Petri nets and algebraic calculi of processes

Gérard Boudol, G. Roucairol, Robert De Simone

KN 276

INRIA

CENTRE
SOPHIA ANTIPOLIS

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P. 105
78153 Le Chesnay Cedex
France
Tél. (3) 954 90 20

Rapports de Recherche

N° 292

PETRI NETS AND ALGEBRAIC CALCULI OF PROCESSES

Gérard BOUDOL
Gérard ROUCAIROL
Robert de SIMONE

Mai 1984

PETRI NETS AND ALGEBRAIC CALCULI OF PROCESSES

Gérard Boudol (*)

Gérard Roucairol (**)

Robert de Simone (***)

(*) INRIA, Sophia Antipolis, 06565 Valbonne, France
(**) LRI, Université Paris Sud, 91405 Orsay, France
(***) CMA-ENSMP, Sophia Antipolis, 06565 Valbonne, France

ABSTRACT

We show that, as transition systems, Petri nets may be expressed by terms of a calculus of processes which is a variant of Milner's SCCS. We then prove that the class of labelled nets forms a subcalculus, thus an algebra, with *juxtaposition, adding condition* and *labelling* as primitive operations.

RESUME

On montre que, en tant que systèmes de transition, les réseaux de Petri peuvent être exprimés par des termes d'un calcul de processus qui est une variante de SCCS de Milner. On prouve ensuite que la classe des réseaux étiquetés forme un sous-calcul, donc une algèbre dont les opérations primitives sont la juxtaposition, les ajouts de place et les étiquetages.

PETRI NETS AND ALGEBRAIC CALCULI OF PROCESSES

Gérard Boudol (*)

Gérard Roucairol (**)

Robert de Simone (***)

(*) INRIA, Sophia Antipolis, 06565 Valbonne, France
(**) LRI, Université Paris Sud, 91405 Orsay, France
(***) CMA-ENSMP, Sophia Antipolis, 06565 Valbonne, France

1 - INTRODUCTION

In the last few years several mathematical models of concurrent and communicating systems have emerged. One of the best known is that of Petri nets, which gave rise to a considerable amount of theoretical developments (see for example [11,2,3]). In some sense a Petri net is a dynamic pictorial description of a non deterministic asynchronous concurrent system. This graphical aspect involves a slight defect : it is not clear how to recognize "subgames" in the picture, which is given as a whole, and reset the components. In other words, we may need some syntax to build nets. Previous efforts in this direction can be found in [4,5].

We aim here at proposing such a syntax, in the framework of a more recent abstract model, that of Milner's algebraic calculi of processes ([7,9,10]). These models consist first of all of a free algebra of terms, with as primitive operations parallel composition, synchronization or desynchronization mechanisms, and so on : whence the algebraic character. Second there are actions labelled transition relations between terms. These relations are structurally defined, that is the behaviour of a compound term is a function of the behaviour of its components. Finally an equivalence relation on terms is given which is a congruence and respects the transitions abilities. Thus we have equational laws, whence a calculus, and each term denotes a *process:* a transition system on equivalence classes. Thus the calculus provides this way a syntax for some universe of processes.

Milner already showed that the static (pictorial) aspect of nets can be modeled in a calculus, that of (free) flow algebra [6]. Concerning dynamics, one of his most fruitful ideas was to realize that the set of actions should be thought of as an algebraic structure also [9,10]. Specifically, actions will form an abelian monoïd : to perform simultaneously two actions one performs their product. Moreover, some actions may have an inverse, in order to handle a rudimentary form of communication.

We set a relationship between Petri's and Milner's mathematical models of processes. The calculus we will use, called MEIJE (cf [1]), lies on an "asynchronous" (not in the technical sense of asynchrony in [10]) parallel composition, together with some synchronization primitives. We present it briefly in the second section. In fact, this calculus is just an equivalent formulation of the "finitary" version of Milner's synchronous calculus (see [1,8]). In the third section we show that, as far as they denote transition systems, Petri nets can be exactly expressed as terms of this calculus. The crucial tool here rests in the monoid of actions, formalizing notions of simultaneity and communication. In the fourth section we present the labelled nets as an algebra, build with juxtaposition, adding conditions and labelling as primitive operations. We discuss the semantical aspects of our propositions in the conclusion.

2 - THE MEIJE CALCULUS

First we describe the monoïd of actions which is the basis of our calculus. This monoïd $\mathbb{M}$ is the product of two others :

(1) the free abelian monoïd generated by some given countable set A of *atomic actions*. Thus, loosely speaking, this monoïd is that of instantaneous events, which do not interact with each other.

(2) The free abelian group generated by some given countable set S of *signals*. Each of the element of S, say s, is a synchronization or communication action, endowed with an inverse $\overline{s}$; the simultaneous occurrence $s.\overline{s}$ of these actions establishes a communication, or a handshake, which is a private act, only showing the unit action 1.

The product of $a \in \mathbb{M}$ and $b \in \mathbb{M}$ will be denoted by $a.b$ and the unit by 1. If B is a subset of $A \cup S$ we shall denote by $\mathbb{M}\backslash B$ the substructure of $\mathbb{M}$ generated by $(A \cup S)-B$.

Now let X be a countable set of variables, which will serve as identifiers in order to define recursive processes. The syntax for the terms of the MEIJE calculus ([1]) is given by the following :

(i) $\mathbb{0}$ is a term *(inaction)*, and a variable is a term ;

(ii) guard or *action* : if $a \in \mathbb{M}$ and p is a term, then $a{:}p$ is a term ;

(iii) *morphism* : if φ is an endomorphism of $\mathbb{M}$ and p is a term, then $<\varphi>p$ is a term ;

(iv) *restriction :* if $\alpha \in S$ and p is a term then $p \backslash \alpha$ is a term ;

(v) *recursive definition :* if $x_1, \ldots, x_n$ are variables and $p_1, \ldots, p_n$ are terms, then for $1 \leqq i \leqq n$ $(x_i \ \underline{where} \ x_1 = p_1, \ldots, x_n = p_n)$ is a term.

These are, with slight lexical variations (but the same semantics, see below), among CCS's or SCCS's primitives.

This is not the case of the following :

(vi) *asynchronous parallel composition :* if p and q are terms, then $(p \parallel q)$ is a term ;

(vii) *triggering :* if $\alpha \in S$ and p is a term, then $\alpha \Rightarrow p$ is a term ;

(viii) *ticking :* if $\alpha \in S$ and p is a term, then $\alpha * p$ is a term.

Free and bound occurrences of variables are defined in the usual way, and we denote by

$$[q_1/x_1, \ldots, q_n/x_n]p$$

the term we get by substituting the q_i's to the x_i's at their free occurences in p (renaming bound variables of p if necessary).

The transitions relations $\xrightarrow{a}$ (where $a \in \mathbb{M}$) between terms are are the least ones satisfying the following rules :

(R1) $a{:}p \xrightarrow{a} p$

(R2) if $p \xrightarrow{a} p'$ then $<\varphi>p \xrightarrow{\varphi(a)} <\varphi>p'$

(R3) if $p \xrightarrow{a} p'$ and $a \in \mathbb{M} \backslash \alpha$ (that is neither α nor $\overline{\alpha}$ appears as an irreducible factor of a) then $p \backslash \alpha \xrightarrow{a} p' \backslash \alpha$

(R4) if, for $i \leqq j \leqq n$, $q_j = (x_j \ \underline{where} \ x_1 = p_1, \ldots, x_n = p_n)$ and $[q_1/x_1, \ldots, q_n/x_n]p_i \xrightarrow{a} p$ then $q_i \xrightarrow{a} p$.

There are three rules defining the semantics of the parallel composition, according to the idea that the components are independant :

(R5) if $p \xrightarrow{a} p'$ then $(p \parallel q) \xrightarrow{a} (p' \parallel q)$

(R6) if $p \xrightarrow{a} p'$ and $q \xrightarrow{b} q'$ then $(p \parallel q) \xrightarrow{a.b} (p' \parallel q')$

(R7) if $q \xrightarrow{b} q'$ then $(p \parallel q) \xrightarrow{b} (p \parallel q')$

Finally for the synchronization primitives of MEIJE :

(R8) if $p \xrightarrow{a} p'$ then $\alpha => p \xrightarrow{\alpha.a} p'$

(R9) if $p \xrightarrow{a} p'$ then $\alpha * p \xrightarrow{\alpha.a} \alpha * p'$.

Thus $t \xrightarrow{a} t'$ *iff* this transition has a proof following these rules. For example, the transition graph of the term $p = ((\alpha=>(a:b:\mathbb{O}) \parallel \alpha=>(c:\mathbb{O})) \parallel \overline{\alpha}:\mathbb{O})\backslash\alpha$ is

$$p \xrightarrow{a} ((b:\mathbb{O} \parallel \alpha => (c:\mathbb{O})) \parallel \mathbb{O})\backslash\alpha \xrightarrow{b} ((\mathbb{O} \parallel \alpha => (c:\mathbb{O})) \parallel \mathbb{O}\backslash\alpha$$
$$p \xrightarrow{c} ((\alpha => (a:b:\mathbb{O}) \parallel \mathbb{O}) \parallel \mathbb{O})\backslash\alpha$$

With each action a we associate a *clock* on a

$$h_a =_{def} (x \ \underline{where} \ x = a:x)$$

and it is easily seen that $h_a \xrightarrow{a} h_a$ (and moreover $h_a \xrightarrow{b} p$ implies $b = a$ and $p = h_a$).

Let $A_{(A,S)}$ be the set of closed terms, that is without free variables : these are the *agents* of the calculus, which denote processes, as we shall just see. Let $\sim$ be the coarsest equivalence relation between agents which satisfy the property of commutation (or compatibility) with transitions :

$$p \sim q \text{ and } p \xrightarrow{a} p' => \exists\, q' \ q \xrightarrow{a} q' \text{ and } p' \sim q'$$

It is a congruence relation over the algebra of agents, called the *strong congruence* (see [10,1]). The set $P_{(A,S)} = A_{(A,S)}/\sim$ of equivalence classes is the set of (MEIJE) *processes*. In this set we still have transition relations :

$$[\![p]\!] \xrightarrow{a} [\![p']\!] \text{ iff } \exists\, q : p \xrightarrow{a} q \text{ and } q \sim p'$$

(where $[\![p]\!]$ denotes the equivalence class of the agent p).

Equational laws where verified somewhere else ([1]), among which :

(L1) $(p \parallel q) \sim (q \parallel p)$ (*commutativity*)

(L2) $(p \parallel (q \parallel r)) \sim ((p \parallel q) \parallel r)$ (*associativity*)

which allows to write $(p_1 \parallel \ldots \parallel p_n)$

(L3) $(\mathbb{O} \parallel p) \sim p$ (*unit*)

And, for $\alpha, \beta \in S$:

(L4) $(p\backslash\alpha)\backslash\alpha \sim p\backslash\alpha$

(L5) $(p\backslash\alpha)\backslash\beta \sim (p\backslash\beta)\backslash\alpha$

Thus for $\{\alpha_1, \ldots, \alpha_n\} \subseteq S$ we may write $p\backslash\{\alpha_1, \ldots, \alpha_n\}$ for $(\ldots(p\backslash\alpha_1)\backslash \cdots \backslash\alpha_n)$.

Another law will be of use : if, as in (R4), we let for $1 \leq i \leq n$

$q_i = (x_i$ <u>*where*</u> $x_1 = p_1,\ldots,x_n = p_n)$ then

(L6) $q_i \sim [q_1/x_1,\ldots,q_n/x_n]p_i$ (fixpoint).

While agents denote processes, the expressions of the calculus, with free variables, denote functions on processes. If t is such an expression, with free variables $x_1,\ldots,x_n$ we define

$$\hat{t}(\llbracket p_1 \rrbracket, \ldots, \llbracket p_n \rrbracket) = \llbracket [p_1/x_1, \ldots, p_n/x_n]t \rrbracket$$

for agents $p_1, \ldots, p_n$.

For example, the *interleaving* operator, specified on $P_{(A,S)}$ by the rules :

(1) if $p \xrightarrow{a} p'$ *then* $(p \mid q) \xrightarrow{a} (p' \mid q)$

(2) if $q \xrightarrow{b} q'$ *then* $(p \mid q) \xrightarrow{b} (p \mid q')$

is defined by the MEIJE expression

$$(x \mid y) = (\alpha^* x \parallel \alpha^* y \parallel h_{\overline{\alpha}}) \setminus \alpha$$

Some other examples of such *derived operators* (for which $\sim$ is still a congruence) are given in [1]. We get a *subcalculus* by taking some of these derived operators as being primitive.

3 - EXPRESSING PETRI NETS IN MEIJE

3.1. - Let us first briefly recall some basic definitions (we assume here some familiarity with Petri nets, see [11,2,3]). We deal with Petri nets allowing multiple arcs between transitions and places. Thus a net r is a structure $(P,T,Pre,Post)$ where

- $P = \{p_1, \ldots, p_K\}$ is the finite non-empty set of *places*
- $T = \{t_1, \ldots, t_N\}$ is the finite non-empty set of *transitions* (with some ambiguity in the use of this word)
- $Pre : P \times T \to \mathbb{N}$ ($\mathbb{N}$ is the set of non-negative integers)

and $Post : T \times P \to \mathbb{N}$ are the numerical functions setting the preconditions and postconditions of the firing of transitions.

A *marking* μ on the net r is a map from P into $\mathbb{N}$: $\mu(p)$ is the number of *tokens* on the place p in this marking (and we note $\mu(p_i) = \mu_i$). A transition t is *enabled* by the marking μ, a property noted $(r,\mu)[t>$ *iff*

$$\forall p \in P \quad \mu(p) \geq Pre(p,t)$$

Under this condition, by *firing* t we get the marking μ' on r such that

$$\forall p \in P \quad \mu'(p) = \mu(p) - Pre(p,t) + Post(t,p)$$

We write that $(r,\mu)[t>(r,\mu')$, which is the transition relation between states of the net.

One may extend this relation in order to define the *simultaneous* firing of transitions. Given a non-empty subset U of T, we say that U is enabled by the marking μ *iff*

$$\forall p \in P \quad \mu(p) \geq \sum_{t \in U} Pre(p,t)$$

and we denote again this fact by $(r,\mu)[U>$. Then, by firing simultaneously (or : in parallel) the transitions of U, we get the marking μ' :

$$\forall p \in P \quad \mu'(p) = \mu(p) + \sum_{t \in U} [Post(t,p) - Pre(p,t)]$$

The extended transition relation is again denoted

$$(r,\mu)\,[U>(r,\mu')$$

(This is the definition of [12]).

3.2. - Now we want to find for each marked Petri net a MEIJE agent which, as far as the two denote processes, is "isomorphic" to this net. The idea is to consider places and transitions of the net as processes, and then to express them by agents and find an appropriate communication structure between them.

A place of a net is no more than a *bag*, into which we can put -and there after remove- tokens. Let us denote for a moment by a and b respectively the atomic actions of putting and removing a token. Then a bag is a process, the behaviour of which is the following :

one can, simultaneously

- put (performing the action a) as many tokens as one wishes
- remove (performing b) no more tokens than there already are in the bag.

Thus, if we denote by $\Omega^{(k)}$ a bag containing initially k tokens, its specification is :

$$\forall m \in \mathbb{N}\ \forall n \in \mathbb{N}\ n \leq k \quad \Omega^{(k)} \xrightarrow{a^m b^n} \Omega^{(k+m-n)}$$

Now let $\omega^{(k)}$ be the sequence of MEIJE agents defined by

$$\omega^{(0)} = (x \ \underline{where}\ x = (a{:}b{:}\mathbb{O} \parallel x))$$
$$\omega^{(k+1)} = (b{:}\mathbb{O} \parallel \omega^{(k)})$$

lemma 1

$$\forall\, k \in \mathbb{N} \quad \omega^{(k)} \sim (a{:}b{:}\mathbb{O} \parallel \omega^{(k)})$$

proof : we proceed by induction on k

- if $k=0$, then we apply the law $(L\,6)$ and immediately get the result

$$-(a{:}b{:}\mathbb{O} \parallel \omega^{(k+1)}) = (a{:}b{:}\mathbb{O} \parallel (b{:}\mathbb{O} \parallel \omega^{(k)})) \ \ (\textit{definition})$$

$$\sim (b{:}\mathbb{O} \parallel (a{:}b{:}\mathbb{O} \parallel \omega^{(k)})) \ \ (\textit{by } L\,1,\, L\,2)$$

$$\sim (b{:}\mathbb{O} \parallel \omega^{(k)}) \ \ (\textit{induction hypothesis})$$

$$= \omega^{(k+1)} \ \ (\textit{definition}) \ \blacksquare$$

lemma 2

$$\llbracket \omega^{(o)} \rrbracket \xrightarrow{c} \llbracket p \rrbracket \ \ \textit{iff}\ \exists\, k \in \mathbb{N}\,,\, k>0 \ \ c=a^k \text{ and } \llbracket p \rrbracket = \llbracket \omega^{(k)} \rrbracket$$

proof :

(1) by induction on k $(k>0)$ we prove that $\omega^{(o)} \xrightarrow{a^k} \omega^{(k)}$

(1.1) - We have (rule $R\,1$) $a{:}b{:}\mathbb{O} \xrightarrow{a} b{:}\mathbb{O}$

whence (rule $R\,5$) $(a{:}b{:}\mathbb{O} \parallel \omega^{(o)}) \xrightarrow{a} (b{:}\mathbb{O} \parallel \omega^{(o)}) = \omega^{(1)}$

Thus (rule $R\,4$) $\omega^{(o)} \xrightarrow{a} \omega^{(1)}$

(1.2) - By induction hypothesis $\omega^{(o)} \xrightarrow{a^k} \omega^{(k)}$. Since $a{:}b{:}\mathbb{O} \xrightarrow{a} b{:}\mathbb{O}$ we have (rule R6)

$$(a{:}b{:}\mathbb{O} \parallel \omega^{(o)}) \xrightarrow{a^{k+1}} (b{:}\mathbb{O} \parallel \omega^{(k)}) = \omega^{(k+1)}$$

Thus (rule $R\,4$) $\omega^{(o)} \xrightarrow{a^{k+1}} \omega^{(k+1)}$

(2) We prove the converse by induction on the definition of the transition relations.

If $\llbracket \omega^{(o)} \rrbracket \xrightarrow{c} \llbracket p \rrbracket$ then (by definition of transitions in $P_{(A,S)}$ and rule $R\,4$)

$$\exists\, q \sim p \quad (a{:}b{:}\mathbb{O} \parallel \omega^{(o)}) \xrightarrow{c} q$$

then

(2.1) - $c = a$ and $q = (b:\mathbb{O} \,\|\, \omega^{(o)}) = \omega^{(1)}$ (rules $R\,1$ and $R\,5$)

or

(2.2) - (rule $R\,7$) $\exists\, q' : \omega^{(o)} \xrightarrow{c} q'$ and $q = (a:b:\mathbb{O} \,\|\, q')$

then by induction hypothesis

$$\exists\, k \in \mathbb{N},\ k>0\ c = a^k \text{ and } q' \sim \omega^{(k)}$$

Thus $q \sim (a:b:\mathbb{O} \,\|\, \omega^{(k)})$ since $\sim$ is a congruence and by lemma 1 $q \sim \omega^{(k)}$

or

(2.3) - (rules $R\,6$ and $R\,1$)

$$\exists\, c'\ \exists\, q' : \omega^{(o)} \xrightarrow{c'} q',\ c = a.c' \text{ and } q = (b:\mathbb{O} \,\|\, q')$$

by induction hypothesis

$$\exists\, k > 0\ \ c' = a^k \text{ and } q' \sim \omega^{(k)}$$

thus $c = a^{k+1}$ and ($\sim$ congruence) $q \sim (b:\mathbb{O} \,\|\, \omega^{(k)}) = \omega^{(k+1)}$ ▪

We get the immediate corollary :

lemma 3

$\forall\, k \in \mathbb{N}\ \ [\![\omega^{(k)}]\!] \xrightarrow{c} [\![p]\!]$ *iff* $\exists\, n,m \in \mathbb{N},\ n \leq k$

and $c = a^m b^n,\ p \sim \omega^{(k+m-n)}$

(*proof* : by induction on k and case analysis, obvious).

This proves that we modeled well the concept of bag by the MEIJE agents $\omega^{(k)}$, that is :

$$[\![\omega^{(k)}]\!] = \Omega^{(k)}$$

In the synchronous calculus of Milner ([9,10]) we should have written

$$\omega^{(k)} = fix_k\ \{x_j = \sum_{n \leq j} ((b^n:1) \times \sum_{m \in \mathbb{N}} a^m : x_{j+m-n}) \,/\, j \in \mathbb{N}\}$$

where $1 = h_1 = (x$ <u>where</u> $x = 1:x)$.

3.3. - The second step consists in building agents for transitions. The semantics of the firing of a transition t is that, simultaneously :

- one remove $Pre(p,t)$ tokens from the input places from t

- one performs t

- one put $Post(t,p)$ tokens in the output places of t.

In order to provide this, we assume that places and transitions communicate by means of signals. Technically we suppose with the notations of (3.1), that:

- the set A of atomic actions includes $\{t_1, \ldots, t_N\}$

- the set S of signals includes $\{\alpha_i, \beta_i \ / \ 1 \leq i \leq K\}$

Then performing $\overline{\alpha}_i$ means putting a token in the place p_i, while p_i performs the action α_i, that is receives this token. When p_i performs $\overline{\beta}_i$, a token is sent from p_i, while performing β_i means receiving this token. Thus each place p_i, with some initial content (marking) is now a bag among

$$\Theta_i^{(o)} = (x \ \underline{where} \ x = (\alpha_i : \overline{\beta}_i : \mathbb{O} \parallel x))$$
$$\Theta_i^{(k+1)} = (\overline{\beta}_i : \mathbb{O} \parallel \Theta_i^{(k)})$$

$(1 \leq i \leq K)$

If we denote $n_{i,j} = Pre(p_i, t_j)$ and $m_{i,j} = Post(t_j, p_i)$ to each transition t_j correspond the instantaneous action

$$\tau_j = \beta_1^{n_{1,j}} \ \cdots \ \beta_K^{n_{K,j}} \ t_j \ \overline{\alpha}_1^{m_{1,j}} \ \cdots \ \overline{\alpha}_K^{m_{K,j}}$$

And the process associated with this transition, which can repeatedly perform this action, is therefore modeled by the clock h_{τ_j}.

Finally for each Petri net r (with the notations of (3.1)) and marking μ on r we build the MEIJE agent

$$\rho_{(r,\mu)} =_{def} ((h_{\tau_1} \mid \ldots \mid h_{\tau_N}) \parallel (\Theta_1^{(\mu_1)} \parallel \cdots \parallel \Theta_K^{(\mu_K)})) \setminus \{\alpha_i, \beta_i / 1 \leq i \leq K\}$$

Here we set an interleaving structure (which, strictly speaking, ought to be expanded in its MEIJE code) on the transitions, according to the fact that only one at a time may fire. Restriction on the signals means that tokens flow between places and transitions, so none can be lost.

lemma 4

$\llbracket \rho_{(r,\mu)} \rrbracket \xrightarrow{c} \llbracket p \rrbracket$ *iff* $\exists \, j \ (1 \leq j \leq N)$ *such that*

1 – $\forall \, i \ (1 \leq i \leq K) \ \mu_i \geq n_{i,j}$ and

2 – $c = t_j$ and

$3 - \; p \sim \rho_{(r,\mu')}$ *where* $\forall_i (1 \leq i \leq K) \;\; \mu'_i = \mu_i + m_{i,j} - n_{i,j}$

proof

(1) Let us assume that $\rho_{(r,\mu)} \xrightarrow{c} p$. Since for all i $(1 \leq i \leq K)$ and $k \in \mathbb{N}$ we have

$$\Theta_i^{(k)} \xrightarrow{c'} q \text{ implies } c' \not\subset \mathbb{M} \setminus \{\alpha_i, \beta_i / 1 \leq i \leq K\}$$

we deduce (using rule $R3$) that

(1.1) - either (rule $R5$) $(h_{\tau_1} | ... | h_{\tau_N}) \xrightarrow{c} q$ for some q such that

$$p = (q \parallel (\Theta_1^{(\mu_1)} \parallel \cdots \parallel \Theta_K^{(\mu_K)})) \setminus \{\alpha_i, \beta_i / 1 \leq i \leq K\}$$

In this case we have, by the definition of the interleaving operator (and the behaviour of the clocks) :

$$\exists j \; (1 \leq j \leq N) \;\; c = \tau_j \text{ and } q \sim (h_{\tau_1} | ... | h_{\tau_N})$$

Moreover $c \in \mathbb{M} \setminus \{\alpha_i, \beta_i / 1 \leq i \leq K\}$ (rule $R3$) thus for all $i (1 \leq i \leq K)\; n_{i,j} = 0 = m_{i,j}$ and we get the desired conclusion in this case.

(1.2) - either (rule $R6$)

$(h_{\tau_1} | ... | h_{\tau_N}) \xrightarrow{c'} q$ and

$(\Theta_1^{(\mu_1)} \parallel \cdots \parallel \Theta_K^{(\mu_K)}) \xrightarrow{c''} q'$

and $c = c'c''$, and $p = (q \parallel q') \setminus \{\alpha_i, \beta_i / 1 \leq i \leq K\}$. Thus $\exists j \; (1 \leq j \leq N) \;\; c' = \tau_j$ and $q \sim (h_{\tau_1} | ... | h_{\tau_N})$

Since $c \in \mathbb{M} \setminus \{\alpha_i, \beta_i / 1 \leq i \leq K\}$ we must have

$$c'' = \overline{\beta}_1^{n_{1,j}} \cdots \overline{\beta}_K^{n_{K,j}} \; \alpha_1^{m_{1,j}} \cdots \alpha_K^{m_{K,j}} \;\; \text{whence} \;\; c = t_j$$

Thus for all i $(1 \leq i \leq K)$ such that $n_{i,j} + m_{i,j} \neq 0$ we have :

$$\exists i : \Theta_i^{(\mu_i)} \xrightarrow{\overline{\beta}_i^{n_{i,j}} \alpha_i^{m_{i,j}}} q_i .$$

and $q' = (q_1 \parallel \cdots \parallel q_K)$ if we let $q_i = \Theta_i^{(\mu_i)}$ if $n_{i,j} + m_{i,j} = 0$.

By lemma 3 $n_{i,j} \leq \mu_i$ and if we let $\mu'_i = \mu_i + m_{i,j} - n_{i,j}$ we have $q_i \sim \Theta_i^{(\mu'_i)}$.

Finally, $p \sim \rho_{(r,\mu')}$ as we meant to prove.

(2) Conversely, assuming that for some j $(1 \leq i \leq N)$ we have $\forall i \; (1 \leq i \leq K) \;\; \mu_i \geq n_{i,j}$, it is easy, by means of lemma 3 and rules defining the transition relations, to show that

$$\llbracket \rho_{(r,\mu)} \rrbracket \xrightarrow{t_j} \llbracket \rho_{(r,\mu')} \rrbracket$$

where $\mu'_i = \mu_i - n_{i,j} + m_{i,j}$ ∎

Our goal is now plainly achieved :

proposition 1

For any net r, marking μ and transition t :

$$(r,\mu)[t>(r,\mu') \text{ iff } \llbracket \rho_{(r,\mu)} \rrbracket \xrightarrow{t} \llbracket \rho_{(r,\mu')} \rrbracket \quad ∎$$

We model the parallel behaviour of a Petri nets in a similar way, by defining

$$\overline{\rho}_{(r,\mu)} =_{def} (((h_{r_1} \| \cdots \| h_{r_N}) \| (\circledS_1^{(\mu_1)} \| \cdots \| \circledS_K^{(\mu_K)})) \setminus \{\alpha_i, \beta_i / 1 \leq i \leq K\}$$

and we get the analogous property :

proposition 2

For all non empty set $U = \{t_{j_1}, \ldots, t_{j_n}\}$ of transitions of r

$$(r,\mu)[U>(r,\mu') \text{ iff } \llbracket \overline{\rho}_{(r,\mu)} \rrbracket \xrightarrow{t_{j_1} \cdots t_{j_n}} \llbracket \overline{\rho}_{(r,\mu')} \rrbracket \quad ∎$$

This last expression seems more convenient if we think about Petri nets as modelling parallel systems. And we get a flow expression in the sense of [6] which depicts the graphical aspect of the net-taking marked places' and transitions' agents as primitive nodes.

We should also emphasize that these results are actually more accurate : they are true for the semantics of agents given by an equational congruence. We find among the equational laws we use the "laws of flow" of Milner [6].

4 - THE ALGEBRA OF LABELLED NETS

So far we have only used a restricted syntax (a subcalculus) from MEIJE to express the behaviour of Petri nets. But even in this restricted syntax (inaction, action, parallel composition, restriction and recursive definition) we could write agents which do not represent Petri nets (see [1]). Thus in this section we face the question of whether there is (inside MEIJE, that is by means of derived operators) a syntax which describes exactly the processes determined by Petri nets.

We do this for a slight extension of these nets, to be precise for *labelled nets*. The idea behind this is that the set of transitions merely is a set of (spatial) occurences of actions. Thus a labelled and marked net is

- as in (3.1) a net $\tau = (P, T, Pre, Post)$ with $T \subseteq A$
- a marking μ *on* τ
- a labelling, that is a mapping $\lambda : T \rightarrow A$.

For a transition $t \in T$, $\lambda(t)$ is the actual action performed when t fires. Thus the transition relation is modified in an obvious way :

$$(\tau, \mu, \lambda)\ [V> (\tau, \mu', \lambda) \ \ iff\ \ \exists\, U \subseteq T \ \ (\tau, \mu)\ [U> (\tau, \mu')$$

$$\text{and } V \text{ is the } \textit{multiset } \lambda(U).$$

A labelling uniquely determines an endomorphism on $\mathbb{M}$ such that $\lambda(a) = a$ for $a \in (S \cup A) - T$ (and we do not distinguish the two in notation). A labelled marked net is modeled, in the same sense than before, by a MEIJE agent

$$< \lambda > \overline{p}_{(\tau, \mu)}$$

Let L_A be the set of such nets' expressions in MEIJE. We may thus define the set of processes determined by labelled marked nets by

$$R_A = L_A \,/\, \sim$$

This is the universe of processes which we want to present as an algebraic calculus.

In the sequel we will note α_1 and β_1 by α and β, and a marked "typical place" will be an agent from the sequence $\{\Theta^{(k)} \,/\, k \in \mathbb{N}\}$ where $\Theta^{(k)} = \Theta_1^{(k)}$.

There exists an operation on marked nets which consists in "adding a condition", that is which adds a new marked place and extends the *Pre* and *Post* functions in a specified manner. Thus such an operation is fully determined by

- an integer $k \in \mathbb{N}$ which is the marking of the new place p
- a map $f : A \rightarrow \mathbb{N} \times \mathbb{N}$ which for each transition t gives

  $(Pre(p, t), Post(t, p)) = f(t)$.

We do not formulate the precise definition, but merely indicate the MEIJE expression of the operation associated with such a "marked condition" (k, f) :

$$\pi_{(k,f)}(x) =_{def} (< \psi > x \parallel \Theta^{(k)}) \setminus \alpha, \beta$$

where ψ is the morphism determined by

$$\forall\, a \in A \ \ \psi(a) = \beta^n\, a\, \overline{\alpha}^m \ \text{ if } \ f(a) = (n, m)$$

$$(\textit{and } \psi(s) = s \text{ for } s \in S)$$

Here we simply call such a derived operator a *constraint*.

proposition 3

Let N_A be the least set of MEIJE agents containing the set $H_A = \{h_a \,/\, a \in A\}$ of clocks (on A) and closed under parallel composition, labellings and constraints. Then $R_A = N_A \,/\!\sim$.

proof (outline)

(1) In one direction we have to prove that each labelled marked net expression $<\lambda>\overline{\rho}_{(r,\mu)}$ may be translated, up to strong equivalence, into a term of the algebra N_A. It suffices to show that $\overline{\rho}_{(r,\mu)}$ is equivalent to a term build on clocks using parallel composition and constraints. And here we simply follow the idea that, in order to build a marked net one

- sets together some transitions,
- then adds one by one places and appropriate arcs.

We use here a slight extension (concerning morphisms) of Milner's laws of flow (which are valid in MEIJE, see [1]) together with :

- for all morphism φ $<\varphi>h_a \sim h_{\varphi(a)}$
- if φ is the morphism given by $\varphi(\alpha_i) = \alpha$ and

$$\varphi(\beta_i) = \beta \;\; then \;\; <\varphi>\Theta_i^{(k)} \sim \Theta^{(k)}$$

We sketch the proof that, with the notations of (3.3)

$$\overline{\rho}_{(r,\mu)} = [(h_{\tau_1} \| \cdots \| h_{\tau_N}) \| (\Theta_1^{(\mu_1)} \| \cdots \| \Theta_K^{(\mu_K)})] \setminus \{\alpha_i, \beta_i \,/\, a \leq i \leq K\}$$

we have

$$\overline{\rho}_{(r,\mu)} \sim \pi^{(K)}(\ldots(\pi^{(1)}(h_{t_1} \| \cdots \| h_{t_N}))\ldots)$$

$$where \;\; \pi^{(i)} = \pi_{(\mu_i, f_i)} \,,\; f_i(t_j) = (n_{i,j}, m_{i,j})$$

By induction on K :

(1.1) - If $K = 1$, we let

$$\psi(t_j) = \beta^{n_{1,j}} \, t_j \, \overline{\alpha}^{m_{1,j}}$$

for $1 \leq j \leq N$. Then,

$$\begin{aligned}(h_{\tau_1} \| \cdots \| h_{\tau_N}) &\sim (<\psi>h_{t_1} \| \cdots \| <\psi>h_{t_N}) \\ &\sim <\psi>(h_{t_1} \| \cdots \| h_{t_N})\end{aligned}$$

(1.2) - At the induction step, where $K = l+1$, we let

- for $1 \leq j \leq N$

$$\psi(t_j) = \beta_K^{n_{Kj}}\, t_j\, \overline{\alpha}_K^{\,m_{Kj}} \text{ and}$$

$$\tau'_j = \beta_1^{n_{1j}} \cdots \beta_l^{n_{lj}}\, t_j\, \overline{\alpha}_1^{\,m_{1j}} \cdots \overline{\alpha}_l^{\,m_{lj}}$$

(thus $\psi(\tau'_j) = \tau_j$)

$$-\ \overline{\rho}_{(\tau',\mu')} = [(h_{\tau'_1} \| \cdots \| h_{\tau'_N}) \| (\Theta_1^{(\mu_1)} \| \cdots \| \Theta_l^{(\mu_l)})] \setminus \{\alpha_i, \beta_i \ / \ 1 \leq i \leq l\}$$

We have

$$\overline{\rho}_{(\tau,\mu)} \sim [(<\psi> h_{\tau'_1} \| \cdots \| <\psi> h_{\tau'_N}) \| (\Theta_1^{(\mu_1)} \| \cdots \| \Theta_K^{(\mu_K)})] \setminus \{\alpha_i, \beta_i \ / \ 1 \leq i \leq K\}$$

$$\sim [(<\psi> (h_{\tau'_1} \| \cdots \| h_{\tau'_N})] \| (\Theta_1^{(\mu_1)} \| \cdots \| \Theta_K^{(\mu_K)})] \setminus \{\alpha_i, \beta_i \ / \ 1 \leq i \leq K\}$$

$$\sim (<\psi> \overline{\rho}_{(\tau',\mu')} \| \Theta_K^{(\mu_K)}) \setminus \alpha_K, \beta_K$$

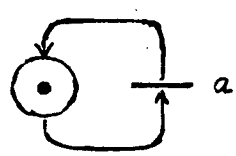
$$\sim \pi^{(K)}(\overline{\rho}_{(\tau',\mu')})$$

(the last two steps might have to be detailed).

And we apply the induction hypothesis

(2) Now in the converse direction we have to prove that each clock is equivalent to a (labelled marked) net expression and then that the operations preserve this property. Here again, apart for the generators, we essentially use laws of flow.

(2.1) - Intuitively a clock h_a is nothing but the net



And in fact one can prove (by induction on the definition of transitions) that adding such an implicit condition has no effect :

if $f : A \to \mathbb{N} \times \mathbb{N}$ is such that, for some $a \in A$

$$f(t) = \begin{cases} (1,1) & \text{if } t = a \\ (0,0) & otherwise \end{cases}$$

then for all net expression $p \quad \pi_{(1,f)}(p) \sim p$

(2.2) - The parallel composition of net expressions represents the operation of *juxtaposition* (and here appears the labelling). Thus the proof that it preserves the property of being equivalent to a term of L_A follows the idea that to juxtapose two labelled nets, one

- renames (by injective labellings) their sets of transitions, in order to disjoin them ;
- puts together these two new disjoint unlabelled nets, obtaining a net ;
- resets the labelling (composing the original one with the inverse renamings).

We omit the technical details.

(2.3) - Applying a constraint to a labelled net we have

$$\pi_{(k,f)}(<\lambda>\overline{\rho}_{(r,\mu)}) \sim <\lambda> (\pi_{(k,g)}(\overline{\rho}_{(r,\mu)}))$$

$$where \quad g = f \circ \lambda' ,\ \lambda'(t) = \begin{cases} \lambda(t) & \text{if } t \in T \\ t & otherwise \end{cases}$$

And as in the point (1) :

$$\exists (r'',\mu'') : \pi_{(k,g)}(\overline{\rho}_{(r,\mu)}) \sim \overline{\rho}_{(r'',\mu'')}$$

(2.4) - The case of labellings is trivial since

$$<\psi>(<\varphi>p) \sim <\psi \circ \varphi>p \quad \blacksquare$$

We may rediscover the interleaved transitions behaviour of labelled Petri nets since

if $f : A \to \mathbb{N} \times \mathbb{N}$ is such that

$$\forall a \in A \quad f(a) = (1,1) \quad then \quad \pi_{(1,f)}(<\lambda>\overline{\rho}_{(r,\mu)}) \sim <\lambda> \rho_{(r,\mu)}$$

5 - CONCLUDING REMARKS

In order to build a true calculus of nets, it remains to find expressions of other interesting operations preserving (or extending) the proposed algebra and, perhaps more important, to discover algebraic properties of these operators. Each equational law may be seen as the validation of transformations or simplifications.

But we must first discuss the semantics. The informal postulate underlying the strong congruence is that nothing can be said about a "state" (of a process) unless it results from the observation of the performed actions. Thus one may disagree with the fact that our semantics is right for Petri nets. For instance the k-boundedness property is not preserved. Nevertheless, this question requires more careful examination : what is the exact meaning of our strong congruence, that is how can we describe our "states" of a net (obviously all that concern transitions is preserved) ? Moreover we could have proposed a stronger equivalence : we may find a set E of equational laws (among them the laws of flow, together with laws concerning the dynamics of generators and primitive operators) so that we may present, with the same proofs, the algebra of labelled nets as $L_A \,/\!=_E$ (where $=_E$ is the least congruence containing the instances of the equalities of E). Here again, to what extent do we get a right semantics for Petri nets ? Certainly such an equational congruence preserves some structural aspects. On the other hand, observational equivalences or bisimulations (see [7,10,1]), which allow one to forget about some transitions, may be better suited for some purposes. Thus the equivalence appears as a parameter of an algebraic calculus of processes, which may be chosen according to the intended semantics.

We leave all these semantical questions for future research, which may bring forth a new point of view on Petri nets.

REFERENCES

[1] D. Austry, G. Boudol : "Algèbre de processus et synchronisation", Theoret. Comput. Sci. 30 (1984)

[2] W. Brauer (editor) : "Advanced course on general net theory of processes and systems", Hamburg 1979. Lecture Notes in Comput. Sci. 84 (Springer, Berlin, 1980)

[3] G.W. Brams : "Réseaux de Petri : théorie et pratique", (Masson, Paris, 1983)

[4] R.H. Campbell, P.E. Lauer : "Formal semantics for a class of high-level primitives for coordinating concurrent processes". Acta Informatica 5 (1975), 247-332

[5] V.E. Kotov : "An algebrea for parallelism based on Petri nets", MFCS 79, Lecture Notes in Comput. Sci. 64 (Springer, Berlin, 1979), 39-55

[6] R. Milner : "Flowgraphs and flow algebras", JACM 26 (1979), 794-818

[7] R. Milner : "A calculus of communicating systems", Lecture Notes in Comput. Sci., 92 (Springer, Berlin, 1980)

[8] R. Milner : "On relating synchrony and asynchrony", Tech. Rept. CSR-75-80, Comput. Sci. Dept., Edinburgh Univ. 1980

[9] R. Milner : "Four combinators for concurrency", 2nd ACM Symp. on Principles of Distributed Computing, Ottawa (1982)

[10] R. Milner : "Calculi for synchrony and asynchrony", Theoret. Comput. Sci. 25 (1983), 267-310

[11] J.L. Peterson : "Petri nets", Comput. Surveys 9 (1977), 223-252

[12] G. Rozenberg, R. Verraedt : "Subset languages of Petri nets", Part I, Theoret. Comput. Sci. 26 (1983), 301-326

Imprimé en France

par

l'Institut National de Recherche en Informatique et en Automatique