Informatique théorique et applications

E. Pelz

Normalization of place/transition-systems preserves net behaviour

Informatique théorique et applications, tome 26, nº 1 (1992), p. 19-44.

<http://www.numdam.org/item?id=ITA_1992__26_1_19_0>

Informatique théorique et Applications/Theoretical Informatics and Applications
(vol. 26, n° 1, 1992, p. 19 à 44)

NORMALIZATION OF PLACE/TRANSITION-SYSTEMS PRESERVES NET BEHAVIOUR (*)

by E. PELZ ([1])

Communicated by Wilfried BRAUER

Abstract. – *In this paper we consider place/transition-systems (abbreviated as P/T-systems) which are λ-free labeled. They are called normalized if their arcs are not weighted and their initial and final markings are subsets of the set of places. We prove that for each general P/T-system there exists a normalized P/T-system having exactly the same concurrent behaviour (in the partial word semantics). The same (constructive) transformation also preserves finitary and infinitary sequential behaviours and the step behaviour. This allows us to consider only normalized P/T-systems when working on net behaviour without loss of generality.*

Résumé. – *Nous considérons dans cet article des réseaux de Petri étiquetés sans λ. On les appelle normalisés si leurs arcs ne sont pas valués et si leurs marquages initiaux et finals sont des sous-ensembles de l'ensemble des places. Nous prouvons que tout réseau de Petri général peut être (effectivement) transformé en un réseau de Petri normalisé ayant exactement le même comportement concurrent. Ses comportements séquentiels finis et infinis ainsi que ses suites de pas sont également préservés. Ceci permet de toujours considérer des réseaux de Petri sous une forme normalisée quand on travaille sur le comportement des réseaux, sans restreindre la généralité des résultats. Ainsi un bon nombre de recherches futures devrait se trouver facilité.*

0. INTRODUCTION

The general definition of place/transitions-systems (P/T-systems) allows arbitrary weights of the arcs of a net and arbitrary initial and final markings. For normalized P/T-systems the weight function on the arcs can take only values in $\{0, 1\}$ and the initial and final markings can only be $\{0, 1\}$-vectors. In this context several problems arise.

The first problem is to transform a general P/T-system into a normalized one having the same finitary behaviour (e. g. the same firing sequences). From 1979 on, some results in this direction have been obtained: in the

(*) Received April 1990, revised April 1991.
Research supported by ESPRIT Basic Research Action, project 3148: DEMON.
([1]) L.R.I., C.N.R.S.-U.A. 410, Université Paris-Sud, Bât. 490, 91405 Orsay Cedex, France.

literature one can find several constructions for the transformation of general P/T-systems into normalized ones but with arbitrary labelings (*cf.* [4], [16], [15]); these transformations preserve the firing sequences and/or the finitary languages of the systems. We mention only two points which their constructions have in common and with which we do not agree:

— the use of invisible actions *i.e.* λ-labeled transitions: as is well known that λ-labeled transitions allow one to obtain quite all r.e. sets, these constructions are quite easy. And they do not necessarily preserve the net class, which can be more embarrassing (*cf.* Hack's theorem [9], stating that the class of finitary sequential behavours of λ-free labeled P/T-systems is strictly contained in that of arbitrary labeled P/T-systems).

— the restriction of these authors' interest to the preservation of sequential behaviour, whereas nets are normally used in order to study concurrency and thus the main interest should center in the "truely" concurrent behaviour. We remind the reader that in the case of general P/T-systems the set of processes (or the set of partial words) cannot be computer for the net from is sequential language.

With this said, we can formulate further problems: the second one is to solve the problem of normalization inside the class of λ-free labeled nets (without adding invisible actions). And the third one is to generalize such a solution to other semantics, especially to partial word semantics.

In an earlier paper [10], a transformation, which preserves sequential finitary languages, from a general P/T-system into a normalized one, both without λ-labels, was presented for the first time. By induction on the number of places q which do not have a desired property (e.g. they have an incident arc with value different from one) we duplicate all places q and the transitions in their pre- and postsets. The duplication factor of each such place q depends on the multiplicities of its incident arcs, that of a transition depends on combinatiorial considerations with respect to the copies of place q.

The algorithms we present in this article are based on the same combinatiorial principles as in the paper quoted. The adaptation to process of partial word semantics implies modifications, namely concerning markings, and the introduction of — what we call — the "initial component". As we want to preserve the partial word language, the proofs that the given transformations are good become non trivial and make use of a result from graph-colouring. With the given algorithms we solve the third problem.

One could be tempted to think that the work done in [6], or in a more elegant way in [18], on the transformation of P/T-systems into 1-safe nets preserving the concurrent behaviour, already yields a solution to the third problem. (Let us recall that a net is called 1-safe if all its reachable markings are $\{0, 1\}$-vectors). But the *P/T*-systems considered in these papers are very restricted: All have finite place capacities. This is a necessary condition for the transformation into a 1-safe net or in other terms, into a *C/E*-system. The difference to our work is that we shall consider *P/T*-systems in their generality and that the normalized nets we obtain are without capacity restrictions, only the specified markings (*i.e.* the initial and finite ones) are 1-safe.

The transformation methods described in the quoted articles cannot be generalized to suit our purposes. For instance the net of figure 1, whose concurrent as well as sequential behaviour is $\{a^n b^n \mid n>0\}$, has an unbounded place, p_2. Generalizations of Goltz's or Vogler's method would transform it into an infinite net (having an infinite number of copies of p_2 and an infinite number of transitions). Our transformation leaves the net unchanged; it is already normalized.

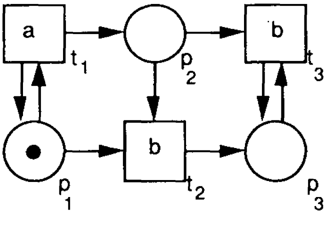


Figure 1. – $M_f = \{p_3\}$.

In the last section we study other semantics. It will be easy to show that our transformation from a *P/T*-system to its normalized version preserves finitary and infinitary sequential languages, as well as step languages.

As a consequence work on behavioural aspects of λ-free labeled *P/T*-systems may become easier: no matter which kind of net behaviour one is interested in – for particular or general investigations – one can always restrict the study to normalized *P/T*-systems without loss of generality.

Further investigations should be facilitated by this result. For instance, by the use of normalized nets the proofs in [11] become much simpler.

1. SOME CLASSICAL DEFINITIONS

We use the standard notations for net theory as in [5] and [2]. Througout this paper Σ denotes a *finite alphabet*, Σ^* the set of *finite words* over Σ, Σ^ω the set of *infinite words* over Σ.

We recall the notion of *partial words* over Σ, *cf.* [8]:

A *partial ordering* on a set is an irreflexive and transitive binary relation; if $<$ is such a relation, we use $\leqq$ for its extension by the identity.

A *labeled partial ordered set* over the alphabet Σ is a triple $(U, <, \beta)$ where U is a finite or infinite set, $<$ is a partial ordering on U and $\beta: U \rightarrow \Sigma$ is a function.

Two labeled partial ordered sets $(U, <, \beta)$ and $(U', <', \beta')$ are said to be *isomorphic* iff there is an order isomorphism

$$f: \quad (U, <) \rightarrow (U', <') \text{ such that } \beta' \circ f = \beta.$$

A *partial word* is an isomorphism class of labeled partial ordered sets over Σ. The class containing $(U, <, \beta)$ is denoted by $[(U, < \beta)]$. The *set of all partial words* over Σ is denoted by $\mathscr{PW}(\Sigma)$.

The *transitive closure of a binary relation* A will be denoted A^+ and its *reflexive and transitive closure*, A^*.

The *restriction of a function* f to a subset X of its domain is denoted $f|_X$.

A *net* N is defined as a triple $N=(P, T, A)$, where P is a set of *places*, T a set of *transitions*, and $A \subseteqq P \times T \cup T \times P$ a set of *arcs*.

The *pre-set* (resp. *post-set*) of an element $x \in P \cup T$ is written ${}^\circ x$ (respectivement x°) and defined by ${}^\circ x = \{ y \in P \cup T \mid (y, x) \in A \}$, (respectivement $x^\circ = \{ y \in P \cup T \mid (x, y) \in A \}$). This notation can be generalized to sets $X \subseteqq P \cup T$ in the following way: ${}^\circ X = \{ {}^\circ x \mid x \in X \}$ and $X^\circ = \{ x^\circ \mid x \in X \}$.

The *input* ${}^\circ N$ *of a net* N, is defined by ${}^\circ N = \{ x \in P \cup T \mid {}^\circ x = \varnothing \}$ and its *output* N° by $N^\circ = \{ x \in P \cup T \mid x^0 = \varnothing \}$.

Place/Transition systems

We call a (general) *P/T-system* a *λ-free labeled Place/Transition-system* (*or Petri net*) $N=(P, T, A, v, h, M_0, F)$

where (P, T, A) is a finite net with $P = \{ p_1, \ldots, p_r \}$ and $T = \{ t_1, \ldots, t_s \}$, $v: P \times T \cup T \times P \rightarrow \mathbb{N}$ is a *weight function on the arcs* satisfying $v(x, y) = 0$ iff $(x, y) \notin A$, $h: T \rightarrow \Sigma$ a *λ-free labeling* from T into a finite alphabet Σ, $M_0 \in \mathbb{N}^r$ an *initial marking* of P, and $F \subseteqq \mathbb{N}^r$ a finite set of *final markings*.

If we consider the *infinite behaviour* of nets, the term F in the definition of a P/T-system will be replaced implicetely by $\mathscr{F} \subseteqq \mathscr{P}(\mathbb{N}^r)$, a finite set of non-empty sets of markings. $\mathscr{F}$ is called the set of *anchor marking sets*.

Remark: As P/T-systems are normally without capacity restrictions, we shall in general identify the set of markings $\{M \mid M : P \to \mathbb{N}\}$ of the net and the set of vectors $\mathbb{N}^r$ of natural numbers of length $r = |P|$.

A P/T-system is *normalized* if it satisfies the following conditions:

(P1): the weight function takes its values in $\{0, 1\}$ (or in other words, arc weights are all unity);

(P2): the initial and final markings are subsets of P (or in other words, subsets of $\{0, 1\}^r$).

Remark: Sometimes, e. g. in [3], nets satisfying (P1) are called "ordinary".

If the labeling is a function from T into $\Sigma \cup \{\lambda\}$, *i.e.* if the label λ representing an invisible action is allowed, we speak of *arbitrary labeled P/T-systems.*

A *partial order on markings* can be defined by $M \leqq M'$ if and only if $\forall p \in P$, $M(p) \leqq M'(p)$.

The labelling of transitions of a net can be canonically extended to sequences of transitions by $h(t_1 t_2 \ldots t_n) = h(t_1) h(t_2) \ldots h(t_n)$.

We have the following *firing rule on P/T-systems*: a *transition t may occur* (or *is firable*) at a marking M if and only if for all places $p \in P$ we have $M(p) \geqq v(p, t)$; we write then $M(t\rangle$.

In a similar way $M(\bar{t}\rangle$ denotes a *sequence of transitions* $\bar{t} = t_0 t_1 t_2 \ldots$ *which may occur at M*. If $\bar{t}$ is finite $\bar{t} = t_0 t_1 \ldots t_n$, and its occurrence yields the new marking M', we write $M(\bar{t}\rangle M'$. This implies the existence of markings $M_i (0 \leqq i \leqq n+1)$ satisfying: $M = M_0$, $M' = M_{n+1}$ and for each $i \leqq n$, $M_i(t_i\rangle M_{i+1}$. M' can also be obtained directly by

$$M'(p) = M(p) - \sum_{i \leqq n} v(p, t_i) + \sum_{i \leqq n} v(t_i, p),$$

for each $p \in P$.

Now we shall consider multisets of transitions. Formally, they are functions from T into $\mathbb{N}$, but we use set terminology for them as well.

Let u be a non-empty multiset of transitions of T. We say that *u may occur concurrently at M* (notation: $M(u\rangle$) if and only if the transitions in u may occur individually at M and together. More precisely, let $\#_t(u)$ be *the number of transitions t in u*. Then $M(u\rangle$ iff

$$\forall p \in P \quad \sum_{t \in u} \#_t(u) \,.\, v(p, t) \leqq M(p).$$

If the occurrence of u changes M to M', we write $M(u\rangle M'$ with $M'(p) = M(p) - \sum_{t \in u} \#_t(u) \,.\, v(p, t) + \sum_{t \in u} \#_t(u) \,.\, v(t, p)$ for all $p \in P$.

As u is a multiset, transitions may occur auto-concurrently. In particular, if t is a transition with an empty pre-set (*i.e.* $^\circ t = \emptyset$), t may occur auto-concurrently as many times as one wants at each marking of the net: $\forall n \in \mathbb{N}$ $\forall M \in \mathbb{N}^r\ M(t^{(n)}\rangle$, where $t^{(n)}$ stands for the multiset containing the transition t exactly n times.

Finite sequential behaviour of a P/T-system $N=(P, T, A, v, h, M_0, F)$

The *finite sequential behaviour of a P/T-system N* will be defined with respect to the final markings F:

$L(N) = \{ w \in \Sigma^+ \mid \exists M_f \in F\ \exists \bar{t} \in T^+$ such that $M_0(\bar{t}\rangle M_f$ and $h(\bar{t}) = w \}$ is called the *language of N*. The corresponding class id noted **L**, precisely $\mathbf{L} = \{ L(N) \mid N$ is a general P/T-system$\}$.

Extending this definition to arbitrary labeled P/T-systems we obtain the class $\mathbf{L}^\lambda = \{ L(N) \mid N$ is an arbitrary labeled P/T-system$\}$.

This class is strictly situated between the class **L** and the class of recursively enumerable languages (**r.e.**), but does not contain the class of context free languages (**CF**), as stated by the following result of [9]:
Fact 1: $\mathbf{L} \subset \mathbf{L}^\lambda \subset$ r.e. and not CF $\subset \mathbf{L}^\lambda$.

Concurrent behaviour of a P/T-system $N=(P, T, A, v, h, M_0, F)$

If we are interested in the concurrent behaviour of a net, one idea is to "run" the net – resolving conflicts in an arbitrary fashion as and when they arise. The resulting objects (called processes) are based on a special kind of net called occurrence nets.

An *occurrence net* is a net (B, E, A) which satisfies

(i) $\forall b \in B, |^\circ b| \leqq 1$ and $|b^\circ| \leqq 1$

(ii) $\forall x, y \in B \cup E, (x, y) \in A^+ \Rightarrow (y, x) \notin A^+$

(iii) $\forall e \in E, |e^\circ| \geqq 1$.

Remark: In order to eliminate any confusion between the objects of a P/T-systems and its run, the places of an occurrence net are called B and its transitions E, as in *EN* systems.

$(B, E, A; \varphi)$ is a *node-labeled occurrence net* if (B, E, A) is an occurrence net and φ is a total function from $B \cup E$ to an alphabet Σ.

We first give the definition of the concurrent behaviour of a P/T-system as a "run", *i.e.* as a certain occurrence net, restricted to unlabeled P/T-systems without specified final-markings (as in [7]).

A *process of an unlabeled P/T-system* $N=(P, T, A, v, M_0)$ is a node-labeled occurrence net $\pi=(B, E, A'; \varphi)$ iff

(i) $\varphi(B) \subseteq P$ and $\varphi(E) \subseteq T$

(ii) $^\circ\pi \subseteq B$ and $\forall p \in P \; M_0(p) = |\varphi^{-1}(p) \cap {}^\circ\pi|$

(iii) $\forall e \in E, \forall p \in P \; v(p, \varphi(e)) = |\varphi^{-1}(p) \cap {}^\circ e|$ and $v(\varphi(e), p) = |\varphi^{-1}(p) \cap e^\circ|$.

We will generalize this definition to general P/T-systems. As transitions may have an empty pre-set (or post-set), the condition $^\circ N \subseteq B$ of (ii) will not be required.

A *process of a P/T-system* $N=(P, T, A, v, h, M_0, F)$ is a node-labeled occurrence net $\pi=(B, E, A'; \varphi)$ iff

(i) $\varphi(B) \subseteq P$ and $\varphi(E) \subseteq T$

(ii) $\forall p \in P \; M_0(p) = |\varphi^{-1}(p) \cap {}^\circ\pi|$

(iii) $\forall e \in E, p \in P \; v(p, \varphi(e)) = |\varphi^{-1}(p) \cap {}^\circ e|$ and $v(\varphi(e), p) = |\varphi^{-1}(p) \cap e^\circ|$

(iv) $\exists M \in F \; \forall p \in P \; M(p) = |\varphi^{-1}(p) \cap \pi^\circ|$.

To have a notion expressing true concurrency which is comparable to languages, we are interested only in recording the transitions which occur concurrently in a process. If $\pi=(B, E, A; \varphi)$ is a process of N we will consider the B-contraction of π. This is the labeled graph $(E, A'; \varphi)$ where a pair $(e, e') \in E^2$ is an arc of A' if and only if there exists a place $b \in B$ of π such that $(e, b) \in A$ and $(b, e') \in A$; it is called the *contracted process of N*. Its transitive closure (E, A'^{+}, φ) is a labeled partial ordered set, whose image in Σ is the labeled partial ordered set (E, A'^{+}, β) with $\beta = h \circ \varphi$. Finally the *partial word associated with* π is its equivalence class $[(E, A'^{+}, \beta)]$.

The set of all partial words associated with processes of N is called the *partial word language of N* and noted $\mathcal{PWL}(N)$. It is a subset of $\mathcal{PW}(\Sigma)$.

Remark: In general we shall not distinguish a graph-theoretic object (like a net or a multigraph or a labeled partial order) and its isomorphism class.

2. NORMALIZATION OF *P/T*-SYSTEMS

In this section we prove that for each general P/T-system there is a normalized system having the same concurrent behaviour.

THEOREM 1: *Let N be a general P/T-system. There is an normalized P/T-system N″ such that* $\mathcal{PWL}(N) = \mathcal{PWL}(N'')$.

The transformation from N to N'' composes two steps. The proof follows immediately from the following two lemmata. Let us recall that a net satisfies property (P1) if and only if each arc of the net has weight unity.

LEMMA A: *Let N be a general P/T-system. There is a P/T-system N' satisfying property* (P1) *such that $\mathscr{PWL}(N)=\mathscr{PWL}(N')$.*

LEMMA B: *Let N be a a P/T-system satisfying property* (P1). *There is an normalized P/T-system N'' such that $\mathscr{PWL}(N)=\mathscr{PWL}(N'')$.*

To prove the lemmata, we decrease the number of places satisfying an undesired property by induction; the structure of these proofs is as follows:

First, we give the transformation which is to be applied at each step of the induction of lemma A, then we prove lemma A. Subsequentely we proceed in the same way for lemma B. Note, in the sequential case [10], it was sufficient to define the transformations, the remaining proofs being omitted. In our case, the transformations are more complicated (e. g. an "initial component" has to be defined) and the proofs become non-trivial (especially for lemma A). (Note also a change of notation: $^\circ q$ was denoted by $I(q)$ and q° by $O(q)$ in [10].)

Transformation A

Input: ***A P/T-system $N=(P, T, A, v, h, M_0, F)$ and a place*** $q \in Q(N)$, where $Q(N) \subseteq P$ is the set of places p for which there is at least one transition satisfying $v(p, t)>1$ or $v(t, p)>1$, .

Output: ***A P/T-system $N'=(P', T', A', v', h', M'_0, F')$ with $Q(N')=Q(N)\setminus\{q\}$ having the same partial word language as N.***

ALGORITHM: Let $n=\max\{i \in \mathbb{N} \mid \exists t \in T\ v(q, t)=i \text{ or } v(t, q)=i\}$.

First we replace q by n new places $q_0, \ldots, q_{n-1}$. We often consider the set $\mathscr{C}_q=\{q_0, \ldots, q_{n-1}\}$, the copies of q.

Each transition connected to q by an arc will be replaced by as many copies as there are possibilities of taking one token from $v(q, t)$ different places in $\mathscr{C}_q$ and adding one token to $v(t, q)$ different places in $\mathscr{C}_q$:

For each transition $t \in {}^\circ q \cup q^\circ$ let $(D_i)_{i<n_t}$ and $(E_j)_{j<m_t}$ be enumerations of the subsets of $\mathscr{C}_q$ having $v(q, t)$ resp. $v(t, q)$ elements, thus $n_t=|\mathscr{P}_{v(q,t)}(\mathscr{C}_q)|=\binom{n}{v(q,t)}$ and $m_t=|\mathscr{P}_{v(t,q)}(\mathscr{C}_q)|=\binom{n}{v(t,q)}$. Note that $v(q, t)=0$ implies $n_t=1$ and $D_0=\varnothing$, and that $v(t, q)=0$ implies $m_t=1$ and $E_0=\varnothing$.

Replace t by $n_t . m_t$ transitions $(t_{i,j})_{i<n_t, j<m_t}$ having the same label as t and such that for each (i, j):

(i) the arcs between $t_{i,j}$ and the old places $p \in P \setminus \{q\}$ are the same as those between t and these places.

(ii) the arcs $r \to t_{i,j}$ and $t_{i,j} \to s$ for every $r \in D_i$ and $s \in E_j$ are added.

Example: These replacements are illustrated in figure 2:

We show the part of a net N which is incident to place q (with $M_0(q)=0$) (*fig.* 2*a*).

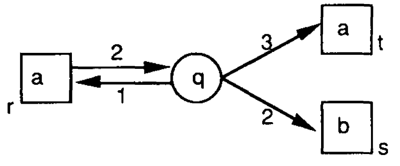


Figure 2*a*

Under transformation A this part will be replaced by the net of figure 2*b* (with $M'_0(q_i)=0$, for $i=0, 1, 2$). The following enumerations have been considered

for r:

$f(q, r)=1 \Rightarrow n_r=3$ and $D_0=\{q_0\}$, $D_1=\{q_1\}$, $D_2=\{q_2\}$

$b(q, r)=2 \Rightarrow m_r=3$ and $E_0=\{q_0, q_1\}$, $E_1\{q_0, q_2\}$, $E_2=\{q_1, q_2\}$

for t:

$f(q, t)=3 \Rightarrow n_t=1$ and $D_0=\{q_0, q_1, q_2\}$

$b(q, t)=0 \Rightarrow m_t=0$

for s:

$f(q, s)=2 \Rightarrow n_s=3$ and $D_0=\{q_0, q_1\}$, $D_1=\{\{q_0, q_2\}, D_2=\{q_1, q_2\}$

$b(q, s)=0 \Rightarrow m_s=0$

With each final marking $M_f \in F$ we associate a set of final markings F_{M_f} in the following way: $M \in F_{M_f}$ iff $M|_{P\setminus\{q\}}=M_f$ and $\sum_{i<n} M(q_i)=M_f(q)$.

Finally we define $F'=\{M \in F_{M_f} \mid M_f \in F\}$.

Now let us consider the initial marking with $M_0(q)=k$. We have to define M'_0. Independently of k we take $M'_0|_{P\setminus\{q\}}=M_0$.

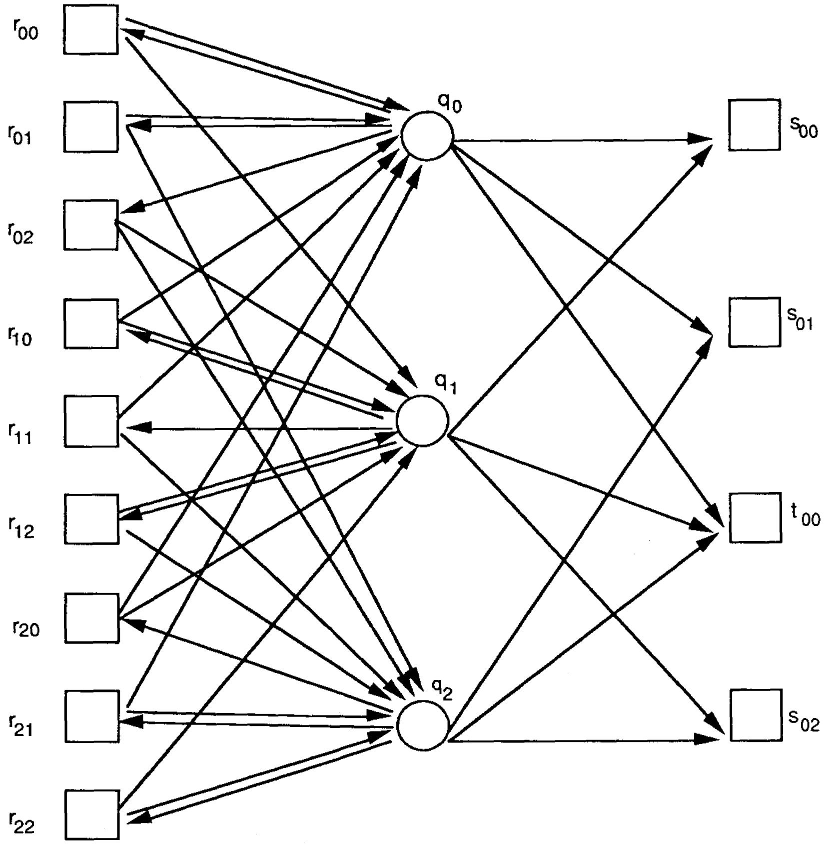


Figure 2*b*

If $k=0$, we set $M'_0(q_i)=0$ for each $i<n$.

If $k>0$, there are some problems: we would need a "set of initial markings" containing all markings M' satisfying $M'|_{P\setminus\{q\}}=M_0$ and $\sum_{i<n} M'(q_i)=M_0(q)$. (The reader will well understand at the end of the proof of lemma A why this is necessary.) But technically it is not possible to have a set of initial markings (rather of a single one), the final net would not be normalized.

The solution we choose consists of the addition of an "initial component" to N': We add k "initial" places $q'_0, \ldots, q'_{k-1}$ marked by one token (*i. e.* $M'_0(q'_i)=1$ for each $i<k$). Each of these places should play the rôle of each one of the places $q_0, \ldots, q_{n-1}$. In particular, if in the system N a

transition t with $q \in {}^\circ t$ may be auto-concurrently enabled m times at M_0, we need that each one of the new transitions $t_{i,j}$ in the system N' may be auto-concurrently enabled m times at M'_0. It is not sufficient to have m different copies of t concurrently enabled at M'_0 (see the end of the proof Lemma A). We call $\mathscr{I}_q$ the set $\{q'_0, \ldots, q'_{k-1}\}$ of initial copies of q.

We also need to add "initial" copies of the transitions $t \in q^\circ$:

We add for each h such that $1 \leqq h = \min(v(q, t), M_0(q)) \leqq k$ a set of copies of t which can be enabled by h tokens from the initial places in $\mathscr{I}_q$ and $v(q, t) - h$ generated tokens from the places in $\mathscr{C}_q$:

Let $(F_l)_{l<h_t}$ be an enumeration of the subsets of $\mathscr{I}_q$ having h elements, $(G_i)_{i<k_t}$ be an enumeration of the subsets of $\mathscr{C}_q$ having $v(q, t) - h$ elements and $(E_j)_{j<m_t}$ be an enumeration of the subsets of $\mathscr{C}_q$ with $v(t, q)$ elements, thus

$$h_t = |\mathscr{P}_h(\mathscr{I}_q)| = \binom{k}{n},$$

$$k_t = |\mathscr{P}_{v(q,t)-h}(\mathscr{C}_q)| = \binom{n}{v(q,t)-h} \quad \text{and} \quad m_t = |\mathscr{P}_{v(t,q)}(\mathscr{C}_q)| = \binom{n}{v(t,q)}.$$

We add $h_t . n_t . m_t$ "initial" transitions $(t'_{l,i,j})_{l<h_t,\, i<k_t,\, j<m_t}$ having the same label as t and such that, for each (l, i, j):

(i) the arcs between $t'_{l,i,j}$ and the old places $p \in P \setminus \{q\}$ are the same as those between t and these places;

(ii) the arcs $p \to t'_{l,i,j}$, $r \to t'_{l,i,j}$ and $t'_{l,i,j} \to s$ for every $p \in F_l$, $r \in G_i$ and $s \in E_j$ are added.

Note firstly that we never add arcs whose targets are in $\mathscr{I}_q$, therefore the places in $\mathscr{I}_q$ are really "initial". Secondly, the addition of this "initial component" to the net not only ensures all combinations of removing and adding tokens to copies of q, but also decreases the number of places with an initial marking greater than one. Thus the nect algorithm (B) should not be applied to the place q (nor to its copies).

end of transformation A

Proof of lemma A: Let $N = (P, T, A, v, h, M_0, F)$ be a general P/T-system and $Q(N) \subseteq P$ the set of places q for which there is at least one transition t such that $v(q, t) > 1$ or $v(t, q) > 1$.

By induction on the cardinality of Q(N), we have to show how to decreases this number by one. Application of transformation A to N and to one place $q \in Q(N)$, yields a P/T-system N' with $Q(N') = Q(N) \setminus \{q\}$. Now we only have to prove that N' and N have the same partial word languages. The result we show is a slightly stronger: we shall prove that N' and N have the "same" processes, where the quotes indicate "up to a little renaming". This renaming associates labels from $\mathscr{C}_q$ with conditions labeled by q and labels from $(t_{i,j})_{i<n_t, j<m_t}$ with transitions labeled by t (for $t \in {}^\circ q \cup q^\circ$) – and vice versa –. The remainder of the process does not change at all.

In performing this renaming, we must ensure that the renamed process of N is a well defined process of N' – in particular, that the set of labels of the pre-set (resp. post-set) of a renamed transition will never contain a multiset from $\mathscr{C}_q$.

Let us now describe precisely how the renaming is defined. We begin with the easy direction:

Let $\pi' = (B, E, F, \varphi')$ be a process of N'. We associate with π' a process $\pi = (B, E, F, \varphi)$ of N where only the labeling changes.

First we consider all places $b \in B$: if $\varphi'(b) \notin \mathscr{C}_q \cup \mathscr{I}_q$ we take $\varphi(b) = \varphi'(b)$, if $\varphi'(b) \in \mathscr{C}_q \cup \mathscr{I}_q$ we take $\varphi(b) = q$.

Next we consider transitions $e \in E$: if there is some t such that $t \in {}^\circ q \cup q^\circ$ and $\varphi'(e) \in (t_{i,j})_{i<h_t, j<m_t} \cup (t'_{i,j})_{i<n_t, j<m_t} \cup (t'_{l,i,j})_{l<h_t, i<k_t, j<m_t}$ we take $\varphi(e) = t$, otherwise $\varphi(e) = \varphi'(e)$.

π is clearly a process of N, the partial words associated with π and π' are the same because $h' \circ \varphi' = h \circ \varphi$.

The proof in the other direction is much harder and makes use of graph colouring:

Assume that $\pi = (B, E, F, \varphi)$ is a process of N. We will define $\pi' = (B, E, F, \varphi')$ as a process of N'.

We consider a subnet $\pi_q = (B_q, E_q, F_q)$ of π containing exactly the places of B labeled by q, the transitions of E labeled by some $t \in {}^\circ q \cup q^\circ$ and the arcs of F restricted to $B_q \times E_q \cup E_q \times B_q$. The labels on $\pi \setminus \pi_q$ remain the same in π'. Thus we only have to rename π_q. The goal is to rename B_q by elements of $\mathscr{C}_q$ such that each pre- and post-set of a transition in E_q is labeled by a subset and not by a multiset of $\mathscr{C}_q$. Then the label of a transition e with old label t will be entirely determined by the labels of ${}^\circ e$ and e°, precisely by the ranks of these sets in the enumeration of all subsets of $\mathscr{C}_q$ with $v(q, t)$, resp $v(t, q)$, elements. We shall see att the end why the labeling of ${}^\circ\pi$ may create some problems and how they can be solved using the "initial component" we added to N'.

Intuitively one might expect to obtain this renaming by a straighforward iterative construction on an enumeration of E_q that is compatible with the order of π_q. But immediately one would find counter-examples for such a construction as the label of a place b in B_q depends not only on the transitions in $^\circ b$ but also on those in b°.

Thus the remaning should be global (on all elements of B_q at the same time). Our solution consists of reducing the renaming to the edge-colouring of a bipartite multigraph in the three steps described below and illustrated in figure 3. First let us consider the order of the net π_q: By the choice of n we know that each transition e in E_q satisfies $|^\circ e| \leqq n$ and $|e^\circ| \leqq n$, or, in graph theoretic terminology, that e has at most n incoming arcs and at most n outgoing arcs. For π_q, being a part of an occurrence net, each place p has at most one incoming arc and at most one outgoing arc.

Step 1: We first transform π_q to a directed arc-labeled multigraph $G=(V, A)$ where $V=E_q \cup ((^\circ\pi_q \cup \pi_q^\circ) \cap B_q)$. Intuitively, we make a kind of B-abstraction: we replace $e \to b \to e'$ by one arc $e \to e'$ labeled by b, but preserve arcs $b' \to e$ if $^\circ b' = \varnothing$ (this arc will be labeled by b') and $e' \to b''$ if $b''^\circ = \varnothing$ (this arc will be labeled by b'').

Precisely, for all $e, e' \in E_q$, $b \in B_q$,

$((e, e')$ labeled $b) \in A$ iff $(e, b) \in F_q$ and $(b, e') \in F_q$,

$((b, e)$ labeled $b) \in A$ iff $(b, e) \in F_q$ and $b \in {}^\circ\pi_q$,

$((e, b)$ labeled $b) \in A$ iff $(e, b) \in F_q$ and $b \in \pi_q^\circ$.

Step 2: Next we associate with G the bipartite multigraph $G^*=(V^*, A^*)$ obtained in the following way. Each vertex $v \in V \cap E_q$ is split into v^+, v^-, where v^+ will become the initial endpoint of all arcs being incident out of v, and v^- will become the terminal endpoint of all arcs incident into v.

The arcs incident to $(^\circ\pi_q \cup \pi_q^\circ) \cap B_q$ will not change. Thus we have a bipartition of V^* into $V^+ \cup (^\circ\pi_q \cap B_q)$ containing only initial endpoints and $V^- \cup (\pi_q^\circ \cap B_q)$ containing only terminal endpoints of arcs.

Step 3: Without loss of information we can remove the direction of the arcs to obtain the undirected bipartite multigraph $\bar{G}^*$.

Example: The graph in figure 3*a* shows the subnet $\pi_q=(B_q, E_q, F_q)$ of a process π. We consider that $\{0, 1, \ldots, 16\} = \mathrm{B_q} \cap (^\circ\pi \cup \pi^\circ)$, the inscriptions of the remaining places are omitted. The inscriptions in the transitions $e \in E_q$ are the labels $\varphi(e)$. Figure 3*b* shows its associated graph G and figure 3*c* the graph $\bar{G}^*$.

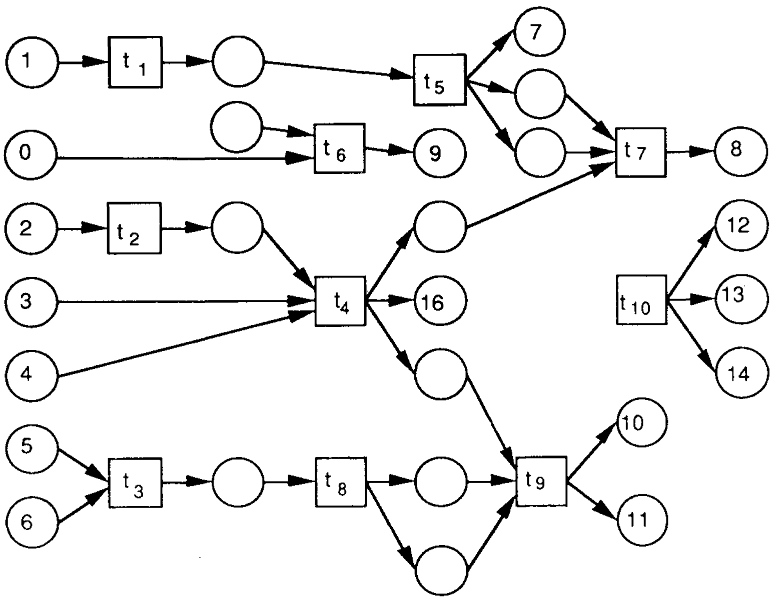


Figure 3*a*

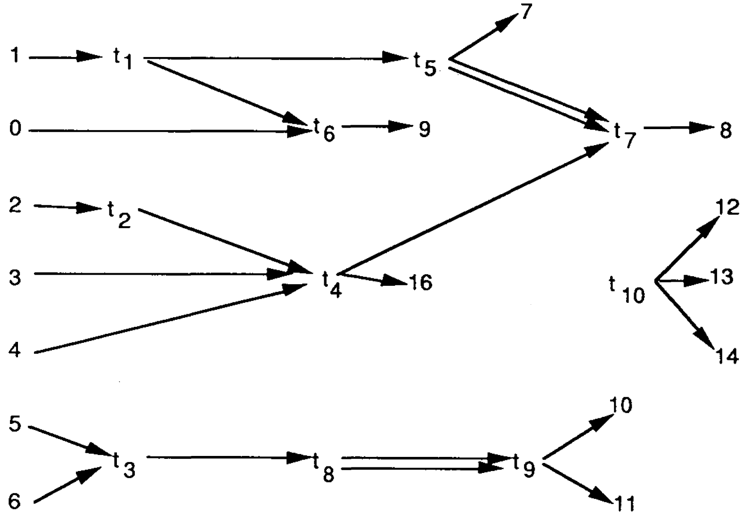


Figure 3*b*

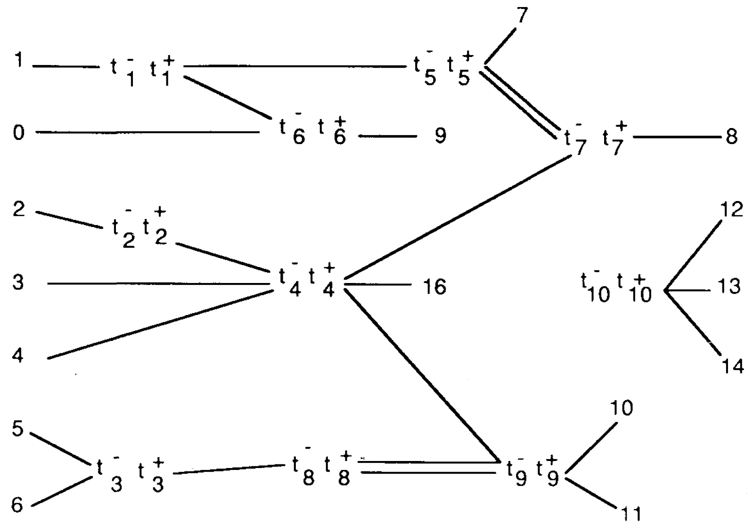


Figure 3 *c*

A well known graph theoretic theorem applies (e.g. [1]). Let us remind that the *chromatic index* $q(G)$ of a graph G is defined to be the smallest number of colours needed to colour the edges of G so that no two adjacent edges have the same colour.

THEOREM: *The chromatic index of a bipartite multigraph G with maximum degree h is $q(G)=h$.*

Remark: The proof of this theorem is important because the pratical method of edge-colouring stems from it. It is sketched in the complete paper [12] and illustrated by a colouring of the graph obtained in figure 3.

Let us continue the proof of lemma A:

As the maximal order of $\bar{G}^*$ is n, $q(\bar{G}^*)=n$, as well as $q(G^*)=n$. Finally $\bar{q}(G)=n$ where $\bar{q}(G)$ is defined to be the smallest number of colours needed to colour the edges of G, so that for each vertex two outgoing edges have the same colour nor two incoming edges have the same colour.

Let $\mathscr{C}_q=\{q_0, \ldots, q_{n-1}\}$ be our set of n colours. The edge-colouring of G yields immediately a good labeling by φ' of the places in π_q: for each $b \in B_q$ and $i<n$, we set $\varphi'(b)=q_i$ if and only if the arc of G labeled b has the colour q_i.

Now each pre- and post-set of a transition in π_q is labeled by a set of $\mathscr{C}_q$ (and not by a multiset of $\mathscr{C}_q$). Before labeling the transitions of π_q by φ', let us have a look at the places in π_q which are in $^\circ\pi$, *i.e.* those corresponding to the initial marking. The labels of these places form an arbitrary multiset of $\mathscr{C}_q$ of maximal size k. We cannot describe it more precisely, it depends on the particular process π and on the colouring obtained for the associated graph. This is the reason which made it impossible to fix uniformely a particular M'_0 on the copies $\mathscr{C}_q$ of q (except for $k=0$); we should take all possible distributions of tokens to $\mathscr{C}^q$. But this solution signifies a set of initial markings instead of a single one, which is out of the scope of normalized nets. These considerations motivated the creation of the "initial component" of N'.

Thus we have to change the labels of $°\pi \cap \pi_q$ into $\mathscr{I}_q$ such that φ' is injective on $°\pi \cap \pi_q$. We can do this in an arbitrary way, e.g. in labeling from top to bottom on the graphical representation of the net $°\pi \cap \pi_q$ by $q'_0, q'_1, q'_2, \ldots$

Now the labeling of B_q and the enumeration of subsets of $\mathscr{C}_q$ (resp. $\mathscr{I}_q$) chosen above entirely determine the labeling of E_q:

$$\begin{aligned} \varphi'(e) &= t_{i,j} && \text{iff } \varphi'(°e) = D_i \text{ and } \varphi'(e°) = E_j, \\ \varphi'(e) &= t'_{i,j} && \text{iff } \varphi'(°e) = F_i \text{ and } \varphi'(e°) = E_j, \\ \varphi'(e) &= t'_{l,i,j} && \text{iff } \varphi'(°e) = G_i \cup F_i \text{ and } \varphi'(e°) = E_j. \end{aligned}$$

This completes the renaming of π. The occurrence net π' is totally defined and is clearly a process of N'. Its associated partial word is the same as the partial word associated with π because $h' \circ \varphi' = h \circ \varphi$.

END OF THE PROOF OF LEMMA A

In order to prove lemma B, where the number of places containing more than one token in the initial or in one the final markings will be decreased inductively, we first present the transformation which will be applied at each step of the induction.

Transformation B

Input: ***A P/T-system*** $N=(P, T, A, v, e, M_0, F)$ ***satisfying property* (P1) *and a place*** $q \in R(N)$ where $R(N) \subseteq P$ is the set of places p for which there is a marking $M \in \{M_0\} \cup F$ satisfying $M(q) > 1$.

Output: ***A P/T-system*** $N''=(P'', T'', A'', v'', e'', M''_0, F'')$ ***satisfying property* (P1),** $R(N'')=R(N) \setminus \{q\}$ ***and having the same partial language as*** N.

ALGORITHM: Let $n = \max\{M(q) \mid M \in \{M_0\} \cup F\}$.

We replace q by n places $q_0, \ldots, q_{n-1}$ and each transition $t \in °q \cup q°$ by n new transitions $t_0, \ldots, t_{n-1}$ having the same label as t and such that for each $i<n$ and for each $t \in {}^{\circ}q \cup q^{\circ}$:

(i) the arcs between t_i and the old places are the same as those between t and these places;

(ii) the arcs between q_i and t_i are the same as those between q and t.

We define the new initial marking M_0'' by $M_0''|_{P \setminus \{q\}} = M_0$, $M_0''(q_i) = 1$ for $i < M_0(q)$ and $M_0''(q_i) = 0$ for $M_0(q) \leqq i < n$.

Each final marking $M_f \in F$ will be replaced by a marking M_f'' satisfying $M_f''|_{P \setminus \{q\}} = M_f$, $M_f''(q_i) = 1$ for $i < M_f(q)$ and $M_f''(q_i) = 0$ for $M_f(q \leqq i < n$. The set of these M_f'' will be F''.

end of transformation B

Example: Figure 4 illustrates transformation B.

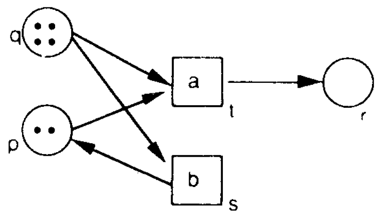


Figure 4 *a*

We show the part of a net N which is incident to transitions of q° with $M_0(p) = 2$, $M_0(q) = 4$ (*fig.* 4 *a*).

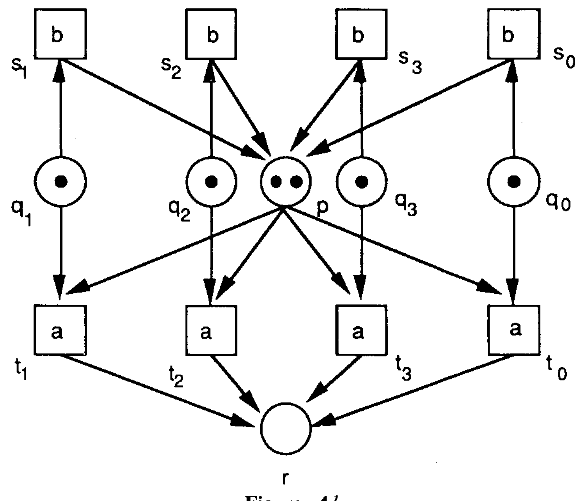


Figure 4 *b*

The application of transformation B to N and the place q gives the subnet of figure 4 *b* from that in figure 4 *a* (with $M_0''(p)=2$, $M_0''(q_i)=1$, for all $i<4$) (*fig.* 4 *b*).

Proof of lemma B: Let $N=(P, T, A, v, e, M_0, F)$ be a P/T-system satisfying property (P1) and $R(N) \subseteq P$ the set of places p for which there is a marking $M \in \{M_0\} \cup F$ satisfying $M(q)>1$.

By induction on the cardinality of $R(N)$ we only have to show how to decrease this number by one. The application of the transformation B to N and to a place $q \in R(N)$, yields a P/T-system $N''=(P'', T'', A'', v'', e'', M_0'', F'')$ satisfying $R(N'')=R(N) \setminus \{q\}$. Now we only have to prove that N'' still satisfies property (P1) and that N'' has the same partial language as N.

The preservation of property (P1) is trivially true: only arcs with multiplicity one have been added to N.

In order to shown that N and N'' have the same partial languages we shall prove that they have the "same" processes – "up to a simple renaming".

We begin with the renaming in the easy direction:

Let $\pi''=(B, E, F, \varphi'')$ be a process of N''. We shall associate with π'' a process $\pi=(B, E, F, \varphi)$ of N where only the labeling changes in a straightforward way:

For each place $b \in B$, if $\varphi''(b) \in \{q_0, \ldots, q_{n-1}\}$ we take $\varphi(b)=q$, otherwise we take $\varphi(b)=\varphi''(b)$. For each transition $e \in E$: if $\varphi''(e) \in \{t_0, \ldots, t_{n-1}\}$ for some t such that $t \in {}^\circ q \cup q^\circ$ we take $\varphi(e)=t$, otherwise we take $\varphi(e)=\varphi''(e)$.

π is clearly a process of N. The partial words associated with π and π'' are the same because $h'' \circ \varphi'' = h \circ \varphi$.

Now let us give the proof for the other direction:

Assume that $\pi=(B, E, F, \varphi)$ is a process of N and let $M \in F$ be the final marking corresponding to π°. We shall define $\pi''=(B, E, F, \varphi'')$ as a process of N''.

As in the proof of lemma A, let $\pi_q=(B_q, E_q, F_q)$ be the subnet of π restricted to places of B labeled by q and to transitions of E labeled by some $t \in {}^\circ q \cup q^\circ$. The labels on $\pi \setminus \pi_q$ will remain the same in π''. Thus we only have to rename π_q.

The situation is much simpler than in the above proof, because N satisfies property (P1): the pre- and post-sets of transitions in π_q are empty or are singletons. This implies that all of the connected components of π_q are chains or "lines". We also know that there are exactly $M_0(q)$ connected components whose source is a place, and exactly $M(q)$ connected components whose sink is a place; all the other connected components have transitions as source and sink. Moreover, $M_0(q) \leqq n$ and $M(q) \leqq n$. We enumerate the $k \leqq M_0(q)$ components whose source and sink are places: $C_0, \ldots, C_k$; those whose source only is a place: $C'_{k+1}, \ldots, C'_{M_0(q)-1}$; those whose sink only is a place: $C''_{k+1}, \ldots, C''_{M(q)-1}$.

For each i, the labels inside the components with index i (i.e. C_i, C'_i and C''_i) will receive index i. Thus label q becomes q_i and each label t becomes t_i.

Each one of the remaining components may receive an arbitrary index $j<n$; we have only to take care that the indexes inside one component are all the same.

Now π is entirely renamed. The resulting occurrence net π'' is clearly a process of N''. Its associated partial word is the same as that of π because $h'' \circ \varphi'' = h \circ \varphi$.

END OF THE PROOF OF LEMMA B

3. COMPLEXITY

This paper presents a structural result on net classes: It is important to know that P/T-systems can always be considered as normalized ones when dealing with behavioural aspects. It may be interesting to know the complexity of our transformations, even if it is too large for the transformation algorithms to be of practical use.

Let us evaluate the complexity of the whole transformation from a general P/T-system N into a normalized P/T-system N''.

The *worst case* is that of a very particular P/T-system

$$N=(P, T, A, v, M_0, F)$$

1. having r places and s transitions (*i.e.* $|P|=r$ and $|T|=s$),

2. which is a complete bipartite multigraph (*i.e.* $A=(P\times T)\cup(T\times P)$), and such that

3. for each place $p\in P$, the value of exactly one of its incoming or outgoing arcs is n and that of all the other adjacent arcs is $n/2$,

4. all places are initially marked by n tokens.

We justify now the last two conditions:

Condition 3 comes from the fact that the duplication of transitions depends on the binomial coefficient. It satisfies $\max_{i\leqq n}\binom{n}{i}=\binom{n}{[n/2]}$.

Note that each place marked in M_0 will be dupliplicated (by factor n) only once: either by Algorithm A if the place is in $Q(N)$, or by Algorithm B otherwise. For each duplicated initially marked place, the duplication-factor for transitions is 2^{n-2} by Algorithm A and $n+1$ by Algorithm B. Condition 4 forces the use of Algorithm A, thus we really have the worst case of possible initial markings.

Also note that for net N as input, the output of Transformation A is already a normalized net. Therefore, we never use Transformation B.

The normalized net N'', which we obtain, is such that
the number of transitions is

$$|T''| \leqq s \, . \, 2^{r(n-1)}$$

and the number of places is

$$|P''| \leqq r(2n+1).$$

Taking $s=r=n$, we finally obtain:

$$|T''| = O(2^{n^2}) \qquad \text{and} \qquad |P''| = O(n^2).$$

The net N'' as bipartite graph is no longer complete, but still has a lot of arcs: $|A''| = O(2^{2n^2})$.

4. SOME REMARKS ABOUT OTHER EQUIVALENCE NOTIONS

In [13], and more completely in [14] a hierarchy of equivalence notions on the behaviour of 1-safe P/T-systems is established.

We may consider the same notions for general P/T-systems. Before presenting here some of them, let us intriduce the notion of step-behaviour:

Steps are non-empty multisets of transitions which may occur concurrently but which will be considered as appearing at the same moment. Let Step(T) be the *set of multisets over* T, Step(Σ) that over Σ, and Step$(T)^*$, resp. Step$(\Sigma)^*$, the set of finite sequences of them.

We say that the finite *step-sequence* $\bar{u}=u_0 u_1 \ldots u_n$ *may occur at* M and change it to M', if there are markings $M_i (0 \leqq i \leqq n+1)$ satisfying: $M=M_0$, $M'=M_{n+1}$ and for each $i \leqq n$, $M_i (u_i \rangle M_{i+1}$; in this case we write $M(\bar{u}\rangle M'$.

The *step-language of the* P/T-*system* N is defined as follows:

$\text{Step}\,L(N) = \{ W \in \text{Step}(\Sigma)^* \mid \exists\, \bar{u} \in \text{Step}(T), \exists\, M_f \in F \text{ such that } M_0 (\bar{u}\rangle M_f \text{ and } h(\bar{u}) = W \}$.

Now we present Pomello's equivalence notions: two P/T-systems N and N' will be called

S-equivalent (noted $N \approx^S N'$) iff $L(N) = L(N')$,

CS-equivalent (noted $N \approx^{CS} N'$) iff $\text{Step}\, L(N) = \text{Step}\, L(N')$,

TCS-equivalent (noted $N \approx^{TCS} N'$) iff $\mathscr{PWL}(N) = \mathscr{PWL}(N')$.

In the three notations *"S"* stands for "string" *"C"* stands for its generalisation to "concurrency" in terms of step semantics and *"TC"* for its generalisation to "true concurrency" in terms of partial words semantics.

In this terminology, we have proved in a previous chapter that each P/T-system N is TCS-equivalent to its normalized net N'' obtained by out transformations.

As $\approx^{TCS}$ is stronger than $\approx^{CS}$, which is stronger than $\approx^S$ [14], our transformation also preserves step-languages and finitary sequential languages.

If we want to consider the different infinitary languages which can be defined for a P/T-system, we have to make some small changes in the transformations A and B: instead of the final marking F we need to treat the set of anchor marking sets $\mathscr{F}$. For transformation B, $R(N)$ will be redefined as set of places p for which there exists at least one marking $M \in \{M_0\} \cup \{M' \mid \exists F \in \mathscr{F}\ M' \in F\}$ such that $M(p) > 1$.

Let us define the notion of *well-distributed marking* with respect to the new places $q_0, \ldots, q_{n-1}$ which replace q in transformation A (resp. B): A marking M' is said to be weel-distributed if there are $m, m' \in \mathbb{N}$, $m' < n$ such that each one of the places $q_o, \ldots, q_{m'-1}$ contains $m+1$ tokens and each one of $q_{m'}, \ldots, q_{n-1}$ contains m tokens. Moreover, to each marking M of the original net there corresponds exactly one well-distributed marking M' in the transformed net satisfying $M'|_{P \setminus \{q\}} = M$ and $\sum_{i<n} M'(q_i) = M(q)$.

In transformations A and B each marking in $\{M \mid \exists F \in \mathscr{F}\ M \in F\}$ will be replaced by the corresponding well-distributed marking M'. The new set of anchor marking sets will be $\mathscr{F}' = \{\{M' \mid M \in F\} \mid F \in \mathscr{F}\}$.

We extend our result to the *infinite sequential behaviour* of P/T-systems as defined in [17], by certain conditions on a set of marking sets $\mathscr{F} \subseteq \mathscr{P}(\mathbb{N}^r)$, called anchor marking sets. We recall these definitions only briefly.

Let N be a P/T-system having $\mathscr{F} \subseteq \mathscr{P}(\mathbb{N}^r)$ as anchor marking sets.

As in [17], the infinite firing sequence $\bar{t}$ is called *i-successful*, if a certain relationship holds between $\mathscr{F}$ and the set of all markings appearing an infinite number of times during the firing of t and which is different for each $i=1, 1', 2, 2', 3, 3'$.

The *i-behaviour of N* (for $i=1, 1', 2, 2', 3, 3'$) is defined by

$$L^i_\omega(N)=\{ w\in\Sigma^\omega \mid \exists\, \bar{t}\in T^\omega \text{ such that } M_0(\bar{t}\rangle,\ h(\bar{t})=w \text{ and } \bar{t} \text{ is } i\text{-successful}\}.$$

The *set of infinite labeled firing sequences of N* is defined by

$$L_\omega(N)=\{ w\in\Sigma^\omega \mid \exists\, \bar{t}\in T^\omega \text{ such that } M_0(\bar{t}> \text{ and } h(\bar{t})=w\}.$$

In order to prove that the infinitary languages are also preserved by the transformations, we consider infinite firing sequences $\bar{t}$ with their associated intermediate markings instead of processes π. The remaining of transitions originally labeled by $t\in{}^\circ q\cup q^\circ$ by corresponding $t_{i,j}$ now becomes trivial. This renaming φ' will ensure that the sequence of markings appearing during the firing of sequence $\bar{t}'$ is the sequence of well-distributed markings corresponding to the sequence of markings of the original firing sequence $\bar{t}$.

Moreover $h\circ\varphi=h'\circ\varphi'$ (resp. $=h''\circ\varphi''$) remains true as above; thus $L_\omega(N)=L_\omega(N')$ (resp. $=L_\omega(N'')$).

By the definition of $\mathscr{F}'$ and φ' (resp. φ''), we observe for each $i=1, 1', 2, 2', 3, 3'$, that the firing sequence $\bar{t}$ of N is i-successful iff the firing sequence $\bar{t}'$ of the transformed net is i-successful. Therefore the i-behaviours of N and N' (resp. of N and N'') are the same, *i.e.* $L^i_\omega(N)=L^i_\omega(N'')$ (resp. $=L^i_\omega(N'')$).

For all the semantics considered up until now, our transformations preserve the behaviour. We can summarize these results in a corollary to the main theorem:

COROLLARY: *Let N be a general P/T-system (having a set F and a set $\mathscr{F}$ specified). There is an normalized P/T-system N'' such that*

$$\mathscr{PWL}(N)=\mathscr{PWL}(N''),\qquad \text{Step}\, L(N)=\text{Step}\, L(N''),\qquad L(N)=L(N''),$$
$$L_\omega(N)=L_\omega(N'') \text{ and } L^i_\omega(N)=L^i_\omega(N'') \quad \text{for each } i=1, 1', 2, 2', 3, 3'.$$

There exist stronger notions of behavioural equivalence for true concurrency based on different kinds of observation, such as exhibited behaviour equivalence, true concurrent bisimulation or – the finest of all these relations – fully concurrent bisimulation. They are discussed in the context of nets in [3], [14], [19].

Without going into the details of these notions and their differences, we explain briefly why we cannot hope to obtain a stronger equivalence between a P/T-system N and its normalized version N'' than TCS-equivalence.

In general, there are various non-isomorphic ways to concatenate isomorphic partial orders. For instance, the notion of concurrent bisimulation requires order isomorphic processes leading to corresponding states and that of true concurrent bisimulation requires the possibility of "isomorphic one-event-extensions" of isomorphic processes.

More precisely, we show the following fact:

Fact: A general P/T-system and its normalized version are not concurrently bisimular (neither are they fully concurrently bisimular).

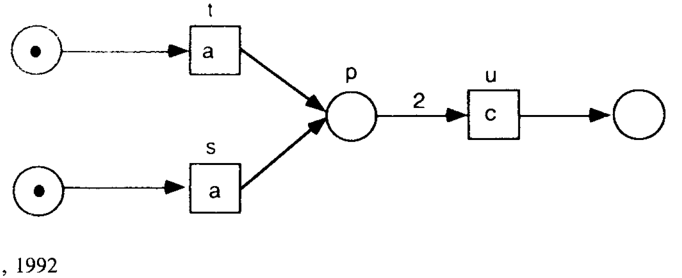
As we have seen in the proof of lemma A, the processes π of N and π' of N' are isomorphic. But the renaming was obtained by a colouring, which is based on successive matchings. In other words, the renaming is global for the whole net and cannot be constructed locally advancing from the input to the output of the net. This implies, in particular, that we have no control over the labels attributed to places in π'° with original label q. Thus if π allows a one-event-extension by a transition $t \in q^{\circ}$, each possibility of taking $f(q, t)$ places labeled q between π° fixes a different concatenation of t to π. All these extensions are possible in π' only if all the corresponding subsets in π'° are labeled by subsets (and not by multisets) of $\mathscr{C}_q$. But normally this will not be the case.

More generally, we have no global control over the labels of intermediate markings obtained during a process (or in other terms, over the labels of its slices). The renaming in transformation A yields a particular set of labels for each slice, ensuring that the whole visible evolution of the process (from the initial marking to a final one) remains possible, but we do not know anything about other possible continuations from any intermediate slice.

The following example will illustrate these remarks:

Example: Consider the nets N and N' of figure 5: N is not normalized and N' is obtained by applying the normalisation algorithm on N.

N:



N':

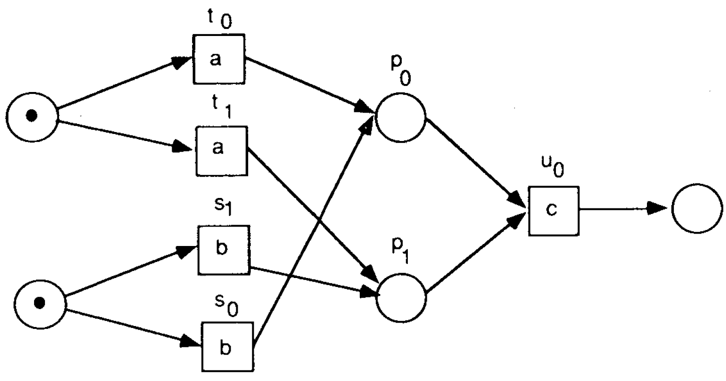
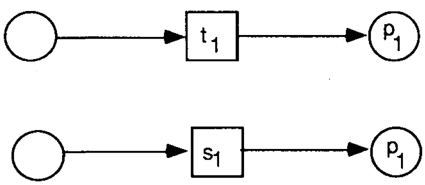


Figure 5

We show that N and N' are not concurrently bisimular (and therefore they are not fully concurrently bisimular):

For the process π' of N', shown in figure 6, there is no process of N such that the visible evolutions would be order isomorphic and lead to corresponding states. In particular, the process π of N can not play this rule as it can be continued by a firing of transition u which is not possible in π'. Figure 6 illustrates this fact.

π':



π:

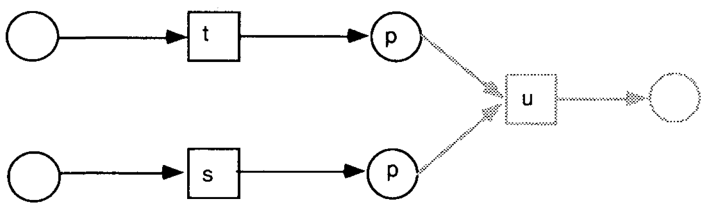


Figure 6

We conclude that TCS-equivalence is the strongest one which holds for the pair (N, N').

For research on formal language related issues of P/T-system behaviour, our result is really strong enough. As stated in the Corollary, all languages usually considered, *i.e.* finitary and infinitary sequential ones, step- and partial word languages, are preserved by the transformation of a λ-labeled P/T-system into a normalized one. Thus we can always restrict ourselves to normalized P/T-systems when working on net behaviour in terms of formal languages, of sequences of transitions, sequences of steps or partial words, without loss of generality. This consequence may facilitate further investigations. Let us quote for instance the definability of partial word languages [11]: the proofs became considerably easier when considering normalized P/T-systems instead of general ones.

ACKNOWLEDGEMENT

I would like to thank Eike Best for his remarks on a previous version of this paper and the members of the research group on graph theory of LRI for our discussions on colourings. My thanks go also to the anonymous referees for some helpful remarks.

REFERENCES

1. C. Berge, Graphs and Hypergraphs, *North Holland*, 1973.
2. E. Best and C. Fernandez, Notations and Terminology on Petri Net Theory, *Arbeitspapiere der GMD* 195, Bonn, 1986.
3. E. Best, R. Devillers, A. Kiehn and L. Pomello, Fully Concurrent Bisimulation Technical report, Université libre de Bruxelles, 1989, and *Acta Informatica*, 1991, *28*, pp. 231-264.
4. G.W. Brams, Réseaux de Petri: Théorie et Pratique, Tome 1 et 2, *Masson*, 1982.
5. W. Brauer, W. Reisig and G. Rozenberg Eds., Petri Nets: Central Models and Their Properties, Part I, *Proceedings*, Advances in Petri Nets 1986, LNCS, *254*, *Springer-Verlag*, 1987.
6. U. Goltz, On Condition/Event Representations of Place/Transition Systems, in: K. Voss, H.J. Genrich and G. Rozenberg Eds, Concurrency and Nets, *Springer-Verlag*, 1987.
7. U. Goltz and W. Reisig, Processes of Place/Transition-Nets in *Proceedings ICALP* 83, LNCS, *154*, *Springer-Verlag*, 1983.
8. J. Grabowski, On partial languages, *Annales Soc. Math. Polonae*, series IV: *Fund. Inform.*, 1981, *IV. 2*, pp. 427-298.
9. M. Hack, Petri net Languages, MIT-Technical Report 159, *MIT Press*, Cambridge, 1975.
10. M. Parigot and E. Pelz, A logical approach of Petri net languages, *T.C.S.*, 1985, *39*, pp. 155-169, and to appear in *Proceedings ESDA'92*, ASME-Volume.

11. E. PELZ, Place/Transitions-systems: Concurrent behaviour and logic, *Rapport* LRI No. 571, Université Paris-Sud, 1990.
12. E. PELZ, Normalisation of Place/Transition-Systems Preserves Net Behaviour, in part II of "Réseaux et Logique", *Thèse d'État*, Université Paris-Sud, 1990.
13. L. POMELLO, Some Equivalence Notions for Concurrent Systems. An Overwiew, in K. VOSS, H.J. GENRICH and G. ROZENBERG Eds, Concurrency and Nets, *Springer-Verlag*, 1987.
14. L. POMELLO and C. SIMONE, A survey of Equivalence Notions for Net Based Systems Systems, *Conference* at ESPRIT-BRA DEMON meeting, Bonn, June 1989, to appear in *Advances in Petri Nets*, 1991.
15. S. SCHWER, Transformation of a Petri Net into a Petri Net with Incidence Matrix on $\{0, 1\}$, *Petri Nets Newsletter*, 1986, *23*, pp. 15-20.
16. J. SIFAKIS, Le contrôle des systèmes asynchrones: Concepts, propriétés, analyse statique, *Thèse*, U.S.M. Grenoble, 1979.
17. R. VALK and H. CARSTENSEN, Infinite Behaviour and Fairness in Petri Nets, in *Advances in Petri nets* 1984, LNCS, 188, G. ROZENBERG Ed., *Springer-Verlag*, 1985.
18. W. VOGLER, Executions: Their relations to Other Partial Order Semantics of Nets and a Compatible C/E-Representation of P/T-systems, *Technical report*, TUM-18806, TU München, 1988.
19. W. VOGLER, Bisimulation and action refinement, in STACS 91, C. CHOFFRUT and M. JANTZEN Eds., LNCS, 480, *Springer-Verlag*, 1991.