Chapter 3
Analysis Methods for Petri Nets

Serge Haddad , François Vernadat[1]

1. Introduction

One of the main advantages of formal models is the possibility to unambiguously define the behaviour of a system, to develop algorithms for properties verification and to integrate them in a dedicated software tool.

The firing rule of Petri nets associates a (finite or infinite) reachability graph with a net. This graph constitutes a formal representation of the net behaviour. Thus we will first define the general and more relevant properties of the net w.r.t. this graph (like liveness or deadlock existence). When it is finite, one can scan it in order to check these properties. The methods based on the construction and the exploration of the whole graph or of some part of it are called behavioural methods. In spite of their relative simplicity and their wide applicability, these methods present some drawbacks: they are only applicable to nets with a finite number of states, their (temporal and spatial) complexity depends on the size of the graph (much bigger than the size of the net) and they require the knowledge the initial marking.

In the next section, we will examine families of alternative methods that take advantage of the net structure in order either to decrease the complexity of the analysis or that apply on a net independently of its initial marking. We wil deepy describe three of these methods called structural methods.

[1]LSV, ENS Cachan, 61, avenue du Président Wilson 94235 Cachan Cedex France, (haddad@lsv.ens-cachan.fr)
LAAS-CNRS, 7, Avenue du Colonel Roche F-31077 Toulouse cedex, (francois@laas.fr)

The state change equation corresponds to the fact that the update of a marking by a firing sequence is exactly the product of the incidence matrix by the vector of transition occurrences in this sequence. By adapting linear algebra techniques, one computes generative families of linear invariants over places or transitions. In the case of places, an invariant is a weighted sum of place markings invariant by transition firing. In the case of transitions, an invariant is an occurrence vector of a firing sequence which do not modify the marking. In addition to this interpretation, this computation has numerous applications, some of them will be detailled in the book.

The reduction technique consists in substituting to a net a smaller one whose behaviour is equivalent w.r.t. a set of relevant properties. A reduction is defined by structural conditions and a transformation method. This technique should be applied before any other one, thus decreasing the subsequent computational complexity.

Since the net is a bipartite graph, its analysis provides interesting informations on the behaviour of the net. Furthermore in the framework of specific modelisations (e.g. manufacturing systems) the structure of the resulting graph is particular and one can associate with a behavioural property a structural characterization. We will briefly describe some of these models and more particularly the free choice Petri nets for which numerous efficient verification algorithms have been designed.

Throughout this chapter, we do not look for an exhaustive overview of the analysis methods. For instance, we skip the analysis by net decomposition ; some of these methods will be illustrated in the chapters devoted to stochastic Petri nets. We skip also the methods that take advantage of the structure of the net in order to build smaller representations of the reachability graph : they will be illustrated in different chapters of the book. Finally the verification methods for unbounded nets are described in the next chapter.

We have chosen to present the proofs of the propositions whenever their size remains reasonnable. It seems that only these descriptions will highlight the reader on the foundations of a method. We restrict the presentation of results without proofs to central ones. The references should enable the reader to access more involved theories in specialized books or in research communications.

General notations

1 Sets and Numbers

- $\mathbb{N}$ is the set of natural integers, $\mathbb{Z}$ is the set of relative integers, $\mathbb{Q}$ est the set of rationals and $\mathbb{R}$ is the set of reals.
- Let X be a set of numbers, X^+ denotes the subset of X restricted to non negative items. $Max(X)$ denotes its smallest upper bound

(possibly ∞) and $Min(X)$ denotes its greatest lower bound (possibly $-\infty$).

- Let E be a set, $|E|$ denotes its cardinality.

2 Vectors and Matrices

- Let E be a set, a vector v with *dimension* E and natural integer coefficients is an application from E to $\mathbb{N}$. For $e \in E$, $v(e)$ denotes the e-component of this vector. The set of vectors is denoted $\mathbb{N}^E$. This notion can be generalized to every set of numbers ($\mathbb{Z}^E$, $(\mathbb{Q}^+)^E$,...). This notion is also applicable to matrices. For instance, the incidence matrice $C \in \mathbb{Z}^{P\times T}$.
- Let $e \in E$, the vecteur $\overrightarrow{e}$ de $\mathbb{N}^E$ is defined by $\overrightarrow{e}(e) = 1$ and $\overrightarrow{e}(e') = 0$ for $e' \neq e$.
- $\overrightarrow{0}$ denotes the null vector whose dimension is fixed by the context.
- Let $(E, <)$ be a totally ordered set, the (total) lexicographical order on X^E (where X is a set of numbers) is defined by:
$$v \prec v' \Leftrightarrow \exists\, e \in E,\ v(e) < v'(e) \text{ and } \forall\, e' < e,\ v(e') = v'(e')$$
- Let A be a matrix with dimension $E \times F$, then A^t is the transposed matrix with dimension $F \times E$ defined by $A^t(i,j) = A(j,i)$. When $E \cap F = \emptyset$, we denote for $e \in E$ (resp. $f \in F$), $A(e)$ (resp. $A(f)$) the row (resp. column) vector of A indexed by e (resp. f). We mainly apply this notation to matrices $Pr\acute{e}, Post$ et C.
- A vector with dimension E can also be viewed like a matrix with dimension $E \times \{1\}$. So the transposition equally applies to vectors.
- Let v be a vector with dimension E, one defines the support of v, denoted $\|v\|$ by: $\|v\| = \{e \in E \mid v(e) \neq 0\}$.
- Let v_1, v_2 be two vectors with same dimension one denotes $v_1 \leq v_2$ *iff* $\forall e,\ v_1(e) \leq v_2(e)$ and $v_1 < v_2$ *iff* ($v_1 \leq v_2$ et $v_1 \neq v_2$). $Sup(v_1, v_2)$ denotes the vector defined by $Sup(v_1, v_2)(e) = Max(v_1(e), v_2(e))$

3 Sequences and Languages

- Let Σ be an alphabet (i.e. a finite set), Σ^* denotes the set of finite words of Σ and Σ^∞ denotes the set of infinite words of Σ.
- Let $\sigma \in \Sigma^*$ and $\sigma' \in \Sigma^* \cup \Sigma^\infty$, $\sigma.\sigma'$ denotes the concatenation of the two words.
- Let $\sigma \in \Sigma^*$, σ^∞ denotes the word (infinite except if σ is the empty word denoted λ) obtained by infinite repetition of σ.
- Let Σ' be a subalphabet of Σ and σ be a word of Σ, the projection of σ on Σ' denoted $\sigma_{\lfloor\Sigma'}$ is recursively defined by:
$$\lambda_{\lfloor\Sigma'} = \lambda \text{ and } (\sigma.a)_{\lfloor\Sigma'} = If\ a \in \Sigma'\ then\ \sigma_{\lfloor\Sigma'}.a\ else\ \sigma_{\lfloor\Sigma'}$$

- Let σ be a word, then $\widetilde{\sigma}$ is the inverse word recursively defined by:
$$\widetilde{\lambda} = \lambda \text{ and } \widetilde{(\sigma.a)} = a.\widetilde{\sigma}$$
- We denote $m \xrightarrow{\sigma}_{P'} m'$ a firing sequence when one restricts the fireability condition to the subset of places P'. When one wants to precise the net R of a firing sequence, one denotes it by: $m \xrightarrow{\sigma}_{R} m'$. The notations can be combined: $m \xrightarrow{\sigma}_{R,P'} m'$.

4 Nets

- Let s be an item of $P \cup T$ then ${}^\bullet s$ denotes the set of predecessors of s in the net and $s^\bullet$ denotes the set of successors of s in the net. Otherwise stated, if s is a transition then ${}^\bullet s = \|Pré(s)\|$ and $s^\bullet = \|Post(s)\|$ and if s is a place then ${}^\bullet s = \|Post(s)\|$ and $s^\bullet = \|Pré(s)\|$.
- This notation is extended to subsets of vertices: ${}^\bullet S = \{t \mid \exists\, s \in S\ t \in {}^\bullet s\}$ and $S^\bullet = \{t \mid \exists\, s \in S\ t \in s^\bullet\}$
- We slightly abuse the language: a Petri net denotes both the structure R and the marked net (R, m_0). The context will allow to deduce which object is denoted.
- In figures representing Petri nets, the double arrow represents superimposed arcs *Pré* and *Post*.

We assume that the reader already knows some basics of graph theory [1, 12]. The main notions that we will discuss are connectivity, strong connectivity, (initial, terminal) strongly connected components, paths, elementary circuits and trees.

At last the theoretical complexity of methods will be discussed anticipating the next chapter where the basics of complexity are presented. The reader can refer to it when necessary.

2. Behavioural analysis of Petri nets

2.1. *Semantics of a net*

The simplest way to define the behaviour of a net is to considere the set of markings reachable from the initial marking.

Definition 1 (Reachability set) *Let (R, m_0) be a Petri net, the reachability set of the net denoted $A(R, m_0)$ is the set of markings reached by a firing sequence:*
$$A(R, m_0) = \{m \mid \exists\, \sigma \in T^* \ t.q.\ m_0 \xrightarrow{\sigma} m\}$$

A more complete way consists to take into account the immediate reachability relation between markings throughout the reachability graph.

Definition 2 (Reachability graph) *Let (R, m_0) be a Petri net, the (directed) reachability graph of the net denoted by $G(R, m_0)$ is defined by:*

- *the set of vertices $A(R, m_0)$*
- *the set of arcs: an arc lebelled by t joins m to m' iff $m \xrightarrow{t} m'$*

If the observation of events is more important than the internal state of the system (represented by the marking) then the language of firing sequences is more appropriate. Often, different transitions model the same event or a transition models an internal action. It is then judicious to introduce a labelling of transitions.

Definition 3 (Language of a net) *Let (R, m_0) be a Petri net, Σ be an alphabet and l be a labelling mapping from T to $\Sigma \cup \lambda$ (the empty word). The labelling is extended to sequences by $l(\lambda) = \lambda$ and $l(\sigma.t) = l(\sigma).l(t)$. Let $Term$ be a finite set of final markings. The language of the net denoted $L(R, m_0, l, Term)$ is defined by:*

$$L(R, m_0, l, Term) = \{w \in \Sigma^* \mid \exists \sigma \in T^*, \exists m_f \in Term,\ m_0 \xrightarrow{\sigma} m_f \wedge w = l(\sigma)\}$$

Other definitions for Petri net languages are possible. For instance, the kind of labelling can be restricted or the final markings can be omitted.

2.2. *Usual properties*

2.2.1. *Properties definition*

The interest of a model is the possibility to formally define properties of the modelled system and to check these properties by algorithms or heuristics. In case of Petri nets, the usual properties are related to the activity of a parallel system. These properties can be specific to the parallelism or simply related to dynamicity.

We illustrate these properties on the net of figure 1. This net models two anonymous processes initially in state $Idle$. Any process may choose between two behaviours: either get the resource A (modelled by $PickA$), and then the resource B and finish ($PickAB$ is an abstraction of these two events) or get the resources in the reverse order.

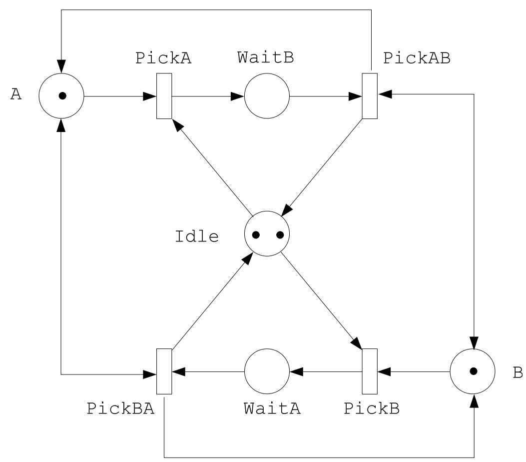

Figure 1: Two processes sharing two resources

The first issue about such a system is whether its behaviour is finite. Otherwise stated, we are looking for an infinite firing sequence.

Definition 4 (Existence of an infinite sequence)
A Petri net (R, m_0) *admits* an infinite sequence $\sigma \in T^\infty$ *if for every* σ' *finite prefix of* σ, σ' *is a firing sequence of* (R, m_0).

Example 1 $(PickA.PickAB)^\infty$ is an infinite sequence of the net of figure 1.

When a net has no infinite sequence, one says that it fulfills the termination property.

An interesting issue is to determine whether the system never stops. For instance, an operating system must never stop whatever the behaviour of its users. Otherwise stated, from any reachable marking one can fire at least one transition.

Definition 5 (Pseudo-liveness)
A Petri net (R, m_0) *is* pseudo-live *if:*

$$\forall\, m \in A(R, m_0)\, \exists\, t \in T \text{ s.t. } m \xrightarrow{t}$$

When a marking has no fireable transition, one says that it is a *dead* marking.

Example 2 The sequence $PickA.PickB$ leads to the dead marking $\overrightarrow{WaitA}+\overrightarrow{WaitB}$.

A frequent error of modelling is to design a net with a transition which is never fireable. It is then important to eliminate such errors.

Definition 6 (Quasi-liveness)
A Petri net (R, m_0) is quasi-live *if:*

$$\forall t \in T\, \exists m \in A(R, m_0)\ s.t.\ m \xrightarrow{t}$$

Example 3 Starting from the initial marking, one can fire the sequence:
$$PickA.PickAB.PickB.PickBA$$
where every transition occurs.

The two previous properties ensure some correctness of the system but they cannot ensure that in every reachable marking, the system keeps all its functionalities. Otherwise stated, one wanders whether every transition can be fired in some future of every state.

Definition 7 (Liveness)
A Petri net (R, m_0) is live *if for every marking $m \in A(R, m_0)$, the net (R, m) is quasi-live. Otherwise stated:*

$$\forall m \in A(R, m_0)\, \forall t \in T\, \exists m' \in A(R, m)\ s.t.\ m' \xrightarrow{t}$$

Example 4 From the dead marking $\overrightarrow{WaitA}+\overrightarrow{WaitB}$, no transition is fireable. Hence the net is not live.

Another interesting property is the possibility to always return to some state corresponding for instance to the reinitialisation of the system. When this marking is the initial one, reinitialisation is identical to initialisation.

Definition 8 (Existence of a home state)
A Petri net (R, m_0) admits a home state m_a *if:*

$$\forall m \in A(R, m_0), \exists \sigma \in T^*\ s.t.\ m \xrightarrow{\sigma} m_a$$

Example 5 From any reachable marking, the dead marking is reachable. Hence the net admits a home state.

Modelling open systems is somewhat different from modelling closed systems. For instance, it may require to model the arrival of a unbounded number of clients leading to the following definition.

Definition 9 (Boundedness of a net)
A Petri net (R, m_0) *is* unbounded *if:*

$$\forall n \in \mathbb{N}, \exists m \in A(R, m_0), \exists p \in P \; t.q. \; m(p) > n$$

R *is* structurally bounded *if it is bounded for every initial marking.*

Example 6 Places contain either resources or processes. Hence the net is bounded.

If the net is unbounded, at least one place may contain a number of tokens as great as possible. Such places are said *unbounded*. If the net is bounded, *a bound of the net* is an integer greater or equal than any possible marking of a place. As will be seen during the study of monotonicity, it is often interesting to modify the initial marking in order to analyze its impact on the behaviour. This explains the interest of structural boundedness.

2.2.2. *Relations between properties*

We establish now simple relations between the different properties.

Proposition 10 *If* (R, m_0) *is pseudo-live or unbounded then* (R, m_0) *admits an infinite sequence.*

Proof
If the net is pseudo-live then one builds the infinite sequence by iteratively firing any transition (there is always at least one). The second part of the proposition will be proved with the help of characterisations of properties by the existence of particular sequences. $\diamondsuit$

Proposition 11 *If* (R, m_0) *is live then* (R, m_0) *is quasi-live and pseudo-live.*

Proof
Assume that the net is live, $m_0 \in A(R, m_0)$ so by definition (R, m_0) is quasi-live. Let t be a transition of T and $m \in A(R, m_0)$, by definition (R, m) is quasi-live, so $\exists\, \sigma \in T^*\ m \xrightarrow{\sigma.t}$. Hence m is not dead. Consequently the net is pseudo-live. $\diamondsuit$

Proposition 12 *If (R, m_0) is quasi-live and admits m_0 as home state then (R, m_0) is live.*

Proof
In order to fire a transition t from a reachable marking, one first returns to m_0 (home state) and then one fires a sequence ended by t (quasi-liveness). $\diamondsuit$

2.2.3. *Monotonicity of properties*

During a modelling, once the structure of the net is defined, the designer modifies the initial marking in order to examine different hypotheses. Often this modification consists in adding tokens in places. So it is interesting to determine whether a property remains fulfilled in the new marked net.

Definition 13 *Let π be a property of Petri nets, π is said monotonic iff:*

$$\forall R\ \forall\, m_0 \leq m'_0,\ \pi \text{ is fulfilled by } (R, m_0) \Rightarrow \pi \text{ is fulfilled by } (R, m'_0)$$

The next lemma justifies the study of monotonicity.

Lemma 14 (Lemma of monotonicity) *Let R be a Petri net,*

- $\forall\, m_1 \leq m'_1\ m_1 \xrightarrow{\sigma} m_2 \Rightarrow\ m'_1 \xrightarrow{\sigma} m'_2$ *with* $m_2 \leq m'_2$
- *Furthermore if there is a place p, $m_1(p) < m'_1(p)$ then $m_2(p) < m'_2(p)$*

Proof
The resultat is obtained by a straightforward recurrence, starting from the case where the sequence σ is reduced to a single transition. In case of a single transition, it is a simple consequence of the firing rule. $\diamondsuit$

Let us examine among the previously defined properties which ones are monotonic.

Proposition 15 *Let (R, m_0) be a Petri net:*

- *"(R, m_0) admits an infinite sequence" is a monotonic property.*
- *"(R, m_0) is pseudo-live" is not a monotonic property.*
- *"(R, m_0) is quasi-live" is a monotonic property.*
- *"(R, m_0) is live" is not a monotonic property.*
- *"(R, m_0) admits a home state" is not a monotonic property.*
- *"(R, m_0) is unbounded" is a monotonic property.*

The properties which are characterized by the existence of firing sequences starting from the initial state are monotonic. For the other ones, we can exhibit elementary counter-examples.

Example 7 The net of figure 1 is live for the initial marking $\overrightarrow{Idle} + \overrightarrow{A} + \overrightarrow{B}$ lower than the original initial marking for which the net is not even pseudo-live.

2.2.4. *Characterization of properties with the help of a finite reachability graph*

The easiest way to check the properties consists in examining the reachability graph whenever it is finite. Hence our first characterizations rely on this graph.

We will illustrate these characterizations on the net of figure 1 whose reachability graph is presented in figure 2.

Proposition 16 *Let (R, m_0) be a Petri net, (R, m_0) is bounded iff $A(R, m_0)$ is finite.*

Proof
Assume (R, m_0) bounded and let n be a bound, then $A(R, m_0)$ is included in $\{m \,|\, m \leq \sum_{p \in P} n.\overrightarrow{p}\}$. Now this set is finite. Assume $A(R, m_0)$ finite, then $Max(\{m(p) \,|\, p \in P \text{ } et \text{ } m \in A(R, m_0)\})$ is finite and constitutes a bound of the net. $\diamondsuit$

In the remainder of the paragraph, we will precise whether the characterization depends on the finiteness of $A(R, m_0)$.

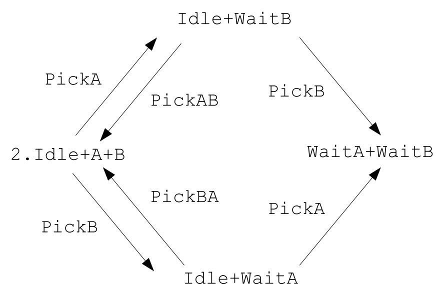

Figure 2: A reachability graph

Proposition 17 *Let (R, m_0) be a bounded Petri net, (R, m_0) admits an infinite sequence iff $G(R, m_0)$ admits a circuit.*

Proof
Assume that (R, m_0) admits an infinite sequence σ, then this sequence goes through some marking at least twice. Hence $\sigma = \sigma'.\sigma''$, $\sigma' = u.v$ with $m_0 \xrightarrow{u} m \xrightarrow{v} m$. So $m \xrightarrow{v} m$ is a circuit of the graph. Assume that $G(R, m_0)$ admits a circuit $m \xrightarrow{v} m$, m is reachable so there exists u such that $m_0 \xrightarrow{u} m$. Consequently, $u.v^\infty$ is an infinite sequence of (R, m_0). $\diamondsuit$

Example 8 The reachability graph of figure 2 has two elementary circuits, hence the Petri net admits an infinite sequence.

Proposition 18 *Let (R, m_0) be a Petri net, (R, m_0) is pseudo-live iff every vertex of $G(R, m_0)$ admits a successor.*

Proof
(R, m_0) is pseudo-live iff every reachable marking of (R, m_0) enables to fire a transition iff every vertex of $G(R, m_0)$ admits a successor. $\diamondsuit$

Example 9 The reachability graph has a vertex without successor, so the Petri net is not pseudo-live (and not live).

Proposition 19 *Let (R, m_0) be a Petri net, (R, m_0) is quasi-live iff every transition labels an arc of $G(R, m_0)$*

Proof
(R, m_0) is quasi-live iff every transition is fireable from a reachable marking iff every transition labels an arc of $G(R, m_0)$. $\diamondsuit$

Example 10 Every transition occurs on the reachability graph, so the net is quasi-live.

The last two properties are characterized with the help of strongly connected components (*s.c.c.*) of the reachability graph.

Proposition 20 *Let (R, m_0) be a bounded Petri net, (R, m_0) is live iff for every terminal s.c.c. $\mathcal{C}$ of $G(R, m_0)$, every transition labels an arc of $\mathcal{C}$.*

Proof
Assume (R, m_0) live and let m belonging to a terminal s.c.c. $\mathcal{C}$. By definition (R, m) is quasi-live hence every transition labels an arc of $G(R, m)$ which is exactly $\mathcal{C}$ since $\mathcal{C}$ is terminal.

Now let m be any reachable marking there exists a path from m to m' belonging to a terminal s.c.c. $\mathcal{C}$ *(property of finite graphs)*. Since $\mathcal{C}$ is included in $G(R, m)$ every transition labels an arc de $G(R, m)$. So (R, m) est quasi-live. $\diamondsuit$

Proposition 21 *Let (R, m_0) be a bounded Petri net, (R, m_0) admits a home state iff there exists a single terminal s.c.c. of $G(R, m_0)$.*

Proof
Let m be a home state of (R, m_0) and $\mathcal{C}$ its s.c.c. then there exists a path from every reachable m' to m. Let $\mathcal{C}'$ the s.c.c. of m'. If $\mathcal{C}'$ is different from $\mathcal{C}$, then $\mathcal{C}'$ is not terminal. In addition, $\mathcal{C}$ is terminal since one can always return to m.

Let $\mathcal{C}$ be the single terminal s.c.c. of $G(R, m_0)$. For every $m' \in A(R, m_0)$, there exists a path from m' to $\mathcal{C}$. Hence every marking of $\mathcal{C}$ is a home state. $\diamondsuit$

Example 11 There are two s.c.c in the graph, one is initial (including the initial marking) and the other is terminal reduced to the dead marking. So this marking is a home state.

2.2.5. *Characterization of properties with the help of particular finite sequences*

For at least two reasons, one wishes to obtain characterizations that do not rely on the reachability graph. First, these characterizations are only effective when the graph is finite and secondly even in this case the size of the graph may forbid the verification. In this paragraph, we take advantage of some general lemmata that we recall now.

Lemma 22 (Koenig lemma) *Let A be a tree with a finite degree (i.e. every vertex admits a finite number of successors) and with an infinite number of vertices. Then A admits an infinite branch.*

Proof
We exhibit the infinite branch as follows. Starting from the root, one selects one successor of the root whose subtree has an infinite number of vertices. There must be at least one since the number of successors is finite. Iterating this process at the level of the current subtree, one builds an infinite branch. $\diamondsuit$

Lemma 23 (Extraction lemma) *Let $m_0, m_1, \ldots$ be an infinite sequence of vectors of $\mathbb{N}^{\{1,\ldots,k\}}$, then this sequence admits a largely increasing sequence.*

Proof
We prove it by recurrence on k. If $k = 1$, then this is a sequence of natural integers. So one selects as first index of the subsequence the index of one minimal item of the sequence. Then one iterates the process starting from the truncated sequence starting from this item. Assume the result holds for $k - 1$; starting from a sequence of $\mathbb{N}^{\{1,\ldots,k\}}$, one extracts an increasing subsequence on the first $k - 1$ components. Applying the process used for $k = 1$ to the last component of the intermediary subsequence, one obtains the wished subsequence. $\diamondsuit$

We immediately apply these lemmata for characterising two properties.

Proposition 24 *(R, m_0) admits an infinite sequence iff (R, m_0) admits a firing sequence $m_0 \xrightarrow{\sigma_1} m_1 \xrightarrow{\sigma_2} m_2$ with $m_1 \leq m_2$.*

Proof
Assume first that the net admits an infinite sequence and consider the infinite sequence of encountered markings. Using lemma 23, one extracts an increasing subsequence. Let us note m_1 et m_2 the two first items of this sequence, then the finite sequence which reaches m_2 is the one we look for.

In the reverse direction, since $m_1 \leq m_2$, applying the lemma about monotonicity σ_2 may be fired from m_2 leading to a marking $m_3 \geq m_2$. Iterating this process, one obtains the infinite sequence $\sigma_1.\sigma_2^\infty$. $\diamondsuit$

Proposition 25 *(R, m_0) is unbounded iff (R, m_0) admits a firing sequence $m_0 \xrightarrow{\sigma_1} m_1 \xrightarrow{\sigma_2} m_2$ with $m_1 < m_2$.*

Proof
Assume first that the net is unbounded and let us consider an infinite tree built starting from the initial marking and such that one adds a son to a marking if from this marking, one can fire a transition leading to a marking not yet present in the tree. There can be several possible trees, but all have exactly as set of vertices the set of reachable markings. This tree has a finite degree since T is finite so using lemma 22, it contains an infinite branch corresponding to an infinite firing sequence. Using lemma 23, one extracts an increasing subsequence. Let us note m_1 and m_2 the two first items of this sequence; then the finite sequence that reaches m_2 is the one we look for. Indeed $m_2 > m_1$ since all the markings are different in the tree.

In the reverse direction, one remarks that since $m_1 \leq m_2$, σ_2 can be fired from m_2 leading to a marking $m_3 \geq m_2$. Iterating this process, one obtains the infinite sequence $\sigma_1.\sigma_2^\infty$. Now, let p be a place such that $m_1(p) < m_2(p)$ then $m_2(p) < m_3(p)$. Consequently the sequence infinitely increases the number of tokens in p. $\diamondsuit$

These two characterisations straightforwardly establish the proof of the second part of proposition 10. We now introduce some kinds of sequences one meets in the analysis of nets.

Definition 26 (Repetitive sequences) *Let R be a Petri net, σ be a sequence of transitions and let m be a marking such that $m \xrightarrow{\sigma} m'$, then:*

- *If $m \leq m'$, σ is said repetitive*
- *If $m = m'$, σ is said repetitive stationary*
- *If $m < m'$, σ is said repetitive increasing*

This definition does not depend on the choice of m and so it is sound.

3. Analysis of nets by linear invariants

3.1. *Definitions and first applications*

The state change equation that we state below has the following interpretation: **the effect** of a firing sequence is determined by the incidence matrix and the vector of transition occurrences in the sequence.

Definition 27 *Let $\sigma \in T^*$ be a sequence of transitions, its occurrence vector $\overrightarrow{\sigma} \in \mathbb{N}^T$ is defined by: $\overrightarrow{\sigma}(t)$ is the number of occurrences of t in σ.*

Proposition 28 (State change equation) *Let R be a Petri net and let $m \xrightarrow{\sigma} m'$ be a firing sequence then:*

$$m' = m + C.\overrightarrow{\sigma}$$

where C, the incidence matrix, is defined by $C = Post - Pré$

Proof
We prove it by recurrence on the length of the sequence. In case of an empty sequence, the result is immediate. The recurrence step is a consequence of the firing definition. $\diamondsuit$

We are looking for invariant quantities with the help of this equation. So it is related to the cancellers of matrix C.

Definition 29 (Flows of a net) *The different cancellers that we consider are:*

- *A P-flow is a non null vector $v \in \mathbb{Z}^P$ which fulfils $v^t.C = \overrightarrow{0}$*
- *A P-semiflow is a non null vector $v \in \mathbb{N}^P$ which fulfils $v^t.C = \overrightarrow{0}$*
- *A T-flow is a non null vector $v \in \mathbb{Z}^T$ which fulfils $C.v = \overrightarrow{0}$*
- *A T-semiflow is a non null vector $v \in \mathbb{N}^T$ which fulfils $C.v = \overrightarrow{0}$*

A P-flow (resp. a P-semiflow) is a weighted sum of places with integer coefficients (resp. natural integers). A P-flow provides mapping from markings to integers by weighting the place markings and summing them. A T-semiflow could be obtained as the occurrence vector of a transition sequence while a T-flow could be obtained as the difference of two occurrence vectors. This yields the first results. Examples will be given later on.

Proposition 30 *Let R be a Petri net,*

- *let v be a P-flow and $m \xrightarrow{\sigma} m'$ be a firing sequence; then:*

$$v^t.m = v^t.m'$$

- *let v be a T-semiflow and σ be a firing sequence such that $\overrightarrow{\sigma} = v$; then:*

$$m \xrightarrow{\sigma} m' \Rightarrow m = m'$$

  otherwise stated, σ is a repetitive stationary sequence

- *let v be a T-flow and σ_1, σ_2 two transition sequences such that $\overrightarrow{\sigma_1} - \overrightarrow{\sigma_2} = v$; then:*

$$m \xrightarrow{\sigma_1} m' \text{ et } m \xrightarrow{\sigma_2} m'' \Rightarrow m' = m''$$

Proof
These assertions are trivial consequences of the state change equation. For instance, the proof of the first point is the following one: $v^t.m' = v^t.m + v^t.C.\overrightarrow{\sigma} = v^t.m$ ◊

Definition 31 (Linear invariants) *Let (R, m_0) be a marked net, a* linear invariant *denotes the equation:*

$$\forall m \in A(R, m_0),\ v^t.m = v^t.m_0$$

where v is a P-flow.
In case of a P-semiflow, one says that it is a positive invariant.

The positive invariants have numerous applications. For instance, every place belonging to the support of a P-flow v are bounded whatever the initial marking, since $m(p) \leq v(p)^{-1}.v^t.m_0$. Similarly, from an invariant $m(p) + m(q) + \cdots = 1$, one deduces that p and q cannot be simultaneously marked.

More generally, invariants are the basis of numerous necessary and/or sufficient conditions of behavioural properties. In order to develop this point, we introduce two structural properties of a Petri net.

Definition 32 (Conservative nets, consistent nets) *Let R be a Petri net,*

- *R is conservative if there exists a P-semiflow v such that $\|v\| = P$*

- *R is consistent if there exists a T-semiflow v such that $\|v\| = T$*

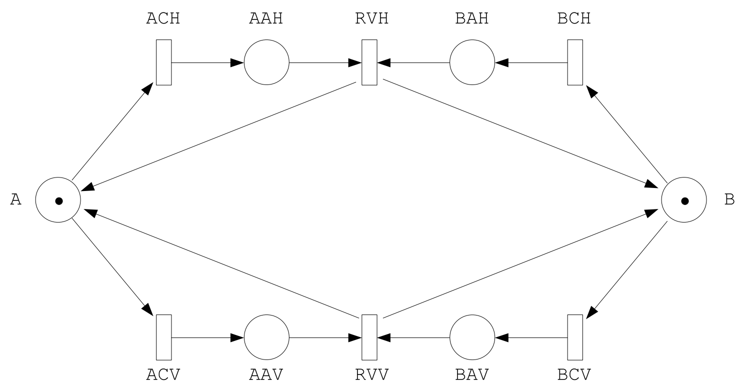

Figure 3: Non deterministic synchronisation of processes

We illustrate this section with the net presented in figure 3. Two processes (A and B) repeatedly execute one of the two local procedures (H or V) then synchronise themselves to exchange their results. The synchronisation is only possible if both processes have chosen the same procedure. Observe that the net is bounded (exactly two tokens in every reachable marking) and not live since different choices lead to a deadlock. We are going to examine information provided by the linear invariants.

We now recall a useful lemma for the analysis of nets by techniques of linear algebra.

Lemma 33 (Duality lemma) *Let p be a place:*

$$\exists v \in \mathbb{N}^P, v^t.C = \overrightarrow{0} \wedge v(p) > 0 \Leftrightarrow \nexists\, w \in \mathbb{Z}^T, C.w \in \mathbb{N}^P \wedge (C.w)(p) > 0$$

Proof
Assume the simultaneous existence of v and w as described in the lemma, then $\forall p' \in P, v(p').(C.w)(p') \geq 0$ et $v(p).(C.w)(p) > 0$. Hence $v^t.C.w > 0$, but $v^t.C.w = \overrightarrow{0}^t.w = 0$, so there is a contradiction. It remains to show that one of this vector always exists.

We prove it by recurrence on $|P|$.

$|P| = 1$**.** Then either C is the null matrix and $v = \overrightarrow{p}$ is appropriate. If C is not null then $\exists t \in T, C.\overrightarrow{t} \neq 0$. If $C.\overrightarrow{t} > 0$ then $w = \overrightarrow{t}$ is appropriate else $w = -\overrightarrow{t}$ is appropriate.

$|P| = n+1$ **and the lemma holds for** $|P| = n$**.** We will try to obtain v or w by two ways.

First attempt. Let p_1 be a place different from p, $P_1 = P \setminus \{p_1\}$ and C_1 be the matrix obtained from C, by deleting the row indexed by p_1. Using the recurrence hypothesis:

- either $\exists v_1 \in \mathbb{N}^{P_1}, v_1^t.C_1 = 0 \wedge v_1(p) > 0$. In this case, v defined by $\forall p' \in P_1, v(p') = v_1(p') \wedge v(p_1) = 0$ is appropriate.
- or $\exists w \in \mathbb{Z}^T, C_1.w \in \mathbb{N}^P \wedge (C_1.w)(p) > 0$. If $C.w(p_1) \geq 0$ then w is appropriate. It remains the unfavourable case where $C.w(p_1) < 0$.

Second attempt. Let W the vectorial subspace of $\mathbb{Q}^P$ generated by the set $\{C(t)\}_{t \in T}$. W can be described by a linear equation $v.D = 0$ where the columns of D are a basis of the orthogonal of W which is exactly the set of P-flows. Moreover, D can be chosen with integer coefficients by some multiplication. One applies the recurrence hypothesis to D_1, the matrix obtained from D by deleting the row $D(p_1)$. Then:

- either $\exists v_1 \in \mathbb{N}^{P_1}, v_1.D_1 = 0 \wedge v_1(p) > 0$. Observe that v defined by $\forall p' \in P_1, v(p') = v_1(p') \wedge v(p_1) = 0$ fulfils $v.D = 0$. So v is generated by $\{C(t)\}_{t \in T}$, i.e. $v = \sum_{t \in T} \lambda_t.C(t)$ with $\lambda_t \in \mathbb{Q}$. Multiplying the λ_t by the lest common multiple of theirr denominator, one obtains a vector $v' = C.w \in \mathbb{N}^P$ with $w \in \mathbb{Z}^T$ and $v'(p) > 0$. Hence w is appropriate.
- or $\exists w, D_1.w \in \mathbb{N}^{P_1} \wedge (D_1.w)(p) > 0$. Define $v = D.w$, by construction $v^t.C = 0$. If $v(p_1) \geq 0$ then w is appropriate. It remains the unfavourable case where $v(p_1) < 0$.

Assume that the two unfavourable cases are simultaneously realised. This means that:

- $\exists w \in \mathbb{Z}^T, \forall p' \notin \{p_1, p\}, (C.w)(p') \geq 0 \wedge (C.w)(p) > 0 \wedge (C.w)(p_1) < 0$ and
- $\exists v \in \mathbb{Z}^P, v^t.C = 0 \wedge \forall p' \notin \{p_1, p\}, v(p') \geq 0 \wedge v(p) > 0 \wedge v(p_1) < 0$.

Let us compute by two ways $v^t.C.w$. First, $v^t.C.w = (v^t.C).w = 0$. Second, $v^t.C.w = \sum_{p \notin \{p_1, p\}} v(p').(C.w)(p') + v(p_1).(C.w)(p_1) + v(p).(C.w)(p)$. This sum is composed from non negative terms whose two last ones are positive, hence $v^t.C.w > 0$. This contradiction achieves the proof. $\diamondsuit$

Lemma 34 (Other kinds of duality) *This duality has numerous features:*

1. $\nexists\ v \in \mathbb{N}^P\ \|v\| = P$ *and* $v^t.C = \overrightarrow{0} \Leftrightarrow \exists\, w \in \mathbb{Z}^T$ *s.t.* $C.w > \overrightarrow{0}$
2. $\nexists\ v \in \mathbb{N}^P\ \|v\| = P$ *and* $v^t.C \leq \overrightarrow{0} \Leftrightarrow \exists\, w \in \mathbb{N}^T$ *s.t.* $C.w > \overrightarrow{0}$

Proof
The first equivalence is a straightforward consequence of the previous lemma.
$\exists w \in \mathbb{Z}^T,\ t.q.\ C.w > \overrightarrow{0} \Leftrightarrow \exists p \in P, \exists w \in \mathbb{Z}^T,\ t.q.\ C.w \geq \overrightarrow{0} \wedge (C.w)(p) > 0$
$\Leftrightarrow \exists p \in P, \nexists\ v \in \mathbb{N}^P, v^t.C = \overrightarrow{0} \wedge v(p) > 0$ (using lemma 33)
It remains to show the equivalence of this last assertion with the left term of the first equivalence.

- Obviously, $\exists p \in P, \nexists\ v \in \mathbb{N}^P, v^t.C = \overrightarrow{0} \wedge v(p) > 0 \Rightarrow \nexists\ v \in \mathbb{N}^P, \|v\| = P$ and $v^t.C = \overrightarrow{0}$

- Assume $\forall p \in P, \exists v_p \in \mathbb{N}^P, v_p^t.C = \overrightarrow{0} \wedge v_p(p) > 0$, then defining $v = \sum_{p\in P} v_p$, one has $v \in \mathbb{N}^P$, $\|v\| = P$ et $v^t.C = \overrightarrow{0}$.

We establish the second equivalence using the first one. Let us show first that v and w cannot simultaneously exist. Assume the contrary and compute by two ways $v^t.C.w$.
$v^t.C.w = v^t.(C.w) > 0$ since the support of v is P and
$v^t.C.w = (v^t.C).w \leq 0$ leading to a contradiction.

Define $T' \subseteq T$ by: $T' = \{t \in T \mid \exists w \in \mathbb{N}^T C.w = \overrightarrow{0} \wedge w(t) > 0\}$.
Let us call w_t the vector, that witnesses that t belongs to T'.
Then $w_0 = \sum_{t\in T'} w_t \in \mathbb{N}^T$ fulfils $C.w_0 = \overrightarrow{0}$ and $\|w_0\| = T'$. Let us note $T'' = T \setminus T'$ and introduce matrix $C_{T''} \in \mathbb{Z}^{(P\cup T'')\times T}$ defined by: $C'(p,t) = C(p,t)$ and if $t' = t$ then $C(t',t) = 1$ else $C(t',t) = 0$.

Assume now that $\nexists\ v \in (\mathbb{N})^P\ \|v\| = P$ fulfilling $v^t.C \leq \overrightarrow{0}$.

Then, a fortiori, by construction of $C_{T''}$,
$\nexists\ v \in (\mathbb{N})^{P\cup T''}\ \|v\| = P \cup T''$ fulfilling $v^t.C_{T''} = \overrightarrow{0}$.
Using the first equivalence:
$\exists w_1 \in \mathbb{Z}^T$ fulfilling $C_{T''}.w_1 > \overrightarrow{0}$ which can be expressed by $C.w_1 \geq \overrightarrow{0}$, $\forall t \in T'', w_1(t'') \geq 0$ and:

- either $C.w_1 > \overrightarrow{0}$. Then for some $\lambda \in \mathbb{N}$ enough great, $w = w_1 + \lambda.w_0 \in \mathbb{N}^T$ and also $C.w = C.w_1 + \lambda(C.w_0) > \overrightarrow{0}$. Hence w is an appropriate vector.

- or $\exists t \in T'', w_1(t) > 0$. Then for some $\lambda \in \mathbb{N}$ enough great, $w = w_1 + \lambda.w_0 \in \mathbb{N}^T$ and also $C.w = C.w_1 + \lambda(C.w_0) \geq \overrightarrow{0}$. Since $t \in \|w\|$, $C.w \neq \overrightarrow{0}$. Hence $C.w > \overrightarrow{0}$ and w is an appropriate vector.

This achieves the proof. $\diamondsuit$

The next proposition points out the relations between behavioural properties (liveness and boundedness) and structural properties (conservation and consistency).

Proposition 35 *Let R be a Petri net,*

- $\exists\, v \in \mathbb{N}^P\ \|v\| = P$ *and* $v^t.C \leq \overrightarrow{0} \Leftrightarrow R$ *structurally bounded*
  In particular, R conservative $\Rightarrow$ R structurally bounded

- (R, m_0) *bounded and live* $\Rightarrow R$ *consistent*

The net of figure 3 illustrates that the second implication is not an equivalence. It is conservative (see below the invariant computation) and consistent (sequence $ACH.BCH.RVH.ACV.ACV.RCV$) but it is not live (and this whatever the initial marking as we will see later).

Proof
If v fulfils the hypothesis of the first assertion then $\forall m \in A(R, m_0)$, $v^t.m \leq v^t.m_0$ and consequently for every place p, $m(p) \leq v(p)^{-1}.v^t.m_0$. R is structurally bounded.

If there does not exist such v fulfilling this hypothesis then this is equivalent to $\exists\, w \in \mathbb{N}^T$ *s.t.* $C.w > \overrightarrow{0}$ (lemma 34). w is then the occurrence vector of increasing repetitive sequence σ. Let m_0 be a marking such that $m_0 \xrightarrow{\sigma}$; then (R, m_0) is bounded so R is not structurally bounded.

Let (R, m_0) be a live net. One builds an infinite sequence as follows: one fires a sequence ending by the first transition (liveness) then one applies the same process with every transition and one starts again with the first transition. Consider the sequence of markings obtained after every iteration. Using lemma 23, one can extract two markings of this sequence such that the second is greater or equal than the first. Hence one has $m_0 \xrightarrow{\sigma_0} m_1 \xrightarrow{\sigma_1} m_2$ with $m_1 \leq m_2$ and $\|\overrightarrow{\sigma_1}\| = T$. If $m_1 \neq m_2$ then the net is unbounded using proposition 25. Consequently $m_1 = m_2$, σ_1 is a stationary repetitive sequence and $\overrightarrow{\sigma_1}$ is a T-semiflow which establishes the consistency. $\diamondsuit$

Let us note that the test of the second assertion stated in the duality lemma is reduced to $|P|$ problems of linear programming $Pb(p) : \exists\, w \in \mathbb{N}^T$ *t.q.* $C.w \geq \overrightarrow{0} \wedge (C.w)(p) \geq 1$. Hence the structural boundedness of a net is a problem which can be solved in polynomial time [19].

3.2. *Flow computations*

We only present the flow computation for P-flows since it is enough to consider the transpose of the incidence matrix to obtain T-flows. Observe first that if the net has a flow, then it has an infinity of them (by multiplying the flow by any scalar). Thus we focus on the computation of a generative family of flows.

Definition 36 *Let R be a Petri net, $\{v_1, \ldots, v_n\}$ a family of flows; this family is generative if:*

$$\forall\, v \textit{ flow } \exists\, \{\lambda_1, \ldots, \lambda_n\} \in \mathbb{Q}^n \textit{ s.t. } v = \sum_{i=1}^{n} \lambda_i.v_i$$

It is a smallest family if it is minimal w.r.t. the number of items among the generative families.

As the coefficients are in $\mathbb{Q}$, we are looking for a basis of the vectorial subspace of left cancellers of C. Thus we can compute this family by some variant of Gauss elimination.

GAUSS ELIMINATION

The algorithm proceeds transition by transition: it starts from a generative family of flows for the matrix reduced to the k first transitions and builds a generative family for the matrix reduced to the $k+1$ first transitions.

Initially ($k = 0$); there is no condition and the generative family is defined by $\{\overrightarrow{p}\}_{p \in P}$.

Let t be the next transition to be examined and $\{v_1, \ldots, v_n\}$ the current family.

Case n$^\circ$ 1 $\forall\, v_i\ v_i^t.C(t) = 0$

In this case, the family of flows is unchanged.

Case n$^\circ$ 2 $\exists\, v_{i0}\ v_{i0}^t.C(t) \neq 0$

In this case the flow v_{i0} will play the role of *pivot* to constitute the new generative family $\{v'_i\}_{i \neq i0}$ with:

$v'_i = (v_{i0}^t.C(t)).v_i - (v_i^t.C(t)).v_{i0}$

So during each transition elimination, either the generative family is unchanged, or its cardinality is decremented by one unit. In practice at each iteration, matrix C is transformed to represent the incidence matrix of the current family of flows on the remaining transitions. The number of arithmetical operations is polynomial since there are $|T|$ eliminations and during each elimination the number of operations is bounded by $3.|P|.(|P| + |T|)$. Nevertheless the coefficients of flows could exponentially increase in theory. So one divides the coefficients of a new flow by their greatest common divisor. With this simplification one can prove the memory size of the coefficients remains polynomial since every coefficient is a fraction of determinants of submatrices of C (the memory size of a determinant is polynomial w.r.t. the memory size of the matrix).

Example 12 We apply Gauss elimination to the incidence matrix of the net of figure 3.

$$\mathbf{C} = \begin{array}{c} \begin{array}{cccccc} \text{ACH} & \text{BCH} & \text{ACV} & \text{BCV} & \text{RVH} & \text{RVV} \end{array} \\ \left(\begin{array}{cccccc} -1 & 0 & -1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & -1 & 0 & -1 & 1 & 1 \\ 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 \end{array} \right) \end{array} \begin{array}{l} \\ \overrightarrow{A} \\ \overrightarrow{AAH} \\ \overrightarrow{AAV} \\ \overrightarrow{B} \\ \overrightarrow{BAH} \\ \overrightarrow{BAV} \end{array}$$

In the first column, two items are non null: those of $\overrightarrow{A}$ and $\overrightarrow{AAH}$. We choose $\overrightarrow{A}$ as pivot. This row is deleted and the row indexed by $\overrightarrow{AAH}$ is combined with the row of the pivot to produce a new row. Other rows are unchanged. The second column is similarly handled with the rows indexed by $\overrightarrow{B}$ and $\overrightarrow{BAH}$. We obtained the matrix presented below. Let us note that the current family indices the rows of this matrix (on the right of the matrix).

$$\begin{array}{c} \begin{array}{cccc} \text{ACV} & \text{BCV} & \text{RVH} & \text{RVV} \end{array} \\ \left(\begin{array}{cccc} -1 & 0 & 0 & 1 \\ 1 & 0 & 0 & -1 \\ 0 & -1 & 0 & 1 \\ 0 & 1 & 0 & -1 \end{array} \right) \end{array} \begin{array}{l} \\ \overrightarrow{A} + \overrightarrow{AAH} \\ \overrightarrow{AAV} \\ \overrightarrow{B} + \overrightarrow{BAH} \\ \overrightarrow{BAV} \end{array}$$

We proceed now to the elimination of the first and second columns of the above matrix. In the first column, only two components are non null: those of $\overrightarrow{A} + \overrightarrow{AAH}$ and $\overrightarrow{AAV}$. As previously, we combine the rows. The second row is similarly handled and leads to the matrix presented below.

$$\begin{array}{cc} \text{RVH} & \text{RVV} \\ \end{array}$$

$$\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \quad \begin{array}{l} \overrightarrow{A} + \overrightarrow{AAH} + \overrightarrow{AAV} \\ \overrightarrow{B} + \overrightarrow{BAH} + \overrightarrow{BAV} \end{array}$$

This matrix is null. Thus the current family is a generative family of flows. The associated invariants express the state of processes A and B.

3.3. *Semiflow computation*

We define the notion of smallest generative family of semiflows w.r.t. linear combinations with non negative rational coefficients.

Definition 37 *Let R be a Petri net, $\{v_1, \ldots, v_n\}$ be a family of semiflows, this family is generative if:*
$\forall\ v\ semiflow\ \exists\ \{\lambda_1, \ldots, \lambda_n\} \in (\mathbb{Q}^+)^n\ t.q.\ v = \sum_{i=1}^{n} \lambda_i.v_i$

It is a smallest generative family if it is minimal w.r.t. the number of items of the family.

Farkas algorithm
In order to compute a generative family, we want again to proceed by transition elimination. The initial family is the same as the one of the flow computation. Let us examine the way to produce a new generative family during the elimination of t. We split the semiflows into three categories:

- $F^+ = \{v \mid v^t.C(t) > 0\}$
- $F^- = \{v \mid v^t.C(t) < 0\}$
- $F^0 = \{v \mid v^t.C(t) = 0\}$

Every vector of F^0 belongs to the new generative family. In order to obtain new semiflows, we must cancel the incidence w.r.t. t by positive combinations. So it is obvious that every combination must include at least a vector of F^+ and a vector of F^-. Thus we take every such pair to produce new items of the family noted F':
$F' = F^0 \cup \{w \mid \exists v_+ \in F^+, \exists v_- \in F^- w = (v_+^t.C(t)).v_- - (v_-^t.C(t)).v_+\}$

We admit that the minimality of the family is ensured by keeping only one semiflow per minimal support [6]. It is more efficient to minimise the family after each elimination since the combinatory explosion of the number of semiflows often occurs. In the worst case, the size of a minimal generative family (independent of the family) is exponential w.r.t. the number of places.

The reader can consult the same paper for a deep discussion about efficient implementations of this algorithm, called Farkas algorithm.

Example 13 We apply Farkas algorithm to the incidence matrix of the net of figure 3. In order to illustrate the different steps of the computation, we have modified the order of columns.

$$\mathbf{C} = \begin{array}{c} \begin{array}{cccccc} RVH & ACH & BCH & ACV & BCV & RVV \end{array} \\ \left(\begin{array}{cccccc} 1 & -1 & 0 & -1 & 0 & 1 \\ -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 \\ 1 & 0 & -1 & 0 & -1 & 1 \\ -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{array}\right) \end{array} \begin{array}{l} \\ \overrightarrow{A} \\ \overrightarrow{AAH} \\ \overrightarrow{AAV} \\ \overrightarrow{B} \\ \overrightarrow{BAH} \\ \overrightarrow{BAV} \end{array}$$

The column indexed by RVH has two positive components and two negative components. We combine the corresponding rows in pairs, cancelling their component relative to RVH. The other rows are unchanged. Thus we obtain the matrix presented below.

$$\begin{array}{c} \begin{array}{ccccc} ACH & BCH & ACV & BCV & RVV \end{array} \\ \left(\begin{array}{ccccc} 0 & 0 & -1 & 0 & 1 \\ -1 & 1 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 & -1 \\ 1 & -1 & 0 & -1 & 1 \\ 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 1 & -1 \end{array}\right) \end{array} \begin{array}{l} \\ \overrightarrow{A} + \overrightarrow{AAH} \\ \overrightarrow{A} + \overrightarrow{BAH} \\ \overrightarrow{AAV} \\ \overrightarrow{B} + \overrightarrow{AAH} \\ \overrightarrow{B} + \overrightarrow{BAH} \\ \overrightarrow{BAV} \end{array}$$

The column indexed by ACH includes a negative component and a positive component. Combining the corresponding rows, we obtain the vector: $\overrightarrow{A} + \overrightarrow{BAH} + \overrightarrow{B} + \overrightarrow{AAH}$. Its support is not minimal. For instance, it strictly includes the support of $\overrightarrow{A} + \overrightarrow{AAH}$. This new vector is thus deleted. Then the algorithm goes on (**in this particular case**) as Gauss elimination and the family of semiflows is identical to the family of flows.

3.4. *Application of invariants to the analysis of a net*

Here we apply the previous techniques to the problem of readers/writers in a database. We consider an abstraction of this problem and we focus on the synchronisation constraints between the operations "read" and "write" described in table 1. As said in chapter 1, the capacity of the reading room is limited to k readers (C_1), at any time at most one write is possible on the database (C_2) and the operations read and write are mutually exclusive (C_3).

C_1 : At most k simultaneous read
C_2 : At most one write
C_3 : No simultaneous read and write

Table 1: Synchronisation conditions between readers and writers

In the real world, one must additionally maintain the consistency of data and for instance ensure that if a value is read, it corresponds to the last written value.

3.4.1. *Modelling of the readers/writers problem*

Figure 4 presents a modelling of this problem by a Petri net. This modelling is somewhat parameterised since the capacity of the lecture room is represented by a variable, the positive integer k. It appears both in the initial marking of the net (place M) and as a valuation of arcs between place M and transitions EnE and SoE. The database system includes two waiting rooms: one for the readers (AL) and the other for the writers (AE), one reading room (L) and one writing room (E). The left part of the net describes the management of readers while the right part is relative to the management of writers. The central part (place M) ensures the synchronisation between readers and writers.

We detail below the dynamic features of the modelling.

Readers: Transition ArL represents the arrival of new readers in the system. They wait in place AL. Transition EnL represents the beginning of a read operation. Place L counts the number of active read operations. Finally transition SoL corresponds to the end of a read operation.

Writers: Without taking into account the synchronisation the management of writers is similar to the one of the readers. Geometrically, this analogy is materialised by a vertical symmetry axis through place M.

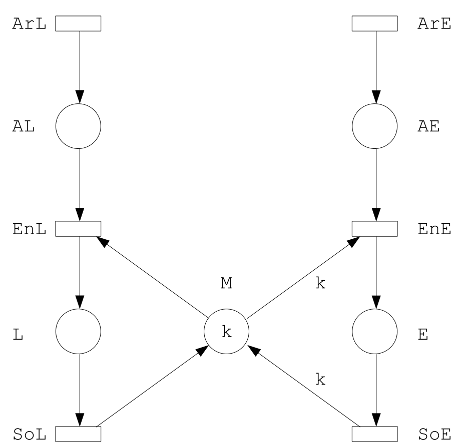

Figure 4: Problem of readers/writers ($k \geq 1$)

Synchronisation: Place M, shared precondition of transitions EnL and EnE, performs the synchronisation between read and write operations:
- The capacity of the reading room is materialised by the marking of place M: activation of a new read is only possible where there are still at least one token in M ($m(M) \geq 1$). At the end of a read operation (transition SoL), the marking of M is incremented.
- To activate a write operation the reading room must be empty ($m(M) = Pre(M, EnE) = k$). This activation consumes the k tokens of place M and produces a token in place E. The end of a write operation (transition SoE) increments by k the marking of place M.

3.4.2. *Verification synchronisation constraints*

Expression of properties

The first step consists in translating the properties of table 1 formulae about the modelling net. More precisely, these properties are relative to the marking of places L and E. Table 2 presents the corresponding formulae. These formulae are (non linear) invariants that must be fulfilled by every reachable marking.

P-flows computation

$\forall m \in A(R, m_0),$

$C_1 : m(L) \leq k$	At most k simultaneous read
$C_2 : m(E) \leq 1$	At most one write
$C_3 : m(E).m(L) = 0$	No simultaneous read and write

Table 2: Translation of synchronisation conditions of table 1

We apply the computation of flows to the incidence matrix described below. Introduction of parameters like k does not rise serious difficulties for Gauss elimination. Instead of operating in the rational field, it operates on a polynomial ring whose variables are the parameters of the net. The single relevant modification consists in maintaining a polynomial "condition" (product of the successive pivots) which ensures that whenever the values of the parameters do not cancel the polynomial, the family of flows is a generative family.

$$\mathbf{C} = \begin{array}{c c} \begin{array}{cccccc} \text{ArL} & \text{EnL} & \text{SoL} & \text{ArE} & \text{EnE} & \text{SoE} \end{array} & \\ \left(\begin{array}{cccccc} 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & -k & k \\ 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{array}\right) & \begin{array}{c} \overrightarrow{AL} \\ \overrightarrow{L} \\ \overrightarrow{M} \\ \overrightarrow{AE} \\ \overrightarrow{E} \end{array} \end{array}$$

The elimination of columns ArL and ArE respectively deletes the vectors $\overrightarrow{AL}$ and $\overrightarrow{AE}$. The elimination of column SoL combines $\overrightarrow{M}$ and $\overrightarrow{L}$ in a partial flow $\overrightarrow{L} + \overrightarrow{M}$. The elimination of column SoE combines it $\overrightarrow{E}$ to produce the flow:

$$\overrightarrow{L} + \overrightarrow{M} + k.\overrightarrow{E}$$

The remaining columns are null and this flow constitutes the generative family. In this particular case, the polynomial condition is the constant 1, which means that the family is valid for every value of the parameter. Applying this flow to the initial marking, one obtains:

$$\forall m \in A(R, m_0),\ m(L) + m(M) + k.m(E) = k$$

Proof of synchronisation constraints

Using the computed invariant, let us prove that conditions of table 2 are fulfilled.

C_1 : Isolating $m(L)$ in the invariant, one obtains:

$$m(L) = k - (m(M) + k.m(E)) \leq k$$

C_2 : Isolating $m(E)$ in the invariant, one obtains:

$$k.m(E) = k - (m(M) + m(L)) \leq k$$

$$k \neq 0 \Rightarrow m(E) \leq 1$$

C_3 : C_3 can be rewritten $m(L) \neq 0 \Rightarrow m(E) = 0$.
Assume $m(L) \neq 0$, then:

$$k.m(E) = k - (m(M) + m(L)) < k$$

$$k \neq 0 \Rightarrow m(E) < 1 \Rightarrow m(E) = 0$$

3.4.3. *Discussion*

Observe that the computation of flows has operated on a parameterised net (this not the case of most of the other analysis methods). So we have established the property for a family (indexed by $k > 0$) of marked nets. For instance, an exhaustive approach based on the construction of the reachability graph or here of the covering tree of Karp and Miller (see the next chapter) since places AL and AE are unbounded would prove it for a fixed value of the parameter k.

Computation of invariants is an efficient approach for verification of *safeness* properties. This class of properties can be informally described by the statement "Nothing bad will happen". In the example, the bad event is the violation of synchronisation conditions.

In practice, model validation requires the satisfaction of *liveness* properties. This class of properties can be informally described by the statement "Something good must happen". In the example, we could check whether a reader will not infinitely wait. The net does not fulfill this property: staring from the reachable marking $\overrightarrow{AL} + k.\overrightarrow{M}$ (one waiting reader) sequence $\sigma_1 = (ArE \,.\, InE \,.\, SoE)^\infty$ is possible and leads to an infinity of write operations while the reader is waiting. The infinite sequence $\sigma_2 = ArE^\infty$ is also possible from this marking. T-semiflows may provide partial hints about such behaviours. For instance, the support of σ_1 is the T-semiflow $\overrightarrow{ArE} + \overrightarrow{EnE} + \overrightarrow{SoE}$. Unfortunately the support of σ_2 is not a T-semiflow. However it is an increasing repetitive sequence and its support could be computed by an adaptation of the T-semiflow computation.

4. Net reductions

A reduction is a net transformation which reduces its size and such that, for a set of properties, the reduced net is equivalent to the initial net [4]. A reduction is characterised by:

- its application conditions,
- the net transformation,
- the preserved properties (i.e. those whose verification can be performed on the reduced net).

From a theoretical point of view, definition of a reduction raises some methodological problems:

- In order to be often satisfied, the application conditions must correspond to a behavioural pattern frequently used in modellings. So one must find structural conditions that ensure that the behaviour of the net fulfils the pattern. For instance, we forbid the case when checking a condition requires the construction of the reachability graph.
- The effect of the transformation must be really efficient. Otherwise stated, what is relevant is the potential reduction of the reachability graph rather than the reduction of the net.
- At last, among the properties one wishes to include the boundedness and the liveness which are the more relevant w.r.t. the behaviour.

In [3] a set of ten reductions is proposed. We only present three of them since, on one side, their application conditions are fully structural and on the other side they cover useful patterns. Other sets of reductions have been defined in [5, 15, 20]. Two of these reductions are related to transitions, the pre-agglomeration and the post-agglomeration and the last one is the deletion of redundant places.

In this paragraph, we note *Prop* the following set of properties: existence of infinite sequence, pseudo-liveness, quasi-liveness, liveness, existence of a home state, boundedness.

4.1. *Pre-agglomeration of transitions*

This reduction is related to a transition h whose firing is *necessary* to the firing of a set of F via a place p. The principle of the pre-agglomeration consists in the reduced net to only consider sequences in which a firing of h is immediately followed by the firing of a transition $f \in F$. In the reduced net, h and F are deleted and a set of transitions $h.f$ (one per transition of F) is added. The structural conditions of the next definition ensure that:

- in a firing sequence where an occurrence of h is later followed by an occurrence of f, one can **postpone** the occurrence of h until the one of f;
- in a firing sequence where an occurrence of h is not later followed by an occurrence of f, one can **delete** the occurrence of h.

These two proprerties (and some additional ones) ensure the equivalence of the two nets w.r.t. *Prop.*

Definition 38 (Pre-agglomerable transitions) *Let (R, m_0) be a Petri net; a set of transitions F is pre-agglomerable with a transition $h \notin F$ iff the following conditions are fulfilled:*

1. *There exists a place p modelling an intermediate state between the firing of h and the one of a transition of F:*
   $m_0(p) = 0$, $Post(p) = \overrightarrow{h}$ *and* $Pre(p) = \sum_{f\in F} \overrightarrow{f}$
2. *h only produces a token in p:* $Post(h) = \overrightarrow{p}$
3. *h does not share its inputs with any other transition:*
   $\forall t' \neq t,\ {}^{\bullet}t' \cap {}^{\bullet}t = \emptyset$
4. *h has at least one input:* ${}^{\bullet}h \neq \emptyset$

The first hypothesis synchronises the firings of the transitions de F with those of h. In a marking, $m(p)$ represents the difference between the number of firings of h and the number of firings of transitions of F. The second hypothesis ensures that the firing of h is only useful for the firing of transitions of F. The next hypothesis implies that if h is fireable then it cannot be disabled. The last hypothesis is required for the equivalence of boundedness of the original net and the one of the reduced net.

Definition 39 (Pre-agglomeration of net) *Let (R, m_0) be a pre-agglomerable Petri net, then (R', m'_0) the reduced net is defined by:*

- $P' = P \setminus \{p\}$ *and* $T' = T \setminus (F \cup \{h\}) \cup \{h.f \mid f \in F\}$
- $\forall t \in T' \cap T,\ Pre'(t) = Pre(t)$ *and* $Post'(t) = Post(t)$

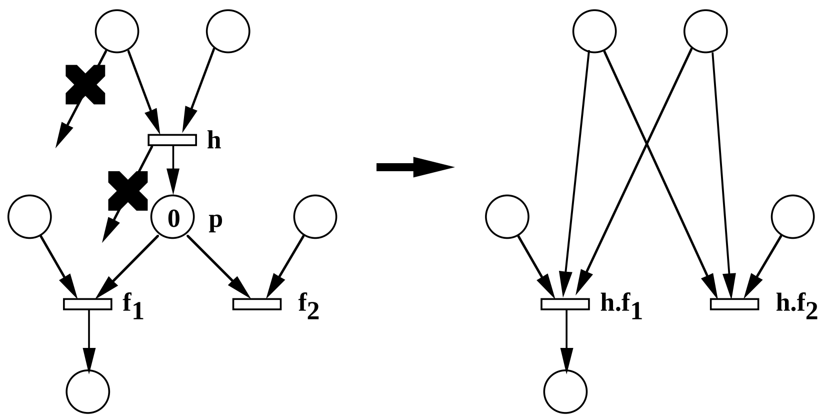

Figure 5: Pre-agglomeration of transitions

- $\forall f \in F,\ Pre'(h.f) = Pre(h) + Pre(f) - \overrightarrow{p}$ *and* $Post'(h.f) = Post(f)$
- $\forall p' \in P',\ m'_0(p') = m_0(p')$

Figure 5 shows the conditions of pre-agglomeration and the transformation of the net.

The theory of reductions establishes that the reachability set of the new net is isomorphic to the reachability set of the original net such that p is unmarked. One empirically observes that this reduction divides the size of the reachability set approximatively by 2. So n consecutive reductions approximatively divide this size by 2^n. This remarks equally applies to the next reduction. Using the application conditions, we obtain:

Proposition 40 (Preservation of properties) *Let* (R, m_0) *be a pre-agglomerable net and* π *be a property of Prop:* (R, m_0) *fulfils* π *iff* (R', m'_0) *fulfils* π*.*

4.2. ***Post-agglomeration of transitions***

This reduction is related to a set of transitions H whose firing of any of these transitions is *necessary and sufficient* for the firing of any of transitions of a set F and this via a place p. The principle of the post-agglomeration consists to only consider in the reduced net sequences in which a firing of a transition $h \in H$ is immediately followed by the firing of any $f \in F$. In the reduced net, H and F are deleted and a set of transitions $h.f$ (one per pair of transitions of $H \times F$) is added. The structural conditions of the next definition ensure that:

- in a firing sequence where an occurrence of h is later followed by an occurrence of f, one can **anticipate** the occurrence of f immediately after the one of h,
- in a firing sequence where an occurrence of h is not later followed by an occurrence of f, one can **add** an occurrence of f

These two properties about sequences (and some additional properties) ensure the equivalence of the two nets w.r.t. *Prop.*

Definition 41 (Transitions post-agglomerables) *Let (R, m_0) be a Petri net, a set of transitions F is post-agglomerable with a set of transitions H disjoint from F iff the following conditions are fulfilled:*

1. *There exists a place p modelling an intermediate state between the firing of h and the one of a transition of F:*
   $m_0(p) = 0$, $Post(p) = \sum_{h \in H} \overrightarrow{h}$ *et* $Pre(p) = \sum_{f \in F} \overrightarrow{f}$
2. *Transitions of F have no other input than p:* $\forall\, f \in F,\ Pre(f) = \overrightarrow{p}$
3. *There exists a transition f of F with at least one output:*
   $\exists f \in F,\ f^{\bullet} \neq \emptyset$.

As for pre-agglomeration, the first hypothesis synchronises the firing of transitions of H and those of F. The second hypothesis ensures that every transition $f \in F$ is fireable as soon as p is marked. The last hypothesis is required for the equivalence of the boundedness of the reduced net and the one of the original net.

Definition 42 (Post-agglomeration of net) *Let (R, m_0) be a post-agglomerable Petri net, then (R', m'_0) the reduced net is defined by:*

- $P' = P \setminus \{p\}$ *and* $T' = T \setminus (F \cup H) \cup \{h.f \mid f \in F\ ,\ h \in H\}$
- $\forall\, t \in T' \cap T,\ Pre'(t) = Pre(t)$ *et* $Post'(t) = Post(t)$
- $\forall\, f \in F,\ \forall\, h \in H,\ Pre'(h.f) = Pre(h)$ *and*
  $Post'(h.f) = Post(h) + Post(f) - \overrightarrow{p}$
- $\forall\, p' \in P',\ m'_0(p') = m_0(p')$

Figure 6 shows the condition of post-agglomeration and the transformation of the net.

Here again we have:

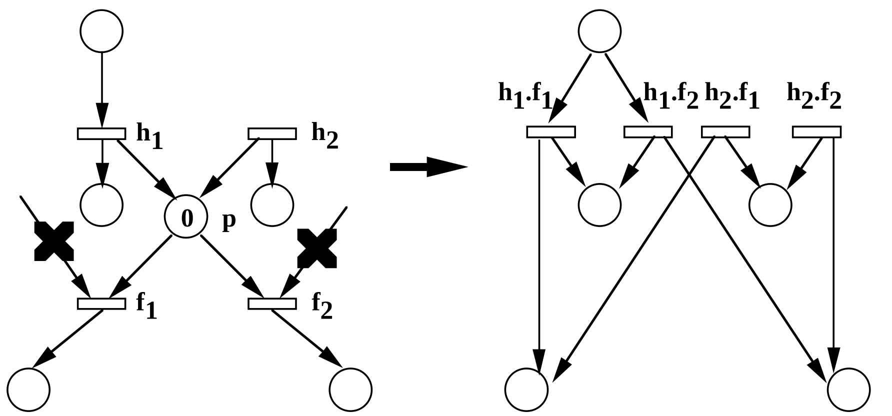

Figure 6: Post-agglomeration of transitions

Proposition 43 (Preservation of properties) *Let (R, m_0) be a post-agglomerable net and π be a property of Prop: (R, m_0) fulfils π iff (R', m'_0) fulfils π.*

4.3. *Deletion of redundant places*

The deletion of redundant places consists in deleting in the net a place which never alone disables the firing of transitions. Usually, this place is a witness of some activities without perturbation on the behaviour of the net. The existence of a redundant place is characterised by the existence of some particular linear invariant. This reduction let unchanged the size of the reachability set but very often **other reductions become applicable** which decrease this size.

Definition 44 (Redundant place) *Let (R, m_0) be a Petri net. A place p_0 is redundant if:*

1. *There exists a P-flow $v = \sum_{p \in P} \lambda_p.\overrightarrow{p}$ with $\lambda_{p_0} > 0$ and $\forall p \neq p_0,\ \lambda_p \leq 0$*
2. *$\forall t \in T,\ v^t.m_0 \geq v^t.Pre(t)$*

The second hypothesis ensures that initially a redundant place cannot disable alone the firing of a transition. The first condition ensures that this hypothesis is valid for every reachable marking.

Definition 45 (Deletion of a redundant place) *Let (R, m_0) be a Petri net where p_0 is a redundant place; then (R', m'_0) the reduced net is defined by deleting p and the adjacent arcs.*

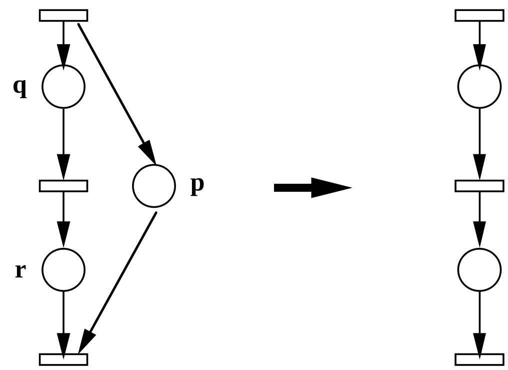

Figure 7: Deletion of a redundant place

Figure 7 shows a case of a redundant place and the transformation of the net. The P-flow of the definition is here: $\overrightarrow{p} - \overrightarrow{q} - \overrightarrow{r}$

Again, one obtains:

Proposition 46 (Preservation of properties) *Let* (R, m_0) *be a Petri net with a redundant place and* π *be a property of Prop:* (R, m_0) *fulfils* π *iff* (R', m'_0) *fulfils* π

A complete analysis of a net with the help of reductions will be developed in the chapter about high level nets.

5. The graph of a Petri net

A Petri net can be viewed as a bipartite graph whose arcs are labelled by integers. In this section, we take advantage of the graph analysis in order to obtain hints about the behaviour of the net. We begin by general results applicable to every net. Then we restrict our attention to subclasses of nets for which a structural analysis gives deeper results.

5.1. *General results*

Let us first examine the influence of the (strong) connectivity on the behaviour of the net. If the net is not connected, every connected component is an independent net. So without loss of generality we restrict our attention to connected nets. Before the presentation of the classical result on the strong connectivity, let us examine the two following nets:

- A net constituted by a place input of a transition. This net (not strongly connected) is bounded and not live.
- A net constituted by a transition input of a place. This net (not strongly connected) is not bounded and live.

These two elementary examples suggest the following proposition.

Proposition 47 *Let (R, m_0) be a Petri net, live and bounded; then R is a strongly connected graph.*

Proof
Assume that R is not a strongly connected graph; then there exists an initial s.c.c. $\mathcal{C}$ which has (at least) an arc leading to another s.c.c. $\mathcal{C}'$. Assume additionally that (R, m_0) is live. The proof depends on the kind of this arc.

This arc is an arc $t \to p$. Since $\mathcal{C}$ is initial, the net restricted to $\mathcal{C}$ is live. So there exists a firing sequence of $\mathcal{C}$ with an infinity of occurrences of t. By definition, place p is the not the input of any transition of $\mathcal{C}$. Its marking infinitely increases during this firing sequence thus (R, m_0) is unbounded.

This arc is an arc $p \to t$. There exists a firing sequence of (R, m_0) including an infinity of occurrences of t. If we project this sequence on transitions of $\mathcal{C}$, then again since $\mathcal{C}$ is initial this projected sequence is a firing sequence. By definition, no transition other than the ones of $\mathcal{C}$ provides tokens to p. Let us note σ_n a finite subsequence of the initial sequence including n firings of t, m_n the reached marking, σ'_n the projection of σ_n on transitions of $\mathcal{C}$ and m'_n the reached marking. Then since t consumes at least one token of p, $m'_n(p) \geq m_n(p) + n \geq n$. So the net is unbounded. $\diamondsuit$

Example 14 Again the non live net of figure 3 fulfils the condition of strong connectivity.

The next result is another necessary condition for boundednes and liveness of a net. The interest of this proposition is twofold. First, it is obtained by combining graph analysis and linear algebra techniques. Moreover, it points out the reason why numerous results have been obtained for the subclasses we present later. We follow [22] for the development of the proof.

Let us recall some elementary notions of linear algebra.

Definition 48 (Independent family , rank of a matrix)
Let $\{v_1, \ldots, v_n\}$ be a family of vectors of $\mathbb{Q}^E$, this family is linearly independent if:

$$\forall \{\lambda_1, \ldots, \lambda_n\} \in \mathbb{Q}^n \left(\sum_{i=1}^{n} \lambda_i.v_i = 0 \Rightarrow \forall i \; \lambda_i = 0 \right)$$

Let A be a matrix. The rank of A, denoted $rank(A)$, is defined as the size of the greatest family of its column vectors linearly independent.

One proves that it is equivalent to define the rank w.r.t. to row vectors.

Observe that if the net is consistent, then $rank(C) < |T|$. Using proposition 35, one deduces that if a net is bounded and live then this condition is fulfilled. We improve this upper bound with the help of an equivalence relation between transitions.

Definition 49 (Relation of equal conflict) *Let R be a Petri net. Two transitions t and t' are in relation of* equal conflit *if: $Pre(t) = Pre(t')$*

This relation is an equivalence relation. We denote Θ the set of equivalence classes.

The key point of this relation is that the transitions of an equivalence class are always simultaneously fireable. We will transform the net in such a way that the equivalence classes are singletons. We proceed iteratively.

Definition 50 *Let R be a Petri net, let $E = \{t_0, \ldots, t_{k-1}\}$ be an equivalence class of Θ with $k > 1$. The net R_E is defined by:*

- $P_E = P \cup \{p_0, \ldots, p_{k-1}\}$ *where p_i are new places,*
- $T_E = T$
- $\forall p \in P \; Pre_E(p) = Pre(p)$, $Post_E(p) = Post(p)$
- $\forall 0 \leq i < k \; Post_E(p_i) = \overrightarrow{t_{(i-1) mod\, k}}$, $Pre_E(p_i) = \overrightarrow{t_i}$

Informally, one superimposes to the initial net, a circuit constituted alternatively by transitions t_i and places p_i. The next lemma justifies the transformation.

Lemma 51 *If (R, m_0) is live and bounded then:*

- $\exists \, m'_0$ *such that (R_E, m'_0) is live and bounded*

- $rank(C_E) = rank(C) + |E| - 1$

Proof
Let m be a reachable marking of (R, m_0) and t be a transition. Since (R, m_0) is live, there exists a sequence σ such that $m \xrightarrow{\sigma.t}$. Define $\Delta(m, t)$ as the number of occurrences of transitions of E in σ and Δ as the maximum of $\Delta(m, t)$ for every reachable m and $t \in T$ (a finite enumeration since the net is bounded). We define m'_0 by:

$$\forall p \in P\ m'_0(p) = m_0(p) \quad et \quad \forall 0 \leq i < k\ m'_0(p_i) = \Delta$$

Let m' a reachable marking of (R_E, m'_0) with the corresponding sequence $m'_0 \xrightarrow{\sigma} m'$. We show first that we can reach a marking where places p_i have their initial marking. Let us call $s()$, the mapping which associates with the transition t_i the sequence of transitions in E: $t_{(i+1) mod\, k} \ldots t_{(i-1) mod\, k}$. The effect of this sequence is to put again in p_i a token "moved" by a firing of t_i. Otherwise stated, on the set of places $\{p_j\}$, sequence $s(t_i)$ cancels the effect of t_i. This mapping can be extended to sequences by the usual way. Let $\sigma_{\lfloor E}$ be the projection of σ on transitions of E. Define $\sigma_1 = s(\widetilde{\sigma_{\lfloor E}})$. Obviously, $m' \xrightarrow{\sigma_1}_{\{p_j\}_{0 \leq j < k}}$ since one cancels the effect of every transition firing in E beginning by the last transition fired. m' restricted to P is a reachable marking of (R, m_0). So there exists a shortest sequence σ_2 in this net in order to fire a transition of E. σ_2 does not include transitions of E and can be fired in (R_E, m') leading to a marking where the first transition of σ_1 is fireable. One fires it and one iterates this process until one fires all transitions of σ_1. The reached marking m'' is identical to the initial marking on places $\{p_j\}_{0 \leq j < k}$ and corresponds on P to a reachable marking m^* of (R, m_0). Pick now any transition t; one can fire in (R_E, m'') the sequence ended by t corresponding to $\Delta(m^*, t)$ occurrences of transitions of E. So (R_E, m'_0) is live.

(R_E, m'_0) is bounded since, on the one side the projection on P of a reachable marking is a reachable marking of (R, m_0) (bounded net) and, on the other side, places $\{p_j\}$ are bounded due to the semiflows $\overrightarrow{p_0} + \cdots + \overrightarrow{p_{k-1}}$.

Let us study the rank of C_E. We reason on the row vectors (i.e. the incidence of places) of C_E. Observe first that C_E has $|E|$ additional rows corresponding to $\{p_j\}_{0 \leq j < k}$ but the associated vectors are not linearly independent due to the semiflow $\overrightarrow{p_0} + \cdots + \overrightarrow{p_{k-1}}$. We conclude that $rank(C_E) \leq rank(C) + |E| - 1$. We can delete any such row and keep the same rank. Let us delete the one indexed by p_0. In order to have a strict inequality, a row vector indexed by some p_i should be a linear combination of the other row vectors. Otherwise stated:

$$C_E(p_i) = \sum_{p \in P} \lambda_p.C(p) + \sum_{p_j \neq p_i, p_0} \lambda_j.C_E(p_j)$$

Since (R, m_0) is live, there exists a shortest sequence in order to fire any transition of E; we fire it followed by t_0 then we do it again with t_1 and again util t_{i-1} where we start again with t_0. Let σ_n be the sequence including n steps of this process and $\overrightarrow{\sigma_n}$ be its occurrence vector then:

- $C_E(p_i)^t.\overrightarrow{\sigma_n} = \sum_{p \in P} \lambda_p.C(p)^t.\overrightarrow{\sigma_n} + \sum_{p_j \neq p_i, p_0} \lambda_j.C_E(p_j)^t.\overrightarrow{\sigma_n}$
- $C_E(p_i)^t.\overrightarrow{\sigma_n} = n$
- $\forall p_j \neq p_i, p_0 \; C_E(p_j)^t.\overrightarrow{\sigma_n} = 0$
- $\forall p \in P, \; -m_0(p) \leq C(p)^t.\overrightarrow{\sigma_n} \leq B - m_0(p)$ where B is a bound of net (R, m_0)

However if n goes to infinity, the left hand of the first equality goes to infinity while the right hand remains bounded. So the equation between ranks is fulfilled. $\diamondsuit$

We can now state a new necessary condition for simultaneous boundedness and liveness.

Proposition 52 *If (R, m_0) is live and bounded then $rank(C) < |\Theta|$*

Proof
We apply the previous construction on all equivalence classes with size greater than 1. Let (R', m'_0) be the obtained net, due to the necessary condition of consistency stated in proposition 35: $rank(C') < |T'| = |T|$ and also $rank(C') = rank(C) + \sum_{E \in \Theta}(|E| - 1) = rank(C) + |T| - |\Theta|$. Substituting $rank(C')$ in the inequality by its expression, one obtains the result. $\diamondsuit$

Example 15 This condition is more discriminating. For instance the bounded and non live net of figure 3 does not fulfil this condition. In fact the rank of matrix C is 4 (a generative famille of 2 P-flows for 6 transitions) and $|\Theta| = 4$ ($\{ACH, ACV\}$, $\{BCH, BCV\}$, $\{RVH\}$ et $\{RVV\}$ are the different equivalence classes). Since this condition is independent from the initial marking and since the net is structurally bounded, one deduces that there does not exist an initial marking such that the net would be live.

5.2. *State machine*

We present three subclasses by increasing order of complexity: state machines, event graphs and free choice nets. For these classes, several behavioural

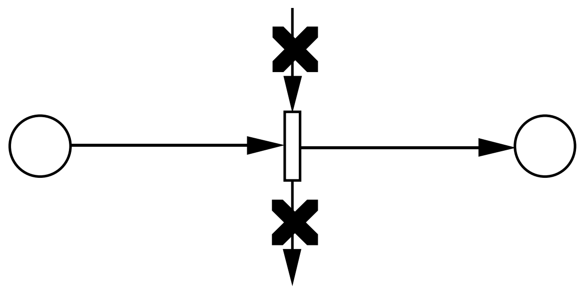

Figure 8: Constraints of state machines

properties are characterized by structural conditions. The verification of these characterisations is performed by efficient algorithms compared to the state based algorithms like those that rely on the reachability graph. As previously, we assume that the net is connected.

A state machine can be viewed as a finite automaton shared by several anonymous processes. Places describe states, transitions represent state changes and the marking of a place indicates the number of processes in the corresponding state (see figure 8).

Definition 53 (State machine) *R, a Petri net, is a state machine if:* $\forall\, t \in T, \exists\, p_{in}, p_{out}$ *with* $Pre(t) = \overrightarrow{p_{in}}$ *and* $Post(t) = \overrightarrow{p_{out}}$

The flow of tokens is extremely simple since consuming of a token is followed the production of another token. So we obtain:

Proposition 54 *Let R be a state machine; then* $\sum_{p \in P} \overrightarrow{p}$ *is a P-semiflow. In particular R is conservative thus structurally bounded.*

Verification of liveness is also easy and is performed in linear tine w.r.t. the size of the net with the help of Tarjan algorithm [1].

Proposition 55 (Liveness of a state machine) *If R is a state machine then:*

$$(R, m_0) \text{ is live iff } R \text{ is strongly connected and } m_0 \neq \overrightarrow{0}$$

Proof

Assume the net is live. So some transition is initially fireable, hence $m_0 \neq \overrightarrow{0}$. Furthermore, since the net is bounded applying proposition 47, one deduces that the net is strongly connected.

Assume the net is initially marked and strongly connected, since the net is

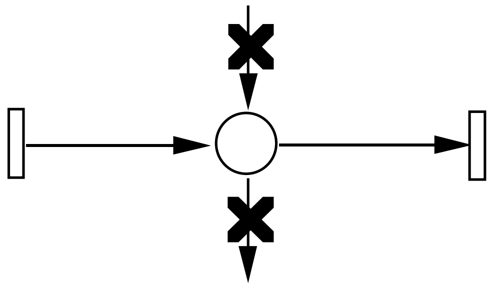

Figure 9: Constraints of event graphs

conservative every marking m has at least one token in place p. Let t be any transition, there exists a path from p to t (strong connectivity). One fires successively the transitions of this path ending by t. $\diamondsuit$

5.3. *Event graph*

An event graph [8] is a net when transitions never conflict, since a place is input (and output) of a single transition. Otherwise stated, there are no real choices in these nets, but rather different schedulings. When one precondition of a transition firing is fulfilled, it remains fulfilled until its firing (see figure 9).

Definition 56 (Event graph) *R, a Petri net, is an event graph if:* $\forall p \in P, \exists\, t_{in}, t_{out}$ *with* $Pre(p) = \overrightarrow{t_{out}}$ *and* $Post(p) = \overrightarrow{t_{in}}$

Recall that an elementary circuit is a path in a graph such that only the first and last vertices are identical. Observe first that the number of tokens of an elementary circuit in an event graph is invariant since there are neither input transitions nor output transitions. So places of an elementary circuit constitute a P-semiflow. This fact is the starting point of the theory of event graphs.

Proposition 57 (Liveness of an event graph) *If R is an event graph, then:*
(R, m_0) *is live iff*
Every elementary circuit of R includes an initially marked place

Proof
Assume that an elementary circuit is initially unmarked. Then no transition of the circuit will never be fired. Hence the net is not live.

Assume that every elementary circuits are initially marked. then for every reachable marking m, they are still marked (see the previous remark). Pick such a marking m, we define the relation t *helps* t' *iff* there exists an unmarked place output of t and input of t' and define the relation t *preceeds* t' as the reflexive and transitive closure of *helps*. Let us prove that *preceeds* is a (partial) order. Assume the contrary, then we have two transitions t and t' such that t *preceeds* t' and t' *preceeds* t. By definition of *preceeds*, it means that there exist paths from t to t' and from t' to t where all places are unmarked. Combining them, one obtains a circuit from which one extracts an elementary circuit unmarked, which is impossible. This partial order can be extended in a total order. Let $t_1, \ldots, t_n$ the ordered sequence of transitions. We claim that $m \xrightarrow{t_1 \ldots t_n}$. In fact t_1 is fireable since all the input places are marked and if $m \xrightarrow{t_1 \ldots t_i} m'$ then all input places of t_{i+1} are marked in m'. Hence the net is live. $\diamondsuit$

To check the liveness, we delete the marked places and check the existence of a circuit in the obtained graph. The time complexity of this search is linear w.r.t. the size of the graph. Also observe that during the proof, we have shown that liveness is equivalent to the existence of a firing sequence starting from the initial marking including exactly an occurrence of every transition. Moreover this sequence is repetitive stationnary, since by definition of an event graph $\sum_{t \in T} \overrightarrow{t}$ is a T-semiflow. Let us study the structural boundedness.

Proposition 58 (Structurally bounded event graph) *If R is an event graph then:*

R is structurally bounded iff R is strongly connected

Proof

If R is strongly connected, every place is covered by a circuit. So the sum of the associated P-semiflows ensure that the net is conservative hence structurally bounded.

Assume that R is not strongly connected. There exists an initial s.c.c. $\mathcal{C}$ with a s.c.c successor $\mathcal{C}'$. Otherwise stated, there exists a vertex $x \in \mathcal{C}$ and a vertex $x' \in \mathcal{C}'$ such that the arc (x, x') belongs to R. If $\mathcal{C}$ est reduced to x, then x is a transition (since every place has an input) without input and with an output x', so R is unbounded. Otherwise every vertex of $\mathcal{C}$ belongs to an elementary circuit. So x, has at least two outputs and it is a transition; x' is a place.

Let us pick the net restricted to $\mathcal{C}$. This subnet is an event graph (since every place belongs to a circuit). Let us choose an initial marking of the subnet with every circuit marked. This subnet is live; so one can fire an infinite sequence including an infinity of occurrences of x. This sequence is also a firing sequence of the initial net which infinitely increases the number of tokens in x'. So R is not structurally bounded. $\diamondsuit$

We achieve the analysis of event graphs by a characterisation of simultaneous liveness and boundedness (we have already established necessary conditions for ordinary nets).

Proposition 59 (Live and bounded event graph)
If R is an event graph, then the following assertions are equivalent:

1 (R, m_0) *is live and bounded.*

2 R *is strongly connected and every elementary circuit is initially marked by* m_0.

3 R *is strongly connected and there exists a firing sequence including exactly an occurrence of every transition.*

Proof
We have already obtain the equivalence of points 2 and 3 and the implication $2 \Rightarrow 1$. For the implication $1 \Rightarrow 2$, it is enough to modify the last part of the previous proof, choosing for initial marking of $\mathcal{C}$, the restriction of m_0 to this component. $\diamondsuit$

5.4. *Free choice net*

In a free choice net, when a place is the input of several transitions all these transitions have the same inputs reduced to this place and thus are always simultaneous fireable justifying the name of the subclass (see figure 10). Observe that the net of figure 3 is also a free choice net.

Definition 60 (Free choice net) *R, a Petri net, is a free choice net if:*
$\forall\, t \in T$,

- $\exists\, P_{in}, P_{out} \subset P$ *with* $Pre(t) = \sum_{p \in P_{in}} \overrightarrow{p}$ *and* $Post(t) = \sum_{p \in P_{out}} \overrightarrow{p}$
- $\forall\, t' \in T,\ {}^\bullet t \cap {}^\bullet t' \neq \emptyset \Rightarrow |P_{in}| = 1$ *et* $Pre(t) = Pre(t')$

We first give a characterisation of liveness of free choice nets. To this aim, we define two properties of a set of places.

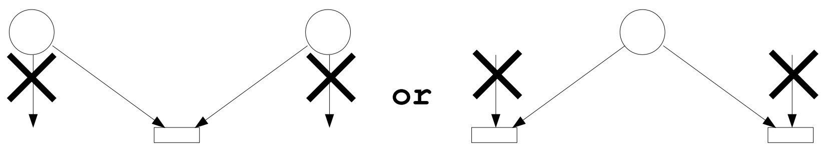

Figure 10: Constraints of a free choice net

Definition 61 *Let R be a Petri net and P' a non empty subset of places then:*

- *P' is a* deadlock *if its inputs are included in its outputs,* $\cup_{p \in P'} {}^{\bullet}p \subset \cup_{p \in P'} p^{\bullet}$
- *P' is a* trap *if its outputs are included in its inputs,* $\cup_{p \in P'} p^{\bullet} \subset \cup_{p \in P'} {}^{\bullet}p$

When a deadlock is unmarked, it will always remain unmarked and every transition output of the deadlock will never fire. When a trap is marked, it will always remain marked. Otherwise stated, a unmarked deadlock is a sufficient condition for non liveness and this cannot happen if the deadlock contains a trap initially marked. This is the starting point of the characterisation of liveness. To this aim, we first define a device to empty places.

Let p be a place of a (non necessarily free choice) Petri net which does not belong to any trap of the net. Then, there exists a sequence of disjoint non empty subsets of places, $P_1, \ldots, P_h$ (determined in a single way by the following construction) such that: $P_h = \{p\}$ and $\forall\, p' \in P_i$, $\exists\, t \in p'^{\bullet}$ s.t. $t^{\bullet} \subset \cup_{j<i} P_j$

The construction proceeds as follows. Let us note $Succ(p)$, the set of places reachable from p by a path in the net (observe that $p \in Succ(p)$). Since $Succ(p)$ is not a trap, the subset of places which have an output transition without output is not empty. If p is such a place, one defines $P_0 = \{p\}$ and one stops. Otherwise P_0 is this subset of places and one considers the subset $Succ(p) \setminus P_0$. This set is not a trap, so the subset of places which have an output transition whose all outputs are in P_0 is not empty. If p is such a place, one defines $P_1 = \{p\}$ and one stops. Otherwise P_1 is this subset and one iterates the process with $Succ(p) \setminus (P_0 \cup P_1)$. Since the set of places is finite, the process must stop.

By construction, every place $p \in P_k$ does not belong to any trap. Given a place p which does not belong to any trap, one denotes by $h(p)$, the number h of the construction. We also define $vide(p)$ as one output transition of p

which has all its outputs $\cup_{j<h(p)} P_j$. Observe that for a place $p' \in \cup_{j<h(p)} P_j$ $h(p') < h(p)$. We now establish the characterisation of live free choice nets.

Theorem 62 (Commoner condition [14, 7])
Let R be a free choice net; then (R, m_0) is live iff every deadlock of R includes a trap initially marked.

Example 16 The net of figure 3 has a deadlock which does not include a trap: $\{A, B, AAH, BAV\}$. So it is a new proof that it cannot be live whatever its initial marking.

Proof
Assume (R, m_0) is not live, and let t be the transition that can never be fired from a reachable marking m. Necessarily from a marking m' reachable from m, one of the input places of t, p will always remains unmarked. Indeed, either t has a single input and the fireability of t is equivalent to the fact that p is marked, or t has several outputs but does not share it *(free choice)* which implies that the number of marked places input of t can only increase and then one picks a marking m' for which this value is maximal. We build a set of unmarked places, initialised to $\{p\}$. Since from m', p is never marked all its input transitions are never fireable. One iterates the previous process and one obtains for all these transitions an input place (possibly p) such that p and these places will remain unmarked from a reachable marking m''. One iterates the process for the new places. Again this process must stop and when it stops all the selected input places are already present in the current (and final) set of places. By construction, this set is unmarked in a reachable marking and it is a deadlock. Using a previous observation on traps, we conclude that it cannot include a trap initially marked.

Assume now that (R, m_0) is live and that there exists a deadlock V not containing a marked trap. We will obtain a contradiction. First, V must be initially marked. Let E be the set of places of V which do not belong to any trap included in V (this set includes the marked places of V). We consider the subnet generated by the places of V and we order places of V, beginning by the places of $V \setminus E$ and ending by places of E, such a place say p is ordered by increasing order of $h(p)$ where h is relative to the subnet. Observe that different orders are possible. Once this order is chosen, we order the vectors of $\mathbb{N}^V$ by the lexicographic order. We are going to prove that one can always decrease the marking restricted to places of V w.r.t. the lexicographic order by a firing sequence. In particular, places of $V \setminus E$ will always remain unmarked during the proces.

Starting from m_0, one fires every possible transition $vide(p)$ for $p \in E$. Any firing consumes (at least) a token of p and produces token in places $p' \in E$ with

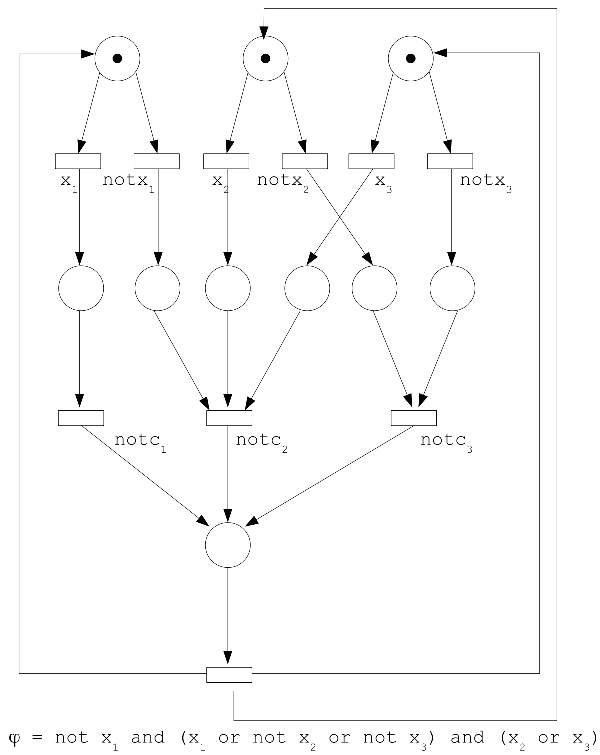

Figure 11: From satisfiability of a formula to non liveness of a free choice net

$h(p') < h(p)$ or in places $p' \in P \setminus V$. The submarking w.r.t. V decreases after every firing until it reaches a marking m_1 where no transition $vide(p)$ is fireable. Let σ be the shortest sequence which enables the firing of a transition $vide(p)$ ($m_1 \xrightarrow{\sigma} m_2$). This sequence cannot provide tokens to V. Indeed by definition of a deadlock, it requires to consume tokens of the deadlock. But all tokens of the deadlock are in places inputs of transitions $vide(p)$ which are not fireable and so these transitions have several inputs. The free choice hypothesis implies that these places are not input of another transition. Thus the submarking of V is unchanged during the firing of σ. From m_2 one fires the transition $vide(p)$ which is become fireable decreasing the submarking of V and one iterates the process. But there are no infinite strictly decreasing sequences in $\mathbb{N}^V$ with lexicographic order pointing out the contradiction. $\diamondsuit$

Using the theorem, one straightforwardly designs a test algorithm for non liveness in $\mathcal{NP}$. One chooses non deterministically a subset of places, then one checks that it is a deadlock and one computes in polynomial time its maximal trap [18] (union of traps included in the deadlock) and one verifies that it is unmarked. Using a (quite simple) reduction of the satisfiability problem for a formula in conjunctive normal form, one proves that the problem of non liveness is $\mathcal{NP}$-complete [16]. We let the reader prove that the net of figure 11 is not live iff the formula stated under the net is satisfiable. Intuitively, the formula is satisfiable if the three negations of its clauses can be simultaneously false. In this case, one can fire three transitions among $\{x_1, notx_1, x_2, notx_2, x_3, notx_3\}$ without enabling any of the three transitions $notc_1, notc_2, notc_3\}$.

We end this study by establishing a characterisation of live and bounded free choice nets. First, the boundedness ensures the equivalence of some behavioural properties.

Proposition 63 *Let (R, m_0) be free choice net strongly connected and bounded then:*

$$(R, m_0) \textit{ is live iff } (R, m_0) \textit{ is pseudo-live}$$

Proof

We have to prove that pseudo-liveness implies liveness. Assume that (R, m_0) is not live; then there exists a reachable marking m and a transition t such that t is never fireable in (R, m). Let p be an input place of t; the marking of p cannot decrease since if p is an input of another transition, these transitions never fire also. Necessarily, the input transitions of p can only occur a finite number of times in an infinite sequence (otherwise p would be unbounded). Starting from m, one can reach a marking m' where these transitions never fire. Iterating this process, one obtains a marking where all transitions with a path to t never fire. Since R is strongly connected, this set of transitions is T and this marking is dead. $\diamondsuit$

The necessary condition of simultaneous liveness and boundedness about the rank of C in Petri nets can be refined for free choice nets.

Proposition 64 *Let R be a free choice net, then:*

$$(R, m_0) \textit{ is live and bounded} \Rightarrow rank(C) = |\Theta| - 1$$

Proof

We reason about the live and bounded (R', m'_0) of proposition 52 obtained after superimposing to (R, m_0) a circuit for every equivalence class of Θ different from a singleton. In this net, we already know that $rank(C') \leq |T| - 1$

and additionally that there exists a T-semiflow v such that $\|v\| = T$ (proposition 35). In order to prove the proposition, we must show that the inequality is an equality. If the inequality is strict, there exists a second T-flow v' (with v, v' linearly independent). W.l.o.g. we assume that there is at least one transition t such that $v(t) > 0$. Among such transitions, let t_0 be a transition that fulfils: $\frac{v'(t_0)}{v(t_0)} = Max(\{\frac{v'(t')}{v(t')} \mid t' \in T', v'(t') > 0\})$. Then $v'' = v'(t_0).v - v(t_0).v'$ is a T-semiflow whose support is **strictly included** in T.

Let t be a transition belonging to the support of a T-semiflow of R'. Due to the additional circuits, every transition of the equivalence class of t' in R also belongs to the support of the T-semiflow. Indeed if this class is not reduced to t, then t produces a token in a place of the circuit only consumed by the next transition of the circuit. Thus this transition must appear in the support of the T-semiflow. By iteration, every transition of the circuit must appear in the support.

Since (R, m_0) is live and bounded, R is strongly connected. Let t be a transition of the support of v'', t' be any transition and consider a path in R from t to t'. We claim that every transition on the path belong to the support of v''. By hypothesis, t belongs to the support. Let $t_1 \neq t'$ be a transition of the path that belongs to the support, let p be the place which follows t' on the path and t_2 the next transition. One of the output transition of p, say t_3, must belong to the support of v'' and so every transition of its equivalence class must also belong to the support. But t_2 belong to this class *(here we have used the hypothesis of the free choice)* so it also belongs to the support. Hence any t' belongs to the support of v'' which contradicts the fact that the support of v'' does not contain every transition. $\diamondsuit$

We establish another necessary condition of simultaneous boundedness and liveness of a free choice net starting from the characterisation of liveness. To this aim, we introduce the notion of subnet and of covering of a net.

Definition 65 *Let (R, m_0) be a Petri net,*

- *let P' be a subset of places; then $(R[P'], m_0[P'])$ is the subnet defined by the subset of places P', the subset of transitions ${}^\bullet P' \cup P'^\bullet$ and the incidence matrices and the initial marking of (R, m_0) restricted to these subsets.*
- *R is covered by marked state machines if every place belongs to a subset P' such that $(R[P'], m_0[P'])$ is a marked state machine.*

One says that a deadlock is *minimal* if it does not contain a strictly smaller deadlock.

Lemma 66 (Characterisation of a minimal deadlock) *Let (R, m_0) be a free choice net and let V be a deadlock then:*

$$V \text{ is minimal iff}$$
$$\forall\, p, p' \in V \text{ there exists a path from } p \text{ to } p' \text{ in } R[V]$$
$$\text{and } \forall\, t \text{ transition of } R[V]\ ,\ |{}^\bullet t| = 1$$

Proof
Let V be a minimal deadlock and let $\mathcal{C}$ be an initial s.c.c. of $R[V]$; $\mathcal{C}$ is not reduced to a transition otherwise V would not be a deadlock. By construction, places of $\mathcal{C}$ constitutes a deadlock thus this set is V which establishes the first condition. Assume that t a transition of $R[V]$ has two inputs. These two places are only inputs of t and deleting any such place one obtains a new deadlock. Any t of $R[V]$ has an input since V is a deadlock.

Assume that V fulfils the characterization of minimality but that there exists V' a deadlock strictly included in V. Let p be a place of $V \setminus V'$ and p' be a place of V'. There exists a path in $R[V]$ from p to p'. Let p'' be the last place belonging to $V \setminus V'$ on the path. The transition which follows p'' is an input of V' and its single input in V, p'', does not belong to V'. Hence V' is not a deadlock. $\diamondsuit$

Lemma 67 (Minimal deadlock of a live and bounded net) *Let (R, m_0) be a live and bounded free choice net and let V be a minimal deadlock then:*

$$R[V] \text{ is a marked state machine}$$

Proof
Since the net is live, V contains a marked trap Tr. Assume that Tr is different from V, then since Tr cannot be a deadlock, there exists a transition t of $R[Tr]$ which has no input. Let us examine the sum of place markings of Tr, this sum cannot decrease since every transition of $R[Tr]$ has at least one output and at most one input. Furthermore the firing of t increases this sum. Since (R, m_0) is live, one can build an infinite sequence including an infinity of occurrences of t contradicting the boundedness of the net. Hence $Tr = V$, V is marked and every transition of $R[V]$ has at least one output and exactly an input. Using the same reasonning as the one for the trap, one establishes that no transition has two outputs and so $R[V]$ is a state machine. $\diamondsuit$

Lemma 68 (Minimal deadlock of a strongly connected net) *Let R be a strongly connected free choice and let p be a place of P then:*

$$p \text{ is contained in a minimal deadlock}$$

Proof
Let p be a place; if $P = \{p\}$ the result is obvious. Otherwise, let $t \in {}^{\bullet}p$. This set is non empty since the net is strongly connected. There exists an elementary path from p to t whose length is minimal. We build the minimal deadlock starting from the places of circuit that we have obtained. We note the current subset of places P'. R' the net restricted to P' and $P'^{\bullet}$. P' fulfils at each step the minimality conditions of a deadlock (without necessarily be a deadlock). Initially, the circuit ensures the strong connectivity between places. Furthermore, no transition cannot have two places of the circuit as input due to the minimality of the path. Suppose that the current subset is not a deadlock of R. Then there exists a transition $t' \in {}^{\bullet}P' \setminus P'^{\bullet}$. Let $p' \in {}^{\bullet}t'$; since the net is strongly connected, there exists an elementary path from any vertex of R' to p'. Let us a choose a path **with minimal length**, every vertex excepted the first one does not belong to R' and (due to the minimality of the path) no place of the path shares its output transitions neither with the other places of the path nor with the places of P'. P' is updated with these new places. One iterates this process which must stop since T is finite. By construction, the final subset P' is a minimal deadlock. $\diamondsuit$

Proposition 69 *Let R be free choice net, then:*
(R, m_0) is live and bounded implies
(R, m_0) is covered by marked state machines

Proof
Using proposition 47 the net is strongly connected. Using the previous lemma, every place p belongs to a minimal deadlock. This deadlock is a marked state machine due to lemma 67. $\diamondsuit$

It turns out that the conjunction of the necessary conditions previously established is a sufficient condition yielding the fundamental theorem about the behaviour of the free choice net called the rank theorem.

Theorem 70 (Rank theorem [11]) *Let R be a free choice net; then (R, m_0) is live and bounded iff the following conditions are met:*

- *R is strongly connected*
- *R is covered by state machines*
- *$rank(C) = |\Theta| - 1$*
- *Every deadlock of R is initially marked*

Proof
Using the previous results, we have only to prove that the condition is sufficient. Since the net is covered by state machines, it is conservative and so structurally bounded.

Let m_1 be a marking that marks every trap of the net; we show that (R, m_1) is live. Suppose the contrary. Using proposition 63 (R, m_1) is not pseudo-live. Let m_2 be a dead marking; every equivalence class of Θ has (at least) one input place unmarked in m_2. We choose such a place per class and we note P', this subset of places.

Since $|P'| = |\Theta|$ and $rank(C) < |\Theta|$, there exists a flow v whose support is included in P'. Let us note $v = v_1 + v_2$ where v_1 is constituted by the positive coefficients of v and $v_2 = v - v_1$. W.l.o.g., one supposes that $v_1 \neq \overrightarrow{0}$. The choice of P' implies that every transition t admits a single input in P'. If this input is not in $\|v_1\|$ then $v_1^t.C(t) \geq 0$. Otherwise this input is not in $\|v_2\|$, but then $v_1^t.C(t) = -v_2^t.C(t) \geq 0$. In conclusion, $v_1^t.C \geq \overrightarrow{0}$. But this is possible only if $\|v_1\|$ is a trap. Due to the choice of m_1 every trap is marked which is contradictory.

Since (R, m_1) is live and bounded, every minimal deadlock de R is a trap (lemma 67). Since in (R, m_0) every deadlock is marked, every deadlock contains a marked trap (i.e. one of its minimal deadlock). Using Commoner condition, (R, m_0) is live. $\diamondsuit$

This result has two outstanding features.

On the one hand, it only relies on the graph structure, the incidence matrix and the initial marking.

On the other hand, one deduces an algorithm which checks the simultaneous boundedness and liveness in polynomial time. The strong connectivity can be checked with the algorithm of Tarjan. The rank of the matrix is computed by a variation of the Gauss elimination. In order to test the covering by state machines, one builds a minimal deadlock containing every place following the proof of lemma 68 and then checks that it is a state machine. At last to check that there does not exist an unmarked deadlock, one restricts the net to the unmarked places and one tests whether the maximal deadlock exists (union of all deadlocks).

Numerous works are relative to generalisations of characterisations to extensions of free choice nets [13, 21, 22, 2]. Similarly, other behavioural properties of free choice nets have been analysed [17, 10]. The reader interested by this class can refer to a book that is devoted to it [9].

References

[1] Alfred V. Aho, John E. Hopcroft, and J. D. Ullman. *The Design and Analysis of Computer Algorithms*. Addison-Wesley, Reading, 1974.

[2] K. Barkaoui, J. M. Couvreur, and C. Dutheillet. On liveness in extended non self-controlling nets. In *Proceeding of the* 16^{th} *International Conference on Application and Theory of Petri Nets*, volume 935 of *LNCS*, pages 25–44, Turin, Juin 1995.

[3] G. Berthelot. Transformations et analyse de reseaux de Petri. application aux protocoles. Thèse d'Etat de l'Université P. et M. Curie Paris, 1983.

[4] G. Berthelot. Transformations and decompositions of nets. In W. Brauer, W. Reisig, and G. Rozenberg, editors, *Advances in Petri Nets '86 - Part I*, volume 254 of *LNCS*, pages 359–376. Springer Verlag, Bad Honnef, Allemagne, Février 1987.

[5] J. M. Colom, J. Martinez, and M. Silva. Packages for validating discrete production systems modeled with Petri nets. In *Proc. of the IMACS-IFAC Symp. Modelling and Simulation for Control of Lumped and Distributed Parameter Systems, Lille*, pages 457–462, 1986.

[6] J. M. Colom and M. Silva. Convex geometry and semiflows in P/T nets. A comparative study of algorithms for computation of minimal P-semiflows. *Lecture Notes in Computer Science; Advances in Petri Nets 1990*, 483:79–112, 1991.

[7] F. Commoner. Deadlocks in Petri nets. Rapport technique CA-7206-2311, Applied Data Research Inc., Wakefield, Massachusetts, 1972.

[8] F. Commoner, A. Holt, S. Even, and A. Pnueli. Marked directed graphs. *Journal of Computer and System Sciences*, 5(5):511–523, Octobre 1971.

[9] J. Desel and Esparza J. *Free-choice Petri Nets*, volume 40 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1995.

[10] J. Esparza. Reachability in live and safe free-choice Petri nets is NP- complete. *Theoretical Computer Science*, 198(1-2):211–224, 1998.

[11] J. Esparza and M. Silva. A polynomial-time algorithm to decide liveness of bounded free choice nets. *Theoretical Computer Science*, 102:185–205, 1992.

[12] S. Even. *Graph Algorithms*. Computer Science Press, Rockville, 1979.

[13] W. Griese. Liveness in NSC-Petri nets. *Discrete Structures and Algorithms*, pages 255–264, 1980.

[14] M. Hack. Analysis of production schemata by Petri nets. Master's thesis, M.I.T., Cambridge, MA, 1972. publié comme rapport de recherche MAC TR-94. Corrigé en 1974.

[15] S. Haddad. Une catégorie régulière de réseau de Petri de haut niveau : définition, propriétés et réductions. application à la validation de systèmes distribués. Thèse de l'Université P. et M. Curie Paris, Chapitre IV, 1987.

[16] N.D. Jones, L.H. Landweber, and Lien Y.E. Complexity of some problems in Petri nets. *Theoretical Computer Science*, 4:277–299, 1977.

[17] D-I. Lee, S. Kumagai, and Kodama S. Handles and reachability analysis of free choice nets. In *Application and Theory of Petri Nets 1995, Proc.* 16^{th} *Intern. Conference*, volume 935 of *LNCS*, pages 298–315, Turin, Italie, June 1995. Springer Verlag.

[18] M. Minoux and K. Barkaoui. Deadlocks and traps in Petri nets as Horn-satisfiability solutions and some related polynomially solvable problems. *Discrete Mathematics*, 29:195–210, 1990.

[19] C. Roos, T. Terlaky, and J.-Ph. Vial. *Theory and Algorithms for Linear Optimization. An Interior Point Approach.* Wiley-Interscience, John Wiley & Sons Ltd, West Sussex, England, 1997.

[20] P. Schnoebelen and N. Sidorova. Bisimulation and the reduction of Petri nets. In *Lecture Notes in Computer Science; Application and Theory of Petri Nets 2000, Proceedings* 21^{st} *International Conference*, volume 1825 of *LNCS*, pages 409–423, Aarhus, Danemark, Juin 2000. Springer Verlag.

[21] E. Teruel and M. Silva. Liveness and home states in equal conflict systems. In *Application and Theory of Petri Nets 1992, Proc.* 14^{th} *Intern. Conference, Chicago, USA*, volume 691 of *LNCS*, Chicago, USA, Juin 1993. Springer Verlag.

[22] E. Teruel and M. Silva. Well-formedness of equal conflict systems. In *Lecture Notes in Computer Science; Application and Theory of Petri Nets 1994, Proceedings* 15^{th} *International Conference, Zaragoza, Spain*, volume 815 of *LNCS*, pages 491–510, Zaragossa, Espagne, Juin 1994. Springer Verlag.

Index