# Recent and Simple Algorithms For Petri Nets*

Alain Finkel[1] and Jérôme Leroux[2]

[1] LSV, ENS Cachan, CNRS & INRIA, France.
firstname.lastname@lsv.ens-cachan.fr
[2] LaBRI, Université de Bordeaux, CNRS
firstname.lastname@labri.fr

**Abstract.** We show how inductive invariants can be used to solve coverability, boundedness and reachability problems for Petri nets. This approach provides algorithms that are conceptually simpler than previously pblished ones.

## 1 Introduction

We revisit the decidability proofs of the following well-known, decidable problems for Petri nets: the *coverability*, the *boundedness*, the *place boundedness* and the *reachability* problem. We present *simple*, *unified* and *generalizable* new or recent algorithms for these four problems.

Our algorithms are *simpler* than the previous ones: this means they can be understood easily, with the background of undergraduate students. We present a proof of decidability for the place boundedness problem that avoids the non-trivial Karp and Miller tree algorithm and only uses invariants. Moreover, our algorithm for deciding reachability is incomparably simpler than all the different existing decidability proofs of Mayr, Kosaraju and Lambert.

The *structure* of these four algorithms consists in finding inductive invariants by simple enumeration. This unifies the proofs and facilitates the understanding.

Finally, our algorithms do not depend on the particularity of Petri nets, hence they can be easily *extended* to apply to other similar models: for instance, our reachability algorithm can be easily adapted to Lossy Channel Systems in replacing Presburger invariants by recognizable invariants. This idea has been used in 2003 by Pachl for (perfect) FIFO Channel Systems in [14].

Our algorithms are not only simple, they also provide *quasi-optimal* complexities for all considered problems, except for the reachability problem whose complexity is not known.

We don't give all the proofs but we give precise references when we omit proofs.

* This work is supported by the french Agence Nationale de la Recherche, REACHARD (grant ANR-11-BS02-001) and by the "Chaire DIGITEO-ENS CACHAN - ECOLE POLYTECHNIQUE".

# 2 Petri Nets

## 2.1 Notations

We denote by $\mathbb{Z}, \mathbb{N}, \mathbb{Q}, \mathbb{Q}_{\geq 0}$ the set of *integers*, *natural numbers*, *rational numbers*, and *non negative rational numbers*. *Vectors* and *sets of vectors* are denoted in bold face. The $i$th *component* of a vector $\boldsymbol{v}$ is written $\boldsymbol{v}(i)$.

We denote by $\leq$ the classical order over $\mathbb{Z}$ and we denote by $\leq$ its component-wise extension over $\mathbb{Z}^d$ defined by $\boldsymbol{x} \leq \boldsymbol{y}$ if $\boldsymbol{x}(i) \leq \boldsymbol{y}(i)$ for every $1 \leq i \leq d$. Given a vector $\boldsymbol{v}$, we write $\|\boldsymbol{v}\|^+$ and $\|\boldsymbol{v}\|^-$ for the sets of indexes $i$ such that $\boldsymbol{v}(i) > 0$ and $\boldsymbol{v}(i) < 0$, respectively.

## 2.2 Petri nets

A *Petri net (net for short)* is a tuple $N = \langle P, T, F \rangle$ where $P$ is a set of *places*, $T$ is a set of *transitions* disjoint from $P$, and $F$ is a *flow function* that maps $(P \times T) \cup (T \times P)$ to $\mathbb{N}$. In the sequel, places are ordered by $P = \{1, \ldots, d\}$. A *marking* $\boldsymbol{m}$ is a vector in $\mathbb{N}^d$. Intuitively, $\boldsymbol{m}(i)$ is the number of *tokens* in the $i$th place.

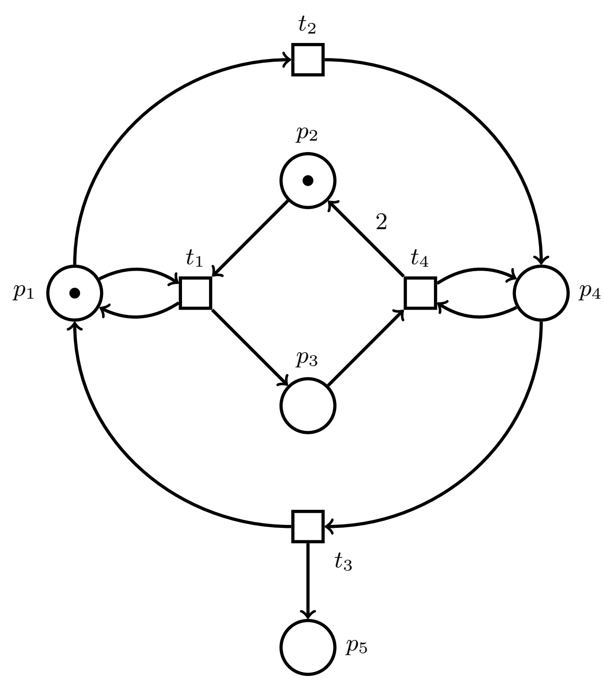


**Fig. 1.** The Hopcroft and Pansiot net.

The operational semantics of a Petri net is formalized by the labeled transition relation defined over markings $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{N}^d$, by $\boldsymbol{x} \xrightarrow{t} \boldsymbol{y}$ where $t \in T$ if $\boldsymbol{x}(p) \geq F(p, t)$

and $\boldsymbol{y}(p) = \boldsymbol{x}(p) - F(p,t) + F(t,p)$ for every $p \in P$. Given a word $w = t_1 \dots t_n$ of transitions $t_j \in T$, we introduce the binary relation $\xrightarrow{w}$ over the markings defined by $\boldsymbol{x} \xrightarrow{w} \boldsymbol{y}$ if there exists a sequence $\boldsymbol{m}_0, \dots, \boldsymbol{m}_n$ of markings such that $\boldsymbol{m}_0 = \boldsymbol{x}$, $\boldsymbol{m}_n = \boldsymbol{y}$, and:

$$\boldsymbol{m}_0 \xrightarrow{t_1} \boldsymbol{m}_1 \dots \xrightarrow{t_n} \boldsymbol{m}_n$$

Given a language $W \subseteq T^*$, we denote by $\xrightarrow{W}$ the union of all $\xrightarrow{w}$, for all $w$, hence, $\xrightarrow{W} = \bigcup_{w \in W} \xrightarrow{w}$. The relation $\xrightarrow{T^*}$ is called the *reachability relation* and it is denoted by $\xrightarrow{*}$, the relation $\xrightarrow{T}$ is called the *one-step reachability relation* and it is also denoted by $\rightarrow$. Note that $\xrightarrow{*}$ is the reflexive and transitive closure of $\rightarrow$. Given a set $\boldsymbol{M} \subseteq \mathbb{N}^d$ of markings, we introduce the following sets:

$$\mathrm{Post}(\boldsymbol{M}) = \bigcup_{\boldsymbol{m} \in \boldsymbol{M}} \{\boldsymbol{y} \in \mathbb{N}^d \mid \boldsymbol{m} \rightarrow \boldsymbol{y}\} \qquad \mathrm{Pre}(\boldsymbol{M}) = \bigcup_{\boldsymbol{m} \in \boldsymbol{M}} \{\boldsymbol{x} \in \mathbb{N}^d \mid \boldsymbol{x} \rightarrow \boldsymbol{m}\}$$
$$\mathrm{Post}^*(\boldsymbol{M}) = \bigcup_{\boldsymbol{m} \in \boldsymbol{M}} \{\boldsymbol{y} \in \mathbb{N}^d \mid \boldsymbol{m} \xrightarrow{*} \boldsymbol{y}\} \qquad \mathrm{Pre}^*(\boldsymbol{M}) = \bigcup_{\boldsymbol{m} \in \boldsymbol{M}} \{\boldsymbol{x} \in \mathbb{N}^d \mid \boldsymbol{x} \xrightarrow{*} \boldsymbol{m}\}$$

In this paper, we present simple algorithms deciding the following classical problems. In these problems, $N$ denotes a Petri net, $\boldsymbol{m}_0, \boldsymbol{m}$ some markings, and $p \in \{1, \dots, d\}$ is a place.

**Reachability** Input: $(N, \boldsymbol{m}_0, \boldsymbol{m})$
**Question**: Is $\boldsymbol{m}_0 \xrightarrow{*} \boldsymbol{m}$ ?

**Coverability** Input: $(N, \boldsymbol{m}_0, \boldsymbol{m})$
**Question**: Does there exist a $\boldsymbol{y}$ such that $\boldsymbol{m}_0 \xrightarrow{*} \boldsymbol{y}$ and $\boldsymbol{y} \geq \boldsymbol{m}$ ?

**Boundedness** Input: $(N, \boldsymbol{m}_0)$
**Question**: Is the set $\mathrm{Post}^*(\boldsymbol{m}_0)$ finite ?

**Place-boundedness** Input: $(N, \boldsymbol{m}_0, p)$
**Question**: Is the set $\{\boldsymbol{m}(p) \mid \boldsymbol{m} \in \mathrm{Post}^*(\boldsymbol{m}_0)\}$ finite ?

*Example 2.1.* The Petri net $N$, depicted in Fig. 1, was introduced in [10] as an example of a Petri net having a reachability set (for some initial marking) that cannot be defined by a Presburger formula. In fact, its reachability set $\mathrm{Post}^*(\{\boldsymbol{m}_0\})$, from the initial marking $\boldsymbol{m}_0 = (1, 1, 0, 0, 0)$, is equal to:

$$\left\{ (x_1, x_2, x_3, x_4, x_5) \in \mathbb{N}^5 \mid \begin{array}{l} (\ x_1 = 1 \wedge x_4 = 0 \wedge 1 \leq x_2 + x_3 \leq 2^{x_5}\ ) \vee \\ (\ x_1 = 0 \wedge x_4 = 1 \wedge 1 \leq x_2 + 2x_3 \leq 2^{x_5+1}\ ) \end{array} \right\}$$

## 3 Upward and Downward Closed Sets

A set $\boldsymbol{U} \subseteq \mathbb{N}^d$ is said to be *upward closed* if for every $(\boldsymbol{u}, \boldsymbol{m}) \in \boldsymbol{U} \times \mathbb{N}^d$, we have $\boldsymbol{u} \leq \boldsymbol{m} \Rightarrow \boldsymbol{m} \in \boldsymbol{U}$. The *upward closure* of a vector $\boldsymbol{u} \in \mathbb{N}^d$ is the set $\{\boldsymbol{m} \in \mathbb{N}^d \mid \boldsymbol{u} \leq \boldsymbol{m}\}$ denoted by $\uparrow\boldsymbol{u}$. The *upward closure* of any set $\boldsymbol{M} \subseteq \mathbb{N}^d$ is the set $\bigcup_{\boldsymbol{m}\in\boldsymbol{M}} \uparrow\boldsymbol{m}$ denoted by $\uparrow\boldsymbol{M}$. Let us observe that $\uparrow\boldsymbol{M}$ is the least upward closed set (for the inclusion) that contains $\boldsymbol{M}$. Since $\mathbb{N}^d$ is well-ordered by $\leq$, for any upward closed set $\boldsymbol{U} \subseteq \mathbb{N}^d$, there exists a finite set $\boldsymbol{F} \subseteq \boldsymbol{U}$ such that $\boldsymbol{U} = \uparrow\boldsymbol{F}$. This means that upward closed sets can be symbolicaly represented by finite sets $\boldsymbol{F} \subseteq \mathbb{N}^d$. Such a set is called an *upward basis* (basis for short) of the upward closed set $\boldsymbol{U}$. One can show that the minimal elements of any basis $\boldsymbol{F}$ of $\boldsymbol{U}$ still form a basis which does not depend on $\boldsymbol{F}$. It is minimal for inclusion among all bases, and is called the *minimal upward basis* (minimal basis for short) of the upward closed set $\boldsymbol{U}$.

*Example 3.1.* Let us consider in $\mathbb{N}^2$ the upward closed set $\big\{(x, y) \in \mathbb{N}^2 \mid x \geq 3 \vee y \geq 1\big\}$. A (non-minimal) basis is $\{(3, 0), (3, 1), (0, 1)\}$. The minimal basis is $\{(3, 0), (0, 1)\}$.

Symmetrically, a set $\boldsymbol{D} \subseteq \mathbb{N}^d$ is a said to be *downward closed* if for every $(\boldsymbol{d}, \boldsymbol{m}) \in \boldsymbol{D} \times \mathbb{N}^d$, we have $\boldsymbol{m} \leq \boldsymbol{d} \Rightarrow \boldsymbol{m} \in \boldsymbol{D}$. The *downward closure* of a vector $\boldsymbol{d} \in \mathbb{N}^d$ is the set $\{\boldsymbol{m} \in \mathbb{N}^d \mid \boldsymbol{m} \leq \boldsymbol{d}\}$ denoted by $\downarrow\boldsymbol{d}$. The downward closure of any set $\boldsymbol{M} \subseteq \mathbb{N}^d$ is the set $\bigcup_{\boldsymbol{m}\in\boldsymbol{M}} \downarrow\boldsymbol{m}$ denoted by $\downarrow\boldsymbol{M}$. Let us observe that $\downarrow\boldsymbol{M}$ is the minimal, for the inclusion, downward closed set that contains $\boldsymbol{M}$. The downward closure of a finite set $\boldsymbol{F} \subseteq \mathbb{N}^d$ is a finite set. Hence, we deduce that, in general, *infinite* downward closed sets, like $\mathbb{N}^d \times \{0\}$, cannot be symbolically represented by finite sets $\boldsymbol{F} \subseteq \mathbb{N}^{d+1}$.

This problem is overcome as follows. Given an ordered set, one may, under suitable assumptions, construct a topological completion of this set to recover a *finite description* of downward closed sets [8,9]. The completion of $\mathbb{N}^d$ is $\mathbb{N}_\omega^d$, with $\mathbb{N}_\omega = \mathbb{N} \cup \{\omega\}$, where we extend $\leq$ by $n \leq \omega$ for all $n \in \mathbb{N}_\omega$. Given $\boldsymbol{x} \in \mathbb{N}_\omega^d$, we denote by $\downarrow\boldsymbol{x}$ the set $\{\boldsymbol{d} \in \mathbb{N}_\omega^d \mid \boldsymbol{d} \leq \boldsymbol{x}\}$. Given a set $\boldsymbol{X} \subseteq \mathbb{N}_\omega^d$, we define the set $\downarrow\boldsymbol{X} = \bigcup_{\boldsymbol{x}\in\boldsymbol{X}} \downarrow\boldsymbol{x}$. The results of [8,9] yield that, if $\boldsymbol{D} \subseteq \mathbb{N}^d$ is downward closed, then $\boldsymbol{D} = \mathbb{N}^d \cap \downarrow\boldsymbol{F}$ for some finite set $\boldsymbol{F} \subseteq \mathbb{N}_\omega^d$ which we call a *downward basis* (basis for short when it is not confusing)) of $\boldsymbol{D}$. One can show that the maximal elements of any basis $\boldsymbol{F}$ of $\boldsymbol{D}$ still form a basis which does not depend on $\boldsymbol{F}$. It is minimal for inclusion among all bases, and is called the *minimal downward basis* (minimal basis for short when it is not confusing) of the downward closed set $\boldsymbol{D}$.

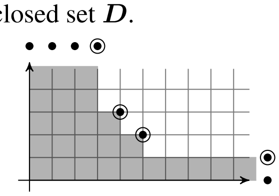

**Fig. 2.** A downward closed set.

*Example 3.2.* Let us consider in $\mathbb{N}^2$ the downward closed set $\{(x,y) \in \mathbb{N}^2 \mid x \leq 3 \vee y \leq 1\} \cup \{(4,2),(4,3),(5,2)\}$ depicted in Fig. 2. A (non-minimal) basis is $(\{0,1,2,3\} \times \{\omega\}) \cup \{(4,3),(5,2)\} \cup (\{\omega\} \times \{0,1\})$. It is shown with dots $\bullet$ in the figure, where elements involving $\omega$ fall beyond the grid. The elements of the minimal basis are circled.

In the following, we will define some algorithms that need the operational semantics of Petri nets to be extended over $\mathbb{N}^d_\omega$. More formally, we have $\boldsymbol{x} \xrightarrow{t} \boldsymbol{y}$ for (extended) markings $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{N}^d_\omega$ if $\boldsymbol{x}(p) \geq F(p,t)$ and $\boldsymbol{y}(p) = \boldsymbol{x}(p) + (F(t,p) - F(p,t))$ for every $p \in P$ where the addition is defined by $\omega + z = \omega$ for every $z \in \mathbb{Z}$. The relations $\xrightarrow{w}$, where $w \in T^*$, $\xrightarrow{*}$ and $\rightarrow$, and the sets $\mathrm{Post}(\boldsymbol{M})$, $\mathrm{Pre}(\boldsymbol{M})$, $\mathrm{Post}^*(\boldsymbol{M})$, and $\mathrm{Pre}^*(\boldsymbol{M})$ where $\boldsymbol{M} \subseteq \mathbb{N}^d_\omega$ are defined as well. The four considered reachability problems are still over $\mathbb{N}^d$.

The *coverability set* (the *cover* for short) is defined as the downward closure of the reachability set: $\mathrm{Cover}(\boldsymbol{m}_0) = \downarrow\mathrm{Post}^*(\boldsymbol{m}_0)$, hence there exists a finite basis $\boldsymbol{F} \subseteq \mathbb{N}^d_\omega$ such that $\mathrm{Cover}(\boldsymbol{m}_0) = \mathbb{N}^d \cap \downarrow\boldsymbol{F}$. For a Petri net $N$ and an initial marking $\boldsymbol{m}_0$, the computation of a finite basis $\boldsymbol{F}$ of the coverability set allows to decide the coverability, the boundedness and the place-boundedness problems (and also other problems like the regularity of the language of traces).

*Example 3.3.* Let us come back to the Petri net $N$ depicted in Fig. 1. Observe that $\mathrm{Cover}(\boldsymbol{m}_0) = \mathbb{N}^d \cap \downarrow\boldsymbol{F}$ where $\boldsymbol{F} = \{(1,\omega,\omega,0,\omega),(0,\omega,\omega,1,\omega)\}$ is the minimal basis (called the clover in [9]).

## 4 Coverability

### 4.1 Coverability is semi-decidable

Recall that a marking $\boldsymbol{x} \in \mathbb{N}^d$ is *coverable* from $\boldsymbol{x}_0$ if, and only if, there exists a *reachable* marking $\boldsymbol{y}$ larger than or equal to $\boldsymbol{x}$. Coverability is semi-decidable for Petri nets. Coverability is also semi-decidable for general machines having a *finite* number of transitions and for which the computation of the one-step reachability set is effective (as Minsky machines and Turing machines).

**Semi-algorithm coverability**$(N, \boldsymbol{x}_0, \boldsymbol{x})$
1. Enumerate all the words $w \in T^*$ and **check**:
2. **if** $\boldsymbol{x}_0 \xrightarrow{w} \boldsymbol{y}$ and $\boldsymbol{y} \geq \boldsymbol{x}$
3. **return** "$\boldsymbol{x}$ is coverable"

To prove that non-coverability is also semi-decidable, we must find a witness in the case where $\boldsymbol{m}$ is not coverable.

### 4.2 Non-coverability is semi-decidable

We may give two different proofs that non-coverability is semi-decidable. The first semi-algorithm is based on the following equivalence:

$$\text{For every } \boldsymbol{y} \text{ such that } \boldsymbol{m}_0 \xrightarrow{*} \boldsymbol{y}, \text{ we have } \boldsymbol{y} \not\geq \boldsymbol{m}$$
$$\Longleftrightarrow$$
$$\text{There exists a finite set } \boldsymbol{F} \subseteq \mathbb{N}^d \text{ such that } \boldsymbol{m}_0 \notin \uparrow\boldsymbol{F},\ \boldsymbol{m} \in \uparrow\boldsymbol{F} \text{ and } \mathrm{Pre}(\uparrow\boldsymbol{F}) \subseteq \uparrow\boldsymbol{F}.$$

($\Leftarrow$) : Naturally, if there exists a finite set $\boldsymbol{F} \subseteq \mathbb{N}^d$ such that $\boldsymbol{m}_0 \notin \uparrow\boldsymbol{F}$, $\boldsymbol{m} \in \uparrow\boldsymbol{F}$, and $\mathrm{Pre}(\uparrow\boldsymbol{F}) \subseteq \uparrow\boldsymbol{F}$, by induction, we may deduce that $\mathrm{Pre}^*(\uparrow\boldsymbol{m}) \subseteq \uparrow\boldsymbol{F}$. Since $\boldsymbol{m}_0 \notin \uparrow\boldsymbol{F}$, we deduce that $\boldsymbol{m}_0 \notin \mathrm{Pre}^*(\uparrow\boldsymbol{m})$. Thus for every $\boldsymbol{y}$ such that $\boldsymbol{m}_0 \xrightarrow{*} \boldsymbol{y}$, we have $\boldsymbol{y} \not\geq \boldsymbol{m}$.

($\Rightarrow$) : In this case, $\boldsymbol{m}_0 \notin \mathrm{Pre}^*(\uparrow\boldsymbol{m})$. Let us remark that for all Well Structured Transition Systems [6] (hence for Petri nets), this set is upward closed. Hence there exists a finite basis, say $\boldsymbol{F} \subseteq \mathbb{N}^d$, such that $\mathrm{Pre}^*(\uparrow\boldsymbol{m}) = \uparrow\boldsymbol{F}$. Now, just observe that $\boldsymbol{m}_0 \notin \uparrow\boldsymbol{F}$ and $\mathrm{Pre}(\uparrow\boldsymbol{F}) \subseteq \uparrow\boldsymbol{F}$.

In order to implement the test $\mathrm{Pre}(\uparrow\boldsymbol{F}) \subseteq \uparrow\boldsymbol{F}$, let us first remark that the set $\mathrm{Pre}(\uparrow\boldsymbol{m})$ is upward-closed for every $\boldsymbol{m} \in \mathbb{N}^d$, and we may effectively compute a finite basis of $\mathrm{Pre}(\uparrow\boldsymbol{m})$ as follows. First notice the following equality:

$$\mathrm{Pre}(\uparrow\boldsymbol{m}) \;=\; \bigcup_{t \in T} \uparrow\boldsymbol{m}_t$$

where $\boldsymbol{m}_t$ is the unique minimum marking such $\boldsymbol{m}_t \xrightarrow{t} \boldsymbol{m}'_t$ for some marking $\boldsymbol{m}'_t \geq \boldsymbol{m}$. The marking $\boldsymbol{m}_t$ satisfies the following equality for every place $p \in P$:

$$\boldsymbol{m}_t(p) = \max\{F(p,t), \boldsymbol{m}(p) + F(p,t) - F(t,p)\}$$

We denote by $\mathrm{pb}(\boldsymbol{m}) = \{\boldsymbol{m}_t \mid t \in T\}$ a finite basis of $\mathrm{Pre}(\uparrow\boldsymbol{m})$. Observe that the basis $\mathrm{pb}(\boldsymbol{m})$ is not necessarily the minimal upward basis. For a given finite set $\boldsymbol{F} \subseteq \mathbb{N}^d$, we denote by $\mathrm{pb}(\boldsymbol{F})$ the set $\bigcup_{\boldsymbol{m} \in \boldsymbol{F}} \mathrm{pb}(\boldsymbol{m})$. Hence, one may deduce the following semi-algorithm:

**Semi-algorithm non-coverability-1**$(N, \boldsymbol{m}_0, \boldsymbol{m})$
1. Enumerate all the finite subsets $\boldsymbol{F} \subseteq \mathbb{N}^d$ and **check**:
2. **if** $\boldsymbol{m}_0 \notin \uparrow\boldsymbol{F}$ and $\boldsymbol{m} \in \uparrow\boldsymbol{F}$ and $\mathrm{pb}(\boldsymbol{F}) \subseteq \uparrow\boldsymbol{F}$
3. **return** "$\boldsymbol{m}$ is not coverable"

The enumeration of line 1 can be algorithmicaly implemented by enumerating for all the natural numbers $n$, all the finite subsets of $\{0, \ldots, n\}^d$. The three tests of line 2 are decidable. For instance $\boldsymbol{m} \in \uparrow\boldsymbol{F}$ is equivalent to $\exists \boldsymbol{f} \in \boldsymbol{F},\ \boldsymbol{m} \geq \boldsymbol{f}$ which can be decided since $\boldsymbol{F}$ is finite. The condition $\mathrm{pb}(\boldsymbol{F}) \subseteq \uparrow\boldsymbol{F}$ is equivalent to $\forall \boldsymbol{x} \in \mathrm{pb}(\boldsymbol{F})\ \exists \boldsymbol{f} \in \boldsymbol{F}\ \boldsymbol{x} \geq \boldsymbol{f}$. When line 3 is executed, the equivalence relation proved at the beginning of the section, ensures correctness of the semi-algorithm.

A second and similar proof for the semi-decidability of non-coverability is based on the following equivalence:

$$\text{For every } \boldsymbol{y} \text{ such that } \boldsymbol{m}_0 \xrightarrow{*} \boldsymbol{y}, \text{ we have } \boldsymbol{y} \not\geq \boldsymbol{m}$$
$$\Longleftrightarrow$$
$$\text{There exists a finite set } \boldsymbol{F} \subseteq \mathbb{N}_\omega^d \text{ such that } \boldsymbol{m}_0 \in {\downarrow}\boldsymbol{F}, \boldsymbol{m} \notin {\downarrow}\boldsymbol{F}, \text{ and } \mathrm{Post}({\downarrow}\boldsymbol{F}) \subseteq {\downarrow}\boldsymbol{F}.$$

$(\Leftarrow)$ : Assume that there exists a finite set $\boldsymbol{F} \subseteq \mathbb{N}_\omega^d$ such that $\boldsymbol{m}_0 \in {\downarrow}\boldsymbol{F}$, $\boldsymbol{m} \notin {\downarrow}\boldsymbol{F}$, and $\mathrm{Post}({\downarrow}\boldsymbol{F}) \subseteq {\downarrow}\boldsymbol{F}$. By induction we deduce that ${\downarrow}\mathrm{Post}^*(\boldsymbol{m}_0) \subseteq {\downarrow}\boldsymbol{F}$. Since $\boldsymbol{m} \notin {\downarrow}\boldsymbol{F}$, we get $\boldsymbol{m} \notin {\downarrow}\mathrm{Post}^*(\boldsymbol{m}_0)$. Hence for every $\boldsymbol{y}$ such that $\boldsymbol{m}_0 \xrightarrow{*} \boldsymbol{y}$, we have $\boldsymbol{y} \not\geq \boldsymbol{m}$.

$(\Rightarrow)$ : Conversely, assume that for every $\boldsymbol{y}$ such that $\boldsymbol{m}_0 \xrightarrow{*} \boldsymbol{y}$, we have $\boldsymbol{y} \not\geq \boldsymbol{m}$. Since ${\downarrow}\mathrm{Post}^*(\boldsymbol{m}_0)$ is downward closed, we deduce that there exists a finite basis $\boldsymbol{F} \subseteq \mathbb{N}_\omega^d$ of this set. Now just observe that $\mathrm{Post}({\downarrow}\boldsymbol{F}) \subseteq {\downarrow}\boldsymbol{F}$.

Testing the inclusion $\mathrm{Post}({\downarrow}\boldsymbol{F}) \subseteq {\downarrow}\boldsymbol{F}$ can be implemented by observing that this condition is equivalent to $\mathrm{Post}(\boldsymbol{F}) \subseteq {\downarrow}\boldsymbol{F}$. Hence, non-coverability is semi-decidable by using the following semi-algorithm:

**Semi-algorithm non-coverability-2**$(N, \boldsymbol{m}_0, \boldsymbol{m})$
1. Enumerate all the finite subsets $\boldsymbol{F} \subseteq \mathbb{N}_\omega^d$ and **check**:
2. **if** $\boldsymbol{m}_0 \in {\downarrow}\boldsymbol{F}$ and $\boldsymbol{m} \notin {\downarrow}\boldsymbol{F}$ and $\mathrm{Post}(\boldsymbol{F}) \subseteq {\downarrow}\boldsymbol{F}$
3. **return** "$\boldsymbol{m}$ is not coverable"

*Remark 4.1.* These two semi-algorithms are very similar since for every partition $(\boldsymbol{X}, \boldsymbol{Y})$ of $\mathbb{N}^d$, we have $\mathrm{Post}(\boldsymbol{X}) \subseteq \boldsymbol{X}$ if, and only if $\mathrm{Pre}(\boldsymbol{Y}) \subseteq \boldsymbol{Y}$.

From these two semi-algorithms (for coverability and for non-coverability), we deduce that coverability is decidable.

### 4.3 An algorithm for deciding coverability

The following algorithm works in $\mathbb{N}^d$ (and not in $\mathbb{N}_\omega^d$). Let us still remark that for two markings $\boldsymbol{m_0}, \boldsymbol{m} \in \mathbb{N}^d$, the property "$\boldsymbol{m}$ is coverable from $\boldsymbol{m}_0$" is equivalent to "$\boldsymbol{m}_0 \in \mathrm{Pre}^*({\uparrow}\boldsymbol{m})$". We now present a backward algorithm to compute a finite basis of the upward closed set $\mathrm{Pre}^*({\uparrow}\boldsymbol{m})$. We will present a forward algorithm for coverability in Section 6.

**Algorithm Coverability**$(N, \boldsymbol{m}_0, \boldsymbol{m})$
1. Let $\boldsymbol{F} := \{\boldsymbol{m}\}$;
2. **while** $\mathrm{pb}(\boldsymbol{F}) \not\subseteq {\uparrow}\boldsymbol{F}$
3. $\boldsymbol{F} := \boldsymbol{F} \cup \mathrm{pb}(\boldsymbol{F})$;
4. **If** $\boldsymbol{m}_0 \in {\uparrow}\boldsymbol{F}$ **then return** "$\boldsymbol{m}$ is coverable" **else return** "$\boldsymbol{m}$ is not coverable"

The termination and the correctness of this algorithm are obtained as follows. Let $\boldsymbol{F}_n$ be the value of the set $\boldsymbol{F}$ at line 2 at the $n^{th}$ iteration of the while-loop. An immediate induction shows the following equality:

$$\uparrow\boldsymbol{F}_n = \bigcup_{\boldsymbol{y} \in \uparrow\boldsymbol{m}} \{\boldsymbol{x} \in \mathbb{N}^d \mid \exists w \in T^*,\ \boldsymbol{x} \xrightarrow{w} \boldsymbol{y} \ \wedge\ |w| \leq n\}$$

We deduce that if the while loop condition "$\mathrm{pb}(\boldsymbol{F}_n) \subseteq \uparrow\boldsymbol{F}_n$" holds then $\uparrow\boldsymbol{F}_n = \mathrm{Pre}^*(\uparrow\boldsymbol{m})$. Hence the algorithm is correct.

For the termination, observe that there exists a finite basis $\boldsymbol{G}$ of $\mathrm{Pre}^*(\uparrow\boldsymbol{m})$. Since every element of $\boldsymbol{G}$ is a member of $\uparrow\boldsymbol{F}_n$ for some large enough $n$, we deduce that there exists an $n_0$ such that $\boldsymbol{G} \subseteq \uparrow\boldsymbol{F}_{n_0}$. In this case, we deduce that $\uparrow\boldsymbol{F}_{n_0} = \mathrm{Pre}^*(\uparrow\boldsymbol{m})$. Therefore $\mathrm{pb}(\boldsymbol{F}_{n_0}) \subseteq \uparrow\boldsymbol{F}_{n_0}$ and the algorithm terminates.

# 5 Boundedness and Place-boundedness

## 5.1 An algorithm for deciding boundedness

Boundedness is semi-decidable for Petri nets. In fact, if the reachability set from an initial marking $\boldsymbol{m}_0 \in \mathbb{N}^d$ is finite, one can effectively compute it. We deduce the following semi-algorithm where the set $\boldsymbol{F}$ is always a finite subset of $\mathbb{N}^d$.

**Semi-algorithm boundedness**$(N, \boldsymbol{m}_0)$
1. $\boldsymbol{F} \leftarrow \{\boldsymbol{m}_0\}$
2. **while** $\mathrm{Post}(\boldsymbol{F}) \not\subseteq \boldsymbol{F}$ **do**
3. $\quad \boldsymbol{F} \leftarrow \boldsymbol{F} \cup \mathrm{Post}(\boldsymbol{F})$
4. **return** “bounded”

The non-boundedness is proved to be semi-decidable as follows. Since the reachability tree is *finite-branching*, a Petri net is not bounded from an initial marking $\boldsymbol{m}_0$ if, and only if, there exist[3] $w, \sigma \in T^*$ such that $\boldsymbol{m}_0 \xrightarrow{w} \boldsymbol{x} \xrightarrow{\sigma} \boldsymbol{y}$ with $\boldsymbol{x} \leq \boldsymbol{y}$ and $\boldsymbol{x} \neq \boldsymbol{y}$. Based on this property, we deduce a semi-algorithm that enumerates the words $w, \sigma \in T^*$ and check the previous conditions.

**Semi-algorithm non-boundedness**$(N, \boldsymbol{m}_0)$
1. Enumerate all the pair of words $(w, \sigma)$ and **check**:
2. $\quad$ **if** one has: $\boldsymbol{m}_0 \xrightarrow{w} \boldsymbol{x} \xrightarrow{\sigma} \boldsymbol{y}$ with $\boldsymbol{x} \leq \boldsymbol{y}$ and $\boldsymbol{x} \neq \boldsymbol{y}$
3. $\quad\quad$ **return** “unbounded”

## 5.2 An algorithm for deciding place boundedness

Place boundedness is proved to be semi-decidable for Petri nets thanks to the following equivalence:

The place $p$ is bounded from the initial marking $\boldsymbol{m}_0 \in \mathbb{N}^d$

$$\Longleftrightarrow$$

There exists a finite set $\boldsymbol{F} \subseteq \mathbb{N}_\omega^d$ such that $\boldsymbol{m}_0 \in \downarrow\boldsymbol{F} \cap \mathbb{N}^d$, $\mathrm{Post}(\boldsymbol{F}) \subseteq \downarrow\boldsymbol{F}$ and $\boldsymbol{m}(p) \neq \omega$ for every $\boldsymbol{m} \in \boldsymbol{F}$.

$(\Rightarrow)$ : Let $\boldsymbol{F} \subseteq \mathbb{N}_\omega^d$ be a finite basis of the coverability set $\mathrm{Cover}(\boldsymbol{m}_0)$ where $\boldsymbol{m}_0 \in \mathbb{N}^d$. Since $p$ is bounded, we deduce that for every $\boldsymbol{m} \in \boldsymbol{F}$, we have $\boldsymbol{m}(p) \in \mathbb{N}$.

[3] Words that can be iterated are denoted with the letter $\sigma$.

Note that $\boldsymbol{F}$ satisfies $\boldsymbol{m}_0 \in \downarrow\boldsymbol{F}$ and $\mathrm{Post}(\boldsymbol{F}) \subseteq \downarrow\boldsymbol{F}$.

($\Leftarrow$) : Assume that there exists a finite set $\boldsymbol{F} \subseteq \mathbb{N}_\omega^d$ such that $\boldsymbol{m}_0 \in \downarrow\boldsymbol{F}$, $\mathrm{Post}(\boldsymbol{F}) \subseteq \downarrow\boldsymbol{F}$, and $\boldsymbol{m}(p) \in \mathbb{N}$ for every $\boldsymbol{m} \in \boldsymbol{F}$. Then from $\mathrm{Post}^*(\boldsymbol{m}_0) \subseteq \mathbb{N}^d \cap \downarrow\boldsymbol{F}$, we deduce that $p$ is bounded. Hence, place boundedness is semi-decided by the following semi-algorithm.

**Semi-algorithm place boundedness**$(N, \boldsymbol{m}_0, p)$
1. Enumerate all the finite subsets $\boldsymbol{F} \subseteq \mathbb{N}_\omega^d$ and **check**:
2. **If** $\boldsymbol{m}_0 \in \downarrow\boldsymbol{F}$ and $\mathrm{Post}(\boldsymbol{F}) \subseteq \downarrow\boldsymbol{F}$ and $\boldsymbol{m}(p) \neq \omega$ for every $\boldsymbol{m} \in \boldsymbol{F}$
3. **return** "place $p$ is bounded"

Place non-boundedness is semi-decidable as follows. Based on Higman's Lemma over the runs, we may prove [4] that a place $p$ is unbounded from $\boldsymbol{m}_0$ if, and only if, there exists a run (where $d$ is the number of places, $w_1, \sigma_1, \ldots, w_d, \sigma_d \in T^*$):

$$\boldsymbol{m}_0 \xrightarrow{w_1} \boldsymbol{x}_1 \xrightarrow{\sigma_1} \boldsymbol{y}_1 \cdots \xrightarrow{w_d} \boldsymbol{x}_d \xrightarrow{\sigma_d} \boldsymbol{y}_d$$

satisfying the two following (set of) inequalities:
(1) $\boldsymbol{y}_1 + \cdots + \boldsymbol{y}_j \geq \boldsymbol{x}_1 + \cdots + \boldsymbol{x}_j$ for every $1 \leq j \leq d$, and
(2) $\boldsymbol{y}_1(p) + \cdots + \boldsymbol{y}_d(p) > \boldsymbol{x}_1(p) + \cdots + \boldsymbol{x}_d(p)$.

We deduce the following semi-algorithm for place non-boundedness:

**Semi-algorithm place non-boundedness**$(N, \boldsymbol{m}_0, p)$
1. Enumerate all the finite sequences of $2d$ words $(w_1, \sigma_1, \ldots, w_d, \sigma_d)$ and **check**:
2. **if** $\boldsymbol{m}_0 \xrightarrow{w_1} \boldsymbol{x}_1 \xrightarrow{\sigma_1} \boldsymbol{y}_1 \cdots \xrightarrow{w_d} \boldsymbol{x}_d \xrightarrow{\sigma_d} \boldsymbol{y}_d$ and
3. $\boldsymbol{y}_1 + \cdots + \boldsymbol{y}_j \geq \boldsymbol{x}_1 + \cdots + \boldsymbol{x}_j$ for every $1 \leq j \leq d$ and
4. $\boldsymbol{y}_1(p) + \cdots + \boldsymbol{y}_d(p) > \boldsymbol{x}_1(p) + \cdots + \boldsymbol{x}_d(p)$
5. **return** "place $p$ is unbounded"

## 6 A Single Algorithm for Solving the Three Previous Problems

We wish to decide the three previous problems and also, for instance, to compute the maximal values of each place. This is possible by computing *a* finite basis of the coverability set $\mathrm{Cover}(\boldsymbol{m}_0)$. The following is a simple algorithm that computes *a* finite basis for $\mathrm{Cover}(\boldsymbol{m}_0)$.

**Procedure clover**$(N, \boldsymbol{m}_0)$
1. $\boldsymbol{F} \leftarrow \{\boldsymbol{m}_0\}$
2. **while** $\mathrm{Post}(\boldsymbol{F}) \not\subseteq \downarrow\boldsymbol{F}$ **do**
   (a) Choose fairly (see below) $(w, \boldsymbol{m}) \in T^* \times \boldsymbol{F}$ such that $\boldsymbol{m} \xrightarrow{w} \boldsymbol{y}$ with $\boldsymbol{m} \leq \boldsymbol{y}$
   (b) $\boldsymbol{F} \leftarrow \boldsymbol{F} \cup \{\boldsymbol{m} + \omega(\boldsymbol{y} - \boldsymbol{m})\}$
3. **return** $\boldsymbol{F}$

In this algorithm, $\boldsymbol{m} + \omega(\boldsymbol{y} - \boldsymbol{m})$ denotes the vector $\boldsymbol{v} \in \mathbb{N}_\omega^d$ defined as $\boldsymbol{v}(i) = \boldsymbol{m}(i)$ if $\boldsymbol{y}(i) = \boldsymbol{m}(i)$ and $\boldsymbol{v}(i) = \omega$ if $\boldsymbol{y}(i) > \boldsymbol{m}(i)$.

The fairness condition that line (2.a) must implement is defined as follows. Let us denote by $\boldsymbol{F}_n$ the set $\boldsymbol{F}$ of vectors at line (2) before the $n^{th}$ execution of the while-loop, and let us introduce $\boldsymbol{M} = \bigcup_{n\in\mathbb{N}} \boldsymbol{F}_n$. The fairness condition means that on every execution, every pair $(w, \boldsymbol{m}) \in T^* \times \boldsymbol{M}$ is chosen at least once at line (2).

When the algorithm terminates, we observe that $\mathbb{N}^d \cap \downarrow\boldsymbol{F}$ is the coverability set. The termination is obtained thanks to the Koenig's and the Dickson's lemmata.

We may also compute *the* (unique) set of maximal elements of the limit of the cover, (called the minimal coverability set in [7] and called the *clover* in [9]). In this case, the set of maximal elements of $\boldsymbol{F}$ is *the* clover.

## 7 Reachability

### 7.1 Reachability is semi-decidable

In this Section, we will stay in the set $\mathbb{N}^d$ and we will not use the $\omega$. The reachability problem is semi-decidable with the following semi-algorithm:

**Semi-algorithm reachability**$(N, \boldsymbol{m}_0, \boldsymbol{m})$
1. Enumerate all the words $w \in T^*$ and **check**:
2. **if** $\boldsymbol{m}_0 \xrightarrow{w} \boldsymbol{m}$
3. **return** "$\boldsymbol{m}$ is reachable"

### 7.2 Non-reachability is semi-decidable

To prove that non-reachability is also semi-decidable, we must find a witness proving that a configuration $\boldsymbol{m}$ is not reachable. Formulas in the decidable logic $\mathrm{FO}(\mathbb{N}, +)$, a.k.a the Presburger arithmetic, will provide these witnesses.

In [12], we proved that reachability sets can be over-approximated in such a way that any non-reachable marking can be witnessed by a Presburger inductive invariant.

**Theorem 7.1 ([12]).** *If a marking $\boldsymbol{m}$ is not reachable from an initial marking $\boldsymbol{m}_0$ then there exists a set $\boldsymbol{M} \subseteq \mathbb{N}^d$ definable in the Presburger arithmetic such that $\mathrm{Post}(\boldsymbol{M}) \subseteq \boldsymbol{M}$, $\boldsymbol{m}_0 \in \boldsymbol{M}$, and $\boldsymbol{m} \notin \boldsymbol{M}$.*

We deduce that the following semi-algorithm "decides" the non-reachability problem. In fact, the conditions on line 2 can be implemented as a satisfiability problem for the Presburger arithmetic.

**Semi-algorithm non-reachability**$(N, \boldsymbol{m}_0, \boldsymbol{m})$
1. Enumerate all the Presburger formulas denoting Presburger sets $\boldsymbol{M} \subseteq \mathbb{N}^d$
2. **if** $\mathrm{Post}(\boldsymbol{M}) \subseteq \boldsymbol{M}$, $\boldsymbol{m}_0 \in \boldsymbol{M}$, and $\boldsymbol{m} \notin \boldsymbol{M}$
3. **return** "$\boldsymbol{m}$ is not reachable"

## 8 Conclusion

The *coverability*, the *boundedness*, and the *place boundedness problems* are proved to be decidable thanks to inductive invariants that are upward or downward closed. The *reachability problem* is proved to be decidable with inductive invariants that are in the richer class of Presburger definable sets (every upward or downward closed set in $\mathbb{N}_\omega^d$ is Presburger definable).

Another kind of result that can be easily deduced from the presented algorithms is that the boundedness and the place-boundedness problems are recursively enumerable for Lossy Channel Systems (to the best of our knowledge, this was not known).

From a complexity point of view, coverability, boundedness, and place-boundedness problems are known to be EXPSPACE-complete with Rackoff's technique. In fact, on the positive instances of the coverability problem, and the negative instances of the boundedness, and place-boundedness problems, there exist runs with lengths bounded by a double exponential in the size of the problem input (encoded in binary) proving that the three semi-algorithms **coverability**, **non-boundedness**, and **place-non-boundedness** can be directly transformed into optimal EXPSPACE *algorithms*. Concerning the other algorithms and semi-algorithms, the following table sums up the sizes (encoded in binary), for some Petri nets, of the structures (words, sequences of words, or finite sets) enumerated or computed by the algorithms or by the semi-algorithms on the terminating instances. Let us analyze the size of the set searched by the semi-algorithm **boundedness** in Section 5.1. We know from [13,5] that there exists an infinite sequence of Petri nets $(N_n)_{n\in\mathbb{N}}$, having *finite* reachability sets, which only contain incomparable elements, such that the number of elements in $Post^*_{N_n}(\boldsymbol{m}_0)$ is not primitive-recursive but Ackermannian in the size of $N_n$. The semi-algorithm **boundedness** seeks an inductive invariant $\boldsymbol{F}$ which must, at least, contain $Post^*_{N_n}(\boldsymbol{m}_0)$ as a subset. If we use this semi-algorithm on the sequence $(N_n)_{n\in\mathbb{N}}$, it will necessarily find a set that is larger than $Ack(size(N_n))$, hence we deduce the complexity. The same reasoning may be done for the semi-algorithm **place boundedness** and for the algorithm **clover**.

All these given sizes are optimal since they can be reached on some instances.

| Section | Algorithm/Semi-algorithm | minimal size for some instances |
|---|---|---|
| 4.1 | **coverability** | 2-EXP [3,15] |
| 4.2 | **non-coverability-1&2** | 2-EXP [3,2] |
| 4.3 | **coverability** | 2-EXP [3,2] |
| 5.1 | **boundedness** | Ackermann [13,5] |
| 5.1 | **non-boundedness** | 2-EXP [3,15] |
| 5.2 | **place boundedness** | Ackermann [13,5] |
| 5.3 | **place non-boundedness** | 2-EXP [15,2] |
| 6 | **clover** | Ackermann [13,5] |

The complexity of the reachability problem is known to be between EXPSPACE-hard [3] and decidable from Mayr and Kosaraju and more recently from Leroux [12]. Reducing this complexity gap is an open problem. Note that even the existence of a primitive recursive upper bound on the complexity of the reachability problem is still open ([1] introduced such a bound but it was proved to be incorrect in [11]).

**Acknowledgements:** We thank Pierre McKenzie and Wolfgang Reisig for their careful reading of the paper.

## References

1. Bouziane, Z.: A primitive recursive algorithm for the general petri net reachability problem. In: FOCS 1998. (nov 1998) 130 –136
2. Bozzelli, L., Ganty, P.: Complexity analysis of the backward coverability algorithm for vass. In Delzanno, G., Potapov, I., eds.: Reachability Problems - 5th International Workshop, RP 2011, Genoa, Italy, September 28-30, 2011. Proceedings. Volume 6945 of Lecture Notes in Computer Science, Springer (2011) 96–109
3. Cardoza, E., Lipton, R.J., Meyer, A.R.: Exponential space complete problems for Petri nets and commutative semigroups: Preliminary report. In: STOC'76, ACM (1976) 50–54
4. Demri, S.: On selective unboundedness of vass. In Chen, Y.F., Rezine, A., eds.: Proceedings 12th International Workshop on Verification of Infinite-State Systems. Volume 39 of EPTCS (2010) 1–15
5. Figueira, D., Figueira, S., Schmitz, S., Schnoebelen, P.: Ackermannian and primitive-recursive bounds with dickson's lemma. In: Logic in Computer Science (LICS), 2011 26th Annual IEEE Symposium on. (june 2011) 269 –278
6. Finkel, A., Schnoebelen, Ph.: Well-structured transition systems everywhere! Theoret. Comput. Sci. **256**(1–2) (2001) 63–92
7. Finkel, A.: The minimal coverability graph for petri nets. In Rozenberg, G., ed.: Advances in Petri Nets 1993. Volume 674 of Lect. Notes Comp. Sci. Springer (1993) 210–243
8. Finkel, A., Goubault-Larrecq, J.: Forward analysis for WSTS, part I: Completions. In Albers, S., Marion, J.Y., eds.: 26th Symp. on Theoretical Aspects of Computer Science, STACS'09, Springer (2009) 433–444
9. Finkel, A., Goubault-Larrecq, J.: Forward analysis for WSTS, Part II: Complete WSTS. In Albers, S., Marchetti-Spaccamela, A., Matias, Y., Nikoletseas, S.E., Thomas, W., eds.: 36th Int. Colloquium on Automata, Languages and Programming, ICALP'09. Volume 5556 of Lect. Notes Comp. Sci., Springer (2009) 188–199
10. Hopcroft, J.E., Pansiot, J.J.: On the reachability problem for 5-dimensional vector addition systems. Theoritical Computer Science **8** (1979) 135–159
11. Jančar, P.: Bouziane's transformation of the petri net reachability problem and incorrectness of the related algorithm. Inf. Comput. **206** (November 2008) 1259–1263
12. Leroux, J.: Vector addition systems reachability problem (a simpler solution). In Voronkov, A., ed.: The Alan Turing Centenary Conference, Turing-100, Manchester UK June 22-25, 2012, Proceedings. Volume 10 of EPiC Series, EasyChair (2012) 214–228
13. Mayr, E.W., Meyer, A.R.: The complexity of the finite containment problem for petri nets. J. ACM **28**(3) (1981) 561–576
14. Pachl, J.K.: Reachability problems for communicating finite state machines. CoRR **cs.LO/0306121** (2003)
15. Rackoff, Ch.: The covering and boundedness problems for vector addition systems. Theoret. Comput. Sci. **6**(2) (1978) 223–231