) = \Phi'(t)$ if $t \in C$, and $\Phi_C(t) = 0$ otherwise. Our goal is to show that $\Delta[\Phi_C] \geq \vec{0}$ since this and the induction hypothesis imply $\Delta[\Phi_1 + \Phi_C] \geq \vec{0}$, i.e., after moving all transitions of C in the ℓ-th step, the resulting Parikh vector is a nonnegative loop.

First notice that for all places $p \notin C$ we have $\Delta[\Phi_C](p) \geq 0$. (This follows from the fact that C is a transition induced SCC, implying ${}^\bullet t \in C$ for all $t \in C$.)

Consider a place $p \in C$. By the induction hypothesis, we have $\Delta[\Phi_1](p) \geq 0$. $\Delta[\Phi_1](p) > 0$ cannot occur since otherwise the ℓ-th step would be of case (i) (applied to a transition $t \in C$ having $p = {}^\bullet t$). Thus, we have $\Delta[\Phi_1](p) = 0$.

Now, observe that for all $t \in \Phi'$ having $t^\bullet \in C$ we have $t \in \Phi_C$ since C is a top component. This implies $\Delta[\Phi_C](p) \geq \Delta[\Phi'](p)$. Combining all these observations we obtain

$$\Delta[\Phi_C](p) = \Delta[\Phi_1 + \Phi_C](p) \geq \Delta[\Phi_1 + \Phi'](p) = \Delta[\Phi](p) \geq 0.$$

Now, we show that Φ' is a nonnegative loop *at the end of the procedure*. Let p be a place, and consider the situation *after the last step*. If $\Delta[\Phi_1](p) > 0$, then there is no transition $t \in \Phi'$ having ${}^\bullet t = p$ since otherwise (i) would be applicable, and the procedure wouldn't have stopped, yet. This implies $\Delta[\Phi'](p) \geq 0$. If $\Delta[\Phi_1](p) = 0$, then $\Delta[\Phi'](p) \geq 0$ follows from $\Delta[\Phi_1 + \Phi'] = \Delta[\Phi] \geq \vec{0}$. As shown above, the case $\Delta[\Phi_1](p) < 0$ cannot occur.

Since ${}^\bullet t \neq \emptyset$ for all transitions t, each top component of $\mathcal{P}_\Phi$ contains at least one place. This implies $k \leq n$, concluding the proof. □

Lemma 3.6. *Let $\mathcal{P} = (P, T, F, \mu_0)$ be a BPP-PN, and Φ, ϑ Parikh vectors such that ϑ is a nonnegative loop, and Φ and $\Phi + \vartheta$ are enabled. Then, for each firing sequence α such that $\mathcal{P}_\Phi$ is a subnet of $\mathcal{P}_{\Psi[\alpha]}$, there are transition sequences $\alpha_1, \ldots, \alpha_{k+1}$ and nonnegative loops $\tau_1, \ldots, \tau_k$, $k \leq n$, such that*

(a) $\alpha = \alpha_1 \cdots \alpha_{k+1}$,

(b) $\vartheta = \tau_1 + \ldots + \tau_k$,

(c) $\mathcal{P}_{\tau_i}$, $i \in [k]$, *has exactly one top component, and this top component is the i-th top component of $\mathcal{P}_\vartheta$ using a properly chosen numbering of the top components, and*

(d) τ_i, $i \in [k]$, *is enabled at marking μ_i where $\mu_0 \xrightarrow{\alpha_1 \cdots \alpha_i} \mu_i$.*

Proof. Consider the decomposition of ϑ by Lemma 3.5 into nonnegative loops $\tau_1, \ldots, \tau_k$, $k \leq n$, such that $\vartheta = \sum_{i=1}^k \tau_i$, and the i-th top component C_i of $\mathcal{P}_\vartheta$ is the unique top component of $\mathcal{P}_{\tau_i}$.

Let $i \in [k]$. Assume that C_i and $\mathcal{P}_\Phi$ are disjoint. Then, C_i is a top component of $\mathcal{P}_{\Phi+\vartheta}$, and C_i is marked at μ_0 by Lemma 3.3 since $\Phi + \vartheta$ is enabled at μ_0. Therefore, by the same lemma, τ_i is enabled at μ_0.

Now, assume that C_i and $\mathcal{P}_\Phi$ are not disjoint, i.e., they share a place p. Since $\mathcal{P}_\Phi$ is a subnet of $\mathcal{P}_{\Psi[\alpha]}$, Lemma 3.4 implies that there are transition sequences α', α'' such that $\alpha = \alpha' \cdot \alpha''$ and p is marked at μ' where $\mu_0 \xrightarrow{\alpha'} \mu'$. Therefore, by Lemma 3.3, τ_i is enabled at μ' .

We conclude that, by splitting the sequence α at appropriate positions, there are transition sequences $\alpha_1, \ldots, \alpha_{k+1}$ such that $\alpha = \alpha_1 \alpha_2 \cdots \alpha_{k+1}$, and $\mu_0 \xrightarrow{\alpha_1} \mu_1 \cdots \xrightarrow{\alpha_k} \mu_k \xrightarrow{\alpha_{k+1}} \mu$, and τ_i is enabled at μ_i where we assume w.l.o.g. that the top components of $\mathcal{P}_\vartheta$ are conveniently numbered. □

Having collected and proven these observations, we can show the following restricted variation of Theorem 3.2.

Theorem 3.7. *Let $\mathcal{P} = (P, T, F, \mu_0)$ be a BPP-Petri net of size s such that for all firing sequences τ, τ' leading to the same marking $\Psi(\tau) = \Psi(\tau')$ holds. For some fixed constants c_1, c_2, d_1, d_2, d_3 independent of s, we can construct in DTIME($2^{c_2 s^3}$) a semilinear reachability set $\mathcal{R}(\mathcal{P}) = \bigcup_{\nu \in B} \mathcal{L}(\nu, \rho_\nu)$ whose size is bounded by $\mathcal{O}(2^{c_1 s^3})$, where*

1. *B is the set of reachable markings with no component larger than $2^{d_1 s^2}$, and*

2. *ρ_ν is the set of all $\vartheta \in \mathbb{N}^k$ such that*

   (a) *ϑ has no component larger than $2^{d_2 s^2}$, and*

   (b) *$\exists\ \sigma, \sigma_1, \sigma_2 \in T^*$, $\exists$ marking μ_1,*

(i) $\mu_0 \xrightarrow{\sigma_1} \mu_1 \xrightarrow{\sigma_2} \nu$,

(ii) $\mu_1 \xrightarrow{\sigma} \mu_1 + \vartheta$,

(iii) $|\sigma|, |\sigma_1\sigma_2| \leq 2^{d_3 s^2}$.

Proof. Assume $B \neq \emptyset$, and let $\nu \in B$, and $\mu \in \mathcal{L}(\nu, \rho_\nu)$ be arbitrarily chosen. Our goal is to show that μ is reachable. W.l.o.g. let $\rho_\nu = \{\vartheta_1, \ldots, \vartheta_\ell\}$. By definition, there are $a_1, \ldots, a_\ell \in \mathbb{N}_0$ such that $\mu = \nu + \sum_{i=1}^{\ell} a_i \vartheta_i$. For ϑ_i, $i \in [\ell]$, let $\sigma_{i,1}$ denote the sequence σ_1 as defined in the theorem. Since all firing sequences leading to ν have the same Parikh image Φ_ν, $\mathcal{P}_{\Psi[\sigma_{i,1}]}$ is a subnet of $\mathcal{P}_{\Phi_\nu}$.

Let α be some firing sequence having Parikh image Φ_ν, and let μ_j, $j \in [|\alpha|]$, be defined by $\mu_0 \xrightarrow{\alpha_1 \cdots \alpha_j} \mu_j$. By applying Lemma 3.6 to $\Psi[\sigma_{i,1}]$, θ_i, and α for all $i \in [\ell]$, we find that for any partial loop of any θ_i there is a $j \in [0, |\alpha|]$ such that the partial loop under consideration is enabled at μ_j. Therefore, the Parikh vector $\Psi[\alpha] + \sum_{i=1}^{\ell} a_i \vartheta_i$ leading to μ is enabled at μ_0. $\square$

We can use this theorem and corresponding construction in a mediate way to construct a semilinear representation of $\mathcal{R}(\mathcal{P})$ in exponential time for every BPP-PN $\mathcal{P}$. For that we need the following definition.

Definition 3.8 (Parikh extension)**.** Let $\mathcal{P} = (P, T, F, \mu_0)$, $P = \{p_1, \ldots, p_n\}$, $T = \{t_1, \ldots, t_m\}$ be a Petri net. The Parikh extension $\mathcal{P}^e = (P^e, T, F^e, \mu_0^e)$ of $\mathcal{P}$ is obtained by adding an unmarked place p_i^* for each transition t_i such that $F(t_i, p_i^*) = 1$.

Figure 2 illustrates the Parikh extension of the net of Figure 1. If we fire a firing sequence σ, $\mu_0' \xrightarrow{\sigma} \mu_1$, in the Parikh extension $\mathcal{P}^e$, then the new place p_i^*, $i \in [m]$, counts how often the transition t_i is fired. In other words, the projection of μ_1 onto the new places equals $\Psi[\sigma]$. Hence, for each marking μ reachable in $\mathcal{P}^e$ all firing sequences leading to μ have the same Parikh image. This allows us to prove the next theorem. We remark that the concept of the Parikh extension is closely related to the concept of extended Parikh maps used in [12] for persistent Petri nets.

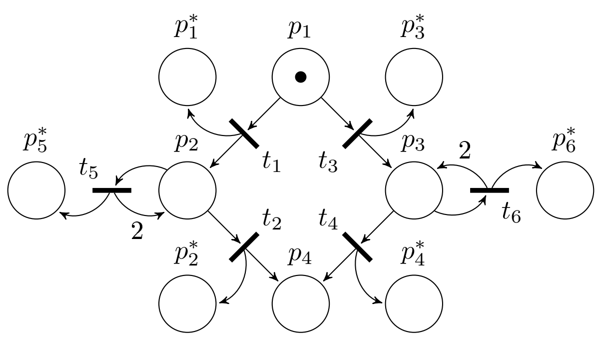

Figure 2: The Parikh extension of the net of Figure 1.

Theorem 3.9. *Let $\mathcal{P} = (P, T, F, \mu_0)$ be a BPP-Petri net of size s. For some fixed constants c_1, c_2, d_1, d_2 independent of s, we can construct in DTIME$(2^{c_2 s^3})$ a semilinear representation of the reachability set $\mathcal{R}(\mathcal{P})$ whose size is bounded by $\mathcal{O}(2^{c_1 s^3})$ where no component of any constant vector is larger than $2^{d_1 s^2}$ and no component of any period is larger than $2^{d_2 s^2}$.*

Proof. We compute the Parikh extension $\mathcal{P}^e$ of $\mathcal{P}$. First notice that $\mathcal{P}^e$ is a BPP-PN. Since all firing sequences of $\mathcal{P}^e$ leading to the same marking have the same Parikh image, we can apply the construction given in [20], which is correct for $\mathcal{P}^e$ by Theorem 3.7, in order to obtain the semilinear representation $\mathcal{SL}(\mathcal{P}^e)$ of $\mathcal{R}(\mathcal{P}^e)$. Now notice that a marking μ is reachable in $\mathcal{P}$ if and only if there is a marking μ' that is reachable in $\mathcal{P}^e$ such that the projection of μ' onto the places of $\mathcal{P}$ equals μ (to see this, simply apply the same firing sequence). Therefore, the projection of $\mathcal{SL}(\mathcal{P}^e)$ onto the places of $\mathcal{P}$ yields the semilinear representation $\mathcal{SL}(\mathcal{P})$ of $\mathcal{R}(\mathcal{P})$.

The running time of this projection is linear in the size of $\mathcal{SL}(\mathcal{P}^e)$. In turn, the size of $\mathcal{P}^e$ is linear in the size of $\mathcal{P}$. Hence, the constants c_1, c_2, d_1, d_2 may be larger for this theorem than for Theorem 3.7 but all specified constants (like the cube of s^3) are not increased. $\square$

In the next sections we investigate several variations of the boundedness, and the liveness problem for BPP-PNs. In addition we show that the covering problem is NP-complete for BPP-PNs.

4 Boundedness Problems for BPP-PNs

We first define the concepts of boundedness we are interested in.

Definition 4.1. Let $\mathcal{P} = (P, T, F, \mu_0)$ be a Petri net, and $R \subseteq P$. A place $p \in P$ with $p \notin R$ is

(i) empty-R-unbounded if, for all $k \in \mathbb{N}$, there is a reachable marking $\mu \in \mathcal{R}(\mathcal{P})$ such that $\mu(p) \geq k$ and $\mu(r) = 0$ for all $r \in R$.

(ii) unbounded if p is empty-$\emptyset$-unbounded.

(iii) unbounded on an ∞-firing sequence σ if, for all $k \in \mathbb{N}$, there is a finite prefix of σ leading to a marking μ such that $\mu(p) \geq k$.

(iv) persistently unbounded if, for all reachable markings $\mu \in \mathcal{R}(\mathcal{P})$, p is unbounded in the Petri net (P, T, F, μ).

For a set $R \subseteq P$, a set $S \subseteq P$ with $S \cap R = \emptyset$ is

(iv) (placewise) empty-R-unbounded if some place (all places, respectively) of S are empty-R-unbounded.

(v) (∞-)unbounded if S contains a place that is unbounded (on an ∞-firing sequence, respectively).

(vi) placewise (∞-)unbounded if all places of S are unbounded (on an ∞-firing sequence, respectively).

(vii) simultaneously unbounded if, for all $k \in \mathbb{N}$, there is a reachable marking $\mu \in \mathcal{R}(\mathcal{P})$ such that $\mu(p) \geq k$ for all $p \in S$.

(viii) simultaneously ∞-unbounded if there is an ∞-firing sequence σ such that, for all $k \in \mathbb{N}$, there is a finite prefix of σ leading to a marking μ satisfying $\mu(p) \geq k$ for all $p \in S$.

We remark that, for a place, "universally unbounded" implies "persistently unbounded" which implies "unbounded on an ∞-firing sequence" which implies "unbounded". Further, by Lemma 3.2 of [12] a set $S \subseteq P$ of places is simultaneously unbounded on some ∞-firing sequence if and only if there is an ∞-firing sequence σ such that all places $p \in S$ are unbounded on (the same sequence) σ. Hence, this on first sight weaker characterization yields another definition for the same concept.

4.1 Concepts of Non-Simultaneously Unboundedness and Related Problems

In this subsection we investigate concepts of unboundedness where the places under consideration are not required to be simultaneously (∞-)unbounded, and provide efficient algorithms for the corresponding decision problems. In addition, efficient algorithms for related problems for Basic Parallel Processes and (commutative) context-free grammars are proposed.

Lemma 4.2. *Let $\mathcal{P} = (P, T, F, \mu_0)$ be a BPP-PN, and $p \in P$ a place. Then the following are equivalent.*

1. *p is unbounded.*

2. *There is a firing sequence σ leading to a marking μ and a positive loop τ enabled at μ such that $\Delta[\tau](p) > 0$.*

3. *p is unbounded on some ∞-firing sequence.*

4. *There are strongly connected components C_1, C_2, C_3, C_4 of $\mathcal{P}$ such that*

   (a) *$p \in C_4$,*

   (b) *$C_1 \geq C_2 \geq C_3 \geq C_4$,*

   (c) *C_1 contains a marked place, and*

   (d) *C_2 contains a transition t with ${}^\bullet t \in C_2$ and $\sum_{p' \in t^\bullet \cap (C_2 \cup C_3)} F(t, p') \geq 2$.*

Proof. $1 \Rightarrow 2$: By definition, there is an infinite sequence of enabled Parikh vectors $(\Phi'_1, \Phi'_2, \ldots)$ such that $\Delta[\Phi'_i](p) < \Delta[\Phi'_{i+1}](p)$, $i \in \mathbb{N}$. It is easy to see that this sequence contains an infinite subsequence $(\Phi_1, \Phi_2, \ldots)$ such that $\Phi_i \leq \Phi_{i+1}$, $i \in \mathbb{N}$ (see, e.g., Lemma 4.1. of [10]). In particular, we have $\Delta[\Phi_1](p) < \Delta[\Phi_2](p)$, i.e., there is a positive loop ϑ such that $\Delta[\vartheta](p) > 0$ and $\Phi_1 + \vartheta = \Phi_2$. Since both Φ_1 and Φ_2 are enabled, we can apply Lemma 3.6 to Φ_1, ϑ and some firing sequence α having Parikh image Φ_1. Let $\alpha_1, \ldots, \alpha_{k+1}$ and $\tau_1, \ldots, \tau_k$ be defined as in the lemma. Then we have $\Delta[\tau_i](p) > 0$ for some $i \in [k]$. Let $\tau := \tau_i$. For $\mu_0 \xrightarrow{\alpha_1 \cdots \alpha_i} \mu$, τ is enabled at μ, concluding the proof.

$1 \Rightarrow 4$: We continue where the proof for $1 \Rightarrow 2$ ended. Let C'_2 be the unique top component of $\mathcal{P}_{\Psi[\tau]}$, and C'_4 the SCC of $\mathcal{P}_{\Psi[\tau]}$ containing p. Since τ is enabled at μ, by Lemma 3.3 there are places p_1 and p_2 such that p_1 is marked at μ_0, $\mathcal{P}$ contains a path from p_1 to p_2, p_2 is contained in C'_2, and $\mu(p_2) > 0$. Define C_1 as the SCC of $\mathcal{P}$ containing p_1.

Since τ is a nonnegative loop, C'_2 contains a transition. If there is a transition t of C'_2 such that $\sum_{p' \in t^\bullet \cap C'_2} F(t, p') \geq 2$, then simply define $C'_3 = C'_2$. Now, assume that such a transition doesn't exist. Then, we have $C'_4 \neq C'_2$ since the total number of tokens in C'_2 cannot increase by firing τ. In particular, there is a path $(p_2, t, p_3, \ldots, p)$ from C'_2 to C'_4 where $p_3 \notin C'_2$. Let C'_3 be the SCC of $\mathcal{P}_{\Psi[\tau]}$ containing p_3. If $t^\bullet \cap C'_2 = \emptyset$, then τ decreases the number of tokens at C'_2, a contradiction to τ being a nonnegative loop. Therefore, $t^\bullet \cap C'_2 \neq \emptyset$, and we obtain $\sum_{p' \in t^\bullet \cap (C'_2 \cup C'_3)} F(t, p') \geq 2$. Now, let C_i for $i \in [2, 4]$ be the SCC of $\mathcal{P}$ containing C'_i, and observe that $C_1, \ldots, C_4$ satisfy the properties (a)–(d).

$2 \Rightarrow 3$: p is unbounded on the ∞-firing sequence $\sigma\tau^\infty$.

$3 \Rightarrow 1$: This follows immediately from the definitions.

$4 \Rightarrow 1$: To mark ${}^\bullet t$, we first fire along a path starting at a marked place of C_1 and ending at ${}^\bullet t$. Then we fire $k \in \mathbb{N}$ times along a cycle containing t. After that, at least k tokens can be transferred to p. □

We remark that a Petri net is unbounded if and only if there is a firing sequence σ, a marking μ, and a positive loop ϑ such that $\mu_0 \xrightarrow{\sigma} \mu$, and ϑ is enabled at μ (see [10]). Lemma 4.2 states that the same principle holds for single places of a BPP-PN. In general, however, this is not true. We further note that (in contrast to, e.g., persistent Petri nets, see [12]) this concept doesn't hold for sets of places of BPP-PNs, i.e., a set $S \subseteq P$ of places of a BPP-PN is not necessarily simultaneously ∞-unbounded if it is simultaneously unbounded. An example is given in Figure 3. We can use the characterization provided by Lemma 4.2 to give efficient algorithms for certain boundedness problems.

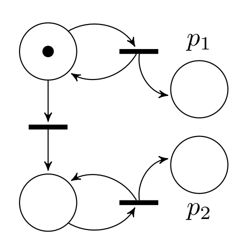

Figure 3: $\{p_1, p_2\}$ is simultaneously unbounded but not simultaneously-∞-unbounded.

Theorem 4.3. *Given a BPP-PN $\mathcal{P} = (P, T, F, \mu_0)$ and sets $S_1, \ldots, S_k \subseteq P$ of places, we can determine in linear time which sets are*

(a) *(∞-)unbounded in $\mathcal{P}$.*

(b) *placewise (∞-)unbounded in $\mathcal{P}$.*

Proof. We give a linear time algorithm that works in six steps.

1. Using Tarjan's modified depth-first search [19], we find the strongly connected components of $\mathcal{P}$.
2. By investigating all SCCs, we determine all SCCs that contain a marked place. We call these components C_1-components.
3. Using a DFS on the shrunken graph we determine all SCCs that contain a place and are reachable by a C_1-component. We call these components C_2-components.
4. We determine all SCCs C that are either C_2-components and contain a transition t satisfying $\sum_{p \in t^\bullet \cap C} F(t,p) \geq 2$, or that are reachable by a single edge emanating from a transition that is contained in a C_2-component. We call these components C_3-components.
5. Then we use a DFS in the shrunken graph to determine all SCCs that are reachable from a C_3-component. These SCCs are called C_4-components. By Lemma 4.2, all places contained in C_4-components are (∞-)unbounded.
6. Now we simply check for (a) if each S_i, $i \in [k]$, contains a place p that is in some C_4-component. For (b) we check for each S_i, $i \in [k]$, if each place of S_i is in some C_4-component.

Notice that each of these steps can be performed in linear time. □

Definition 4.4 (Boundedness problem for BPP-PNs)**.** Given a BPP-PN $\mathcal{P}$, are all places of $\mathcal{P}$ bounded?

Corollary 4.5. *The boundedness problem for BPP-PNs is decidable in linear time.*

Proof. Apply Theorem 4.3 to the set of all places. □

We remark that in [11] was shown that boundedness of Basic Parallel Processes can be decided in polynomial time.

Interestingly, a slight relaxation of BPP-PNs leads to a class of nets for which the boundedness problem is coNP-hard.

Theorem 4.6. *Let multiplicity generalized BPP-PNs be the Petri net class consisting of all Petri nets $\mathcal{P} = (P, T, F, \mu_0)$ which satisfy $|^\bullet t| = 1$ for all $t \in T$. The boundedness problem for multiplicity generalized BPP-PNs is coNP-hard.*

Proof. We reduce `3-SAT` in logspace to the unboundedness problem which is the complement of the boundedness problem. The reduction is illustrated in Figure 4. A similar reduction was used by Esparza [3] to show the NP-hardness of the reachability problem for BPP-PNs. Let $C_1 \wedge C_2 \wedge \ldots \wedge C_\ell$ be a formula in 3-CNF with k variables $x_1, \ldots, x_k$ and ℓ clauses $C_1, \ldots, C_\ell$.

We create a BPP-PN $\mathcal{P}$ as follows. Each variable is represented by a place containing one token, and each clause C_j is represented by a place c_j containing 3 tokens. For each variable x_i, there are two transitions x_i and $\overline{x_i}$ representing the truth assignment of x_i where x_i ($\overline{x_i}$, resp.) puts a token to place c_j if the literal x_i ($\overline{x_i}$, resp.) is contained in C_j.

Then, there is a counting place p which counts how many clauses are satisfied. If clause C_j is satisfied by an assignment, c_j contains at least 4 and at most 6 tokens which allows us to transfer exactly one token from c_j to p. If C_j is unsatisfied, c_j contains 3 tokens, and we cannot transfer any token to p.

Therefore, $\mathcal{P}$ is unbounded if and only if p is unbounded if and only if all ℓ clauses can be satisfied. This shows the NP-hardness of the unboundedness problem and therefore the coNP-hardness of the boundedness problem. □

In order to decide the next variation of the boundedness problem in linear time, we need some observations about traps in BPP-PNs.

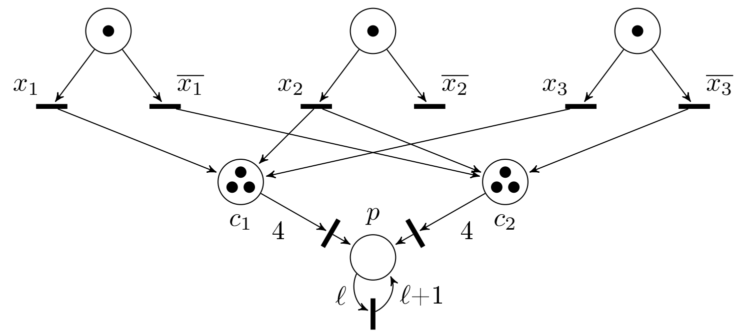

Figure 4: The formula $C_1 \wedge C_2 = (x_1 \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee x_2 \vee \overline{x_3})$ can be satisfied if and only if this BPP-PN is unbounded.

Lemma 4.7. *Let $\mathcal{P} = (P, T, F, \mu_0)$ be a BPP-PN and $R \subseteq P$ be a set of places. The trap $Q \subseteq R$ of maximum cardinality w.r.t. R can be determined in linear time.*

Proof. Apply the following procedure. Let $Q' := R$. While there is a transition $t \in T$ such that ${}^\bullet t \in Q'$ and $t^\bullet \cap Q' = \emptyset$, remove ${}^\bullet t$ from Q'. Let Q denote the resulting set. Q must be a trap since otherwise the procedure wouldn't have stopped. Furthermore, Q is maximal w.r.t. inclusion since the procedure can't remove a place from a maximal trap. There is exactly one maximal trap w.r.t. inclusion which therefore is a maximum trap.

We can implement the procedure in linear time as follows. We use two arrays A and N, and a list L, as well as a collection Q. The collection Q is initialized with the set R. The array A has length $|T|$ and $A[i]$ is initialized with $|t_i{}^\bullet \cap R|$. The array N has length $|P|$ and $N[i]$ is initialized with an empty list if $p_i \notin R$, and otherwise with a list of all transitions t_j such that $t_j \in {}^\bullet p_i$. The list L is initialized with all transitions t_i having ${}^\bullet t_i \in Q$ and $A[i] = 0$. It's not hard to see that these data structures can be initialized in linear time.

Now, as long as L is not empty, we do the following. First, we pop some transition t_i from the list, and let $p_j = {}^\bullet t_i$. Then, if $p_j \in Q$, we remove p_j from Q, and, for each t_k contained in the list stored at $N[j]$, we decrease $A[k]$ by 1, and add t_k to L if $A[k] = 0$ after the decreasing step.

When L is empty, $Q \subseteq R$ is the trap of maximum cardinality w.r.t. R. The running time of this procedure is linear.

□

Lemma 4.8. *Let $\mathcal{P} = (P, T, F, \mu_0)$ be a BPP-PN, and $R \subseteq P$ be a subset of places such that no set $Q \subseteq R$ is a trap. Then, there is a firing sequence σ leading to a marking where no place of R is marked such that $\Delta[\sigma](p) \geq 0$ for all $p \notin R$.*

Proof. By definition, if a set $Q \subseteq P$ is not a trap, then there is a transition t with ${}^\bullet t \in Q$ and $t^\bullet \cap Q = \emptyset$. Define the transitions $t_1, \ldots, t_{|R|}$ and the sets $R_0, \ldots, R_{|R|+1}$ recursively as follows. We start with $R_1 = R$. Given R_i for $i \in [|R|]$, then t_i is a transition with ${}^\bullet t_i \in R_i$ and $t_i{}^\bullet \cap R_i = \emptyset$, and $R_{i+1} = R_i - {}^\bullet t_i$. In other words, $R_{|R|} \subsetneqq \ldots \subsetneqq R_1$, and we can successively empty $R_{|R|}, \ldots, R_1$ by firing the transitions $t_{|R|}, \ldots, t_1$ each an appropriate number of times. Since these transitions don't remove tokens from places outside of R, the displacement of the firing sequence at these places is nonnegative. □

Lemma 4.9. *Let $\mathcal{P} = (P, T, F, \mu_0)$ be a BPP-PN, and $Q \subseteq R \subseteq P$ be the trap of maximum cardinality w.r.t. R. Then there is a firing sequence σ leading to μ with $\mu(p) = 0$ for all $p \in R$ if and only if all places of Q are unmarked.*

Proof. "$\Rightarrow$": If Q is marked, then R will always be marked, regardless of the transitions fired.

"$\Leftarrow$": Notice that $R' := R \setminus Q$ doesn't contain a trap by the maximality of Q. Consider the BPP-PN $\mathcal{P}'$ which emerges from $\mathcal{P}$ by removing Q and all transitions incident to Q. R' also doesn't contain a trap w.r.t. $\mathcal{P}'$. By Lemma 4.8, R' can be emptied in $\mathcal{P}'$. Therefore, R can be emptied in $\mathcal{P}$. $\square$

These observations enable us to prove the following theorems.

Theorem 4.10. *Given a BPP-PN $\mathcal{P} = (P, T, F, \mu_0)$ and a place $p \in P$, we can decide in linear time if p is persistently unbounded.*

Proof. We use the terminology of Lemma 4.2. Let C_4 be the SCC containing p. For the marking μ_0' having exactly one token at each place, we determine the set $R \subseteq P$ of all places contained in SCCs C_1 for which SCCs C_2 and C_3 exist such that C_1, C_2, C_3, and C_4 satisfy the properties mentioned in Lemma 4.2. By this lemma, p is unbounded at each marking μ such that there is a place $r \in R$ with $\mu(r) > 0$.

Therefore, p is not persistently unbounded if and only if there is a marking reachable from μ_0 where no place of R is marked. By Lemma 4.9, we only have to determine if the maximum trap $Q \subseteq R$ w.r.t. R is marked. By Lemma 4.7, this can be done in linear time. $\square$

Theorem 4.11. *Given a BPP-PN $\mathcal{P} = (P, T, F, \mu_0)$ and two disjoint sets $S, R \subseteq P$ of places, we can decide in linear time if S is (placewise) empty-R-unbounded.*

Proof. Let $Q \subseteq R$ denote the maximum trap w.r.t. R. Consider the BPP-PN $\mathcal{P}'$ which emerges from $\mathcal{P}$ by removing Q and all transitions incident to Q. S is (placewise) empty-R-unbounded in $\mathcal{P}$ if and only if S is (placewise, respectively) unbounded in $\mathcal{P}'$ and Q is unmarked in $\mathcal{P}$ since, by Lemma 4.8, we can empty all places of $R \setminus Q$ without decreasing the number of tokens at other places. By Theorem 4.3, these conditions can be checked in linear time. $\square$

Another useful Lemma can be proven in a similar way as Theorem 4.11.

Lemma 4.12. *Let $\mathcal{P} = (P, T, F, \mu_0)$ be a BPP-PN, and $Q \subseteq R \subseteq P$ be the maximum trap w.r.t. R. Then, there is a firing sequence leading to a marking where no place of R is marked if and only if Q is not marked. Furthermore, this can be decided in linear time.*

Esparza et al. [4] provided a generic algorithm deciding in quadratic time if the language of a given context-free grammar is finite. In the same paper, they mentioned that a careful implementation of the algorithm in [6] could possibly achieve linear time. Using our results, we can decide a generalization of the finiteness problem of commutative and non-commutative context-free grammars in linear time.

Theorem 4.13. *Given a (commutative) context-free grammar $G = (V, T, P, S)$ and a set $U \subseteq T$ with variables V, terminal symbols T, productions P (free commutative monoids in the case of commutative grammars), and start variable $S \in V$, we can decide in linear time if $L(G)[U]$ is finite. $L(G)[U]$ denotes the set of all words $x \in U^*$ for which a word y of the language $L(G)$ of G exists such that x is obtained by deleting all symbols from y which are not in U.*

Proof. We define a BPP-PN as follows. Variables and terminal symbols are represented by places, where $\tilde{X}$ denotes the place corresponding to $X \in V \cup T$. A production $X \to Y_1 Y_2 \cdots Y_k$, where $X \in V$ and $Y_i \in V \cup T$ is represented by a transition t such that $\tilde{X} = {}^\bullet t$, and $F(t, \tilde{Y}_i)$ equals the number of occurrences of Y_i in $Y_1 \cdots Y_k$. The initial marking contains a token at place $\tilde{S}$. Let $\tilde{U}$ denote the set of places that correspond to the terminal symbols of U. Further, let $\tilde{V}$ be the set of places corresponding to variables. $L(G)[U]$ is infinite if and only if the set $\tilde{U}$ is empty-$\tilde{V}$-unbounded. By Theorem 4.11, this can be decided in linear time. Note that for non-commutative c.f. grammers, this reduction does not take the order of symbols on the right-hand side of productions into account which is irrelevant for this problem. $\square$

An advantage of our algorithm compared to the one in [6] is that it does not require the grammar being in Chomsky normal form. In [4], the authors also provided linear time algorithms for the emptiness problem and the problem of finding nullable variables of context-free grammars. Our results provide alternative linear time algorithms for these problems as well as for corresponding problems of *commutative* context-free grammars.

Theorem 4.14. *Given a (commutative) context-free grammar $G = (V, T, P, S)$, we can decide in linear time if $L(G)$ is empty.*

Proof. Consider the same BPP-PN as in Theorem 4.13. Then $L(G) = \emptyset$ if and only if $\tilde{V}$ cannot be emptied. By Lemma 4.12, this can be decided in linear time. □

Theorem 4.15. *Given a (commutative) context-free grammar $G = (V, T, P, S)$, we can find in linear time all variables $X \in V$ for which the empty word ε is in $L(G_X)$ where G_X is the (commutative) context-free grammar (V, T, P, X).*

Proof. Consider the same BPP-PN as in Theorem 4.13. Then, by Lemma 4.12, $\varepsilon \in L(G_X)$ if and only if $\tilde{V} \cup \tilde{T}$ can be emptied in the net where the only marked place is $\tilde{X}$. By the same lemma, this applies to exactly those places $\tilde{X} \in \tilde{V}$ which are not part of the maximum trap w.r.t. $\tilde{V} \cup \tilde{T}$. By Lemma 4.7, all such places and therefore the set of all nullable variables can be found in linear time. □

In the same way, the zero reachability problem which, in general, is as hard as the reachability problem (see [5]), is decidable in linear time for BPP-PNs.

Theorem 4.16. *Given a BPP-PN $\mathcal{P} = (P, T, F, \mu_0)$, we can decide in linear time if the zero marking $(0, \ldots, 0)$ is reachable.*

Proof. The zero marking is reachable if and only if P can be emptied. By Lemma 4.12, this can be decided in linear time. □

4.2 Simultaneously Unboundedness and the Covering Problem

In this subsection, we consider the covering problem as well as boundedness problems where we ask if many places are simultaneously (∞-)unbounded.

Definition 4.17 (`SU`)**.** Given a BPP-PN $\mathcal{P} = (P, T, F, \mu_0)$ and a subset $S \subseteq P$ of places, is S simultaneously unbounded?

Definition 4.18 (`S`-∞-`U`)**.** Given a BPP-PN $\mathcal{P} = (P, T, F, \mu_0)$ and a subset $S \subseteq P$ of places, is S simultaneously ∞-unbounded?

Definition 4.19 (Covering problem for BPP-PNs (`covering`))**.** Given a BPP-PN $\mathcal{P}$, and a marking μ of $\mathcal{P}$, is there a reachable marking $\mu' \geq \mu$?

In order to prove the next theorems, we need the following corollary which is an immediate consequence of Lemma 2 of [20].

Corollary 4.20. *Let $\mathcal{P} = (P, T, F, \mu_0)$ be a BPP-PN with $m = |T|$ and largest edge multiplicity W, and let μ be a reachable marking. Then there is a firing sequence $\sigma = \pi_1\alpha_1\pi_2\alpha_2 \cdots \pi_m\alpha_m$ leading from μ_0 to μ such that, for all $i \in [m]$, π_i is a nonnegative loop, and α_i satisfies $-mW \leq \Delta[\alpha_i](p) \leq mW$ for all $p \in P$.*

Theorem 4.21. *`SU` and `S`-∞-`U` are NP-complete even if we restrict the input to BPP-PNs $\mathcal{P} = (P, T, F, \mu_0)$ with $|t^\bullet| = 1$ and $F(t, t^\bullet) \leq 2$ for all $t \in T$.*

Note that a further restriction to $F(t, t^\bullet) = 1$ leads to S-Systems, a subclass of BPP-PNs, which are always bounded.

Theorem 4.22. *The covering problem for BPP-PNs is NP-complete.*

Proof. For two problems A and B let $A \prec_{\log} B$ denote the existence of a logspace many-one reduction from A to B.

We first show the NP-hardness of `SU` and `S`-∞-`U` by showing `3-SAT` $\prec_{\log}$ `SU` and `3-SAT` $\prec_{\log}$ `S`-∞-`U`.

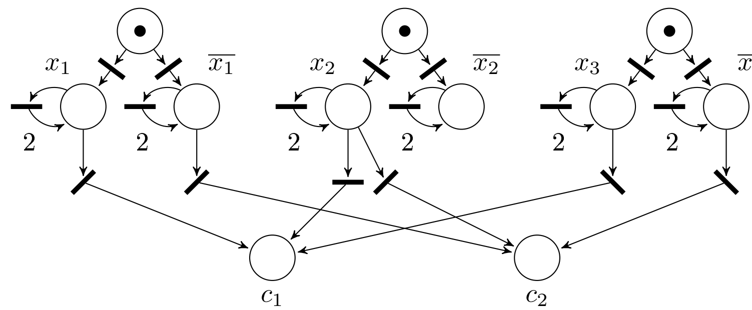

Figure 5: The formula $C_1 \wedge C_2 = (x_1 \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee x_2 \vee \overline{x_3})$ can be satisfied if and only if $\{c_1, c_2\}$ is simultaneously (∞-)unbounded.

Given a formula F in 3-CNF over the variables $x_1, \ldots, x_k$ and clauses $C_1, \ldots, C_\ell$, we construct a BPP-PN such that a certain subset $S = \{c_i \mid i \in [\ell]\}$ of places is simultaneously (∞)-unbounded if and only if F can be satisfied. An example is illustrated in Figure 5 (cf. the reduction in the proof of Theorem 4.6).

Next, we show `covering` $\in$ NP by reducing `covering` in logspace to the reachability problem. We modify $\mathcal{P}$ by adding, for each $p \in P$, a transition t_p having $F(p, t_p) = 1$. Call the resulting BPP-PN $\mathcal{P}'$. Notice that a marking μ can be covered in $\mathcal{P}$ if and only if μ can be covered in $\mathcal{P}'$ if and only if μ is reachable in $\mathcal{P}'$.

Now, we show `SU` $\in$ NP and that `covering` is NP-hard by showing `SU` $\prec_{\log}$ `covering`. Again consider the net $\mathcal{P}'$. Let W be the largest edge multiplicity of $\mathcal{P}'$. Define the marking μ by $\mu(p) = \mu_0(p) + (n+m)^2 W + 1$ if $p \in S$, and $\mu(p) = 0$ otherwise. Notice that μ has polynomial encoding size.

Assume that S is simultaneously unbounded in $\mathcal{P}$. Then μ is coverable in $\mathcal{P}'$. Now, assume that μ is coverable in $\mathcal{P}'$, i.e., μ is reachable in $\mathcal{P}'$. In accordance with Corollary 4.20, let $\pi_1 \alpha_1 \cdots \pi_{n+m} \alpha_{n+m}$ be a firing sequence of $\mathcal{P}'$ leading from μ_0 to μ. Then $-(n+m)^2 W \leq \sum_{i \in [n+m]} \Delta[\alpha_i](p) \leq (n+m)^2 W$ for all $p \in P$. Hence, each place $p \in S$ has some i such that $\Delta[\pi_i](p) > 0$. Therefore, for each $k \in \mathbb{N}$, the marking μ'_k reached in $\mathcal{P}'$ by the firing sequence $\pi_1^{(n+m)^2 W + k} \cdot \alpha_1 \cdots \pi_{n+m}^{(n+m)^2 W + k} \cdot \alpha_{n+m}$ satisfies $\mu'_k(p) \geq k$ for all $p \in S$. By removing all transitions from this firing sequence that are not part of $\mathcal{P}$, we obtain a firing sequence of $\mathcal{P}$ leading to a marking μ_k of $\mathcal{P}$ with $\mu_k(p) \geq k$ for all $p \in S$. Therefore, S is simultaneously unbounded in $\mathcal{P}$.

It remains to be shown that `S-`∞`-U` $\in$ NP. To this end, we will describe a nondeterministic procedure accepting if and only if the given set $S \subseteq P$ is simultaneously ∞-unbounded. Suppose, the latter is the case. Then there is an ∞-firing sequence on which S is simultaneously unbounded. A similar argument as in the proof of Lemma 4.2 shows that there are transitions sequences σ, τ such that $\sigma\tau^\infty$ is an ∞-firing sequence and τ is a positive loop having $\Delta[\tau](p) \geq 1$ for all $p \in S$.

By Lemma 3.3, τ is enabled at exactly those markings μ where all top components of $\mathcal{P}_\tau$ are marked. Therefore, there is a marking μ^* such that each place has either zero or one tokens and such that τ is enabled at μ^*. We will use the existence of μ^* later.

Let $D \in \mathbb{Z}^{n \times m}$ be the displacement matrix of $\mathcal{P}$, i.e., the i-th column of D equals $\Delta[t_i]$. Consider the system $D\Phi \geq \mathbf{0}$ of linear diophantine inequalities. Obviously, the set L of nontrivial nonnegative integral solutions of this system equals the set of nonnegative loops having at least one transition. Now, consider the system $(D, -I_n)y = 0$ having the set L' of nontrivial solutions where I_n is the $n \times n$-identity matrix. The set of projections of the elements of L' onto the first m components equals L. By Theorem 1 of [18], this system has a set $\mathcal{H}(D, -I_n)$ of minimal solutions (called the Hilbert basis) having the following properties:

(i) Each nontrivial solution can be expressed as a linear combination of the elements of $\mathcal{H}(D, -I_n)$ with nonnegative integral coefficients.

(ii) Each element of $\mathcal{H}(D, -I_n)$ has a component sum of at most $(mW+2)^n$.

W.l.o.g., we assume $n, m, W \geq 1$. Let $L_{\min} = \{\Phi_1, \ldots, \Phi_r\} \subseteq L$ denote the projection of $\mathcal{H}(D, -I_n)$ onto the first m components. From (ii) we immediately obtain $r \leq ((mW+2)^n + 1)^n \leq (2mW)^{cn^2}$ for some constant $c > 0$.

Since τ is a nonnegative loop, we can write $\Psi[\tau] = \sum_{i \in [r]} a_i \Phi_i$ for suitable $a_i \in \mathbb{N}_0$. Now, define $a'_i := \min\{a_i, 1\}$ and $\Phi := \sum_{i \in [r]} a'_i \Phi_i$. For each $p \in S$, we have $\Delta[\tau](p) > 0$, implying the existence of an i with $a_i > 0$ and $\Delta[\Phi_i](p) > 0$. Therefore, Φ is a nonnegative loop with $\Delta[\Phi](p) > 0$ for all $p \in S$. Furthermore, by Lemma 3.3, Φ is enabled at μ since $\mathcal{P}_\Phi = \mathcal{P}_{\Psi[\tau]}$. (ii) and $r \leq (2mW)^{cn^2}$ imply that the largest component of Φ is at most $(2mW)^{dn^2}$ for some constant $d > 0$. Therefore, the encoding size of Φ is polynomial.

Now, we can describe the nondeterministic procedure which accepts if and only if S is simultaneously unbounded on some ∞-firing sequence: We guess μ^* and Φ in polynomial time and check nondeterministically and in polynomial time if μ^* can be covered and if Φ is enabled at μ^*.

This completes the proof. □

We note that it can be decided in linear time if the set P of all places is simultaneously (∞-)unbounded. This is the case if and only if all top components C contain a marked place and a transition t with $\sum_{p \in t^\bullet \cap C} F(t,p) \geq 2$. Hence, the problems `SU` and `S-`∞`-U` are hard only if the input set S satisfies $1 < |S| < |P|$.

5 Liveness Problems for BPP-PNs

Many different notions of liveness can be found in literature. We are mainly interested in the following.

Definition 5.1. Let $\mathcal{P} = (P, T, F, \mu_0)$ be a Petri net. A transition t is

- L_0-*live* or *dead* if there is no firing sequence containing t.
- L_1-*live* or *potentially fireable* if it isn't dead.
- L_2-*live* or *arbitrarily often fireable* if for each $k \in \mathbb{N}$ there is a firing sequence containing t at least k times.
- L_3-*live* or *infinitely often fireable* if there is an ∞-firing sequence containing t infinitely often.
- L_4-*live* or *live* if t is potentially fireable at each reachable marking.

A subset $S \subseteq T$ of transitions is called L_x-live, $x \in [0, 4]$, if all transitions of S are L_x-live.

The concepts of $L_0, \ldots, L_4$-liveness are referred to in [17]. Notice, that L_i-liveness implies L_j-liveness, where $4 \geq i \geq j \geq 1$. Using the results of Section 4, we can efficiently solve many decision problems involving these notions of liveness.

Theorem 5.2. *Given a BPP-PN $\mathcal{P} = (P, T, F, \mu_0)$ and sets $S_1, \ldots, S_k \subseteq T$ of transitions, we can determine in linear time which sets are*

(a) L_0-live. (b) L_1-live. (c) L_2-live. (d) L_3-live.

Proof. Consider the Parikh extension $\mathcal{P}^e = (P^e, T, F^e, \mu_0^e)$ of $\mathcal{P}$ (see Definition 3.8). A transition t_i is not L_0-live iff t_i is L_1-live iff for the SCC C_i containing p_i^* there is a marked SCC C such that $C_i \leq C$ (see Lemma 3.3). Hence, we can answer (a) and (b) in linear time by computing $\mathcal{P}^e$, collecting the SCCs of $\mathcal{P}^e$ and investigating the found SCCs in a similar fashion as in Theorem 4.3.

For (c) and (d) notice that t_i is L_2-live iff p_i^* is unbounded iff p_i^* is unbounded on some ∞-firing sequence (see Lemma 4.2) iff t_i is L_3-live. Hence, we simply apply the algorithm of Theorem 4.3 to $\mathcal{P}^e$ and the sets $S_1^*, \ldots, S_k^*$, where $S_i^* = \{p_j^* \mid t_j \in S_j\}$. □

Corollary 5.3. *Given a BPP-PN $\mathcal{P} = (P, T, F, \mu_0)$, we can decide in linear time, if (a transition t of) $\mathcal{P}$ is*

(a) L_0-live. (b) L_1-live. (c) L_2-live. (d) L_3-live.

Theorem 5.4. *Given a BPP-PN $\mathcal{P} = (P, T, F, \mu_0)$ and a transition $t \in T$, we can decide in linear time if t is L_4-live.*

Proof. As before, consider the Parikh extension $\mathcal{P}^e = (P^e, T, F^e, \mu_0^e)$ of $\mathcal{P}$. It is easy to see that a transition t_i is L_4-live iff p_i^* is persistently unbounded. □

In [14], Mayr showed that L_4-liveness is decidable in polynomial time for Basic Parallel Processes. Our results imply a quadratic time algorithm.

Corollary 5.5. *Given a BPP-PN $\mathcal{P} = (P, T, F, \mu_0)$, we can decide in quadratic time, if $\mathcal{P}$ is L_4-live.*

In the same paper, other interesting notions of liveness were investigated, namely the partial deadlock reachability problem and the partial livelock reachability problem. For both problems polynomial time algorithms were proposed for PA-processes in general. Using our results, linear time algorithms can be given for BPPs/BPP-PNs.

Theorem 5.6 (deadlock)**.** *Given a BPP-PN $\mathcal{P} = (P, T, F, \mu_0)$ and a set S of transitions, we can decide in linear time if there is a reachable marking μ such that $\mu(^\bullet t) = 0$ for all $t \in S$.*

Proof. Let $R = \bigcup_{t \in S} {}^\bullet t$. By Lemma 4.7, we determine in linear time the maximum trap $Q \subseteq R$ w.r.t. R. By Lemma 4.9, R can be emptied if and only if Q is unmarked which can be checked in linear time. □

Theorem 5.7 (livelock)**.** *Given a BPP-PN $\mathcal{P} = (P, T, F, \mu_0)$ and a set S of transitions, we can decide in linear time if there is a reachable marking μ such that for all markings μ' reachable from μ, we have $\mu'(^\bullet t) = 0$ for all $t \in S$.*

Proof. We introduce a counting place p and an edge from each transition $t \in S$ to p. A marking μ as defined in the lemma exists if and only if p is not persistently unbounded. By Theorem 4.10, this can be decided in linear time. □

6 Conclusion

We showed in conjunction with [20] that the equivalence problem is decidable in doubly exponential time. Furthermore, we investigated several boundedness and liveness problems for BPP-PNs. For some of them, as well as for the covering problem, NP-completeness was shown. For most of the other problems, linear time could be achieved implying linear time algorithms for many problems in related areas. Open problems include:

- Is the equivalence problem complete for some known complexity class?
- Are the problems `SU` and `S-`∞`-U` decidable in polynomial time for sets S of constant size?
- How do the complexities of the reachability and the covering problem, and of variations of the boundedness and liveness problem behave if we consider different generalizations of BPP-PNs?

Acknowledgements. We thank Javier Esparza for many helpful discussions.

References

[1] S. Christensen. Distributed bisimularity is decidable for a class of infinite state-space systems. In W. Cleaveland, editor, *CONCUR '92*, volume 630 of *Lecture Notes in Computer Science*, pages 148–161. Springer Berlin / Heidelberg, 1992.

[2] S. Christensen, Y. Hirshfeld, and F. Moller. Bisimulation equivalence is decidable for basic parallel processes. In E. Best, editor, *CONCUR'93*, volume 715 of *Lecture Notes in Computer Science*, pages 143–157. Springer Berlin / Heidelberg, 1993.

[3] J. Esparza. Petri nets, commutative context-free grammars, and basic parallel processes. *Fundamenta Informaticae*, 31(1):13–25, 1997.

[4] J. Esparza, P. Rossmanith, and S. Schwoon. A uniform framework for problems on context-free grammars. *EATCS Bulletin*, 72:169–177, October 2000.

[5] M. Hack. The recursive equivalence of the reachability problem and the liveness problem for Petri nets and vector addition systems. In *Switching and Automata Theory, 1974., IEEE Conference Record of 15th Annual Symposium on*, pages 156 –164, oct. 1974.

[6] J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 1979.

[7] D. T. Huynh. The complexity of semilinear sets. In J. de Bakker and J. van Leeuwen, editors, *Automata, Languages and Programming*, volume 85 of *Lecture Notes in Computer Science*, pages 324–337. Springer Berlin / Heidelberg, 1980.

[8] D. T. Huynh. Commutative grammars: The complexity of uniform word problems. *Information and Control*, 57(1):21 – 39, 1983.

[9] D. T. Huynh. A simple proof for the sum upper bound of the inequivalence problem for semilinear sets. *Elektronische Informationsverarbeitung und Kybernetik*, pages 147–156, 1986.

[10] R. M. Karp and R. E. Miller. Parallel program schemata. *Journal of Computer and System Sciences*, 3(2):147 – 195, 1969.

[11] A. Kučera. Regularity is decidable for normed PA processes in polynomial time. In V. Chandru and V. Vinay, editors, *Foundations of Software Technology and Theoretical Computer Science*, volume 1180 of *Lecture Notes in Computer Science*, pages 111–122. Springer Berlin / Heidelberg, 1996.

[12] L. H. Landweber and E. L. Robertson. Properties of conflict-free and persistent Petri nets. *J. ACM*, 25:352–364, July 1978.

[13] E. W. Mayr. An algorithm for the general Petri net reachability problem. In *Proceedings of the thirteenth annual ACM symposium on Theory of computing*, STOC '81, pages 238–246, New York, NY, USA, 1981. ACM.

[14] R. Mayr. Tableau methods for PA-processes. In D. Galmiche, editor, *Automated Reasoning with Analytic Tableaux and Related Methods*, volume 1227 of *Lecture Notes in Computer Science*, pages 276–290. Springer Berlin / Heidelberg, 1997.

[15] R. Mayr. *Decidability and Complexity of Model Checking Problems for Infinite-State Systems*. PhD thesis, Technische Universität München, 1998.

[16] R. Mayr. On the complexity of bisimulation problems for basic parallel processes. In U. Montanari, J. Rolim, and E. Welzl, editors, *Automata, Languages and Programming*, volume 1853 of *Lecture Notes in Computer Science*, pages 329–341. Springer Berlin / Heidelberg, 2000.

[17] T. Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541 –580, apr 1989.

[18] L. Pottier. Minimal solutions of linear diophantine systems : bounds and algorithms. In R. Book, editor, *Rewriting Techniques and Applications*, volume 488 of *Lecture Notes in Computer Science*, pages 162–173. Springer Berlin / Heidelberg, 1991.

[19] R. Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160, 1972.

[20] H.-C. Yen. On reachability equivalence for BPP-nets. *Theoretical Computer Science*, 179(1-2):301 – 317, 1997.