Fundamenta Informaticae XXI (2001) 1001–1029
IOS Press

Accelerations for the coverability set of Petri nets with names *

Fernando Rosa-Velardo
Facultad de Informática
fernandorosa@sip.ucm.es

María Martos-Salgado
Facultad de Informática
mrmartos@estumail.ucm.es

David de Frutos-Escrig
Facultad de CC. Matemáticas
defrutos@sip.ucm.es

Abstract. Pure names are identifiers with no relation between them, except equality and inequality. In previous works we have extended P/T nets with the capability of creating and managing pure names, obtaining ν-PNs and proved that they are strictly well structured (WSTS), so that coverability and boundedness are decidable. Here we use the framework recently developed by Finkel and Goubault-Larrecq for forward analysis for WSTS, in the case of ν-PNs, to compute the cover, that gives a good over approximation of the set of reachable markings. We prove that the least complete domain containing the set of markings is effectively representable. Moreover, we prove that in the completion we can compute least upper bounds of simple loops. Therefore, a forward Karp-Miller procedure that computes the cover is applicable. However, we prove that in general the cover is not computable, so that the procedure is non-terminating in general. As a corollary, we obtain the analogous result for Transfer Data nets and Data Nets. Finally, we show that a slight modification of the forward analysis yields decidability of a weak form of boundedness called width-boundedness, and identify a subclass of ν-PN that we call dw-bounded ν-PN, for which the cover is computable.

Keywords: Petri nets, pure names, infinite state systems, decidability, well structured transition systems, forward analysis, accelerations

Address for correspondence: Facultad de Informática. C\Prof. José García Santesmases, s/n - 28040 Madrid (Spain)
*This paper is a revised and extended version of [22]. Authors partially supported by the Spanish projects DESAFIOS10 TIN2009-14599-C03-01, PROMETIDOS S2009/TIC-1465 and TESIS TIN2009-14312-C02-01.

1. Introduction

Pure names have been extensively studied in the fields of security and mobility, because they can be used to represent different entities widely used in them. For instance, names can represent communicating channels in π-calculus terms, computing boundaries in the Ambient Calculus or ciphering keys in the spi Calculus [14]. In previous works we have extended P/T nets with a primitive to create fresh names, defining ν-PNs.[1] Names are represented as tokens, that are no longer indistinguishable. These tokens can move along the places of the net and be used to restrict the firing of some transitions, imposing for instance that two certain names at the preconditions match.

In [19] we proved that ν-PNs are Well Structured Transition Systems (WSTS). For WSTS it is possible to perform a backward analysis that computes the set $\uparrow Pre^*(\uparrow M)$ [1, 9], the set of predecessors of an upward-closed set $\uparrow M$. An effective representation of that set allows us to decide the coverability problem, by checking whether the initial marking $M_0 \in \uparrow Pre^*(\uparrow M)$. However, the construction of such sets is extremely expensive, with a non primitive recursive complexity [24].

Very recently, Finkel and Goubault-Larrecq have laid the foundation of a theory supporting forward analysis of WSTS [11, 12], computing $\downarrow Post^*(\downarrow M_0)$, the so called *cover* of the transition system. The cover provides a good over approximation of the set of reachable states, and its construction is generally more efficient in practice than that of $\uparrow Pre^*(\uparrow M)$. However, it is not always possible to obtain an effective representation of the cover [3]. The paper [11] establishes a theory for the completion of *well quasi orders* (wqos), so that we can always represent downward-closed sets by means of their least upper bounds. There it is proved that the least completion of X (that contains an adequate domain of limits, in the sense of [13]) is the so called ideal completion of X, or equivalently, the sobrification of X [15].

We will see here that the ideal completion of the set of markings can be effectively represented by mapping markings to the domain $\mathcal{MS}(\mathcal{MS}(P))$ of finite multisets of finite multisets of places. For that purpose we introduce the domain of ω-markings (analogous to the classical notion of ω-markings for P/T nets). In an ω-marking, not only some identifiers may appear an unbounded number of times in some places, as happens in classical ω-markings, but also an unbounded number of different identifiers may occur in a marking.

Assuming a complete domain (thus containing an adequate domain of limits), a generic Karp-Miller procedure to compute the cover is presented in [12]. This procedure is correct provided the WSTS is ∞-effective, which intuitively means that we can accelerate simple loops (flat loops, in the sense of [4]). We will see that ν-PNs are ∞-effective when we restrict the non-determinism arising in loops, so that we can apply to them the generic Karp-Miller procedure. Unfortunately, the procedure is not guaranteed to terminate for ν-PN. We will see that this is unavoidable, since we can reduce a problem related to boundedness which we call depth-boundedness, which is undecidable [23], to the computation of the cover.

Data nets [17] are Petri nets in which tokens are taken from a linearly ordered and dense domain, and capable of performing whole place operations, such as transfers or resets. Transfer Data Nets is the subclass of Data nets in which no resets are allowed, and Petri Data Nets is the subclass of Data Nets (and of Transfer Data Nets) in which no whole-place operation is allowed. Petri Data nets subsume ν-PNs [17], so that as a corollary, there cannot be an algorithm computing (a finite basis of) the cover of a Petri Data net, and therefore neither for a Transfer Data net, thus answering negatively to a question

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

posed in [12].

But even if there is no algorithm for the computation of the cover, we can use a slight modification of the forward Karp-Miller procedure to decide width-boundedness of ν-PNs [20, 5]. A net is width-bounded if only a bounded number of different names appear in each reachable marking. The paper [5] also establishes the decidability of width-boundedness (called m-boundedness there), but we claim that the algorithm presented there does not properly work in all the cases. This is because the algorithm stops whenever unboundedness is detected. However, width-unbounded nets may be bounded or not, so that we need to further explore the reachability graph to decide width-boundedness. For that purpose, we need ways to finitely represent downward-closed sets of reachable markings, our ω-markings. We already knew [20] that width-boundedness is decidable, but we obtain the result here as a simple application of our forward analysis.

Finally, we identify some subclasses for which the procedure does terminate. Width-bounded nets is one of such subclasses. We also prove that we can compute the cover of depth-bounded nets, though depth-boundedness is undecidable. Last, we consider a weaker form of boundedness, that we call dw-boundedness. A net is dw-bounded whenever only a bounded number of names are allowed to appear an unbounded number of times in each reachable marking. Again, we will see that we can compute the cover of a dw-bounded net, though, as we will see, dw-boundedness is undecidable.

The rest of the paper is structured as follows. Section 2 introduces our notations and some basic concepts. In Section 3 we present ν-PNs. In Section 4 we show how ν-PNs fit in the general framework for forward analysis of WSTS in [11, 12]. Section 5 proves the viability of a forward Karp-Miller procedure for ν-PNs, and the non-computability of the cover. Section 6 considers several subclasses of ν-PN: it proves decidability of width-boundedness, undecidability of dw-boundedness and computability of the cover for all the subclasses. Finally, Section 7 presents our conclusions and some directions for further work.

2. Preliminaries

Well quasi orders, directed complete partial orders

A quasi order $\leq$ is a reflexive and transitive binary relation on a set X. A partial order is an antisymmetric quasi order. A poset is a set endowed with a partial order. We write $a < b$ if $a \leq b$ and $b \not\leq a$. A sequence $(a_i)_{i\in\mathbb{N}}$ is increasing if $a_i \leq a_{i+1}$ for each $i \in \mathbb{N}$, and strictly increasing if $a_i < a_{i+1}$ for each $i \in \mathbb{N}$. A quasi order is simply said well (wqo) [10], if for every infinite sequence $a_0, a_1, ...$ there are i and j with $i < j$ such that $a_i \leq a_j$. Equivalently, an order is a wqo if every sequence has an increasing subsequence.

The downward closure $\downarrow E$ of $E \subseteq X$ is $\{y \in X \mid y \leq x \text{ for some } x \in E\}$. A set is downward closed iff $\downarrow E = E$. A basis of a downward closed set E is a set A such that $\downarrow A = E$. An element $x \in X$ is an upper bound of E if $y \leq x$ for all $y \in E$. We write $lub(E)$ to denote the least upper bound of E, when it exists. An element $x \in E$ is maximal if $x = y$ whenever $x \leq y \in E$; $MaxE$ is the set of maximal elements of E. A subset D of X is said to be directed if $lub(\{x, y\})$ exists for all $x, y \in D$. A poset is *directed complete* (dcpo) if every directed subset has a least upper bound. For an arbitrary subset E, $Lub(E) = \{lub(D) \mid D \text{ directed}, D \subseteq E\}$. The set $Lub(E)$ can be thought of as E together with all its limits. For a dcpo X, we write $x \ll y$ whenever $y \leq lub(D)$ implies $x \leq z$ for some $z \in D$, for all directed subset D. X is continuous if for all $x \in X$, $x = lub\{y \in X \mid y \ll x\}$.

Well Structured Transition Systems

A labelled transition system is a tuple $N = (X, \rightarrow, Act)$ with a set X of states, Act a set of actions and a transition relation $\rightarrow = \bigcup_{a \in Act} \xrightarrow{a}$, with $\xrightarrow{a} \subseteq X \times X$. We denote by $\xrightarrow{a}^*$ (resp. $\rightarrow^*$) the reflexive and transitive closure of $\xrightarrow{a}$ (resp. $\rightarrow$). $Post_{a,N}(M)$ (or just $Post_a(M)$) is the set $\{M' \mid M \xrightarrow{a} M'\}$ of immediate a-successors of M. $Post^*(M) = \{M' \mid M \rightarrow^* M'\}$ is the set of states reachable from M. Both $Post_a$ and $Post^*$ are extended pointwise to sets of states. A Well Structured Transition System (WSTS) is a tuple $N = (X, \rightarrow, Act, \leq)$, where $(X, \rightarrow, Act)$ is a labelled transition system, and $(X, \leq)$ is a wqo, satisfying the following monotonicity condition[2]: $M_1 \geq M_2 \xrightarrow{a} M_2'$ implies the existence of M_1' such that $M_1 \xrightarrow{a} M_1' \geq M_2'$. Given a state M, the *cover* of M is the set $\downarrow Post^*(M)$ (or equivalently, $\downarrow Post^*(\downarrow M)$ because of monotonicity), and we will denote it by $Cover_N(M)$ (or just $Cover(M)$ if there is no confusion). Given an initial state M_0, the cover of N is the cover of M_0. N is said to be *effective* if $Post_a(M)$ is finite and computable for all M, and $\leq$ is decidable. A WSTS $(X, \rightarrow, Act, \leq)$ is complete whenever $(X, \leq)$ is a continuous dcpo and for every $a \in Act$, $Post_a(Lub(E)) = Lub(Post_a(E))$ for every set E.

An ideal is a downward closed directed subset. The ideal completion $\overline{X}$ of a wqo X is the set of ideals of X, ordered by inclusion. Given a WSTS $N = (X, \rightarrow, Act, \leq)$, the ideal completion of N is the transition system $\overline{N} = (\overline{X}, \mapsto, Act)$, where $F \overset{a}{\mapsto} F' = \downarrow\{s' \mid s \xrightarrow{a} s',\ s \in F\}$. $(\overline{X}, \subseteq)$ is a continuous dcpo. However, $\overline{N}$ is not a WSTS in general. The Rado's structure is the set $R = \{(a, b) \in \mathbb{N} \times \mathbb{N} \mid a < b\}$ with the wqo $\leq_r$ given by $(a_1, b_1) \leq_r (a_2, b_2) \Leftrightarrow a_1 = a_2$ and $b_1 \leq b_2$, or $a_1 < a_2$. A wqo is an ω^2-wqo if it does not contain an isomorphic copy of the Rado's structure, and an ω^2-WSTS is a WSTS with an underlying ω^2-wqo [16]. Then, $\overline{N}$ is a WSTS iff N is a ω^2-WSTS [12]. There is a finer notion called better quasi order(bqo), whose definition is quite intricate. For a detailed definition see [18]. Bqos are closed under the multiset construction and they are ω^2-wqos [16].

Multisets

Given an arbitrary set A, we will denote by $\mathcal{MS}(A)$ the set of finite multisets of A, that is, the mappings $m : A \rightarrow \mathbb{N}$ with finite support, meaning that $S(m) = \{a \in A \mid m(a) > 0\}$ is finite. When needed, we identify each set with the multiset defined by its characteristic function, and use set notation for multisets when convenient. We denote by $|m| = \sum_{a \in S(m)} m(a)$ the cardinality of m. Given two multisets $m_1, m_2 \in \mathcal{MS}(A)$ we denote by $m_1 + m_2$ the multiset defined by $(m_1 + m_2)(a) = m_1(a) + m_2(a)$. We will write $m_1 \subseteq m_2$ if $m_1(a) \leq m_2(a)$ for every $a \in A$. Then, we can define $m_2 - m_1$, taking $(m_2 - m_1)(a) = m_2(a) - m_1(a)$. We will denote by $\emptyset \in \mathcal{MS}(A)$ the empty multiset. If $f : A \rightarrow B$ and $m \in \mathcal{MS}(A)$, we define $f(m) \in \mathcal{MS}(B)$ 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 $\mathcal{MS}(A)$ of multisets of A, given by $\{a_1, ..., a_n\} \sqsubseteq \{b_1, ..., b_m\}$ if there is an injective function $\iota : \{1, ..., n\} \rightarrow \{1, ..., m\}$ such that $a_i \leq b_{\iota(i)}$ for all i. If we do not demand ι to be injective we obtain the powerdomain order $\leq_{\exists}^{\forall}$. We write $\sqsubseteq_\iota$ and $\leq_\iota^{\forall}$ to stress the use of the mapping ι. It is well known [18] that the multiset order induced by a wqo is also a wqo.

[2] Different monotonicity notions are considered in [10].

3. ν-Petri Nets

In this section we present ν-PNs; the reader is referred[3] to [21] for more details. In ν-PNs names can be created, communicated and matched. We can use this mechanism to deal with authentication issues [19], correlation or instance isolation [6]. We formalize name management by replacing ordinary tokens by distinguishable tokens. 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.

As an example, the net in the top of Fig. 1 is a simple ν-PN with a single transition. When fired, it moves one token from p_1 to q_1 (because of variable x labelling both arcs), removes a token from p_2 (variable y does not appear in any outgoing arc) and a new name is created in q_2 (because of variable ν). Instead, the net in the bottom of Fig. 1 uses the same variable x to label the two arcs incoming its only transition. In that case, the transition must take two tokens carrying the same name from p_1 and p_2, so that the transition is not enabled.

Definition 3.1. A *ν-PN* is a tuple $N = (P, T, F)$, where P and T are finite disjoint sets, and

$$F : (P \times T) \cup (T \times P) \to \mathcal{MS}(Var)$$

is such that for every $t \in T$, $\Upsilon \cap pre(t) = \emptyset$ and $post(t) \setminus \Upsilon \subseteq pre(t)$, where $pre(t) = \bigcup_{p \in P} S(F(p,t))$ and $post(t) = \bigcup_{p \in P} S(F(t,p))$.

The set of pairs (x, y) such that $F(x, y) \neq \emptyset$ defines the set of arcs of N. We also take $Var(t) = pre(t) \cup post(t)$, $fVar(t) = Var(t) \cap \Upsilon$ and $nfVar(t) = Var(t) \setminus fVar(t)$. To avoid tedious definitions, along the paper we will consider a fixed ν-PN $N = (P, T, F)$.

Definition 3.2. A *marking* of N is a function $M : P \to \mathcal{MS}(Id)$. We denote by $Id(M)$ the set of names in M, that is, $Id(M) = \bigcup_{p \in P} S(M(p))$.

Like for other classes of higher-order nets, transitions are fired with respect to a mode, that chooses which tokens are taken from the preconditions and which are put in the postconditions. Given a transition t of N, a mode for t is an injection $\sigma : Var(t) \to Id$ that instantiates each variable to a different identifier. Thus, by using the same variable we force the equality of names taken from preconditions, and because modes are injections, we also check the inequality of names by using different variables. We will use $\sigma, \sigma', \sigma_1 \ldots$ to range over modes.

Definition 3.3. Let M be a marking, t a transition 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 $\nu \in fVar(t)$. The *marking reached after the firing of* t *with mode* σ is M', given by $M'(p) = (M(p) - \sigma(F(p,t))) + \sigma(F(t,p))$ for all $p \in P$.

In the definition of firing we demand that $\sigma(\nu) \notin Id(M)$, for every special variable ν, that is, that every such ν is instantiated to a different fresh name, not in the current marking. Moreover (and unlike in [21]) we demand modes to be injective, which amounts to being able to check for inequality of names

[3]We present here a more general version, that allows weights in arcs and check for inequality. The results in [19, 21, 20] can be easily transferred to this extended version.

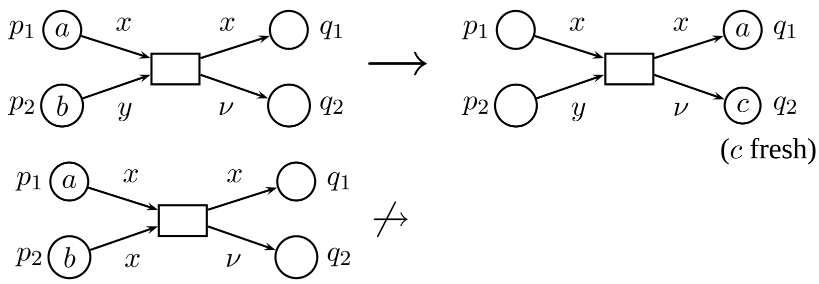

Figure 1. Two simple ν-PN

(not only for equality, by using the same variable in different arcs). We will write $M \xrightarrow{t} M'$, $M \xrightarrow{t(\sigma)} M'$, $M \rightarrow M'$ and $M \xrightarrow{\tau} M'$ with $\tau = t_1(\sigma_1) \cdots t_n(\sigma_n)$, saying that τ is a transition sequence, with their obvious meanings.

Let us now define the natural order between markings, that induces the coverability problem in ν-PN. We define $M_1 \sqsubseteq_\alpha M_2$ if there is an injection $\iota : Id(M_1) \rightarrow Id(M_2)$ such that $\iota(M_1(p)) \subseteq M_2(p)$, for all $p \in P$. We take $\equiv_\alpha$ as $\sqsubseteq_\alpha \cap {}_\alpha\sqsupseteq$ and identify markings up to $\equiv_\alpha$, that allows renaming of names. The relation $\sqsubseteq_\alpha$ is a wqo [19]. We will sometimes write $M_1 \sqsubseteq_\iota M_2$ to emphasize the use of ι.

4. Forward analysis for ν-PNs

The state space of a P/T net is the set $\mathbb{N}^k$. However, that set is not complete. For instance, the increasing chain $(n)_{n=1}^\infty$ does not have a least upper bound in $\mathbb{N}$. For that purpose, the classical Karp-Miller construction for P/T nets works instead with the domain $(\mathbb{N} \cup \{\omega\})^k$, which is the completion of $\mathbb{N}^k$. In particular, the least upper bound of the previous chain is just ω. In general, a generic Karp-Miller procedure needs to work with the completion of the domain of the WSTS, in case it is not already complete.

In this section we build the completion of the transition system defined by a ν-PN. In [11] it is proved that the ideal completion[4] of a poset is effective (ideals can be finitely represented, and inclusion is decidable) whenever the poset is built up from some basic data type constructions, among which are finite domains, with any order, and multisets of elements in a domain with effective ideal completion. Let us see that we can build our markings using these two constructions.

The behavior of ν-PNs is invariant under $\equiv_\alpha$ [19]. When working modulo $\equiv_\alpha$ we can represent markings as multisets of multisets of places, where each multiset represents the projection of the marking over some identifier. For instance, the marking M given by $M(p) = \{a\}$ and $M(q) = \{a, b\}$ can be equivalently represented by the multiset $\{\{p, q\}, \{q\}\}$ in $\mathcal{MS}(\mathcal{MS}(P))$. In general, for a marking M, its multiset representation is given by $\{M^a \mid a \in Id(M)\}$, where $M^a(p) = M(p)(a)$. We can also denote the previous multiset by the expression $pq + q$, where pq represents the identifier a, which is both in p and in q, and q represents the identifier b, which is only in q. In the following, $\sqsubseteq$ will denote the natural order over $\mathcal{MS}(\mathcal{MS}(P))$ (induced by the equality in P).

Lemma 4.1. Let M_1 and M_2 be two markings, and $\overline{M}_1$ and $\overline{M}_2$ their multiset representation. Then we have $M_1 \sqsubseteq_\alpha M_2$ iff $\overline{M}_1 \sqsubseteq \overline{M}_2$.

[4]Actually, the authors work with the equivalent concept of sobrification.

Proof:

$M_1 \sqsubseteq_\alpha M_2 \Leftrightarrow$
$\exists \iota : Id(M_1) \to Id(M_2)$ injective such that $\iota(M_1) \sqsubseteq M_2 \Leftrightarrow$
$\exists \iota : Id(M_1) \to Id(M_2)$ injective such that for all p, $\iota(M_1(p)) \subseteq M_2(p) \Leftrightarrow$
$\exists \iota : Id(M_1) \to Id(M_2)$ injective such that for all p and $b = \iota(a)$, $M_1^a(p) \leq M_2^b(p) \Leftrightarrow$
$\exists \iota : Id(M_1) \to Id(M_2)$ injective such that for all $b = \iota(a)$, $M_1^a \subseteq M_2^b \Leftrightarrow \overline{M}_1 \sqsubseteq \overline{M}_2$. □

In particular, $M_1 \equiv_\alpha M_2$ iff their multiset representations coincide. Since we are interested in the abstract treatment of pure names, our set of configurations will be just the set of finite multisets of finite multisets of places[5].

Next we define ω-markings, the analogous concept of the classical ω-markings of P/T nets in the case of ν-PN. We use a terminology inspired by the Simple Regular Expressions of [3]. We denote by $\mathbb{N}_\omega$ the set $\mathbb{N} \cup \{\omega\}$, and extend the natural order and the usual arithmetic to $\mathbb{N}_\omega$. Next we will consider a fixed enumeration of the places of the net, $P = \{p_1, ..., p_n\}$.

Definition 4.1. A *product* is an expression $p_1^{i_1} \cdots p_n^{i_n}$ with $i_1, ..., i_n \in \mathbb{N}_\omega$. A *sum* is an expression of the form $E_1 + ... + E_m$, where each E_i is a product. An ω*-marking* is an expression $\mathcal{A} + \infty(\mathcal{B})$, with $\mathcal{A}$ and $\mathcal{B}$ sums.

Intuitively, ω-markings are markings (modulo $\equiv_\alpha$) in which some identifiers may appear an unbounded number of times, and also an unbounded number of different identifiers may appear. Notice that each product corresponds to an ordinary ω-marking of a P/T net. For instance, the ω-marking $pq^\omega + \infty(p^\omega)$ represents the marking in which an identifier appears once in p and infinitely often in q, and infinitely many other different identifiers appear infinitely often in p. We will say a product $p_1^{i_1} \cdots p_n^{i_n}$ is bounded if $i_j \neq \omega$ for every j. Clearly, plain markings are a particular class of ω-markings, those in which $\mathcal{B}$ is the empty sum and all the products in $\mathcal{A}$ are bounded. Sometimes, for an ω-marking $\mathcal{M} = \mathcal{A} + \infty(\mathcal{B})$ we will refer to $\mathcal{A}$ as the bounded part of $\mathcal{M}$ and to $\mathcal{B}$ as the unbounded part of $\mathcal{M}$.

We denote by $\emptyset$ the empty sum, and we will simply write $\mathcal{A}$ instead of $\mathcal{A} + \infty(\emptyset)$ and $\infty(\mathcal{B})$ instead of $\emptyset + \infty(\mathcal{B})$. We will often omit places p with a null exponent, and expand exponential factors, writing for instance qq instead of $p^0 q^2$ (assuming $P = \{p, q\}$).

We define $|p_1^{i_1} \cdots p_n^{i_n}|_\omega = |\{k \mid i_k = \omega\}|$, and $(p_1^{i_1} \cdots p_n^{i_n})^\omega = p_1^{j_1} \cdots p_n^{j_n}$, where $j_k = 0$ if $i_k = 0$, and $j_k = \omega$ otherwise (e.g., $(ppq^\omega)^\omega = p^\omega q^\omega$). Given two products $E_1 = p_1^{i_1} \cdots p_n^{i_n}$ and $E_2 = p_1^{j_1} \cdots p_n^{j_n}$ we take $E_1 \sqsubseteq E_2 \Leftrightarrow i_k \leq j_k$ for all $k \in \{1, ..., n\}$, and we define $E_1 \oplus E_2 = p_1^{i_1+j_1} \cdots p_n^{i_n+j_n}$, and whenever $E_2 \sqsubseteq E_1$, $E_1 \ominus E_2 = p_1^{i_1-j_1} \cdots p_n^{i_n-j_n}$, provided $j_k \neq \omega$ for all $k \in \{1, ..., n\}$. Finally, $(\mathcal{A} + \infty(\mathcal{B})) + (\mathcal{A}' + \infty(\mathcal{B}'))$ is the ω-marking $(\mathcal{A} + \mathcal{A}') + \infty(\mathcal{B} + \mathcal{B}')$.

Let us now define the order between ω-markings, that extends the natural one for markings.

Definition 4.2. Given two ω-markings $\mathcal{M} = E_1 + ... + E_m + \infty(E_{m+1} + ... + E_k)$ and $\mathcal{M}' = E_1' + ... + E_{m'}' + \infty(E_{m'+1} + ... + E_{k'})$ we define $\mathcal{M} \sqsubseteq \mathcal{M}'$ if there is $\iota : \{1, ..., k\} \to \{1, ..., k'\}$ such that:

- If $\iota(i) = \iota(j)$ and $\iota(j) \leq m'$ then $i = j$ (it is partially injective),
- If $i > m$ then $\iota(i) > m'$,

[5] Notice that $\mathcal{MS}(P)$ is isomorphic to $\mathbb{N}^{|P|}$, so that alternatively we could have considered $\mathcal{MS}(\mathbb{N}^{|P|})$ instead of $\mathcal{MS}(\mathcal{MS}(P))$.

- $E_i \sqsubseteq E_{\iota(i)}$ for all $i \in \{1, ..., k\}$.

As for multisets, we use a mapping ι to specify which product of $\mathcal{M}'$ is used to bound each product in $\mathcal{M}$. Products in the bounded part of $\mathcal{M}$ can be mapped to products in the bounded or in the unbounded part of $\mathcal{M}'$, though products in the unbounded part of $\mathcal{M}$ can only be mapped to products that are also in the unbounded part of $\mathcal{M}'$. Intuitively, infinitely many copies of a product can only be bounded by an infinite number of products. Products in the bounded part of $\mathcal{M}'$ can only be used once to bound products in $\mathcal{M}$, while this is not the case for products in the unbounded part. Alternatively, we could have defined $\mathcal{A} + \infty(\mathcal{B}) \sqsubseteq \mathcal{A}' + \infty(\mathcal{B}')$ if we can split $\mathcal{A}$ into $\mathcal{A}_1$ and $\mathcal{A}_2$ so that[6] $\mathcal{A}_1 \sqsubseteq \mathcal{A}'$, $\mathcal{A}_2 \leq_{\exists}^{\forall} \mathcal{B}'$, and $\mathcal{B} \leq_{\exists}^{\forall} \mathcal{B}'$. The products in $\mathcal{A}_1$ are mapped to the bounded part, while the ones in $\mathcal{A}_2$ and in $\mathcal{B}$ are mapped to the unbounded part. Notice that, in this case, we are using the order $\leq_{\exists}^{\forall}$ since, intuitively, we have infinitely many copies of the products in $\mathcal{B}'$, so that we can choose any of them to bound as many sums as needed, so that the mapping needs not be injective. For instance, it holds $p + q + qq + \infty(q) \sqsubseteq pq + \infty(qq)$ because $p \sqsubseteq pq$, $q + qq \leq_{\exists}^{\forall} qq$ and $q \leq_{\exists}^{\forall} qq$.

We take $\equiv$ as $\sqsubseteq \cap \sqsupseteq$ and identify ω-markings up to $\equiv$. We take as ω-Markings the set of ω-markings identified up to $\equiv$. As for plain markings, we will also use the notation $\sqsubseteq_\iota$. When there is no confusion, we will write $\iota(E_i)$ instead of $E_{\iota(i)}$. For instance, $p+qq+\infty(q) \sqsubseteq_\iota p+\infty(qq)$ with $\iota(p) = p$, $\iota(qq) = qq$ and $\iota(q) = qq$. The following equivalences will be used along the rest of the paper.

Lemma 4.2. Let $E_1 \sqsubseteq E_2$.

- $E_1 + \infty(E_2) \equiv \infty(E_2)$
- $\infty(E_1 + E_2) \equiv \infty(E_2)$

Proof:

- Clearly $\infty(E_2) \sqsubseteq_\iota E_1 + \infty(E_2)$ with $\iota(E_2) = E_2$. Conversely, $E_1 + \infty(E_2) \sqsubseteq_\iota \infty(E_2)$ with $\iota(1) = \iota(2) = 2$. The first two conditions in Def. 4.2 are trivially satisfied because both 1 and 2 are mapped to the unbounded part. The other condition holds by hypothesis.
- Clearly, $\infty(E_2) \sqsubseteq \infty(E_1 + E_2)$. Conversely, $\iota(1) = \iota(2) = 2$ satisfies the three conditions. In particular, $E_1 \sqsubseteq E_{\iota(1)} = E_2$ by hypothesis, and trivially $E_2 \sqsubseteq E_{\iota(2)} = E_2$.

$\square$

Thus, for instance we have that $p+q+\infty(pq) \equiv \infty(pq) \equiv \infty(p+q+pq)$. Though ω-markings can be intuitively seen as markings in which some identifiers appear infinitely often, and in which an infinite number of different identifiers can appear, technically they represent sets of markings, those bounded by them as expressed by their denotations.

Definition 4.3. The *denotation of a product* $E = p_1^{k_1} \cdots p_n^{k_n}$ is the set of multisets of places $[\![E]\!] = \{A \in \mathcal{MS}(P) \mid A(p_i) \leq k_i \text{ for all } i = 1, ..., n\}$. The *denotation of a sum* $\mathcal{A} = \sum_{i=1}^{m} E_i$ is given by $[\![\mathcal{A}]\!] = \{\{A_i \mid A_i \in [\![E_i]\!], i \in I\} \mid I \subseteq \{1, ..., m\}\}$. We define the *denotation of an ω-marking* $\mathcal{M} = \mathcal{A} + \infty(\mathcal{B})$ as the set of markings $[\![\mathcal{M}]\!] = \{M + \sum_{i=1}^{k} M_i \mid k \geq 0, M \in [\![\mathcal{A}]\!], M_i \in [\![\mathcal{B}]\!]\}$.

[6] Abusing notation, we are considering sums to be multisets of products.

Take the ω-marking $pq + \infty(qq)$. The denotation of pq is the set $\{\emptyset, p, q, pq\}$, and $[\![qq]\!] = \{\emptyset, q, qq\}$. Thus, $[\![pq + \infty(qq)]\!]$ is the set of markings of the form $M + \underbrace{q + ... + q}_{n_1} + \underbrace{qq + ... + qq}_{n_2}$ with $n_1, n_2 \geq 0$ and $M \in [\![pq]\!]$. Notice that $[\![\mathcal{M}]\!]$ is a downward closed and directed set, that is, an ideal.

With these notations we can effectively represent the completion of $\mathcal{MS}(\mathcal{MS}(P))$, as proved next.

Proposition 4.1. The ideal completion of the poset $(\mathcal{MS}(\mathcal{MS}(P)), \sqsubseteq)$ can be effectively represented as (ω-Markings$, \sqsubseteq$).

Proof:
It is enough to consider that in our case ω-markings are a friendly presentation of the $\circledast$-products of $\circledast$-products in P, which according to [11, Theorem 5.3] is the ideal completion of $\mathcal{MS}(\mathcal{MS}(P))$. □

In particular, given two ω-markings $\mathcal{M}_1$ and $\mathcal{M}_2$ it holds that $\mathcal{M}_1 \sqsubseteq \mathcal{M}_2 \Leftrightarrow [\![\mathcal{M}_1]\!] \subseteq [\![\mathcal{M}_2]\!]$, so that ($\omega$-Markings,$\sqsubseteq$) is a continuous dcpo.

Now we need to lift the transition relation to the completed domain of ω-markings. More precisely, for each ω-marking $\mathcal{M}$ we need to effectively compute the set $\downarrow Post([\![\mathcal{M}]\!])$. First, let us introduce some notations: Given a transition t and a variable x, we will denote by $pre_t(x)$ the product $p_1^{i_1} \cdots p_n^{i_n}$, with $i_k = F(p_k, t)(x)$, and $post_t(x) = p_1^{i_1} \cdots p_n^{i_n}$, with $i_k = F(t, p_k)(x)$. In particular, the products $post_t(\nu)$, that correspond to the special variables $\nu \in \Upsilon$, are the "fresh" products created by the transition t. For instance, the net in the bottom of Fig. 1 satisfies $pre_t(x) = p_1 p_2$, $post_t(x) = q_1$ and $post_t(\nu) = q_2$.

Definition 4.4. Let $\mathcal{M} = E_1 + \cdots + E_m + \infty(E_{m+1} + \cdots + E_k)$ be an ω-marking, and t a transition. An *ω-mode* for t is any mapping $\sigma : nfVar(t) \to \mathbb{N}$ such that:

- If $\sigma(x) = \sigma(y)$ and $\sigma(y) \leq m$ then $x = y$, and
- $pre_t(x) \sqsubseteq E_{\sigma(x)}$ for all $x \in Var(t)$.

Then we write $\mathcal{M} \xrightarrow{t(\sigma)} \mathcal{A} + \infty(\mathcal{B})$, where $\mathcal{B} = E_{m+1} + \cdots + E_k$ and

$$\mathcal{A} = \sum_{x \in nfVar(t)} ((E_{\sigma(x)} \ominus pre_t(x)) \oplus post_t(x)) + \sum_{i \notin \sigma(Var(t))} E_i + \sum_{\nu \in fVar(t)} post_t(\nu)$$

We define $\overline{Post}_t(\mathcal{M}) = \{\mathcal{M}' \mid \mathcal{M} \xrightarrow{t(\sigma)} \mathcal{M}' \text{ for some } \sigma\}$, and extend it pointwise to sets of ω-markings.

We will write $\sigma(x) = E$ to denote that the product E is used by variable x in mode σ. For all $x \in Var(t)$, we will write $\nabla_t(x) = (\sigma(x) \ominus pre_t(x)) \oplus post_t(x)$. Notice that if $\nu \in fVar(t)$ then $\nabla_t(\nu)$ is simply $post_t(\nu)$. We will also write $\mathcal{M} \xrightarrow{t} \mathcal{M}'$, $\mathcal{M} \to \mathcal{M}'$, $\mathcal{M} \xrightarrow{\tau} \mathcal{M}'$ and $\mathcal{M} \to^* \mathcal{M}'$ as with plain markings, with their obvious meanings. Moreover, if the product E in $\mathcal{M}$ evolves to E' in $\mathcal{M}'$ we will also write $E \xrightarrow{t(\sigma)} E'$ or $E \to E'$. With these notations, $\sigma(x) \xrightarrow{t(\sigma)} \nabla_t(x)$ holds for each $x \in nfVar(t)$.

Whenever $\mathcal{M} \to \mathcal{M}'$, the unbounded part of $\mathcal{M}$ and $\mathcal{M}'$ coincide. However, new products may appear in the bounded part of $\mathcal{M}'$, like those in the unbounded part of $\mathcal{M}$ involved in the firing of the transition. For instance, the net in Fig. 8 can fire $p + \infty(q) \xrightarrow{t_2(\sigma)} p + qq + \infty(q)$ with $\sigma(x) = q$. Intuitively, one of the infinitely many names in q has been chosen, and put twice in q by the transition.

Let us see that we can compute a finite representation of $\downarrow Post_t([\![\mathcal{M}]\!])$. First, we rephrase the definition of firing of a transition when using the multiset representation (for more details see for instance [19]).

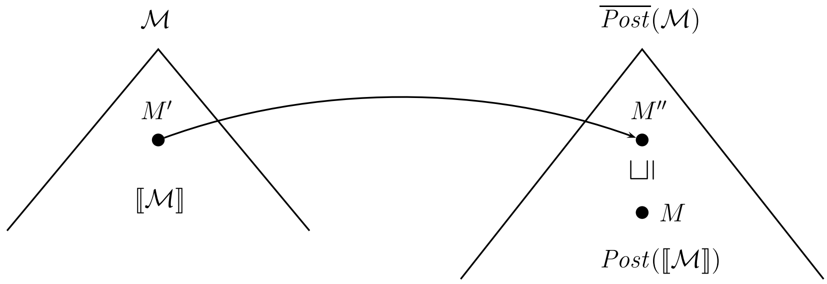

Figure 2. Computation of $Post([\![\mathcal{M}]\!])$

Definition 4.5. Given a marking $M = E_1 + ... + E_m$, a mode for t is an injection $\sigma : nfVar(t) \to \{1, ..., m\}$. Then t is enabled for M with mode σ if $pre_t(x) \sqsubseteq E_{\sigma(x)}$ for all $x \in Var(t)$. The reached marking after the firing of t with mode σ is

$$M' = \sum_{x \in nfVar(t)} ((E_{\sigma(x)} \ominus pre_t(x)) \oplus post_t(x)) + \sum_{\nu \in fVar(t)} post_t(\nu) + \sum_{i \notin \sigma(Var(t))} E_i$$

Let us now see that we can use $\overline{Post}_t(\mathcal{M})$ to compute $\downarrow Post_t([\![\mathcal{M}]\!])$. For that purpose, we need the following lemma. Moreover, we will denote just by $[\![\overline{Post}_t(\mathcal{M})]\!]$ the set $\bigcup_{\mathcal{M}' \in \overline{Post}_t(\mathcal{M})} [\![\mathcal{M}']\!]$.

Lemma 4.3. The following conditions hold:

- If $M \in [\![\mathcal{M}]\!]$ and $M \xrightarrow{t} M'$ then we have $M' \in [\![\overline{Post}_t(\mathcal{M})]\!]$.
- If $M \in [\![\overline{Post}_t(\mathcal{M})]\!]$ then there are $M' \in [\![\mathcal{M}]\!]$ and $M'' \in [\![\overline{Post}_t(\mathcal{M})]\!]$ such that $M \sqsubseteq M''$ and $M' \xrightarrow{t} M''$.

Proof:

- Let $\mathcal{M} = E_1 + ...E_m + \infty(E_{m+1} + ...E_k)$ and $M \in [\![\mathcal{M}]\!]$. By definition of denotation of an ω-marking,

  $$M = (A_1^0 + ... + A_m^0) + (A_{m+1}^1 + ... + A_k^1) + ... + (A_{m+1}^r + ... + A_k^r)$$

  with $A_i^j \in [\![E_i]\!]$ for all $(i, j) \in I = (\{1, ..., m\} \times \{0\}) \cup (\{m+1, ..., k\} \times \{1, ..., r\})$.

  By hypothesis, there is a mode $\sigma : nfVar(t) \to I$ (injective) such that $M \xrightarrow{t(\sigma)} M'$. By definition of firing, we have the following:

  - $pre_t(x) \sqsubseteq A_i^j$ if $\sigma(x) = (i, j)$,
  - $M' = \sum_{\substack{x \in nfVar(t) \\ \sigma(x)=(i,j)}} ((A_i^j \ominus pre_t(x)) \oplus post_t(x)) + \sum_{(i,j) \notin \sigma(nfVar(t))} A_i^j + \sum_{\nu \in fVar(t)} post_t(\nu)$.

  Let us define the ω-mode $\bar{\sigma} : nfVar(t) \to \{1, ..., k\}$ as $\bar{\sigma}(x) = i$ whenever $\sigma(x) = (i, j)$, which is such that $\mathcal{M} \xrightarrow{t(\bar{\sigma})} \mathcal{M}'$ and $M' \in [\![\mathcal{M}']\!]$, thus concluding the proof.

- We use $A \sqcup B$, when A and B are multisets, to denote the multiset given by $(A \sqcup B)(p) = max(A(p), B(p))$. Let $\mathcal{M} = \mathcal{A} + \infty(\mathcal{B})$ with $\mathcal{A} = E_1 + ... + E_m$ and $\mathcal{B} = E_{m+1} + ... + E_k$, and $\sigma : nfVar(t) \rightarrow \{1, ..., k\}$ an ω-mode for t such that $\mathcal{M} \xrightarrow{t(\sigma)} \mathcal{M}'$ and $M \in [\![\mathcal{M}']\!]$. Then, $\mathcal{M}' = \mathcal{A}' + \infty(\mathcal{B})$, where

$$\mathcal{A}' = \sum_{x \in nfVar(t)} ((E_{\sigma(x)} \ominus pre_t(x)) \oplus post_t(x)) + \sum_{i \notin \sigma(nfVar(t))} E_i + \sum_{\nu \in fVar(t)} post_t(\nu)$$

Since $M \in [\![\mathcal{M}']\!]$,

$$M = \sum_{x \in nfVar(t)} E'_x + \sum_{i \notin \sigma(nfVar(t))} E'_i + \sum_{\nu \in fVar(t)} E'_\nu + \sum_{i=1}^{l} M_i$$

where $E'_x \in [\![(E_{\sigma(x)} \ominus pre_t(x)) \oplus post_t(x)]\!]$ for every $x \in nfVar(t)$, $E'_i \in [\![E_i]\!]$ for every $i \notin \sigma(nfVar(t))$, $E'_\nu \in [\![post_t(\nu)]\!]$ for all $\nu \in fVar(t)$ and $M_i \in [\![\mathcal{B}]\!]$ for all $i \in \{1, ..., l\}$.

Let us define $E_x = E'_x \sqcup post_t(x)$ for all $x \in Var(t)$, and define

$$M'' = \sum_{x \in nfVar(t)} E_x + \sum_{i \notin \sigma(nfVar(t))} E'_i + \sum_{\nu \in fVar(t)} E_\nu + \sum M_i$$

The marking M'' satisfies $M \sqsubseteq M'' \in [\![\mathcal{M}']\!]$. Moreover, for M'' there is $M' \in [\![\mathcal{M}]\!]$ such that $M' \rightarrow M''$. It suffices to consider $E''_x = (E_x \ominus post_t(x)) \oplus pre_t(x)$ and to take

$$M' = \sum_{x \in nfVar(t)} E''_x + \sum_{i \notin \sigma(nfVar(t))} E'_i + \sum_{i=1}^{l} M_i$$

$\square$

The first part of the previous lemma states that $Post_t([\![\mathcal{M}]\!]) \subseteq [\![\overline{Post}_t(\mathcal{M})]\!])$. For a better insight of the second part, see Fig. 2. Both allow us to prove the following result.

Proposition 4.2. $\downarrow Post_t([\![\mathcal{M}]\!]) = [\![\overline{Post}_t(\mathcal{M})]\!]$

Proof:
Let us see that both inclusions hold:

$\subseteq$ Let $M \in\downarrow Post_t([\![\mathcal{M}]\!])$. Then there is $M' \in Post_t([\![\mathcal{M}]\!])$ such that $M \sqsubseteq M'$. That means that there is $M'' \in [\![\mathcal{M}]\!]$ such that $M'' \xrightarrow{t} M'$. By applying the first result of the previous lemma, $M' \in [\![\overline{Post}_t(\mathcal{M})]\!]$, so that by downward-closure we conclude that $M \in [\![\overline{Post}_t(\mathcal{M})]\!]$.

$\supseteq$ Let $M \in [\![\overline{Post}_t(\mathcal{M})]\!]$. By the second part of the previous lemma, there are $M' \in [\![\mathcal{M}]\!]$ and $M'' \in [\![\overline{Post}_t(\mathcal{M})]\!]$ such that $M \sqsubseteq M''$ and $M' \xrightarrow{t} M''$. Then $M'' \in Post_t([\![\mathcal{M}]\!])$, and therefore $M \in\downarrow Post_t([\![\mathcal{M}]\!])$.

$\square$

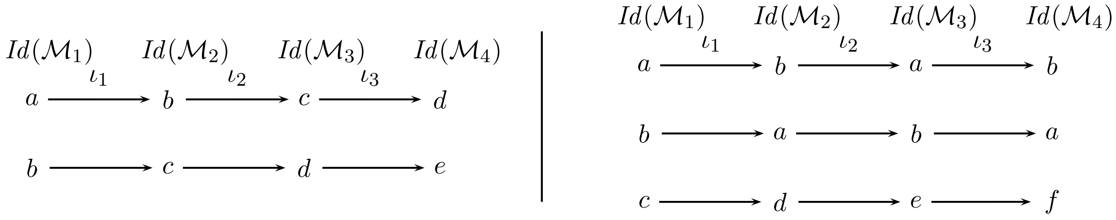

Figure 3. Example of construction of the sequences $(\iota_i)_{i=1}^{\infty}$

Corollary 4.1. The completion $\overline{N}$ of a ν-PN N is an effective complete WSTS.

Proof:
It is effective and complete by Prop. 4.1 and Prop. 4.2, so that it only remains to see that it is a WSTS. It is well known [2] that bqos are closed under the multiset construction, and that bqos are ω^2-wqos [16]. Therefore, the order $\sqsubseteq$ is an ω^2-wqo, and by Theorem 1 in [12], we obtain that $\overline{N}$ is a WSTS. □

For a complete WSTS, and a marking M, the *clover* [12] of M is defined by $Clover(M) = Max\ Lub(Cover(M))$. The clover of a state is finite because our order is well. It holds that $\downarrow Clover(M) = Lub(Cover(M))$, so that the clover is a finite basis of the cover (together with all the limits). Moreover, if $\overline{N}$ is the completion of $N = (X, \rightarrow, \leq)$ then $Cover_N(M) = Cover_{\overline{N}}(M) \cap X = \downarrow Clover_{\overline{N}}(M) \cap X$, so that the clover of the completion is a basis of the cover (once we remove the limits by intersecting with X).

Now let us see that we can apply a forward Karp-Miller algorithm to compute the clover of N (although, as we will see, it will not terminate in general). For that purpose, we will need to compute the least upper bounds of all the ω-markings produced in a loop, that is, we need to accelerate loops.

5. Accelerations

In the previous section we have mostly seen how ν-PNs fit in the general framework of [11, 12]. In the classic construction of the Karp-Miller tree for P/T nets, every time a transition sequence τ such that $M \xrightarrow{\tau} M'$ with $M(p) \leq M'(p)$ for all p and $M(q) < M'(q)$ for some q, we know that the transition sequence τ can be repeated arbitrarily often, so that the number of tokens in q can be considered to be unbounded. In other words, we can replace M' by the least upper bound of the markings obtained by repeating τ an arbitrary number of times.

In order to translate the Karp-Miller procedure to ν-PNs, we need to prove that the completion of a ν-PN is ∞-effective, meaning that we can compute the least upper bound of the markings obtained by repeating a transition sequence, that is, that we can accelerate loops. In the previous section we have shown how we can effectively represent the completed domains, so that the limit of an increasing chain (and more generally, of a directed set) always exists. However, the *double infiniteness* in ω-markings makes the task of computing those limits a non trivial one. We now specify what will it mean in our setting to repeat a transition sequence.

We will discuss the case in which τ is a single transition t, because the general case would only obscure the presentation. Later we will see how the general case can also be considered. Let us suppose that $\mathcal{M}_1 \xrightarrow{t(\sigma_1)} \mathcal{M}_2$ and $\mathcal{M}_1 \sqsubseteq_{\iota_1} \mathcal{M}_2$. Intuitively, because of monotonicity we can repeat the firing of t in

$\mathcal{M}_2$. However, the occurrence of a token a in p is bounded by the occurrence of $\iota_1(a)$ in p. Therefore, if t used a token a because $\sigma_1(x) = a$ for some variable x, then now t must use $\iota(a)$ instead, thus taking $\sigma_2(x) = \iota(a)$. We follow these intuitions in the next definition.

Definition 5.1. Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be ω-markings such that $\mathcal{M}_1 \xrightarrow{t(\sigma_1)} \mathcal{M}_2$ and $\mathcal{M}_1 \sqsubseteq_{\iota_1} \mathcal{M}_2$. We define the sequences $(\sigma_i)_{i=1}^{\infty}$, $(\mathcal{M}_i)_{i=1}^{\infty}$ and $(\iota_i)_{i=1}^{\infty}$ of ω-modes, ω-markings and mappings, respectively, as follows:

- $\sigma_{i+1}(x) = \iota_i(\sigma_i(x))$, for $i \geq 1$,
- $\mathcal{M}_i \xrightarrow{t(\sigma_i)} \mathcal{M}_{i+1}$ for $i \geq 1$, and
- $\iota_{i+1}(E) = \begin{cases} E' & \text{if } F' \xrightarrow{t(\sigma_i)} E \text{ and } \iota_i(F') \xrightarrow{t(\sigma_{i+1})} E' \text{ for } F' \text{ in } \mathcal{M}_i \\ & E \text{ and } E' \text{ in the bounded part,} \\ E & \text{otherwise} \end{cases}$ for $i \geq 1$.

σ_{i+1} is defined following the previous intuitions: if a variable x is first instantiated by a product E, in the next step it is instantiated by $\iota_i(E)$. $\mathcal{M}_{i+1}$ is simply obtained by letting $\mathcal{M}_i$ evolve with mode σ_i. The definition of the mappings ι_i requires further explanations. The mappings ι_i map products to products, but their definition is better understood by considering not the products themselves, but the identifier that each product represents. Consider the left handside of the diagram in Fig. 3, where a is mapped to b by ι_1, and b is mapped to a fresh identifier c. The definition of ι_2 above simply states that now (the product representing) b is mapped to (the product representing) c, because b was mapped to c by ι_1. Accordingly, since ι_1 mapped b to a fresh identifier (represented by a product $E = post_t(\nu)$ for some $\nu \in \Upsilon$), ι_2 must map c to another fresh identifier (which is represented by the same product $E = post_t(\nu)$). In the same way, if E is in the unbounded part of $\mathcal{M}_i$ then it is also in the unbounded part of $\mathcal{M}_{i+1}$, and $\iota_{i+1}(E) = E$.

We will denote by $t(\sigma_1)_\iota^k$ the sequence $t(\sigma_1) \cdots t(\sigma_k)$, where ι is as above. In general, for a transition sequence τ we can define as above the sequences of ω-modes, ω-markings and mappings. This is because we can always simulate the effect of the firing of a transition sequence using some given modes with the firing of a single transition.

Lemma 5.1. Let $\tau = t_1(\sigma_1)t_2(\sigma_2)$ be a transition sequence of a ν-PN $N = (P, T, F)$. Then there is a ν-PN $N' = (P, \{\bar{t}\}, F')$ such that $\mathcal{M}_1 \xrightarrow{\tau} \mathcal{M}_2$ in N iff $\mathcal{M}_1 \xrightarrow{\bar{t}(\sigma)} \mathcal{M}_2$ in N' for some mode σ of $\bar{t}$.

Proof:
We assume without loss of generality that $Var(t_1) \cap Var(t_2) = \emptyset$. Let $s : Var(t_2) \to Var(t_1) \cup Var(t_2)$ be defined as

$$s(y) = \begin{cases} x \text{ if } \sigma_1(x) = \sigma_2(y) \text{ for some } x \in Var(t_1) \\ y \text{ if } \sigma_1(x) \neq \sigma_2(y) \text{ for all } x \in Var(t_1) \text{ or } y \in \Upsilon \end{cases}$$

Then we define F' as

$$F'(p, \bar{t}) = F(p, t_1) + (s(F(p, t_2)) \dot{-} F(t_1, p))$$

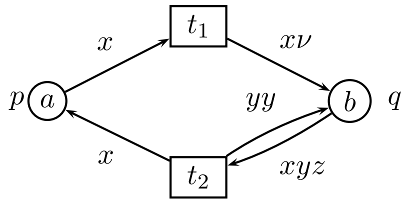

Figure 4. From transition sequences to transitions

$$F'(\bar{t},p) = s(F(t_2,p)) + (F(t_1,p) \dot{-} s(F(p,t_2)))$$

where $(A \dot{-} B)(a) = \begin{cases} A(a) - B(a) \text{ if } B(a) \leq A(a) \\ 0 \text{ otherwise} \end{cases}$

Then, for all $x \in Var(\bar{t})$, it is enough to take $\sigma(x) = \begin{cases} \sigma_1(x) \text{ if } x \in Var(t_1) \\ \sigma_2(x) \text{ otherwise} \end{cases}$ $\square$

Now we generalize the previous result to transition sequences of an arbitrary length.

Lemma 5.2. Let τ be a transition sequence of a ν-PN $N = (P,T,F)$. Then there is a ν-PN $N' = (P, \{\bar{t}\}, F')$ such that $\mathcal{M}_1 \xrightarrow{\tau} \mathcal{M}_2$ in N if and only if $\mathcal{M}_1 \xrightarrow{\bar{t}(\sigma)} \mathcal{M}_2$ in N' for some mode σ of $\bar{t}$.

Proof:
Let $\tau = t_1(\sigma_1) \cdots t_n(\sigma_n)$ with $n \geq 1$ be a transition sequence. We proceed by induction on n. If $n = 1$ there is nothing to prove. If $n > 1$, let $\tau' = t_1(\sigma_1) \cdots t_{n-1}(\sigma_{n-1})$. By induction there is $N'' = (P, t, F'')$ such that $\mathcal{M}_1 \xrightarrow{\tau'} \mathcal{M}_2$ in N if and only if $\mathcal{M}_1 \xrightarrow{t(\sigma')} \mathcal{M}_2$ for some σ'. By the previous lemma the thesis follows. $\square$

We will call the net N' given by the previous result a τ-*contraction* of N. We will also write τ_ι^k to denote the transition sequence $\bar{t}(\sigma)_\iota^k$, where $\bar{t}(\sigma)$ is the only transition of its τ-contraction and ι is as in Def. 5.1 . Consider for instance the net in Fig. 4 and the transition sequence $\tau = t_1(\sigma_1)t_2(\sigma_2)$, where $\sigma_1(x) = a$, $\sigma_1(\nu) = c$, $\sigma_2(x) = b$, $\sigma_2(y) = c$ and $\sigma_2(z) = a$. The τ-contraction of that net is the net N_2 depicted in Fig. 5. Notice that the modes σ_1 and σ_2 are such that $\sigma_1(x) = \sigma_2(z)$. Accordingly, since t_1 puts once $\sigma_1(x)$ in q, and t_2 removes $\sigma_2(z)$ from q, in N_2 the token a is neither put nor removed from q.

We are now ready to define in our setting what it means to accelerate a simple loop. The sequence $(\mathcal{M}_i)_{i=1}^{\infty}$ is an increasing sequence, so that the following definition makes sense.

Definition 5.2. Let $\overline{N}$ be the completion of a ν-PN N. We say $\overline{N}$ is *deterministically* ∞*-effective* if it is effective and whenever $\mathcal{M}_1 \xrightarrow{\tau} \mathcal{M}_2$ with $\mathcal{M}_1 \sqsubseteq_\iota \mathcal{M}_2$ we can compute

$$acc_\iota(\mathcal{M}_1 \xrightarrow{\tau} \mathcal{M}_2) = lub\{\mathcal{M} \mid \mathcal{M}_1 \xrightarrow{\tau_\iota^n} \mathcal{M},\ n > 0\}$$

Let us see that we can compute that least upper bound. In the first place, we can compute the τ-contraction of the net, and work with it instead. Therefore, we can always assume that we want to accelerate a single transition. Let us consider the nets N_1 and N_2 in Fig. 5. Notice that both nets can fire the run $p + q \xrightarrow{t} p + qq$, and $p + q \sqsubseteq_\iota p + qq$ with $\iota(p) = p$ and $\iota(q) = qq$. However, the result of an

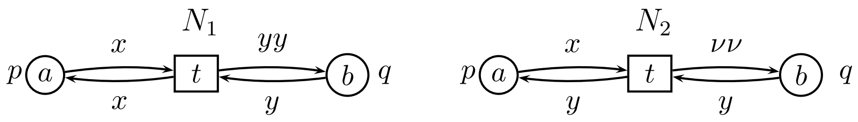

Figure 5. w-accelerations and d-accelerations

acceleration in both cases is very different: for N_1, every marking of the form $p+q^n$ is reachable; for N_2, every marking $p+qq+q+\cdots+q$ is reachable. Intuitively, the difference between both situations is that in N_1 each product is mapped to itself (the product p evolves to $\iota(p)=p$ and the product q evolves to $\iota(q)=qq$). However, that is not the case for N_2, where the product q evolves to $\iota(p)=p$. If we consider not products, but the identifiers they represent, then the difference becomes clearer. In N_1 both a and b are mapped to themselves by ι, while in N_2, a is mapped to b, and b is mapped to a fresh identifier. We formalize the behavior of N_1 in the following definition.

Definition 5.3. We say $\mathcal{M}_1 \xrightarrow{t(\sigma)} \mathcal{M}_2$ is *properly increasing* if $\mathcal{M}_1 \sqsubseteq_\iota \mathcal{M}_2$ and for all products E_2 in $\mathcal{M}_2$ there are no different products E_1 and E_1' in the bounded part of $\mathcal{M}_1$ such that $E_1 \xrightarrow{t(\sigma)} E_2$ and $\iota(E_1') = E_2$.

The firing $p+q \xrightarrow{t} p+qq$ is properly increasing in N_1, but not in N_2, because there is a product p in $p+qq$, and two different products in $p+q$, namely p and q, such that p is mapped to p by ι_1 and q evolves to p. However, every increasing firing can be unrolled into a properly increasing one. Indeed, consider again the diagrams in Fig. 3. In both parts of the diagrams, there is a natural k so that each identifier is mapped in k steps either to itself, or to a fresh identifier. In the left handside, after two steps, both a and b are mapped to fresh identifiers. In the right handside, after three steps, both a and b are mapped to themselves, but c is mapped to a fresh identifier. This happens in general, as we will see in the next lemma.

Lemma 5.3. If $\mathcal{M} \xrightarrow{t(\sigma)} \mathcal{M}'$ and $\mathcal{M} \sqsubseteq_\iota \mathcal{M}'$ then there is $k>0$ such that the firing of the $\bar{t}(\sigma)^k_\iota$-contraction of N is properly increasing.

Proof:
It is a direct consequence of the following fact. Let $\iota=\iota_0 : I_0 \to I_1$ be an injection (with I_0 and I_1 finite). Let us define the sequence of finite sets I_i for $i>0$ and the sequence of injections $\iota_i : I_i \to I_{i+1}$ for $i>0$, so that

$$\iota_{i+1}(a) = \begin{cases} \iota_i(a) & \text{if } a \in I_i \\ b \notin I_0 \cup \ldots \cup I_{i+1}, & \text{otherwise} \end{cases}$$

Then there is a number k such that that for all $i \in I_k$, either $(\iota_k \circ \cdots \circ \iota_1)(i) = i$ or $i \notin I_1$. Indeed, let $I_0 = (I_0 \cap I_1) \cup \{a_1, \ldots, a_m\}$ and $I_1 = (I_0 \cap I_1) \cup \{b_1, \ldots, b_m\}$. Let us consider the permutation $\rho_\iota : I_0 \to I_0$ such that $\rho_\iota(a) = \begin{cases} \iota(a) \text{ if } \iota(a) \in I_0 \\ a_i \text{ if } \iota(a) = b_i \end{cases}$. Since ρ_ι is a permutation it has an order $k = o(\rho_\iota)$, so that $\rho_\iota^k = id_{I_0}$. Then, k satisfies the conditions of the lemma. Moreover, k can be computed as the least common multiple of the lengths of the cycles of ρ_ι. □

We call order of ι, that we will denote as $o(\iota)$, the natural k given by the previous result, which can be effectively computed. Moreover, we will write $\bar{t}(\sigma)$ instead of $t(\sigma)^{o(\iota)}_\iota$, when it is clear from the context. Clearly, $acc_\iota(\mathcal{M}_1 \xrightarrow{t(\sigma)} \mathcal{M}_2) = acc_\iota(\mathcal{M}_1 \xrightarrow{t(\sigma)^k_\iota} \mathcal{M})$ for any $k > 0$ so that, in particular, we can take $k = o(\iota)$. Moreover, by Lemma 5.2 we can work with the $\bar{t}(\sigma)$-contraction of the net instead.

As an example, consider the nets in Fig. 5. In N_1 after one step each identifier can be mapped to itself, that is, $\mathcal{M}_1 \xrightarrow{t(\sigma)} \mathcal{M}_2$ with $\mathcal{M}_1 \sqsubseteq_\iota \mathcal{M}_2$, $\iota(a) = a$ and $\iota(b) = b$, so that $o(\iota) = 1$. In N_2 we find the situation in the left of Fig. 3, so that $o(\iota) = 2$. Thus, we need to consider the transition sequence $\tau = t(\sigma)t(\sigma')$, with $\sigma'(x) = b$ and $\sigma'(y) = \sigma(\nu)$. In turn, in order to compute the acceleration we can consider its τ-contraction, depicted in Fig. 6.

d-acceleration.

Using properly increasing sequences has the advantage that whenever a product E_x (with $\sigma(x) = E_x$) evolves to some E'_x in the range of ι, then necessarily $E_x \sqsubseteq E'_x$. Then, by repeating the firing of t we will obtain products of the form $E_x \oplus \Delta_t(x)^k$, for some *increment* $\Delta_t(x)$ (which will be formally defined later in all cases), with least upper bound $E_x \oplus \Delta_t(x)^\omega$. This is the situation for N_1 in Fig. 5 and $p + q \xrightarrow{t} p + qq$, that is properly increasing. Using the previous notations, $E_x = p$ and $E_y = q$, so that $\Delta_t(x) = \emptyset$ and $\Delta_t(y) = q$. Therefore, $acc_\iota(p + q \xrightarrow{t(\sigma)} p + qq) = p + q^\omega$.

w-acceleration.

However, in N_2 we cannot apply the previous acceleration. In this case the $\bar{t}(\sigma)$-contraction of N_2 is given by the net in Fig. 6. In it, every product is mapped to a fresh one, and every marking of the form $pq+qq+q+...+q$ is reachable. If we take $\Delta^\iota_t(x) = post_t(\nu_1) \ominus pre_t(x)$ and $\Delta^\iota_t(y) = post_t(\nu_2) \ominus pre_t(y)$ then $acc_\iota(p + q \xrightarrow{t(\sigma)} p + qq) = pq + qq + \infty(q) \equiv post_t(\nu_1) + post_t(\nu_2) + \infty(\Delta^\iota_t(x) + \Delta^\iota_t(y))$.

A simpler case in which a w-acceleration can be applied appears in the net in Fig. 8. The first firing that takes place is $p \xrightarrow{t_1} p + q$, so that $p \sqsubseteq_\iota p + q$ with $\iota(p) = p$. Notice that there is a fresh product, namely q, not in the range of ι, so that any marking of the form $p + q + ... + q$ is reachable, and $acc_\iota(p \xrightarrow{t_1} p + q) = p + \infty(q)$.

Following these intuitions, if $\mathcal{M}_1 \xrightarrow{t(\sigma)} \mathcal{M}_2$ is properly increasing we partition $nfVar(t)$ as follows:

$$
\begin{aligned}
V_{un} &= \{x \in nfVar(t) \mid \sigma(x) \text{ in the unbounded part of } \mathcal{M}_1\},\\
V_d &= \{x \in nfVar(t) \mid \sigma(x) \xrightarrow{t(\sigma)} \iota(\sigma(x))\},\\
V^\nu_w &= \{x \in nfVar(t) \mid \iota(\sigma(x)) = post_t(\nu_x) \text{ for } \nu_x \in fVar(t)\},\\
V^{un}_w &= \{x \in nfVar(t) \mid \iota(\sigma(x)) = \nabla_t(y_x) \text{ for some } y_x \in V_{un}\}.
\end{aligned}
$$

Moreover, we can define two injections: $h_\nu : V^\nu_w \to fVar(t)$ and $h_{un} : V^{un}_w \to V_{un}$ given by $h_\nu(x) = \nu_x$ and $h_{un}(x) = y_x$. Let us write $V^\nu_r = fVar(t) \setminus h_\nu(V^\nu_w)$, $V^{un}_r = V_{un} \setminus h_{un}(V^{un}_w)$, $V_w = V^\nu_w \cup V^{un}_w$, $V_b = V_d \cup V_w$ and $V_r = V^\nu_r \cup V^{un}_r$.

For all $x \in V_{un}$, $\sigma(x)$ is a product in the unbounded part. For all $x \in V_d$, the products $\sigma(x)$ are mapped to themselves by ι, so that $\nabla_t(x)$ will be used instead in the following firing of t. They will be responsible for d-accelerations. Products $\sigma(x)$ with $x \in V^\nu_w$ are those mapped by ι to fresh products. Therefore, $post_t(\nu_x)$ will be used instead in the next firing, so that it will leave some garbage that will

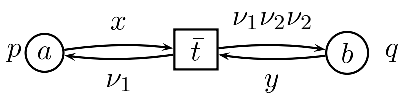

Figure 6. Contraction of the net N_2 in Fig. 5

cause a w-acceleration. Other products of the form $post_t(\nu)$ will not be used later, those with $\nu \in V_r^\nu$, so that they will also contribute to the w-acceleration.

Variables in V_w^{un} and V_r^{un} have an effect analogous to those in V_w^ν and V_r^ν. Products $\sigma(x)$ with $x \in V_w^{un}$ are mapped by ι to a product $\nabla_t(y_x)$ that has evolved from a product in the unbounded part. As before, $\nabla_t(y_x)$ will be used instead in the next firing, leaving again some garbage. Moreover, some products $\nabla_t(y)$ that come from a product in the unbounded part (those with $y \in V_r^{un}$) will also remain and contribute to the w-acceleration.

Definition 5.4. Let $\mathcal{M}_1 \stackrel{t(\sigma)}{\rightarrow} \mathcal{M}_2$ be a properly increasing sequence. We define the following products, that we will generically call *increments*:

- For all $x \in V_d$, $\Delta_t(x)$ is any product such that $\sigma(x) \stackrel{t(\sigma)}{\rightarrow} \sigma(x) \oplus \Delta_t(x)$,
- For all $x \in V_w^\nu$, $\Delta_t^\iota(x) = post_t(h_\nu(x)) \ominus pre_t(x)$,
- For all $x \in V_w^{un}$, $\Delta_t^\iota(x) = \nabla_t(h_{un}(x))) \ominus \sigma(x)$.

Finally, we are ready to prove the main result of this section, which determines which is the acceleration of a properly increasing firing $\mathcal{M}_1 \stackrel{t(\sigma)}{\rightarrow} \mathcal{M}_2$. Therefore, after this result we will be able to state the deterministically ∞-effectiveness of the completion of a ν-PN.

Proposition 5.1. If $\mathcal{M}_1 \stackrel{t(\sigma)}{\rightarrow} \mathcal{M}_2$ is properly increasing with $\mathcal{M}_1 \sqsubseteq_\iota \mathcal{M}_2$ then there is $\mathcal{M}$ such that $\mathcal{M}_1 \equiv \sum_{x \in V_b} \sigma(x) + \infty(\sum_{x \in V_{un}} \sigma(x)) + \mathcal{M}$, and $acc_\iota(\mathcal{M}_1 \stackrel{t(\sigma)}{\rightarrow} \mathcal{M}_2)$ is the ω-marking

$$\sum_{x \in V_d} \sigma(x) \oplus \Delta_t(x)^\omega + \infty\Big(\sum_{x \in V_w^\nu} post_t(x) \oplus \Delta_t^\iota(x) + \sum_{x \in V_w^{un}} \nabla_t(x) \oplus \Delta_t^\iota(x) + \sum_{x \in V_r} \nabla_t(x) + \sum_{x \in V_{un}} \sigma(x)\Big) + \mathcal{M}'$$

with $\mathcal{M}' = \mathcal{M} + \sum_{x \in V_w^\nu} (post_t(h_\nu(x)) + \sum_{x \in V_w^{un}} (\sigma(x) \oplus \Delta_t^\iota(x)) + \sum_{x \in V_w} \nabla_t(x)$. Moreover, the computation of the acceleration does not depend on the increments $\Delta_t(x)$ chosen.

Proof:
In the first place, $\mathcal{M}_1 = \sum_{x \in V_d} \sigma(x) + \sum_{x \in V_w} \sigma(x) + \infty(\sum_{x \in V_{un}} \sigma(x)) + \mathcal{M}$, for some ω-marking $\mathcal{M}$ that is not involved in the firing of $t(\sigma)$. Let us assume that $\mathcal{M}$ is the empty ω-marking; otherwise, we have to sum $\mathcal{M}$ to all the computed ω-markings.

Let us consider the following facts:

- $\sum_{x \in V_d} \nabla_t(x) \oplus k \cdot \Delta_t(x) = \sum_{x \in V_d} \sigma(x) \oplus (k+1) \cdot \Delta_t(x)$,

- $\sum_{x\in V_{un}} \nabla_t(x) = \sum_{x\in V_w^{un}} \nabla_t(h_{un}(x)) + \sum_{x\in V_r^{un}} \nabla_t(x) = \sum_{x\in V_w^{un}} \sigma(x)\oplus\Delta_t(x) + \sum_{x\in V_r^{un}} \nabla_t(x)$,
- $\sum_{\nu\in fVar(t)} \nabla_t(\nu) = \sum_{x\in V_w^{\nu}} \nabla_t(h_\nu(x)) + \sum_{\nu\in V_r^{\nu}} \nabla_t(\nu) = \sum_{x\in V_w^{\nu}} pre_t(x)\oplus\Delta_t(x) + \sum_{\nu\in V_r^{\nu}} \nabla_t(\nu)$.

Let us define $\mathcal{R}_k$ for $k \geq 2$ as follows:

$$\sum_{x\in V_w} \nabla_t(x) + (k-2)\cdot(\sum_{x\in V_w^{un}} \nabla_t(x)\oplus\Delta_t(x) + \sum_{x\in V_w^{\nu}} post_t(x)\oplus\Delta_t(x)) + (k-1)\cdot\sum_{x\in V_r}\nabla_t(x)$$

It holds that

$$\mathcal{R}_{k+1} = \mathcal{R}_k + \sum_{x\in V_w^{un}} \nabla_t(x)\oplus\Delta_t(x) + \sum_{x\in V_w^{\nu}} post_t(x)\oplus\Delta_t(x)) + \sum_{x\in V_r^{un}}\nabla_t(x) + \sum_{x\in V_r^{\nu}} post_t(x)$$

Let us prove that $\mathcal{M}_k$ is equivalent to

$$\sum_{x\in V_d}\sigma(x)\oplus(k-1)\cdot\Delta_t(x) + \sum_{x\in V_w^{un}}\sigma(x)\oplus\Delta_t(x) + \sum_{x\in V_w^{\nu}} pre_t(x)\oplus\Delta_t(x) + \infty(\sum_{x\in V_{un}}\sigma(x)) + \mathcal{R}_k$$

First we see it for $k = 2$. By definition of firing (see Def. 4.5)

$$\mathcal{M}_2 = \sum_{x\in V_d}\nabla_t(x) + \sum_{x\in V_w}\nabla_t(x) + \sum_{x\in V_{un}}\nabla_t(x) + \sum_{\nu\in fVar(t)} post_t(\nu) + \infty(\sum_{x\in V_{un}}\sigma(x))$$

By the previous comments,

$$\mathcal{M}_2 = \sum_{x\in V_d}\sigma(x)\oplus\Delta_t(x) + \sum_{x\in V_w^{un}}\sigma(x)\oplus\Delta_t(x) + \sum_{x\in V_w^{\nu}} pre_t(x)\oplus\Delta_t(x) + \infty(\sum_{x\in V_{un}}\sigma(x)) + \mathcal{R}_2$$

For $k > 2$, and assuming that $\mathcal{M}_{k-1}$ is equivalent to

$$\sum_{x\in V_d}\sigma(x)\oplus(k-2)\cdot\Delta_t(x) + \sum_{x\in V_w^{un}}\sigma(x)\oplus\Delta_t(x) + \sum_{x\in V_w^{\nu}} pre_t(x)\oplus\Delta_t(x) + \infty(\sum_{x\in V_{un}}\sigma(x)) + \mathcal{R}_{k-1}$$

it is enough to consider that it evolves to

$$\sum_{x\in V_d}\nabla_t(x)\oplus(k-2)\cdot\Delta_t(x) + \sum_{x\in V_w^{un}}\nabla_t(x)\oplus\Delta_t(x) + \sum_{x\in V_w^{\nu}} post_t(x)\oplus\Delta_t(x) + \infty(\sum_{x\in V_{un}}\sigma(x)) + \mathcal{R}_{k-1}$$

that, again, thanks to the previous comments, is equivalent to

$$\sum_{x\in V_d}\sigma(x)\oplus(k-1)\cdot\Delta_t(x) + \sum_{x\in V_w^{un}}\sigma(x)\oplus\Delta_t(x) + \sum_{x\in V_w^{\nu}} pre_t(x)\oplus\Delta_t(x) + \infty(\sum_{x\in V_{un}}\sigma(x)) + \mathcal{R}_k$$

To finish, it is enough to consider that the least upper bound of the set of ω-markings $\{\mathcal{M}_k \mid k \geq 2\}$ is indeed

Procedure `Clover`(M_0)
$\Theta \leftarrow \{M_0\}$
while $\overline{Post}(\Theta) \not\sqsubseteq \Theta$ **do**
  Choose fairly $\mathcal{M} \in \Theta$, τ and ι such that $\mathcal{M} \xrightarrow{\tau} \mathcal{M}'$
  if $\mathcal{M} \not\sqsubseteq_\iota \mathcal{M}'$ **then**
    $\Theta \leftarrow \Theta \cup \{\mathcal{M}'\}$
  else
    $\Theta \leftarrow \Theta \cup \{acc_\iota(\mathcal{M} \xrightarrow{\tau} \mathcal{M}')\}$
return $Max\ \Theta$

Procedure `width-Clover`(M_0)
$\Theta \leftarrow \{M_0\}$, $bounded$ $\leftarrow$**true**
while $\overline{Post}(\Theta) \not\sqsubseteq \Theta$ **and** $bounded$ **do**
  Choose fairly $\mathcal{M} \in \Theta$, τ and ι such that $\mathcal{M} \xrightarrow{\tau} \mathcal{M}'$
  if $\mathcal{M} \not\sqsubseteq_\iota \mathcal{M}'$ **then**
    $\Theta \leftarrow \Theta \cup \{\mathcal{M}'\}$
  else
    $\mathcal{M}' \leftarrow acc_\iota(\mathcal{M} \xrightarrow{\tau} \mathcal{M}')$
    if `x-bounded`$(\mathcal{M}')$ **then**
      $\Theta \leftarrow \Theta \cup \{\mathcal{M}'\}$
    else
      $bounded$ $\leftarrow$**false**
return $(bounded, Max\ \Theta)$

Figure 7. Karp-Miller procedure (left) and algorithm deciding width-boundedness (right)

$$\sum_{x\in V_d}(\sigma(x) \oplus \Delta_t(x)^\omega) + \sum_{x\in V_w^\nu}(post_t(h_\nu(x)) + \sum_{x\in V_w^{un}}(\sigma(x) \oplus \Delta_t^\iota(x)) + \sum_{x\in V_w}\nabla_t(x) +$$
$$\infty(\sum_{x\in V_w^\nu}(post_t(x) \oplus \Delta_t^\iota(x))) + \sum_{x\in V_w^{un}}(\nabla_t(x) \oplus \Delta_t^\iota(x)) + \sum_{x\in V_r}\nabla_t(x) + \sum_{x\in V_{un}}\sigma(x))$$

The last remark is a consequence of the fact that any two different increments $\Delta_t(x)$ and $\Delta_t'(x)$ coincide in the places where $\sigma'(x)$ is not ω. Therefore, $\sigma(x) \oplus \Delta_t(x)^\omega$ and $\sigma(x) \oplus \Delta_t'(x)^\omega$ coincide in every place. □

Corollary 5.1. The completion of a ν-PN is a deterministically ∞-effective (complete) WSTS.

Because it is deterministically ∞-effective, it makes sense to apply the procedure `Clover`(M_0) in the left of Fig. 7. Fairness in the choosing of the tuples $(\mathcal{M}, \tau, \iota)$ ensures that in every infinite run, every such tuple will eventually be chosen at a later stage. We know that the cover is effectively representable, so that there is a finite set of ω-markings Θ such that $\downarrow Post^*(\downarrow M_0) = \bigcup_{\mathcal{M}\in\Theta} [\![\mathcal{M}]\!]$.

Example 5.1. Let us see with detail how the algorithm behaves for the ν-PN N_2 in Fig. 5. The initial ω-marking is $\mathcal{M}_0 = p + q$, that is, $\Theta = \{\mathcal{M}_0\}$. The only possible mode that enables t is given by $\sigma_1(x) = p$ and $\sigma(y) = q$, which produces the ω-marking $p + qq$. Notice that:

- $p + q \sqsubseteq_{\iota_1} p + qq$, with $\iota_1(p) = p$ and $\iota_1(q) = qq$,
- the product p in $\mathcal{M}_0$ disappears,
- the product q in $\mathcal{M}_0$ evolves to p,
- the product qq in $p + qq$ is fresh.

Therefore, the firing is not properly increasing, because there is a product p in $p + qq$, and two different products in $\mathcal{M}_0$, namely p and q, such that p is mapped to p by ι_1 and q evolves to p. Actually, we are

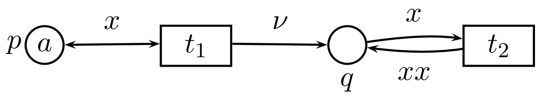

Figure 8. dw-accelerations

exactly in the situation of the diagram in the left of Fig. 3. Therefore, we need to unroll the transition sequence $t(\sigma_1)$, with contraction depicted in Fig. 6. There, the firing $p + q \to pq + qq$ can take place, which is properly increasing. Moreover, $V_d = V_{un} = V_w^{un} = \emptyset$ and $V_w^{\nu} = \{x, y\}$ with $h_\nu(x) = \nu_1$ and $h_\nu(y) = \nu_2$.

- $\Delta_{\bar{t}}(x) = post_{\bar{t}}(\nu_1) \ominus pre_{\bar{t}}(x) = q$,
- $\Delta_{\bar{t}}(y) = post_{\bar{t}}(\nu_2) \ominus pre_{\bar{t}}(y) = q$,
- $\nabla_{\bar{t}}(x) = (\sigma(x) \ominus pre_{\bar{t}}(x)) \oplus post_{\bar{t}}(x) = \emptyset$,
- $\nabla_{\bar{t}}(y) = (\sigma(y) \ominus pre_{\bar{t}}(y)) \oplus post_{\bar{t}}(y) = \emptyset$.

Then, according to Prop. 5.1, the acceleration is $pq + qq + \infty(q + q) \equiv pq + qq + \infty(q) = \mathcal{M}_1$, so that $\Theta = \{\mathcal{M}_0, \mathcal{M}_1\}$. Starting from $\mathcal{M}_1$ we could fire $t(\sigma)$ with $\sigma(x) = pq$ and $\sigma(y) = qq$, that produces again the ω-marking $\mathcal{M}_1$. We can also fire t from $\mathcal{M}_1$ with a different mode σ, with $\sigma(x) = pq$ and $\sigma(y) = q$, which yields the ω-marking $\mathcal{M}_2 = p+qq+qq+\infty(q)$. Since $\mathcal{M}_1 \not\sqsubseteq \mathcal{M}_2$ no acceleration is performed, and $\Theta = \{\mathcal{M}_0, \mathcal{M}_1, \mathcal{M}_2\}$.

Let us now see what happens if we fire the transition starting from $\mathcal{M}_2$. We could fire it using a mode such that $\sigma(x) = p$ and $\sigma(y) = qq$. The corresponding firing is increasing, but not properly increasing. As happened before, the order of the mapping ι is 2, and the contraction of the unrolling is given again by Fig. 6. The acceleration is analogous to the one obtained from $\mathcal{M}_0$, and produces again the ω-marking $\mathcal{M}_1$.

The other way in which t can be fired from $\mathcal{M}_2$ is more interesting, namely in a mode σ with $\sigma(x) = p$ and $\sigma(y) = q$. Notice that y is instantiated to a product in the unbounded part of $\mathcal{M}_2$. Using that mode, the firing $p + qq + \infty(q) \to p + qq + qq + \infty(q)$ can happen. Moreover, that firing is properly increasing. Indeed, $\iota(p) = p$, $\iota(qq) = qq$ (for both occurrences of qq) and $\iota(q) = q$ and, although the product $\iota(p) = p$ is the result of the evolution of a product different from p, namely q, that product is in the unbounded part. Now we have $V_d = V_w^{\nu} = V_r^{un} = \emptyset$, $V_{un} = \{y\}$, $V_w^{un} = \{x\}$ and $V_r^{\nu} = \{\nu\}$, with $h_{un}(x) = y$. Moreover, $\Delta_t^{\iota}(x) = \nabla_t(x) = \emptyset$, so that the acceleration is $(p \oplus \emptyset) + \emptyset + \infty(\emptyset \oplus \emptyset + qq + q) + qq + qq \equiv p + \infty(qq) = \mathcal{M}_3$.

Similarly, from $\mathcal{M}_3$ we can obtain the ω-marking $\mathcal{M}_4 = pq + \infty(qq)$, thus obtaining the set of ω-markings $\Theta = \{\mathcal{M}_0, \mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3, \mathcal{M}_4\}$. From Θ, no other ω-marking can be obtained. Thus, the algorithm returns the maximal ω-marking $\mathcal{M}_4$ (because $\mathcal{M}_i \sqsubseteq \mathcal{M}_4$ for all i), so that the cover is the set of markings $[\![pq + \infty(qq)]\!]$. In particular, every reachable marking M has one identifier a and a (possibly empty) set of identifiers $\{b_1, ..., b_m\}$ such that $M^a \subseteq \{p, q\}$ and $M^{b_i} \subseteq \{q, q\}$.

It is easy to see that the procedure $\texttt{Clover}(M_0)$ does not terminate in general. Consider the net in Fig. 8. First, $p \xrightarrow{t_1} p + q$ and we can apply a w-acceleration as previously explained, thus obtaining $p+\infty(q)$. Now we can fire transition t_2, $p+\infty(q) \xrightarrow{t_2} p+qq+\infty(q)$. Notice that $p+\infty(q) \sqsubseteq_\iota p+qq+\infty(q)$

with $\iota(p) = p$ and $\iota(q) = q$. The algorithm could then replace $p+qq+\infty(q)$ by its acceleration $p+\infty(qq)$. In the same way, all the ω-markings $p + \infty(q^n)$ are produced by the algorithm.

We could consider yet another type of acceleration, that we could call *dw-acceleration*. Instead of firing t_2 again using one of the infinitely many q's, we could fire it using qq. If we repeat this process, every marking $p + \sum_{i=1}^{m} q^i + \infty(q)$ becomes reachable, and their least upper bound is $p + \infty(q^w)$. We will later make more precise what we call a dw-acceleration.

It is true that dw-accelerations give a better approximation of the clover. However, they are not enough, neither any other acceleration we could imagine, since, in general, it is not possible to compute the clover. To prove it, we will reduce the property of depth-boundedness [20] to computability of the clover. Intuitively, a ν-PN is depth-bounded if every name in every reachable marking appears a bounded number of times. More precisely, N is depth-bounded if there is $n \in \mathbb{N}$ such that whenever E is a product of a reachable marking then $|E| \leq n$. In [23] we prove that depth-boundedness is undecidable by reducing boundedness of reset nets (which is undecidable [7]) to it.

Proposition 5.2. N is depth-bounded iff every product in every ω-marking of $Clover(N)$ is bounded.

Proof:
If N is depth-unbounded then for every $n \in \mathbb{N}$ there is M_n with a product E_n such that $|E_n| > n$. Since ν-PNs are finitely branching, by König's lemma we can assume that $M_n \rightarrow^* M_{n+1}$ and $E_n \rightarrow^* E_{n+1}$ for each $n \in \mathbb{N}$. Moreover, because of the wqo property, there are $i < j$ such that $M_i \sqsubset_\iota M_j$ and $E_i \sqsubset E_j$. By Lemma 5.3 there is a properly increasing unrolling of $M_i \rightarrow^* M_j$, which satisfies $\iota(E_i) = E_j$. By Prop. 5.1, $acc_\iota(M_i \rightarrow^* M_j)$ contains a product of the form $\sigma(x) \oplus \Delta(x)^\omega$ with $\Delta(x) = E_j \ominus E_i \neq \emptyset$. Since $acc_\iota(M_i \rightarrow^* M_j) \sqsubseteq \mathcal{M}$ for some $\mathcal{M} \in Clover(N)$ we conclude. The converse is a consequence of the definition of the clover of a net. $\square$

Thus, just by inspection of the clover of a net, we can decide whether it is depth-bounded.

Corollary 5.2. There is a ν-PN for which the clover is not computable.

In particular, since Petri Data Nets [17] subsume ν-PN, there is no procedure computing the cover of a Petri Data Net, neither for a Transfer Data Net, thus answering negatively to a question posed in [12].

6. dw-bounded nets

We have proved that the cover is not computable in general. In this section we identify a subclass of ν-PN for which the cover is computable, the class of dw-bounded nets. A net is dw-bounded whenever only a bounded number of products can grow arbitrarily. This boundedness property is weaker than the property of depth-boundedness we considered before, that is, if a net is depth-bounded then it is also dw-bounded. Moreover, we will see that, as depth-boundedness, dw-boundedness is undecidable.

We will also consider another property related to boundedness, called width-boundedness in [20]. Intuitively, a net is width-bounded if only a bounded number of different names can appear in every reachable marking. It is also the case that width-bounded nets are also dw-bounded. Moreover, we will see that, unlike for depth-boundedness, we can use the generic Karp-Miller procedure (or a slight variation) to decide width-boundedness.

Definition 6.1. N is *width-bounded* if there is $n \in \mathbb{N}$ such that for all reachable M, $|Id(M)| \leq n$.

Let us characterize width-boundedness in terms of the ω-markings in the clover.

Proposition 6.1. A ν-PN N is width-bounded iff every $\mathcal{M} \in Clover(N)$ if of the form $\mathcal{A} + \infty(\emptyset)$.

Proof:
Suppose that every ω-marking in $Clover(N)$ is of the form $\mathcal{M} = \mathcal{A} + \infty(\emptyset)$. Let n be the maximum number of products in ω-markings in $Clover(N)$. Then for every reachable marking M, $|Id(M)| \leq n$ and the net is width-bounded.

Conversely, suppose there is an ω-marking $\mathcal{M}_c = \mathcal{A} + \infty(\mathcal{B})$ and $E \neq \emptyset$ is a product of $\mathcal{B}$. For each $n \in \mathbb{N}$ let $\mathcal{M}_n = \sum_{i=1}^{n} E$, which satisfies $\mathcal{M}_n \sqsubseteq \mathcal{M}_c$. By definition of cover, for each $n \in \mathbb{N}$ there is a reachable marking M_n with n products, so that the net is width-unbounded. □

Let us see that the forward analysis, though non-terminating in general, can decide width-boundedness. Let us define the predicate over ω-markings `width-bounded`, given by `width-bounded`($\mathcal{M}$) iff $\mathcal{M} = \mathcal{A} + \infty(\emptyset)$. To detect width-boundedness it is enough to stop whenever an ω-marking $\mathcal{M}$ such that $\neg$`width-bounded`($\mathcal{M}$) is found. In this way we can slightly modify the procedure `Clover`, obtaining the algorithm in the right of Fig. 7, `width-Clover`(M_0). The modified algorithm always terminates, returning true iff the net is width-bounded, in which case the clover is computed.

Proposition 6.2. Width-boundedness is decidable for ν-PN. Moreover, if N is a width-bounded ν-PN then its clover is computable.

Proof:
Let us see that `width-Clover`(M_0) terminates and returns true iff the net is width-bounded. If N is width-unbounded, since the choice is fair, it will eventually find an ω-marking $\mathcal{M}$, a transition sequence τ and a mapping ι such that $\neg$`width-bounded`($acc_\iota(\mathcal{M} \xrightarrow{\tau} \mathcal{M}')$), and the algorithm will halt returning false, which is correct by the previous result. If N is width-bounded, let us see that it halts returning true. Let us suppose that it runs forever, building an increasing set Θ of ω-markings, so that `width-bounded`($\mathcal{M}$), for all $\mathcal{M} \in \Theta$. By the wqo property, there is an infinite sequence $(\mathcal{M}_i) \subseteq \Theta$ such that $\mathcal{M}_{i+1} = acc(\mathcal{M}_i \xrightarrow{\tau_i} \mathcal{M}'_i)$. Since we have `width-bounded`($\mathcal{M}_{i+1}$), the corresponding acceleration is only a d-acceleration. By applying the notations in Prop. 5.1, we have that $V_w = V_r = V_{un} = \emptyset$. Therefore, there is an m such that $\mathcal{M}_i = E_1^i + ... + E_m^i$ for all $\mathcal{M}_i \in \Theta$. Moreover, we can assume that $\mathcal{M}_i \sqsubset \mathcal{M}_{i+1}$ or, again by fairness, we would eventually have $\overline{Post}(\Theta) \sqsubseteq \Theta$, thus halting. Then, for each $i \in \mathbb{N}$, if $\mathcal{M}_i = E_1^i + ... + E_m^i$ and $\mathcal{M}_{i+1} = E_1^{i+1} + ... + E_m^{i+1}$, there are j and k such that $\iota(E_j^i) = E_k^{i+1}$ and $E_j^i \sqsubset E_k^{i+1}$. Since $\mathcal{M}_{i+1} = acc(\mathcal{M}_i \xrightarrow{\tau_i} \mathcal{M}'_i)$ and because of Prop. 5.1, if $E_j^i = p_1^{e_1} ... p_{|P|}^{e_{|P|}}$ and $E_k^{i+1} = p_1^{e'_1} ... p_{|P|}^{e'_{|P|}}$ then there is an l such that $e_l \neq \omega$ and $e'_l = \omega$. Then we can build the strictly increasing sequence of naturals $k_i = \sum_{j=1}^{m} |E_j^i|_\omega$ which should be bounded by $m \cdot |P|$, thus reaching a contradiction. To conclude, it is enough to consider that both algorithms in Fig. 7 coincide for width-bounded nets. □

Let us now define dw-boundedness. The following definition is given using the multiset representation of markings. Therefore, the products E_i in the next definition are all bounded.

Definition 6.2. N is *dw-bounded* if there are $b, n \in \mathbb{N}$ such that for every marking M reachable, $M \equiv E_1 + E_2 + ... + E_n + E_{n+1} + ... + E_m$, such that for $i > n$, $E_i = p_1^{i_1} \cdots p_l^{i_l}$ with $i_j \leq b$ for $j = 1...l$.

Intuitively, a ν-PN is dw-bounded if there is a bounded amount of names that can appear as tokens an arbitrary number of times in every reachable marking. Next we characterize dw-boundedness in terms of ω-markings. First we prove the following lemma, that states that from an increasing sequence of ω-markings we can always build a properly increasing one.

Lemma 6.1. Let $(\mathcal{M}_i)_{i=1}^{\infty}$ be an increasing sequence of ω-markings such that $\mathcal{M}_i \rightarrow^* \mathcal{M}_{i+1}$ for all $i \in \mathbb{N}$. Then there is a properly increasing sequence $(\mathcal{M}'_i)_{i=1}^{\infty}$ such that $\mathcal{M}'_i \rightarrow^* \mathcal{M}'_{i+1}$ and $\mathcal{M}_i \sqsubseteq \mathcal{M}'_i$ for all $i \in \mathbb{N}$.

Proof:
We build such a properly increasing sequence by applying the following inductive rules:

- $\mathcal{M}'_1 = \mathcal{M}_1$.
- For all $i \geq 1$, if $\mathcal{M}_i \xrightarrow{\tau_i} \mathcal{M}_{i+1}$, then if $t_i(\sigma_i)$ is the properly increasing contraction obtained by Lemma 5.3 from the contraction of τ_i, then $\mathcal{M}'_{i+1}$ is the only marking $\mathcal{M}$ such that $\mathcal{M}'_i \xrightarrow{t_i(\sigma_i)} \mathcal{M}$.

To prove that this construction is correct it is enough to prove that $\mathcal{M}_i \sqsubseteq \mathcal{M}'_i$ for all $i \in \mathbb{N}$, because in that case, $t_i(\sigma_i)$ would be fireable from $\mathcal{M}_i$. Let us prove it by induction: Obviously $\mathcal{M}_1 \sqsubseteq \mathcal{M}'_1$. Now suppose that $\mathcal{M}_{i-1} \sqsubseteq \mathcal{M}'_{i-1}$. Then, as $\mathcal{M}_{i-1} \xrightarrow{t_i(\sigma_i)} \mathcal{M}$, where $\mathcal{M}_i \sqsubseteq \mathcal{M}$, by monotonicity $t_i(\sigma_i)$ is fireable from $\mathcal{M}'_{i-1}$, reaching a marking $\mathcal{M}'_i$, such that $\mathcal{M}_i \sqsubseteq \mathcal{M} \sqsubseteq \mathcal{M}'_i$. □

Proposition 6.3. N is dw-bounded if and only if every ω-marking $\mathcal{M} = \mathcal{A} + \infty(\mathcal{B})$ in $Clover(N)$ is such that all the products in $\mathcal{B}$ are bounded.

Proof:
Suppose N is dw-bounded and let n and b be the bounds of the definition of dw-boundedness for N. Now suppose that one of the ω-markings in its clover is of the form $\mathcal{M} = \mathcal{A} + \infty(\mathcal{B})$, with $\mathcal{B} = E_1 + ... + E_n$ and there is an $i \in \{1...n\}$ such that $E_i = p_1^{i_1} \cdots p_l^{i_l}$ with $i_k = \omega$ for some $k \in \{1...l\}$. As E_i is a product in the sum $\mathcal{B}$, for each $j \in \mathbb{N}$, we can fire a sequence of transitions of N from its initial marking reaching a certain marking $\mathcal{M}_j$ such that there exists at least j products such that, for each of these products, there is at least j tokens of the corresponding name in a place. Therefore, if we consider $m \in \mathbb{N}$ such that $m > n$ and $m > b$, then we have a reachable marking with m products with at least m tokens in one place for each of these products, which is a contradiction.

Conversely, let us suppose that N is dw-unbounded. Then, there is a sequence of reachable markings $\mathcal{M}_1, \mathcal{M}_2, ...$ such that for each $i \in \mathbb{N}$, $\mathcal{M}_i = E_1 + E_2 + ... + E_i + E_{i+1} + ... + E_f$, where for all $j \leq i$, E_j has more than i tokens. Because of the wqo property and König's lemma, we can assume that for all $i \in \mathbb{N}$, $\mathcal{M}_i \sqsubseteq \mathcal{M}_{i+1}$ and $\mathcal{M}_i \rightarrow^* \mathcal{M}_{i+1}$. Now we can apply Lemma 6.1 to this sequence, so we can suppose that $(\mathcal{M}_i)$ is a properly increasing sequence. Suppose that the number of products in $\mathcal{M}_1$ is n and $i \geq 2n + 1$. Then, the number of products in $\mathcal{M}_i$ with at least $2n + 1$ tokens in a place is greater than $2n$ (if there are exactly $2n$ products, then it could be the case that every product in $\mathcal{M}_1$ is mapped to a fresh product). Let us consider $t_i(\sigma_i)$ the contraction of the sequence of firings $\mathcal{M}_1 \rightarrow^* \mathcal{M}_i$, which is properly increasing. By applying the notations in Prop. 5.1, we have that $V_r^\nu \neq \emptyset$ because there is at least one fresh product in $\mathcal{M}_i$ which is not in $h_\nu(V_w^\nu)$. Therefore, $V_r \neq \emptyset$ and because of Prop. 5.1, when we accelerate the transition $t_i(\sigma_i)$, we obtain an ω-marking with a product in the unbounded part, which has at least i tokens in a place. Therefore, if we consider the previous acceleration for all $i' \geq i$, we obtain a sequence of ω markings $\mathcal{M}'_1, ..., \mathcal{M}'_p, ...$ such that for each $j \in \mathbb{N}$, $\mathcal{M}'_j$ has a product in its unbounded part with more than i tokens in a place. Therefore, the clover must contain an ω-marking with an unbounded product in its unbounded part, and we conclude. □

Next, we use the previous result to prove the computability of the clover for dw-bounded ν-PN and finally, the undecidability of dw-boundedness for ν-PN.

Proposition 6.4. The clover is computable for dw-bounded ν-PN.

Proof:
Let N be dw-bounded, and let us suppose that the procedure $\mathtt{Clover}(M_0)$ does not terminate for N. Then there is an increasing set Θ of ω-markings of N such that $\overline{Post}(\Theta) \not\sqsubseteq \Theta$. Since N is dw-bounded, the unbounded part of every ω-marking in Θ is below some sum $\mathcal{B}$. By the wqo property, there is an infinite sequence of ω-markings in Θ, $\mathcal{M}_1, \mathcal{M}_2, ...$, such that $\mathcal{M}_{i+1} = acc(\mathcal{M}_i \xrightarrow{\tau_i} \mathcal{M}'_i)$. Moreover, we can take this sequence so that from some point on all the ω-markings are of the form $\mathcal{M}_i = \mathcal{A}_i + \infty(\mathcal{B})$. We assume for the sake of readability that this property holds for each i (otherwise, we take the corresponding subsequence). Moreover, we can assume that each product in $\mathcal{A}_i$ is not smaller or equal than any product in $\mathcal{B}$ (otherwise, they could be removed by Lemma 4.2). In that case, since $\mathcal{A}_i + \infty(\mathcal{B}) \sqsubset \mathcal{A}_{i+1} + \infty(\mathcal{B})$ holds, we also have $\mathcal{A}_i \sqsubset \mathcal{A}_{i+1}$ for all i. Then, one of the following facts must hold:

- The number of products in $\mathcal{A}_i$ grows arbitrarily. For each $i \in \mathbb{N}$, let (M_j^i) be a sequence of markings with $\mathcal{A}_i = lub\{M_j^i \mid j \geq 1\}$. Since $M_j^i \sqsubseteq \mathcal{M}_i$ and $\mathcal{M}_i \in \Theta$, by definition of cover, there is a transition sequence τ_j^i and a marking $\overline{M}_j^i \in [\![\mathcal{M}_1]\!]$ such that $\overline{M}_j^i \xrightarrow{\tau_j^i} M_j^i$. By monotonicity, $\mathcal{M}_1 \xrightarrow{\tau_j^i} \mathcal{M}'^i_j$ with $M_j^i \sqsubseteq \mathcal{M}'^i_j$. Let $\overline{t}_j^i$ be the transition of the τ_j^i-contraction of N. Since the number of products grows arbitrarily, there is $i \in \mathbb{N}$ such that for any $\iota : Id(\mathcal{M}_1) \to Id(M_j^i)$, some variable $\nu_j \in fVar(\overline{t}_j^i)$ is not in the range of the mapping h_ν induced by ι. For each j we fix some ι_j and consider the properly increasing sequence given by Lemma 5.3. By construction, $\nu_j \in V_r$ and $E_j = \nabla(\nu_j) = post(\nu_j)$ is a product not less or equal than any product in $\mathcal{B}$. By Prop. 5.1 the corresponding acceleration adds E_j to the unbounded part. By fairness for any j the acceleration that adds E_j to the unbounded part will eventually be performed by the algorithm. Since E_j converges to a product not below $\mathcal{B}$ (because the number of products is now fixed) we reach a contradiction with the fact that the unbounded part of every ω-marking obtained by $\mathtt{Clover}(M_0)$ is below or equal to $\mathcal{B}$.

- For all $j > i$ there is a $k > j$ such that $\mathcal{A}_{k-1} = E_1 + ... + E_n$ and $\mathcal{A}_k = E'_1 + ... + E'_n$, with $E_l \sqsubseteq E'_l$ for $l < n$ and there are m, t such that $E_m = p_1^{e_1} \dots p_f^{e_f}$, $E'_m = p_1^{e'_1} \dots p_f^{e'_f}$, $t < f$, $e_t \neq \omega$ and $e'_t = \omega$. Therefore we have an increasing amount of exponents in the products of $\mathcal{A}$ set to ω so we have an increasing unbounded sum of products in $\mathcal{A}$, which is not possible, as we saw in the proof of Prop. 6.2.

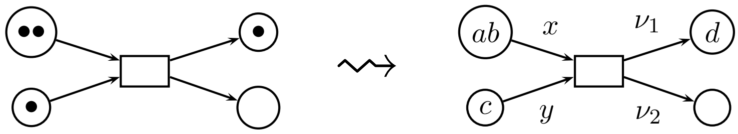

Figure 9. From P/T nets to depth-bounded ν-PN

Therefore, we reach a contradiction in both cases, so that the procedure $\texttt{Clover}(M_0)$ must terminate. □

As a consequence of the previous proposition we have the following result:

Proposition 6.5. dw-boundedness is undecidable for ν-PN.

Proof:
We reduce depth-boundedness, which is undecidable [23], to dw-boundedness. Let N be a ν-PN and let us suppose we can decide if N is dw-bounded. If N is dw-bounded then by the previous result we can compute its clover, which gives us the answer to the depth-boundedness problem (Prop. 5.2). On the contrary, if N is dw-unbounded then it is also depth-unbounded. □

Summing up, the cover is computable for width, depth and dw-bounded nets, though only the first property can be decided. Moreover, as one could expect, its computation has non primitive recursive complexity.

Proposition 6.6. The computation of the cover has non primitive recursive complexity for width-, depth- and dw-bounded ν-PN.

Proof:
P/T nets are in particular width-bounded ν-PN, so that the complexity of the decision procedure for width-boundedness and for the computation of the cover for width-bounded ν-PN is non primitive recursive [8]. Since width-bounded ν-PN are dw-bounded, so is for dw-bounded nets. The case of depth-bounded ν-PN is not so straightforward, since P/T nets are not depth-bounded ν-PN in general (only if they are bounded). However, given a P/T net N we can build in polynomial time a depth-bounded ν-PN N', so that the cover of N can be computed in polynomial time from the cover of N'. Let $N = (P, T, F, M_0)$ be a P/T net, and let $x_p \in Var \setminus \Upsilon$ and $\nu_p \in \Upsilon$ be different variables for each $p \in P$ (that is, $x_p \neq x_q$ and $\nu_p \neq \nu_q$ for $p \neq q$). We build $N' = (P, T, F', M_0')$ (see Fig. 9), where $F'(p,t) = \{x_p\}$ if $(p,t) \in F$, and $F'(p,t) = \emptyset$, otherwise; $F'(t,p) = \{\nu_p\}$ if $(t,p) \in F$, and $F'(t,p) = \emptyset$, otherwise. For the initial marking, we build M_0' by replacing each token in M_0 by a different identifier. Notice that $|M_0^a| = 1$ for each $a \in Id(M_0)$. Moreover, by construction this invariant holds in every reachable marking. Therefore, N' is depth-bounded. Let $Clover(N') = \{\mathcal{M}_1, ..., \mathcal{M}_k\}$. By construction of N', each $\mathcal{M}_i$ is of the form $p_1^i + ... + p_{k_i}^i + \infty(q_1^i + ... + q_{l_i}^i)$. Then, $Clover(N) = \{M_1, ..., M_k\}$, where $M_i = p_1^i ... p_{k_i}^i (q_1^i ... q_{l_i}^i)^w$. □

dw-accelerations and non-determinism.

Let us now make precise what we meant in the previous section by dw-accelerations.

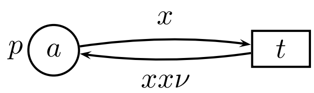

Figure 10. Accelerations and non-determinism

Proposition 6.7. Let $\mathcal{M}_0$ and $\mathcal{M}_1$ be two sums. If $acc(\mathcal{M}_0 \xrightarrow{t} \mathcal{M}_1) = \mathcal{A} + \infty(\mathcal{B})$ and $\mathcal{A}' + \infty(M_0) \sqsubseteq \mathcal{M}$ for some $\mathcal{M} \in Clover(N)$ then there exists $\mathcal{M}' \in Clover(N)$ such that $\mathcal{A}' + \infty(\mathcal{A} + \mathcal{B}) \sqsubseteq \mathcal{M}'$

Proof:
Let $\mathcal{M}_k$ be as defined in Def. 5.1 starting from $\mathcal{M}_0$. Then, $lub\,\{\mathcal{M}_k\} = \mathcal{A} + \infty(\mathcal{B})$. Let us show that firing t repeatedly from $\mathcal{A}' + \infty(\mathcal{M}_0)$, we can reach a set of markings $\{\mathcal{M}'_i \mid i \in \mathbb{N}\}$ with lub equal to $\infty(\mathcal{A} + \mathcal{B})$. If we fire t from $\mathcal{A}' + \infty(\mathcal{M}_0)$ then we reach the ω-marking $\mathcal{M}'_1 = \mathcal{A}' + \mathcal{M}_1 + \infty(\mathcal{M}_0)$. Then, if we fire t two times from this new ω-marking we can reach $\mathcal{M}'_2 = \mathcal{A}' + \mathcal{M}_2 + \mathcal{M}_1 + \infty(\mathcal{M}_0)$. By repeating this procedure, we reach a sequence of ω-markings $\mathcal{M}'_i = \mathcal{A}' + \mathcal{M}_i + \mathcal{M}_{i-1} + \ldots + \mathcal{M}_1 + \infty(\mathcal{M}_0)$ such that $lub(\{\mathcal{M}'_i \mid i \in \mathbb{N}\}) = \mathcal{A}' + \infty(\mathcal{A} + \mathcal{B})$, as we wanted to prove. □

Using this result we could enrich the algorithm in order to perform dw-accelerations. More precisely, it could be modified so as to choose fairly ω-markings $\mathcal{M} = \mathcal{A} + \infty(\mathcal{B})$, then to accelerate starting from $\mathcal{B}$ (actually, we could start from any number of copies of $\mathcal{B}$) and then to add the ω-marking given by the previous result. Notice that this ω-marking can have unbounded products in the unbounded part. As an example, let us again consider the ν-PN in Fig. 8, for which we obtained after the first acceleration the ω-marking $p + \infty(q)$. It is easy to see that $q \xrightarrow{t_2} qq$, and accelerating this transition yields the ω-marking q^ω, so that by the previous result it is correct to add to Φ the ω-marking $p + \infty(q^\omega)$.

In Def. 5.1 and Def. 5.2 we are fixing by means of the mapping ι the relation between names in $\mathcal{M}_1$ and names in $\mathcal{M}_2$. In particular, we are choosing among one of such possible relations, and forcing that the chosen relation is kept between all the markings in the generated increasing chain. Thus, we are removing part of the inherent non-determinism in ν-PN that arises in the non-deterministic choosing of consumed names by transitions. For instance, the net in Fig. 10 can fire its only transition $p \xrightarrow{t} p + pp$ and $p \sqsubseteq p + pp$, but we can choose two different ways to map products to products, namely $\iota_1(p) = p$ and $\iota_2(p) = pp$. In the first case, the result of accelerating is the ω-marking $\mathcal{M}_1 = \infty(pp)$ (we are always consuming the just created name), while in the second case we obtain $\mathcal{M}_2 = p^\omega + \infty(p)$ (we are always taking the name that appeared already in the initial marking).

If we do not impose any particular relation between the names, then at any point any token could be chosen, so that starting from the initial marking, any marking of the form $p^{n_1} + ... + p^{n_k}$ can be reached, with least upper bound $\infty(p^\omega)$. Therefore, any acceleration schema that does not impose any mapping ι relating names should compute $\infty(p^\omega)$ as acceleration.

In general, if we can choose between several mappings $\iota_1, ..., \iota_k$, because of monotonicity, in each of the limits $\mathcal{M}_1, ..., \mathcal{M}_k$ we can again choose between those mappings. Actually, if we choose again the mapping ι_i starting from $\mathcal{M}_i$, the obtained marking is again $\mathcal{M}_i$, by definition of acceleration. However, we could use a different ι_j to accelerate starting from $\mathcal{M}_i$, with $i \neq j$. In our previous example, we can again accelerate starting from $\mathcal{M}_1 = \infty(pp)$ and from $\mathcal{M}_2 = p^\omega + \infty(p)$. In the case of $\mathcal{M}_1$, we reach $\mathcal{M}_3 = ppp + \infty(pp)$, and ppp is obtained from one of the infinitely-many pp. If we apply a dw-acceleration we obtain $\infty(p^\omega)$. Moreover, this is also what we obtain if we accelerate starting from $\mathcal{M}_2$.

In the previous example we have managed to accelerate (using dw-accelerations) without restricting ourselves to a given mapping ι. However, it remains to see that we can do it in general, that is, that we can still accelerate any loop even if we remove the hypothesis of accelerating with respect to a given mapping ι. In this case the situation becomes much more complex. In particular, it is no longer true that we can contract a sequence of transitions into a single equivalent transition.

7. Conclusions and future work

In this paper we have established a forward analysis for ν-PNs, an extension of P/T nets with pure name management and creation, with the goal of computing a finite basis of its cover, that is, of the set $\downarrow Post^*(M_0)$. For that purpose, we have applied the results and techniques developed in [11, 12] for WSTS. We have defined a friendly presentation of the completion of a ν-PN by means of ω-markings, a natural extension of the analogous concept for P/T nets. We have seen that the transition relation, lifted to the completion, is effective, that is, we can effectively compute successors. Moreover, we have seen that if we restrict the non determinism, they are ∞-effective (we can compute the least upper bounds of the sets of ω-markings produced by simple loops). This ensures that it makes sense to apply a forward Karp-Miller procedure. However, we have proved that such procedure cannot terminate in general, or we could decide the property of depth-boundedness, which is undecidable. As a corollary, a finite basis of the cover is not computable for the class of Transfer Data Nets, not even for the class of Petri Data Nets. Nevertheless, we can slightly modify that algorithm to get a procedure to decide width-boundedness and to compute a finite basis of the cover of a width-bounded net.

The d-accelerations and w-accelerations in Sect. 5 appear naturally when computing the least upper bound of simple loops. We have identified a subclass of ν-PN, namely those that are dw-bounded, for which those accelerations are enough to compute the cover. Thus, for dw-bounded nets (a property which is in turn undecidable) we can decide for instance the place-boundedness problem, which is undecidable in general.

If we consider dw-accelerations, then we can also compute the cover of some dw-unbounded nets. It would certainly be interesting to characterize the subclass of ν-PN (larger than dw-bounded nets) for which `Clover` terminates. In general, it would be interesting to see if a non-deterministic version of accelerations, in which we do not restrict the modes by the relation between the different names involved represented by the mapping ι, is computable. More precisely, it would be interesting to study the structure of the set of markings $\{\mathcal{M} \mid \mathcal{M}_1 \xrightarrow{t^k} \mathcal{M}\}$ (without restricting the modes), to see if it is a directed set, and computing its least upper bound in that case.

References

[1] P.A. Abdulla, K. Cerans, B. Jonsson, and Y. Tsay. Algorithmic Analysis of Programs with Well Quasi-ordered Domains. Inf. and Comp. 160(1-2):109-127 (2000)

[2] P.A. Abdulla, and A. Nylén. Better is Better than Well: On Efficient Verification of Infinite-State Systems. 15th Annual IEEE Symp. on Logic in Computer Science, LICS'00 (2000) 132-140.

[3] P.A. Abdulla, A. Collomb-Annichini, A. Bouajjani, and B. Jonsson. Using Forward Reachability Analysis for Verification of Lossy Channel Systems. Formal Methods in System Design 25(1): 39-65 (2004)

[4] S. Bardin, A. Finkel, J. Leroux, and Ph. Schnoebelen. Flat Acceleration in Symbolic Model Checking. 3rd Int. Symposium on Automated Technology for Verification and Analysis, ATVA'05. LNCS vol. 3707. Springer (2005) 474-488.

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

[6] G. Decker, and M. Weske. Instance Isolation Analysis for Service-Oriented Architectures. *Int. Conference on Services Computing, SCC'08*, IEEE Computer Society (2008) 249-256.

[7] C. Dufourd, A. Finkel, and Ph. Schnoebelen. Reset Nets Between Decidability and Undecidability. 25th Int. Automata, Languages and Programming Colloquium, ICALP'98. LNCS vol. 1443. Springer (1998) 103-115.

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

[9] A. Finkel, and Ph. Schnoebelen. Fundamental Structures in Well-Structured Infinite Transition Systems. In 3rd Latin American Symposium on Theoretical Informatics, LATIN'98. LNCS vol. 1380. Springer (1998) 102-118.

[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 vol. 5556. Springer (2009) 188-199.

[13] G. Geeraerts, J.-F. Raskin, and L. van Begin. Expand, enlarge and check: New algorithms for the coverability problem of WSTS. *J.Comp. Sys. Sci.*,72(1):180-203 (2006)

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

[15] J. Goubault-Larrecq. On Noetherian spaces. 22nd IEEE Symposium on Logic in Computer Science, LICS'07. IEEE Computer Society (2007) 453-462.

[16] P. Jančar. A note on well quasi-orderings for powersets. Information Processing Letters 72(5-6):155-160 (1999)

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

[18] 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.

[19] F. Rosa-Velardo, D. de Frutos-Escrig, and O. Marroquín-Alonso. On the expressiveness of Mobile Synchronizing Petri Nets. In 3rd International Workshop on Security Issues in Concurrency, *SecCo'05*. ENTCS 180(1). Elsevier (2007) 77-94.

[20] F. Rosa-Velardo and D. de Frutos-Escrig. Name creation vs. replication in Petri Net systems. *Fundamenta Informaticae* 88(3). IOS Press (2008) 329-356.

[21] F. Rosa-Velardo and D. de Frutos-Escrig. Name creation vs. replication in Petri Net systems. ATPN'07. LNCS vol. 4546. Springer (2007) 402-422.

[22] F. Rosa-Velardo and D. de Frutos-Escrig. Forward analysis for Petri Nets with Name Creation. PETRI NETS 2010. LNCS vol. 6128. Springer (2010) 185-205.

[23] F. Rosa-Velardo and D. de Frutos-Escrig. Decidability and Complexity of Petri Nets with Unordered Data. Theoretical Computer Science (to appear)

[24] Ph. Schnoebelen. Verifying lossy channel systems has nonprimitive recursive complexity. *Inf. Process. Lett.* 83(5) (2002) 251-261.