Decision Problems for Petri Nets with Names

Fernando Rosa-Velardo, David de Frutos-Escrig

Dpto. de Sistemas Informáticos y Computación
Universidad Complutense de Madrid

Abstract

We prove several decidability and undecidability results for ν-PN, an extension of P/T nets with pure name creation and name management. We give a simple proof of undecidability of reachability, by reducing reachability in nets with inhibitor arcs to it. Thus, the expressive power of ν-PN strictly surpasses that of P/T nets. We prove that ν-PN are Well Structured Transition Systems. In particular, we obtain decidability of coverability and termination, so that the expressive power of Turing machines is not reached. Moreover, they are strictly Well Structured, so that the boundedness problem is also decidable. We consider two properties, width-boundedness and depth-boundedness, that factorize boundedness. Width-boundedness has already been proved to be decidable. We prove here undecidability of depth-boundedness. Finally, we obtain Ackermann-hardness results for all our decidable decision problems.

Keywords: Petri nets, pure names, Well Structured Transition Systems, decidability

1. Introduction

Pure names are identifiers with no relation between them other than equality [14]. Dynamic name generation has been thoroughly studied, mainly in the field of security and mobility [14] because they can be used to represent channels, as in π-calculus [24], ciphering keys, as in spi-Calculus [2] or computing boundaries, as in the Ambient Calculus [4].

In previous works we have studied a very simple extension of P/T nets [6], that we called ν-PN [25, 28], for name creation and management.[2] Tokens in ν-PN are pure names, that can be created fresh, moved along the net and be used to restrict the firing of transitions with name matching. They essentially

---

Email addresses: `fernandorosa@sip.ucm.es` (Fernando Rosa-Velardo),
`defrutos@sip.ucm.es` (David de Frutos-Escrig)

[2]Actually, we used the term ν-APN, where the A stands for *Abstract*, though we prefer to use this simpler acronym.

correspond to the minimal OO-nets of [18], where names are used to identify objects.

In this paper we prove several (un)decidability and complexity results for some decision problems in ν-PN. In [18] the author proved that reachability is undecidable for minimal OO-nets, thus proving that the model surpasses the expressive power of P/T nets. The same result was obtained independently in [25] for ν-PN. Both undecidability proofs rely on a weak simulation of a Minsky machine that preserves reachability. We present here an alternative and simpler proof of the same result, based on a simulation of Petri nets with inhibitor nets (thus, with a much smaller representation gap) that reduces reachability in the latter (which is undecidable) to reachability in ν-PN.

In [25] we proved well structuredness [3, 10] of a class of nets we called MSPN. It is easy to see that ν-PN can easily encode MSPN. We present here full details of the proof of well structuredness for ν-PN instead of for MSPN, since the former is a much more cleaner formalism. This gives us decidability of coverability (which is an important property, since safety properties can be specified in terms of it) and termination [3, 10]. We also prove that the well structuredness of ν-PN is strict, so that boundedness (whether there are infinitely many reachable markings) is also decidable [10]. Moreover, we work with an extended version of ν-PN, in which we allow weights in arcs, simultaneous creation of several fresh names and checks for inequality.

ν-PN can represent infinite state systems that can grow in two orthogonal directions: On the one hand, markings may have an unbounded number of different names; On the other hand, each name may appear in markings an unbounded number of times. In the first case we will say the net is width-unbounded, and in the second we will say it is depth-bounded. In [27] we proved decidability of width-boundedness by performing a forward analysis that, though incomplete in general for the computation of the cover, can decide width-boundedness. In particular, we instantiated the general framework developed in [11, 12] for forward analyses of WSTS in the case of ν-PN.

Here we prove undecidability of depth-boundedness. Thus, though both boundedness concepts are closely related, they behave very differently. The proof reduces boundedness in reset nets, which is known to be undecidable [8], to depth-boundedness in ν-PN. This result can be rather surprising. Actually, the paper [7] erroneously establishes the decidability of depth-boundedness (called t-boundedness there).

Related work. Another model based on Petri nets that has names as tokens are *Data Nets*, which are also WSTS [20]. In Data Nets, tokens are not pure in general, but taken from a linearly-ordered infinite domain. Names can be created, but they can only be guaranteed to be fresh by explicitly using the order in the data domain, by taking a datum which is greater than any other that has been used. Thus, in an unordered version of Data Nets, names cannot be guaranteed to be fresh.

Other similar models include Object Nets [30, 31], that follow the so called nets-within-nets paradigm. In Object Nets, tokens can themselves be Petri nets that synchronize with the net in which it lies. This model is supported by

the RENEW tool [19], a tool for the edition and simulation of Object Petri Nets. Moreover, the RENEW tool can represent ν-PN and, therefore, be used to simulate them.

Several papers study the expressive power of Object Nets. The paper [16] considers a two level restriction of Object Nets, called Elementary Object Nets (EON), and proves undecidability of reachability for them. This result extends those in [15]. Moreover, some subclasses are proved to have decidable reachability. In [17] it is shown that, when the synchronization mechanism is extended so that object tokens can be communicated, then Turing completeness is obtained. However, in all these models processes (object nets) do not have identities.

Nested Petri Nets [21] also have nets as tokens, that can evolve autonomously, move along the system net, synchronize with each other or synchronize with the system net (vertical synchronization steps). Nested nets are more expressive than ν-PN. Indeed, it is possible to simulate every ν-PN by means of a Nested Petri Net which uses only object-autonomous and horizontal synchronization steps. In Nested Petri Nets, reachability and boundedness are undecidable, although other problems, like termination, remain decidable [22]. Thus, decidability of termination can also be obtained as a consequence of [22]. Here we obtain decidability of termination on the way of the proof of decidability for boundedness and coverability.

Outline. The rest of the paper is structured as follows. Section 2 presents some basic results and notations we will use throughout the paper. Section 3 defines ν-PN. Section 4 proves undecidability of reachability. In Sect. 5 we prove decidability of coverability, termination and boundedness, and we give non-primitive recursive lower bounds for their decision procedures. Section 6 presents further results about boundedness and in Section 7 we present our conclusions.

2. Preliminaries

Multisets. Given an arbitrary set A, we will denote by $A^{\oplus}$ the set of finite multisets of A, that is, the set of mappings $m : A \to \mathbb{N}$. We will identify each set with the multiset given by its characteristic function, and use set notation for multisets when convenient. We denote by $supp(m)$ the support of m, that is, the set $\{a \in A \mid m(a) > 0\}$ and by $|m| = \sum_{a \in supp(m)} m(a)$ the cardinality of m. Given two multisets $m_1, m_2 \in A^{\oplus}$ we denote by $m_1 + m_2$ and $m_1 \sqcup m_2$ the multisets defined by $(m_1 + m_2)(a) = m_1(a) + m_2(a)$ and $(m_1 \sqcup m_2)(a) = max\{m_1(a), m_2(a)\}$, respectively. We will write $m_1 \subseteq m_2$ if $m_1(a) \leq m_2(a)$ for every $a \in A$. In this case, we can define $m_2 - m_1$, given by $(m_2 - m_1)(a) = m_2(a) - m_1(a)$. We will denote by $\sum$ the extended multiset sum operator and by $\emptyset \in A^{\oplus}$ the multiset $\emptyset(a) = 0$, for every $a \in A$. If $f : A \to B$ and $m \in A^{\oplus}$, then we define $f(m) \in B^{\oplus}$ by $f(m)(b) = \sum_{f(a)=b} m(a)$. Every partial order $\leq$ defined over A induces a partial order $\sqsubseteq$ in the set $A^{\oplus}$, given by $\{a_1, \ldots, a_n\} \sqsubseteq \{b_1, \ldots, b_m\}$ if there is $\iota : \{1, \ldots, n\} \to \{1, \ldots, m\}$ injective such that $a_i \leq b_{\iota(i)}$ for all i. We

will write $\sqsubseteq_\iota$ to stress out the use of the mapping ι.

wqo. A quasi order is a reflexive and transitive binary relation on a set A. A partial order is an antisymmetric quasi order. A quasi order $\leq$ is decidable if for every $a, b \in A$ we can effectively decide if $a \leq b$. All the quasi orders in this paper are trivially decidable. For a quasi order $\leq$ we write $a < b$ if $a \leq b$ and $b \not\leq a$. A set $B \subseteq A$ is said to be a minor set of A if it does not contain comparable elements and for all $a \in A$ there is $b \in B$ such that $b \leq a$. We will write $min(A)$ to denote a minor set of A. The upward closure of a subset B is $\uparrow B = \{a \in A \mid \exists b \in B \text{ st } a \leq b\}$. A subset B is upward closed iff $B = \uparrow B$. A quasi order is well (wqo) [23] if for every infinite sequence $a_0, a_1, \ldots$ there are i and j with $i < j$ such that $a_i \leq a_j$. In a wqo $min(B)$ is always finite.

Transition systems. A transition system is a tuple $(S, \rightarrow, s_0)$, where S is a (possibly infinite) set of states, $s_0 \in S$ is the initial state and $\rightarrow \subseteq S \times S$. We denote by $\rightarrow^*$ the reflexive and transitive closure of $\rightarrow$. Given $S' \subseteq S$ we denote by $Pred(S')$ the set $\{s \in S \mid s \rightarrow s' \in S'\}$.

The reachability problem in a transition system consists in deciding for a given states s_f whether $s_0 \rightarrow^* s_f$. The termination problem consists in deciding whether there is an infinite sequence $s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow \cdots$. The boundedness problem consists in deciding whether the set of reachable states is finite. For any transition system $(S, \rightarrow, s_0)$ endowed with a quasi order $\leq$ we can define the coverability problem, that consists in deciding, given a state s_f, whether there is $s \in S$ reachable such that $s_f \leq s$.

WSTS. A Well Structured Transition System (WSTS) is a tuple $(S, \rightarrow, s_0, \leq)$, where $(S, \rightarrow, s_0)$ is a transition system, $\leq$ is a decidable wqo compatible with $\rightarrow$ (meaning that $s_1' \geq s_1 \rightarrow s_2$ implies that there is $s_2' \geq s_2$ with $s_1' \rightarrow s_2'$), and so that for all $s \in S$ we can compute $min(Pred(\uparrow s))$. We will refer to these properties as monotonicity of $\rightarrow$ with respect to $\leq$, and effective $Pred$-basis, respectively.[3] For WSTS, the coverability and the termination problems are decidable [3, 10]. A WSTS is said to be strict if it satisfies the following strict compatibility condition: $s_1' > s_1 \rightarrow s_2$ implies that there is $s_2' > s_2$ with $s_1' \rightarrow s_2'$. For strict WSTS, also the boundedness problem is decidable [10].

Petri Nets. Next we define P/T nets in order to set our notations. A P/T net is a tuple $N = (P, T, F)$ where P and T are disjoint finite sets of places and transitions, respectively, and $F : (P \times T) \cup (T \times P) \rightarrow \mathbb{N}$. A marking M of N is a finite multiset of places of N, that is, $M \in P^\oplus$.

As usual, we denote by $t^\bullet$ and $^\bullet t$ the multisets of postconditions and preconditions of t, respectively, that is, $t^\bullet(p) = F(t, p)$ and $^\bullet t(p) = F(p, t)$. A

---

[3] Strictly speaking, decidability of the wqo and effective $Pred$-basis are not part of the definition of WSTS, but of the so called *effective* WSTS. These properties are needed to ensure decidability of coverability.

transition t is enabled at marking M if ${}^\bullet t \subseteq M$. The reached state of N after the firing of t is $M' = (M - {}^\bullet t) + t^\bullet$.

We will write $M \xrightarrow{t} M'$ if M' is the reached marking after the firing of t at marking M. We also write $M \rightarrow M'$ if there is some t such that $M \xrightarrow{t} M'$. The reflexive and transitive closure of $\rightarrow$ is denoted by $\rightarrow^*$. For a transition sequence $\tau = t_1 \ldots t_m$ we will write $M \xrightarrow{\tau} M'$ to denote the consecutive firing of transitions t_1 to t_m, as expected.

3. Petri nets with name creation

Let us now extend P/T nets with the capability of name management by defining ν-PN. In a ν-PN names can be created, communicated and matched. We can use this mechanism to deal with authentication issues [25], correlation or instance isolation [5]. We formalize name management by replacing ordinary tokens by distinguishable ones, thus adding colours to our nets. We fix a set Id of names, that can be carried by tokens of any ν-PN. In order to handle these colors, we need matching variables labelling the arcs of the nets, taken from a fixed set Var. Moreover, we add a primitive capable of creating new names, formalized by means of special variables in a set $\Upsilon \subset Var$, ranged by $\nu, \nu_1, \ldots$ that can only be instantiated to fresh names.

Definition 1. A ν-PN is a tuple $N = (P, T, F)$, where P and T are finite disjoint sets, $F : (P \times T) \cup (T \times P) \rightarrow Var^\oplus$ is such that for every $t \in T$, $\Upsilon \cap pre(t) = \emptyset$ and $post(t) \setminus \Upsilon \subseteq pre(t)$, where $pre(t) = \bigcup_{p \in P} supp(F(p, t))$ and $post(t) = \bigcup_{p \in P} supp(F(t, p))$.

We also take $Var(t) = pre(t) \cup post(t)$. To avoid tedious definitions, along the paper we will consider a fixed ν-PN $N = (P, T, F)$.

Definition 2. A marking of N is a function $M : P \rightarrow Id^\oplus$. We denote by $Id(M)$ the set of names in M, that is, $Id(M) = \bigcup_{p \in P} supp(M(p))$.

We will assume a fixed initial marking M_0 of N. Like in other classes of high-order nets, transitions are fired with respect to a mode, that chooses which tokens are taken from preconditions and which are put in postconditions. Given a transition t of a net N, a mode of t is an injection $\sigma : Var(t) \rightarrow Id$, that instantiates each variable to an identifier. We will use $\sigma, \sigma', \sigma_1 \ldots$ to range over modes.

Definition 3. Let M be a marking, $t \in T$ and σ a mode for t. We say t is enabled with mode σ if for all $p \in P$, $\sigma(F(p, t)) \subseteq M(p)$ and $\sigma(\nu) \notin Id(M)$ for all $\nu \in \Upsilon \cap Var(t)$. The reached state after the firing of t with mode σ is the marking M', given by

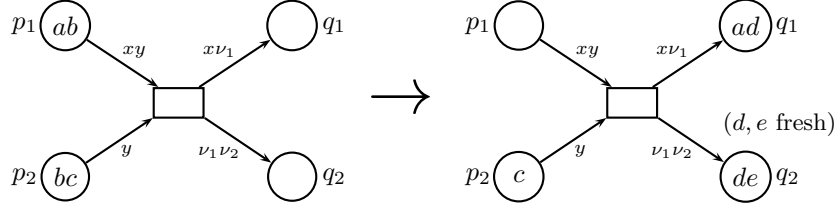$$M'(p) = (M(p) - \sigma(F(p, t))) + \sigma(F(t, p)) \ \ for \ all \ p \in P.$$

Figure 1: A simple ν-PN

We will write $M \xrightarrow{t(\sigma)} M'$ to denote that M' is reached from M when t is fired with mode σ, and extend the notation as done for P/T nets. In particular, for a sequence $\tau = t_1(\sigma_1) \ldots t_m(\sigma_m)$ we will write $M \xrightarrow{\tau} M'$ to denote the consecutive firings of $t_1(\sigma_1)$ to $t_m(\sigma_m)$. We will denote by $Reach(N)$ the set of reachable markings of N. Finally, we will assume that $\bullet \in Id$, so that we can also have ordinary tokens in our nets.

Figure 1 depicts a simple ν-PN with four places and a single transition. This transition moves one token from p_1 to q_1 (because of variable x labelling both arcs), removes a token from p_1 and p_2 provided they carry the same name (variable y appears in both incoming arcs but it does not appear in any outgoing arc), and two different names are created, one appears both in q_1 and q_2 (because of variable $\nu_1 \in \Upsilon$) and the other appears only in q_2 (because of variable $\nu_2 \in \Upsilon$).

Notice that we demand modes to be injections (unlike in [28]), which formalizes the fact that we can check for inequality. For instance, in the example in Fig. 1 the two tokens taken from p_1 must carry different names because we are labelling the arc from p_1 to t with two different variables, namely x and y. The capability of checking for inequality among all the names involved in the firing of a transition improves the expressive power of the model (see Fig. 2). The problem of proving that this improvement is strict is still open.

If a ν-PN has no arc labelled with variables from Υ then only a finite number of identifiers (those in the initial marking) can appear in any reachable marking. It is easy to see that these nets can be expanded to an equivalent P/T net. In particular, reachability is decidable for any such net, as it is for P/T nets [9], unlike for ν-PN [18].

We will work with a subclass of ν-PN without weights and in which transtions can at most create one fresh name.

Definition 4. A ν-PN $N = (P, T, F)$ is normal if there is $\nu \in \Upsilon$ such that:

- for every pair $(x, y) \in (P \cup T) \times (T \cup P)$, $|F(x, y)| \leq 1$,

- if $F(x, y) \cap \Upsilon \neq \emptyset$ then $F(x, y) = \{\nu\}$.

Every ν-PN can be simulated by a normal ν-PN. Intuitively, the simulation considers for each transition several transitions that must be fired consecutively, whenever the original net takes several tokens from the same place. Since the firing of a transition in the original net becomes non-atomic in the simulation, it can introduce deadlocks (whenever the "transaction" cannot be accomplished).
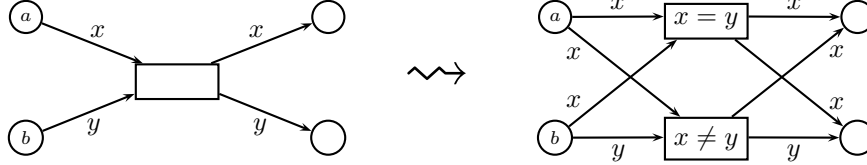
6

Figure 2: The net on the left cannot check for inequalities (it can fire its transition when $a = b$ or $a \neq b$). The net on the right can fire the transition in the top when $a = b$, and the one in the bottom when $a \neq b$.

However, it preserves all the properties we will consider in this paper. Therefore, from now on we will assume that ν-PNs are normal when needed.

4. Undecidability of reachability for ν-PN

Let us now prove that reachability is undecidable for ν-PN. In [18] (and independently in [25]) undecidability of reachability is proved by reducing reachability of the final state with all the counters containing zero in Minsky machines to reachability in ν-PN. In this section we prove that same result in a more simple way, by reducing reachability of inhibitor nets (that allow to check for zero) to reachability in ν-PN.

An inhibitor net is a tuple $N = (P, T, F, F_{in})$, where P and T are disjoint sets of places and transitions, respectively, $F \subseteq (P \times T) \cup (T \times P)$, and $F_{in} \subseteq P \times T$. Pairs in F_{in} are inhibitor arcs. For a transition $t \in T$ we write $\bullet t = \{p \in P \mid (p, t) \in F\}$, $t^\bullet = \{p \in P \mid (t, p) \in F\}$ and $^i t = \{p \in P \mid (p, t) \in F_{in}\}$. In figures we will draw a circle instead of an arrow to indicate that an arc is an inhibitor arc.

A *marking* of an inhibitor net N is a multiset of places of N. A transition t of N is enabled if $M(p) > 0$ for all $p \in \bullet t$ and $M(p) = 0$ for all $p \in {}^i t$. In that case t can be fired, producing $M' = (M - {}^\bullet t) + t^\bullet$.

Proposition 1. *Reachability is undecidable for ν-PN.*

PROOF. Given an inhibitor net $N = (P, T, F, F_{in})$ we build a ν-PN $N^* = (P \cup \bar{P}, T, F^*)$ that simulates it as follows:

- If $(p, t) \in F$ then $F^*(p, t) = F^*(\bar{p}, t) = F^*(\bar{p}, t) = \{x_p\}$ (and analogously for $(t, p) \in F$),

- If $(p, t) \in F_{in}$ then $F^*(\bar{p}, t) = \{x_p\}$ and $F^*(t, \bar{p}) = \{\nu\}$.

- $F^*(x, y) = \emptyset$ elsewhere.

Moreover, if M_0 is the initial marking of N, we consider a different identifier a_p for each place p of N. Then, we define the initial marking of N^* as $M_0^*(\bar{p}) = \{a_p\}$ and $M_0^*(p) = \{a_p, \overset{M_0(p)}{\ldots}, a_p\}$, for each $p \in P$.

Intuitively, for each place p of N we consider a new place $\bar{p}$ in N^*. The construction of N^* is such that $\bar{p}$ contains a single token at any time. The firing
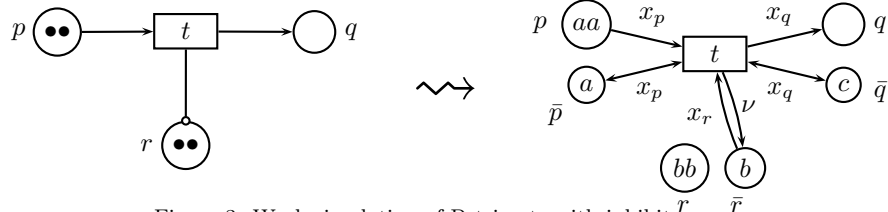
7

Figure 3: Weak simulation of Petri nets with inhibitor arcs

of any transition ensures that the token being used in p coincides with that in $\bar{p}$. Every time a transition checks the emptyness of a place p, the content of $\bar{p}$ is replaced by a fresh token, so that no token remaining in p can be used. In this way, our simulation introduces some garbage tokens whenever it cheats, that once become garbage, always stay like that. Moreover, notice that any marking of N^* of the form M^* for some marking M of N does not contain any garbage, so that it comes from a correct simulation. Fig. 3 depicts a simple inhibitor net and its simulation. Then M is reachable in N from M_0 if and only if M^* is reachable in N^* from M_0^*. Thus, we have reduced reachability in inhibitor nets, which is undecidable [9], to reachability in ν-PN.

5. Strict well structuredness of ν-PN

In this section we prove that the transition sytem generated by a ν-PN is strictly well structured [3, 10]. This will imply decidability of coverability, boundedness and termination. For that purpose, we can proceed following the next steps. In the first place, we need to define an order in the set of configurations, markings in our case, that induces the property of coverability. This order must be a decidable wqo. Then we must prove that this order is strictly monotonic with respect to the transition relation. Finally, we have to prove that it has effective *Pred*-basis.

5.1. Defining the order

One could think that the order we are interested in for ν-PN is the following:

$$M_1 \sqsubseteq M_2 \Leftrightarrow M_1(p) \subseteq M_2(p) \text{ for all } p \in P$$

This order is not a well quasi-order: it suffices to consider a ν-PN with a single place p and a sequence of pairwise different identifiers $(a_i)_{i=1}^{\infty}$, ant define $M_i(p) = \{a_i\}$ for all $i = 1, 2, \ldots$ which trivially satisfies that for all $i < j$, $M_i \not\sqsubseteq M_j$.

However, this order is too restrictive, since it does not take into account the abstract nature of pure names. Indeed, whenever a new name is created, actually any other fresh name could have been created. Therefore, reachability (or coverability) of a given marking is equivalent to reachability (or coverability) of any marking produced after consistently renaming the new names in it. For homogeneity, we will suppose that we can rename every name, even those appearing in the initial marking (which, after all, are a fixed number of names). To capture these intuitions, we identify markings up to renaming of names.

Definition 5. Given two markings M and M' we say that they are α-equivalent, and we write $M \equiv_\alpha M'$, if there is a bijection $\iota : Id(M) \to Id(M')$ such that $M'(p)(\iota(a)) = M(p)(a)$ for all $p \in P$ and $a \in Id(M)$.

We will write $M \equiv_\iota M'$ to stress the use of the particular mapping ι in the previous definition. Moreover, for a marking M and set of identifiers A, any bijection $\iota : Id(M) \to A$ defines a marking that we denote as $\iota(M)$, given by $\iota(M)(p)(\iota(a)) = M(p)(a)$, which is α-equivalent to M.

Proposition 2. *The behavior of ν-PNs is invariant under α-conversion. More specifically, let $M_1 \overset{t(\sigma)}{\to} M_1'$:*

- *If $M_1 \equiv_\alpha M_2$ then there is M_2' and σ' such that $M_1' \equiv_\alpha M_2'$ and $M_2 \overset{t(\sigma')}{\to} M_2'$.*

- *If $M_1' \equiv_\alpha M_2'$ then there is M_2 and σ' such that $M_1 \equiv_\alpha M_2$ and $M_2 \overset{t(\sigma')}{\to} M_2'$.*

PROOF. Let $A = Id(M_1) \setminus Id(M_1')$ and B the set of names created by $t(\sigma)$. Then, $Id(M_1') = (Id(M_1) \setminus A) \cup B$. Notice that $B \subseteq \{b\}$, for some $b \in Id$, assuming N is normal.

- Assume $M_1 \equiv_\iota M_2$ and let $\sigma' = \iota \circ \sigma$. Transition t can be fired from M_2 with mode σ', obtaining M_2' with $Id(M_2') = (Id(M_1') \setminus \iota(A)) \cup B'$ for some B' of the same cardinality than B. We define ι' by extending ι to B so that $\iota(B) = B'$, which verifies $M_2 \equiv_{\iota'} M_2'$.

- Assume now that $M_1 \equiv_{\iota'} M_2$ and let us define $\iota : Id(M_1) \to Id(M_2') \cup A$ by $\iota(a) = \iota'(a)$ if $a \in Id(M_1')$, and $\iota(a) = a$ if $a \in A$. Then $M_2 = \iota(M_1)$ and $\sigma' = \iota \circ \sigma$ satisfy $M_1 \equiv_{\iota'} M_2$ and $M_2 \overset{t(\sigma')}{\to} M_2'$.

For instance, if we represent a marking M of the net in Fig. 1 by a tuple $(M(p_1), M(p_2), M(p_3), M(p_4))$, then $M_1 = (\{a, b\}, \{b, c\}, \emptyset, \emptyset)$ and $M_2 = (\{a, c\}, \{b, c\}, \emptyset, \emptyset)$ are two α-equivalent markings of that ν-PN. Indeed, $M_1 \equiv_\iota M_2$ with $\iota(a) = a$, $\iota(b) = c$ and $\iota(c) = b$. M_1 can evolve to the marking $M_1' = (\emptyset, \{c\}, \{a, d\}, \{d, e\})$ when it fires t and M_2 can evolve to $M_2' = (\emptyset, \{b\}, \{a, e\}, \{d, e\})$. Notice that also $M_1' \equiv_\alpha M_2'$.

Let us now define the order we are interested in, by modifying the order $\sqsubseteq$ between markings with the help of the α-equivalence relation $\equiv_\alpha$.

Definition 6. Let M_1 and M_2 be markings of N. We will write $M_1 \sqsubseteq_\alpha M_2$ if there is a marking M_1' such that $M_1' \equiv_\alpha M_1$ and $M_1' \sqsubseteq M_2$.

Then, $M_1 \sqsubseteq_\alpha M_2$ when there is ι such that $M_1 \equiv_\iota M_1' \sqsubseteq M_2$, or equivalently, when $\iota(M_1) \sqsubseteq M_2$. We will write $M_1 \sqsubseteq_\iota M_2$ to emphasize on the use of ι. Clearly, $\sqsubseteq_\alpha$ is a decidable quasi order. Moreover, the kernel of $\sqsubseteq_\alpha$ is $\equiv_\alpha$, that is, $M_1 \sqsubseteq_\alpha M_2$ and $M_2 \sqsubseteq_\alpha M_1$ iff $M_1 \equiv_\alpha M_2$.

We will now see that the set of markings, ordered by $\sqsubseteq_\alpha$, is a wqo. In particular, notice that the counterexample we saw to prove that $\sqsubseteq$ is not a wqo is no longer valid, since all those markings are α-equivalent. In order to prove that $\sqsubseteq_\alpha$ is a wqo we map it to a multiset order which is known to be a wqo.

A marking is a mapping $M : P \to Id \to \mathbb{N}$ that says, for a given place p and an identifier a, how many times the token a can be found in place p. However, we can also currify those mappings as $M : Id \to P \to \mathbb{N}$. Since the behavior of a net is invariant under renaming, as we proved in Prop. 2, we can represent markings (modulo $\equiv_\alpha$) as multisets in $(P \to \mathbb{N})^\oplus$, that is, in $(P^\oplus)^\oplus$.

In this way, we represent markings by means of multisets, with a cardinality that equals the number of different identifiers appearing in it.

As an example, let us consider a net with only two places p_1 and p_2, and a marking M such that $M(p_1) = \{a, a, b, c\}$ and $M(p_2) = \{b, c\}$. We can represent that marking by the multiset of cardinality 3, since there are 3 different identifiers in M, namely by the multiset $\{\{p_1, p_1\}, \{p_1, p_2\}, \{p_1, p_2\}\}$, where the multiset $\{p_1, p_1\}$ represents identifier a, one of the two multisets $\{p_1, p_2\}$ represents b and the other $\{p_1, p_2\}$ represents c. Let us see it formally:

Definition 7. For a marking M of N, we define $M_a \in P^\oplus$ by $M_a(p) = M(p)(a)$ and $\overline{M} = \{M_a \mid a \in Id(M)\} \in (P^\oplus)^\oplus$.

Let us denote by $\ll$ the canonic order in $(P^\oplus)^\oplus$. It is well known that $\ll$ is a wqo. Moreover, it coincides with $\sqsubseteq_\alpha$, as we prove next.

Lemma 1. *Let M_1 and M_2 be two markings. Then $M_1 \sqsubseteq_\alpha M_2$ iff $\overline{M}_1 \ll \overline{M}_2$.*

PROOF. Let $\overline{M}_1 = \{A_1, \ldots, A_n\}$ and $\overline{M}_2 = \{B_1, \ldots, B_n\}$ with $A_i = M_i^{a_i}$ and $B_j = M_2^{b_j}$. If $M_1 \sqsubseteq_\iota M_2$ then define $h(i)$ such that $B_{h(i)} = M_2^{\iota(a_i)}$. Then $A_i(p) = M_1(p)(a_i) \leq M_2(p)(\iota(a_i)) = B_{h(i)}(p)$, so that $A_i \subseteq B_{h(i)}$ and therefore $\overline{M}_1 \ll \overline{M}_2$.

Conversely, since $\overline{M}_1 \ll \overline{M}_2$, there is $h : \{1, \ldots, n\} \to \{1, \ldots, m\}$ such that $A_i \subseteq B_{h(i)}$. Let us define $\iota : Id(M_1) \to Id(M_2)$ by $\iota(a_i) = b_{h(i)}$. Then we have $M_1(p)(a_i) = M_1^{a_i}(p) \leq M_2^{b_{h(i)}}(p) = M_2^{\iota(a_i)}(p) = M_2(p)(\iota(a_i))$. Therefore, $M_1(p)(a) \leq M_2(p)(\iota(a))$ for all $a \in Id(M_1)$ and the thesis follows.

Finally, we can conclude that the order $\sqsubseteq_\alpha$ is, indeed, a wqo.

Proposition 3. *$\sqsubseteq_\alpha$ is a wqo.*

PROOF. Let $M_0, M_1, M_2, \ldots$ be an infinite sequence of markings. Let us consider the sequence $\overline{M}_0, \overline{M}_1, \overline{M}_2, \ldots$. Since $\ll$ is a wqo, there are two indices $i < j$ such that $\overline{M}_i \ll \overline{M}_j$. By Lemma 1 we have that $M_i \sqsubseteq_\alpha M_j$, from which the thesis follows.

5.3. Strict monotonicity

Now let us see the next condition for strict well-structuredness, namely strict monotonicity of the firing relation with respect to $\sqsubseteq_\alpha$. As a first step, let us see it for $\sqsubseteq$.

Lemma 2. *The firing relation of ν-PN is strictly monotonic with respect to $\sqsubseteq$.*

PROOF. Let us suppose that $M_1 \overset{t(\sigma)}{\to} M_2$ and $M_1 \sqsubset M_1'$. From the former, we know in the first place that $\sigma(F(p,t)) \in M_1(p)$ for all p because that firing is enabled, and $M_2(p) = M_1(p) - \{\sigma(F(p,t))\} + \{\sigma(F(t,p))\}$ by definition of firing. The latter implies $M_1(p) \subset M_1'(p)$. Then, for all p, $\sigma(F(p,t)) \in M_1(p) \subseteq M_1'(p)$ and, therefore, the transition is enabled in M_1'. So that t can be fired to obtain $M_2'(p) = M_1'(p) - \{\sigma(F(p,t))\} + \{\sigma(F(t,p))\}$. Since $M_1(p) \subset M_1'(p)$ we have that $M_2(p) = M_1(p) - \{\sigma(F(p,t))\} + \{\sigma(F(t,p))\} \subset M_1'(p) - \{\sigma(F(p,t))\} + \{\sigma(F(t,p))\} = M_2'(p)$ and the thesis follows.

Proposition 4. *The firing relation of ν-PN is strictly monotonic with respect to $\sqsubseteq_\alpha$.*

PROOF. It is a direct consequence of the previous lemma and Prop. 2.

5.4. Effective Pred-basis

Let us now move to the last condition we must check, effective *Pred*-basis. Let us denote by $\uparrow M$ and $\uparrow_\alpha M$ the upward closure of M with respect $\sqsubseteq$ and $\sqsubseteq_\alpha$, respectively.

Definition 8. Given a transition t of N and σ a mode for t, we define $Pred_t$ and $Pre_{t(\sigma)}$ as the functions mapping markings to sets of markings, defined by $Pred_t(M) = \{M' \mid \exists \sigma\ M' \overset{t(\sigma)}{\to} M\}$ and $Pred_{t(\sigma)}(M) = \{M' \mid M' \overset{t(\sigma)}{\to} M\}$, and extend them pointwise to sets of markings.

With these notations we need to compute $min(Pred_t(\uparrow_\alpha M))$ for each marking M and $t \in T$. By Prop. 2 it is enough to compute $min(Pred_t(\uparrow M))$. Notice that the minor set of $Pred_t(\uparrow M)$ is still considered with respect to $\sqsubseteq_\alpha$, so that it is finite.

When computing the predecessors of $\uparrow M$, it may be the case that M itself has no predecessors, but some other markings in $\uparrow M$ do. In the next definition we identify the least marking in $\uparrow M$ with predecessors. We will use the following notation: Given two markings M_1 and M_2 we will denote by $M_1 \sqcup M_2$ the marking given by $(M_1 \sqcup M_2)(p) = M_1(p) \sqcup M_2(p)$.

Definition 9. Let t be a transition of N, σ a mode of t and M a marking of N. We define $min_{t(\sigma)}(M) = M \sqcup \sigma(F(t,-))$, where $\sigma(F(t,-))$ is the marking of N defined by $\sigma(F(t,-))(p) = \sigma(F(t,p))$.

Indeed, $min_{t(\sigma)}(M)$ is a marking in $\uparrow M$ with some predecessors. Moreover, is the least such marking, as proved next.
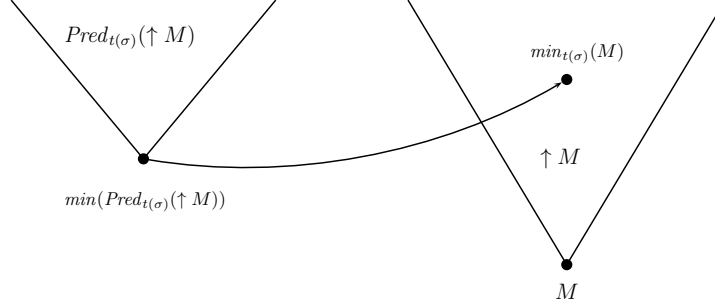
Figure 4: Computation of $Pred_{t(\sigma)}(\uparrow M)$

Lemma 3. *Let M be a marking of N, t a transition of N and σ a mode of t. Then $min_{t(\sigma)}(M)$ is the least M' such that $M \sqsubseteq M'$ and $Pre_{t(\sigma)}(M') \neq \emptyset$.*

PROOF. Let us write $\bar{M} = min_{t(\sigma)}(M)$. Trivially, $M \sqsubseteq \bar{M}$. Let us see that $Pre_{t(\sigma)}(\bar{M}) \neq \emptyset$. For that purpose, let M_0 be the marking defined by $M_0(p) = (M \sqcup \sigma(F(t, -)))(p) - \{\sigma(F(t, p))\} + \{\sigma(F(p, t))\}$ and let us see that $M_0 \overset{t(\sigma)}{\to} \bar{M}$. In the first place, $t(\sigma)$ is enabled in M_0, since $\sigma(F(p, t)) \in M_0(p)$ for each place p. Then the transition can be fired in mode σ and $M_0(p) - \sigma(F(p, t)) + \sigma(F(t, p)) = \bar{M}(p)$. Finally, if $M_1 \overset{t(\sigma)}{\to} M_2$ and $M \sqsubseteq M_2$ let us see that $\bar{M} \sqsubseteq M_2$. Since $M \sqsubseteq M_2$ it holds that $M(p) \subseteq M_2(p)$, for all p. Then $\bar{M}(p) = M(p) \sqcup \sigma(F(t, p)) \subseteq M_2(p) \sqcup \sigma(F(t, p)) \subseteq M_2(p)$, and the thesis follows.

Finally, let us see that we can use $min_{t(\sigma)}(M)$ to compute $min(Pred_t(\uparrow M))$.

Proposition 5. $Pred_{t(\sigma)}(\uparrow M) = \uparrow Pred_{t(\sigma)}(min_{t(\sigma)}(M))$

PROOF. Let $\overline{M}$ such that $Pred_{t(\sigma)}(\uparrow M) = \uparrow \overline{M}$. Since $min_{t(\sigma)}(M) \in \uparrow M$, $Pred_{t(\sigma)}(min_{t(\sigma)}(M)) \in \uparrow \overline{M}$, so that $\overline{M} \sqsubseteq Pred_{t(\sigma)}(min_{t(\sigma)}(M))$. Let us see that also $Pred_{t(\sigma)}(min_{t(\sigma)}(M)) \sqsubseteq \overline{M}$ holds. Indeed, $\overline{M} \in Pred_{t(\sigma)}(\uparrow M)$, so there is $M' \in \uparrow M$ such that $\overline{M} \overset{t(\sigma)}{\to} M'$. By the previous lemma, since M' has predecessores, $min_{t(\sigma)}(M) \sqsubseteq M'$, which entails by the previous lemma that the same relation also holds for their predecessors (because the effect of $t(\sigma)$ is constant), and hence the thesis.

Fig. 4 can give you some insight about the proof of the previous result. A marking M induces an upwards closed set, the cone in the right handside of Fig. 4. We want to compute (a finite representation of) the set of the predecessors of the markings in that cone. For that purpose, we first obtain $min_{t(\sigma)}(M)$, which is known to have a predecessor, according to Prop. 3, that is trivially computable. Therefore, every marking in the left handside cone can reach in one step the cone in the right.

Let us now see that in order to compute $min(Pred_t(\uparrow M))$ it is enough to consider a finite ammount of modes.

12

Proposition 6. *Let M be a marking, t a transition and O a set of identifiers with $|O| = |Var(t)|$. If $M' \in Pred_t(\uparrow M)$ then there is $\sigma : Var(t) \to Id(M) \cup O$ and $M'' \equiv_\alpha M'$ such that $M'' \in Pred_{t(\sigma)}(\uparrow M)$.*

PROOF. Let σ' such that $M' \overset{t(\sigma')}{\to} \overline{M}$ with $M \sqsubseteq \overline{M}$. Because of the latter, $Id(\overline{M}) = Id(M) \cup O'$ for some set of identifiers O'. Let us write $\sigma'(x) = o'_x$ whenever $\sigma'(x) \in O'$. For each such $x \in Var(t)$, choose a different $o_x \in O$ (notice that this can be done because $|O| = |Var(t)|$). Let us define $\sigma : Var(t) \to Id(M) \cup O$ as follows: $\sigma(x) = \sigma'(x)$ if $\sigma'(x) \in Id(M)$, and $\sigma(x) = o_x$ if $\sigma'(x) \in O'$. Let also $\iota : Id(M') \to (Id(\overline{M}) \setminus O') \cup O$ defined by $\iota(o'_x) = o_x$ and $\iota(a) = a$ elsewhere. Finally, let us take $M'' = \iota(M')$ and $\overline{M}'$ such that $M'' \overset{t(\sigma)}{\to} \overline{M}'$. It holds that $\overline{M}' \in \uparrow M$ and the thesis follows.

Therefore, in order to compute $Pred_t(\uparrow M)$ we can fix a set of names O with as many names as variables in $Var(t)$, and consider only modes mapping variables to names in $Id(M)$ or in O. Notice that there are finitely many such modes.

Proposition 7. *For each M, the set $min(Pred_t(\uparrow M))$ is computable.*

PROOF. We can compute $min(Pred_t(\uparrow M))$ as follows:

$$min(Pred_t(\uparrow M)) = min\Big(\bigcup_\sigma Pred_{t(\sigma)}(\uparrow M)\Big) = min\Big(\bigcup_\sigma min(Pred_{t(\sigma)}(\uparrow M))\Big)$$

By Prop. 5 the last term can be computed as $min\big(\bigcup_\sigma Pred_{t(\sigma)}(min_{t(\sigma)}(M))\big)$. Each $Pred_{t(\sigma)}(min_{t(\sigma)}(M))$ is computable, and because by Prop. 6 it is enough to consider finitely many modes, we conclude.

We have proved that ν-PNs are strictly well structured transitions systems.

Proposition 8. *Coverability, boundedness and termination are decidable for ν-PN.*

One can think that we have proved decidability of a weak version of the coverability problem, that in which we allow arbitrary renaming of identifiers. For instance, if we consider the net in the left of Fig. 2, and we ask whether the marking M given by $M(p_0) = M(p_1) = \emptyset$, $M(p_3) = \{b\}$ and $M(p_4) = \{a\}$ can be covered, the result would be affirmative, since the marking obtained by exchanging a and b in M (which is α-equivalent to M) is reachable in one step.

However, we can use this apparently weak version to decide a more restricted version of coverability: Let M_0 and M_f be two markings of a ν-PN $N = (P, T, F)$. We want to decide if we can cover M_f from M_0 without allowing renaming of names. Thus, if a name a appears both in M_0 and in M_f we want to reach a marking M such that $M_f \sqsubseteq_\iota M$ with ι satisfying $\iota(a) = a$. Since $R = Id(M_0) \cap Id(M_f)$ contains only a finite number of names, we can add new
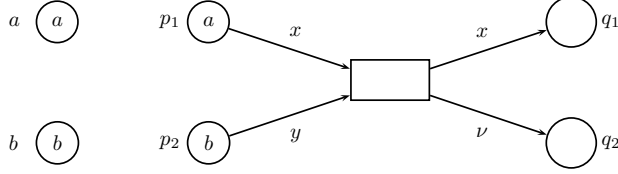
13

Figure 5: The ν-PN in the left of Fig. 2 extended to decide a restricted version of coverability

places in order to ensure the latter. We define the ν-PN $N^* = (P \cup R, T, F)$. For any marking M we define $M^*(p) = M(p)$ if $p \notin R$ and $M^*(r) = \{r\}$ for all $r \in R$. By construction of N^*, places in R are isolated, so that their tokens are never moved or removed. In particular, for any reachable M with $M_f \sqsubseteq_\iota M$ it holds $\iota(a) = a$ for every $a \in R$.

Let us again consider the example in Fig. 2. Following the previous construction, that can be seen in Fig. 5, we add a place for a and another one for b. When we execute this new net, the reasoning we followed before now fails. In one step we can reach M' with $M'(p_0) = M'(p_1) = \emptyset$, $M'(p_3) = M'(a) = \{a\}$ and $M'(p_4) = M'(b) = \{b\}$. However, thanks to the newly added places, it is not true that M' equals the result of exchanging a and b in M^* (using the notations of the proof of the previous result).

We could ask ourselves whether we can consider a ligther version of the reachability problem in which we allow renaming of names, as we are doing with coverability, that allows us to obtain decidability. However, decidability of α-reachability implies the decidability of reachability, by using the same trick we have used for coverability.

5.5. Complexity of the decision procedures

Now we obtain hardness results for the decision problems shown to be decidable in Prop. 8. We do it by means of a simulation of reset nets by ν-PN. The construction is very similar to the one we used in Sect. 4 to simulate inhibitor nets with ν-PN.

A reset net is a tuple $N = (P, T, F, F_r)$, where P and T are disjoint sets of places and transitions, respectively, $F \subseteq (P \times T) \cup (T \times P)$, and $F_r \subseteq P \times T$. Pairs in F_r are reset arcs. For a transition $t \in T$ we write ${}^\bullet t = \{p \in P \mid (p, t) \in F\}$ and ${}^r t = \{p \in P \mid (p, t) \in F_r\}$, and analogously for $t^\bullet$. For simplicity, and without loss of generality, we assume that ${}^r t \cap t^\bullet = \emptyset$ for every $t \in T$.

A marking of a reset net N is a multiset of places of N. A transition t is enabled in M if $M(p) > 0$ for all $p \in {}^\bullet t$. In that case t can be fired, producing M' defined as[4]

- $M'(p) = (M(p) - F(p, t)) + F(t, p)$ for all $p \notin {}^r t$,

- $M'(p) = 0$ for all $p \in {}^r t$.

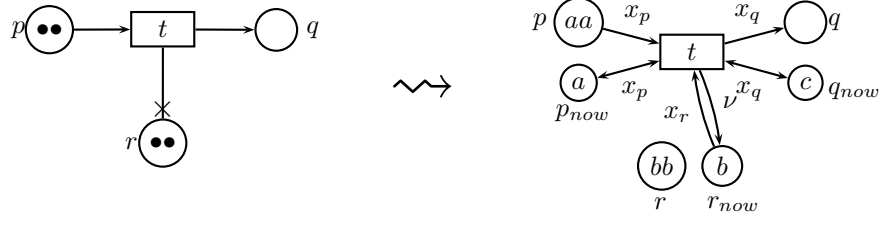[4]Note that we are identifying F with its characteristic funcion.

14

Figure 6: Simulation of reset nets

Proposition 9. *Given a reset net $N = (P, T, F, F_r, M_0)$ we can build in polynomial time a ν-PN $N^* = (P \cup \bar{P}, T, F^*, M_0^*)$ such that:*

- *If M is reachable in N then there is M^* reachable in N^* such that for every $p \in P$ there is $a_p \in Id$ with $M^*(\bar{p}) = \{a_p\}$ and $M^*(p)(a_p) = M(p)$.*

- *If M^* is reachable in N^* then there is M reachable in N and $a_p \in Id$ for every $p \in P$ such that $M^*(\bar{p}) = \{a_p\}$ and $M^*(p)(a_p) = M(p)$.*

In particular,

- *N terminates iff N^* terminates,*

- *Given M we can also build M^* such that M can be covered in N iff M^* can be covered in N^*.*

PROOF. Let $N = (P, T, F, F_r)$ be a reset net. We consider a different variable x_p for each $p \in P$. Then we define $N^* = (P \cup \bar{P}, T, F^*)$ as follows:

- If $(p, t) \in F$ then $F^*(p, t) = F^*(\bar{p}, t) = F^*(\bar{p}, t) = x_p$ (analogously for $(t, p) \in F$),

- If $(p, t) \in F_r$ then $F^*(\bar{p}, t) = x_p$ and $F^*(t, \bar{p}) = \nu$.

- $F^*(x, y) = \emptyset$ elsewhere.

Moreover, if M_0 is the initial marking of N, we consider a different identifier a_p for each place p of N. Then, we define the initial marking of N^* as $M_0^*(p_{now}) = \{a_p\}$ and $M_0^*(p) = \{a_p, \overset{M_0(p)}{\ldots}, a_p\}$, for each $p \in P$.

Intuitively, for each place p of N we consider a new place $\bar{p}$ in N^*. The construction of N^* is such that $\bar{p}$ contains a single token at any time. The firing of any transition ensures that the token being used in p coincides with that in $\bar{p}$. Every time a transition resets a place p, the content of $\bar{p}$ is replaced by a fresh token, so that no token remaining in p can be used. In this way, our simulation introduces some garbage tokens, that once become garbage, always stay like that. Fig. 6 depicts a simple reset net and its simulation.

Proposition 10. *Coverability, boundedness and termination for ν-PN are not primitive recursive.*

15

PROOF. Since coverability and termination are Ackermand-hard for reset nets [29], the previous construction entails Ackerman-hardness for coverability and termination in ν-PN. This hardness extends to boundedness by means of a very simple reduction: given a ν-PN N it is enough to build N' by adding to N a place in which an ordinary token is put in every firing. Clearly, N terminates iff N' is bounded.

6. Weaker forms of boundedness

Let us now discuss weaker forms of boundedness. In the first place, we characterize boundedness (finiteness of the reachability set) in terms of the form of every reachable marking, as is usual in Petri nets.

Lemma 4. *Given a ν-PN with an initial marking, the set of reachable markings is finite (up to $\equiv_\alpha$) if and only if there is $n \geq 0$ such that every reachable marking M satisfies $M(p)(a) \leq n$ for all $p \in P$ and $a \in Id$.*

PROOF. If *Reach* is finite we can define $s = max\{|Id(M)| \mid M \in Reach\}$ and $k = max\{M(p)(a) \mid M \in Reach, \ p \in P, \ a \in Id(M)\}$. Then, for each reachable M, $|M(p)| = |\sum_{a \in supp(M(p))} M(p)(a)| \leq k \cdot s$ and the net is bounded. Conversely, if the net is unbounded then for each n there is a reachable M_n such that $|M_n(p)| > n$ for all p, which implies the thesis.

We will use the previous characterization in order to factorize the property of boundedness. Unlike ordinary P/T nets, that only have one infinite dimension, ν-PNs have two different sources of infinity: the number of different identifiers and the number of times each of those identifiers appear. Consequently, several different notions of boundedness arise, in one of the dimensions, in the other or in both.

Definition 10. Let N be a ν-PN.

- We say N is *width-bounded* if there is $n \in \mathbb{N}$ such that for all reachable M, $|Id(M)| \leq n$.

- We say N is *depth-bounded* if there is $n \in \mathbb{N}$ such that for all reachable M, for all $p \in P$ and for all $a \in Id, M(p)(a) \leq n$.

Indeed, width and depth-boundedness factorize boundedness.

Proposition 11. *N is bounded iff it is width-bounded and depth-bounded.*

PROOF. It is enough to consider that $|M(p)| = |\sum_{a \in Id(M)} M(p)(a)| \leq |Id(M)| \cdot max\{M(p)(a) \mid a \in Id\}$. If there is $n \in \mathbb{N}$ such that $|M(p)| \leq n$ then $|\sum_{a \in Id(M)} M(p)(a)| \leq n$ and since $Id(M) = \{a \in Id \mid M(p)(a) > 0$ for some $p\}$ we have that $|supp(M(p))| \leq n$ and and for all $a \in supp(M(p))$, $M(p)(a) \leq$

Figure 7: Width-bounded but not depth-bounded ν-PN (left) and viceversa (right)

n. Conversely, let us assume there are n and m such that $|supp(M(p))| \leq n$ and $M(p)(a) \leq m$. From the latter if follows that $max\{M(p)(a) \mid a \in supp(M(p))\} \leq m$. Then, by the previous observation, $|M(p)| \leq n \cdot m$ and the thesis follows.

Thanks to the previous result we know that if a ν-PN is bounded then it is width-bounded and depth-bounded. However, if it is unbounded it could still be the case that it is width-bounded (see left of Fig. 7) or depth-bounded (see right of Fig. 7), though not simultaneously width and depth-bounded.

In [27] we prove decidability of width-boundedness for ν-PN. The proof relies on the results in [11, 12] that establish a framework for forward analysis for WSTS. We do not show the details here, since they are rather involved.

Though width and depth-boundedness seem to play a dual role, the proof of decidability of width-boundedness can not be adapted in the case of depth-boundedness. Actually, depth-boundedness turns out to be undecidable, though this fact could be considered to be rather anti intuitive (actually, in the paper [7] there is a wrong decidability proof).

Proposition 12. *Depth-boundedness is undecidable for ν-PN.*

PROOF. Given a ν-PN N, let us consider the reset net N^* built in Prop. 9. Notice that N is bounded iff N^* is depth-bounded. Since boundedness in reset nets is undecidable [8] we can conclude.

7. Conclusions and Future Work

In this paper we have studied the expressive power of a simple extension of P/T nets with a primitive that creates fresh names. We knew that the expressive power of P/T nets is strictly increased because, unlike for P/T nets, reachability is undecidable. However, Turing-completeness is not reached. We have seen it by proving that ν-PNs are strictly well-structured systems. In particular, we obtain that coverability is still decidable for them, as well as boundedness. Therefore, ν-PN is in the class of models whose expressive power lies somewhere in between P/T nets and Turing machines, like Lossy FIFO channel systems [1] or reset nets [8].

We have also defined two orthogonal notions of boundedness. Since our nets have names as tokens, it can be the case that a bounded number of different names appear in every reachable marking. In that case (independently of the number of times that those each of those names appears) we say the net is width-bounded. Dually, if every name that appears in every reachable marking

appears only a bounded number of times (independently of how many different names appear) then we say that the net is depth-bounded. Though width-boundedness is decidable, we have proved undecidability of depth-boundedness by reducing boundedness in reset nets to it.

Many well structured transition systems have undecidable reachability, except some notable exceptions. Moreover, we know that coverability is always decidable for them. Thus, in order to compare the expressive power of different formalisms that lie in this class, reachability and coverability are not enough. One could consider other properties, as different notions of boundedness, though we have seen that boundedness properties tend to be rather tricky. A different option is to consider the languages generated when we label transitions with labels taken from a finite set. Because of the undecidability of reachability, if we accept words that can be recognized when reaching a given marking, then we generally obtain the set of recursively enumerable languages. In [13] the authors propose to use coverability as accepting condition instead. This yields a better framework to relate well structured transition systems. In [26] such framework is used in order to compare ν-PN with other Petri net extensions as Affine Well Nets or Data Nets. However, the distinction between ν-PN and Data Nets remains an open problem.

References

[1] P. A. Abdulla, and B. Jonsson. *Verifying Programs with Unreliable Channels.* Information and Computation, 127(2):91-101, 1996.

[2] M. Abadi, and A.D. Gordon. *A Calculus for Cryptographic Protocols: The Spi Calculus.* Fourth ACM Conference on Computer and Communications Security, ACM Press, 1997, 36-47.

[3] P. A. Abdulla, K. Cerans, B. Jonsson, and Y. Tsay. *Algorithmic analysis of programs with well quasi-ordered domains.* Information and Computation 160:109-127. Academic Press Inc., 2000.

[4] L. Cardelli, and A.D. Gordon. *Mobile Ambients.* 1st Int. Conf. on Foundations of Software Sciences and Computation Structures, FOSSACS'98. LNCS 1387, pp. 140-155. Springer, 1998.

[5] G. Decker, and M. Weske. *Instance Isolation Analysis for Service-Oriented Architectures.* In Proceedings of the 2008 IEEE International Conference on Services SCC'08, pp. 249-256. IEE Computer Society, 2008.

[6] Jörg Desel and Wolfgang Reisig. *Place/transition petri nets.* Lectures on Petri Nets I: Basic Models, LNCS 1491, pp.122–173. Springer, 1998.

[7] R. Dietze, M. Kudlek and O. Kummer. Decidability Problems of a Basic Class of Object Nets. *Fundamenta Informaticae* 79. IOS Press (2007) 295-302.

[8] C. Dufour, A. Finkel and Ph. Schnoebelen. *Reset nets between decidability and undecidability.* 25th Int. Col. on Automata, Languages and Programming, ICALP'98. LNCS 1443, pp. 103-115. Springer, 1998.

[9] J. Esparza and M. Nielsen. *Decidability issues for Petri Nets - a survey.* Bulletin of the EATCS 52:244-262 (1994).

[10] A.Finkel, and P.Schnoebelen. *Well-Structured Transition Systems Everywhere!* Theoretical Computer Science 256(1-2):63-92 (2001).

[11] A. Finkel and J.Goubault-Larrecq. Forward analysis for WSTS, Part I: Completions. *In Proceedings of the 26th International Symposium on Theoretical Aspects of Computer Science, STACS'09* (2009) 433-444.

[12] A. Finkel and J.Goubault-Larrecq. Forward analysis for WSTS, Part II: Complete WSTS. *In 36th International Colloquium on Automata, Languages and Programming, ICALP'09.* LNCS 5556. Springer (2009) 188-199.

[13] G. Geeraerts, J. Raskin, and L. Van Begin. *Well-structured languages.* Acta Informatica, 44:249-288. Springer, 2007.

[14] A. Gordon. *Notes on Nominal Calculi for Security and Mobility.* Foundations of Security Analysis and Design, FOSAD'00. LNCS 2171, pp. 262-330. Springer, 2001.

[15] M. Köhler, and H. Rölke. Properties of Object Petri Nets. 25th Int. Conf. on Petri Nets, ICATPN'04. LNCS 3099, pp. 278-297. Springer, 2004.

[16] M. Köhler. Reachable markings of object Petri nets. Fundamenta Informaticae 79(3-4):401-413 (2007)

[17] M. Köhler, and F. Heitmann. On the expressiveness of communication channels for object nets. Fundamenta Informaticae 93(13):205-219 (2009)

[18] O. Kummer. *Undecidability in object-oriented Petri nets.* Petri Net Newsletter, 59:18-23, 2000.

[19] O. Kummer, F. Wienberg, M. Duvigneau, J. Schumacher, M. Köhler, D. Moldt, H. Rölke, and R. Valk. An Extensible Editor and Simulation Engine for Petri Nets: Renew. In 25th Int. Conf. on Petri Nets, ICATPN'04. LNCS 3099, pp. 484-493. Springer, 2004.

[20] R. Lazic, T.C. Newcomb, J. Ouaknine, A.W. Roscoe and J. Worrell. *Nets with Tokens Which Carry Data.* Fundamenta Informaticae 88 (3):251-274. IOS Press, 2008.

[21] I. Lomazova. *Nested petri nets - a formalism for specification and verification of multi-agent distributed systems.* Fundamenta Informaticae 43(1-4):195-214. IOS Press, 2000.

[22] I. Lomazova, and Ph. Schnoebelen. *Some Decidability Results for Nested Petri Nets.* 3rd Int. Andrei Ershov Memorial Conf. on Perspectives of System Informatics, PSI'99. LNCS vol. 1755, pp. 208-220. Springer, 2000.

[23] E.C. Milner. Basic WQO- and BQO-theory. In I. Rival (ed.). Graphs and Order. The Role of Graphs in the Theory of Ordered Sets and Its Applications. D. Reidel Publishing Co.. pp. 487-502.

[24] R. Milner, J. Parrow, and D. Walker. *A Calculus of Mobile Processes, I.* Information and Computation 100(1): 1-40 (1992).

[25] F. Rosa-Velardo, D. de Frutos-Escrig, and O. Marroquín-Alonso. *On the expressiveness of Mobile Synchronizing Petri Nets.* SecCo'05. ENTCS 180(1):77-94. Elsevier, 2007.

[26] F. Rosa-Velardo, and G. Delzanno. Language-based Comparison of Nets with Black Tokens, Pure Names and Ordered Data. 4th Int. Conf. on Language and Automata Theory and Applications, LATA 2010. LNCS 6031, pp. 524-535. Springer, 2010.

[27] F. Rosa-Velardo, D. de Frutos-Escrig. Forward analysis for Petri nets with name creation. 31st Int. Conf. on Applications and Theory of Petri nets and other models of concurrency, PETRI NETS 2010. LNCS vol. 6128, pp. 185-205. Springer, 2010.

[28] F. Rosa-Velardo, D. de Frutos-Escrig. Name Creation vs. Replication in Petri Net Systems. Fundamenta Informaticae 88(3):329-356. IOS Press, 2008.

[29] P Schnoebelen. Revisiting Ackermann-Hardness for Lossy Counter Machines and Reset Petri Nets. In 355h Int. Symposium on Mathematical Foundations of Computer Science, MCFS'10. LNCS vol. 6281, pp. 616-628. Springer (2010)

[30] R. Valk. Petri Nets as Dynamical Objects. 16th Int. Conf. on Application and Theory of Petri Nets. Workshop proceedings, 1995.

[31] R. Valk. Petri Nets as Token Objects - An Introduction to Elementary Object Nets. 19th Int. Conf. on Applications and Theory of Petri Nets, ICATPN'98, LNCS 1420, pp. 1-25. Springer, 1998.