Theoretical Computer Science 14 (1981) 317–336
North-Holland Publishing Company

COLOURED PETRI NETS AND THE INVARIANT-METHOD

Kurt JENSEN
Computer Science Department, Aarhus University, DK-8000 Aarhus C, Denmark

Communicated by R. Milner
Received September 1979
Revised March 1980

Abstract. In many systems a number of different processes have a similar structure and behaviour. To shorten system description and system analysis it is desirable to be able to treat such similar processes in a uniform and succinct way. In this paper it is shown how Petri nets can be generalized to allow processes to be described by a common subnet, without losing the ability to distinguish between them. Our generalization, called coloured Petri nets, is heavily influenced by predicate/transition-nets introduced by H.J. Genrich and K. Lautenbach. Moreover our paper shows how the invariant-method, introduced for Petri nets by K. Lautenbach, can be generalized to coloured Petri nets.

1. Introduction

Petri nets [4, 5, 6] have proved to be a valuable tool in the description and analysis of systems with concurrent actions. The purpose of this paper is to introduce a generalization of Petri nets and to show how it can be used to describe and analyse complex systems. In coloured Petri nets each token has attached a colour, indicating the identity of the token. Moreover each place and each transition has attached a set of colours. A transition can fire with respect to each of its colours. By a firing of a transition, tokens are removed and added at the input and output places in the normal way, except that a functional dependency is specified between the colour of the transition firing and the colours of the involved tokens. The colour attached to a token may be changed by a transition firing and it often represents a complex data-value.

Our definition of coloured Petri nets is heavily influenced by the definition of 'predicate/transition-nets' in [1, 8] and thus by the definition of 'CP-nets' in [7]. The main idea is essentially the same, but our formalization seems to be simpler and more suitable for mathematical analysis of the described systems.

In [3] place/transition-nets are analysed by means of system invariants in the form of weighted sets of places. When the weights are taken into account, these places together hold an invariant number of tokens. The method builds upon linear algebra, especially matrix-multiplication.

In [1, 8] it is proposed to generalize the invariant-method to predicate/transition-nets. The main extension is that matrices of integers are replaced by matrices of *formal sums over colours*. The invariants of [1, 8] may contain free variables (over the se of colours). To interpret the invariants it seems necessary to bind the free variables via a substitution, where at least partial knowledge about the firing sequence leading to the marking in question must be used.

Our paper proposes to replace matrices of integers by matrices of *linear functions between sets of colours*. Then invariants can be established directly without the need of substitutions.

It should be stressed that we do not claim that our 'coloured Petri nets' are more convenient for *description* of systems than the 'predicate/transition-nets' of [1]. In this respect the two approaches are very similar, and the differences are mainly a matter of personal taste or convenience for the respective applications.

What we do claim is that we have developed an alternative method for the *analysis* of these kinds of nets. The method is directly inspired by [1, 8], but it does not involve substitutions for free variables in the invariants. In our opinion this makes the method more transparent and we give an example where a proof in [1] is simplified considerably.

In Section 2 place/transition-nets and the invariant-method are defined. As a simple example, we consider the well known system consisting of readers and writers.

In Section 3 coloured Petri nets are motivated and informally introduced by means of the well known system, consisting of five dining philosophers.

In Section 4 coloured Petri nets and the invariant-method for them are formally defined. The philosopher-system is analysed.

In Section 5 a more complex system, consisting of database managers and message buffers, is described and analysed. This example is taken from [1], where it is shown how to complete a marking from partial knowledge of it. The proof in [1] uses one page of rather complicated equations and moreover part of the predicate/transition-net is unfolded to a complicated place/transition-net. In our formalism, the similar proof can be done in a few lines and without unfolding the coloured Petri net. Moreover we show how complicated invariants can be constructed from simpler ones.

2. Place/transition-nets

In this section we introduce a kind of Petri nets called place/transition-nets, and we show how these can be analysed by constructing system-invariants as proposed in [3]. Place/transition-nets is one of the most used and well known kinds of Petri nets. However, to ease our later generalization to 'coloured Petri nets' we shall present the definition of place/transition-nets in a terminology which differs slightly from the usual one.

Let $\mathbb{Z}$, $\mathbb{N}$ and $[A \to B]$ denote integers, nonnegative integers, and total functions from A to B respectively.

A *place/transition-net* is a 4-tuple $\mathrm{PTN} = (P, T, W, m_0)$ (fixed for the rest of this section, except for examples), where

(1) P is a set of *places*,
(2) T is a set of *transitions*,
(3) $P \cap T = \emptyset$, $P \cup T \neq \emptyset$,
(4) $W \in [P \times T \to \mathbb{Z}]$ is the *incidence-function*,
(5) $m_0 \in [P \to \mathbb{N}]$ is the *initial marking*.

A *marking* of PTN is a function in $[P \to \mathbb{N}]$. A place p is an *input place* (*output place*) for a transition t iff $W(p, t) < 0$ ($W(p, t) > 0$).

A place/transition-net can be represented as a directed graph. As an example the incidence-function and initial marking in Fig. 1 define the Petri net represented graphically in Fig. 2, where by convention $|W(p, t)| = 1$ for all unlabelled arcs. For the moment ignore the three invariants.

P \ T	t_1	t_2	t_3	t_4	t_5	t_6	m_0	Invariants		
								i1	i2	i3
LF	-1	-1			1	1	n	1		-1
WR	1		-1					1		-1
WW		1		-1				1		-1
R			1		-1			1	1	
W				1		-1		1	n	(n-1)
S			-1	-n	1	n	n		1	1

Fig. 1. Incidence-function and initial marking for place/transition-net.

The intuition behind place/transition-nets is that transitions may fire thereby removing tokens from their input places and adding tokens to their output places. The numbers of tokens removed or added are specified by W. Transitions may fire concurrently (simultaneously) iff they involve disjoint sets of tokens.

For the rest of this section we assume that the sets of places and transitions are finite and of the form $P = \{p_1, p_2, \ldots, p_n\}$ and $T = \{t_1, t_2, \ldots, t_m\}$, where $n, m \in \mathbb{N}$. It is then possible to consider the incidence-function as an *incidence-matrix* containing n rows and m columns. Analogously each function from P or T can be considered as a vector with n and m elements respectively.

To formalize the firing rule we need a few definitions: A *weighted set of transitions* is a function $x \in [T \to \mathbb{Z}]$. It is *positive* iff $x(t) \geq 0$ for all $t \in T$ and $x(t) > 0$ for at least one $t \in T$. '+' and '*' denote matrix-addition and matrix-multiplication respectively.

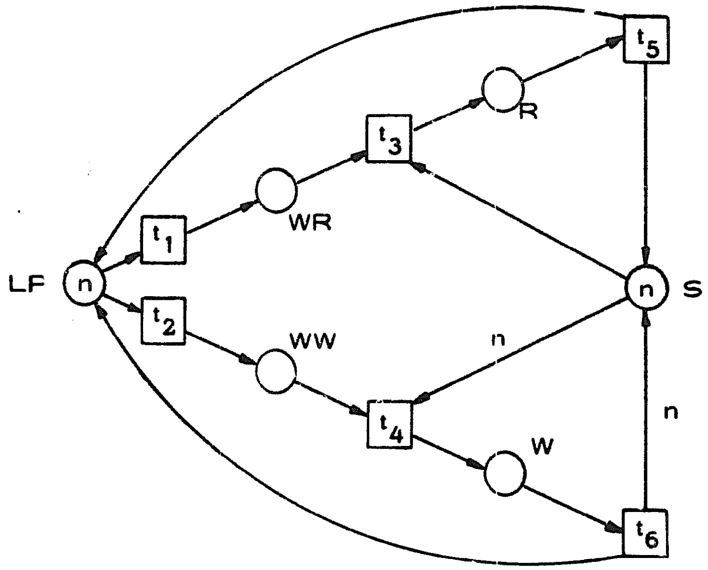

Fig. 2. Graphical representation of the place/transition-net in Fig. 1.

Two markings m and m' are in the relation $m \geqslant m'$ iff $\forall p \in P\,[m(p) \geqslant m'(p)]$. W^- is a matrix, constructed from W by

$$\forall (p, t) \in P \times T \quad \left[W^-(p, t) = \begin{cases} -W(p, t), & \text{if } W(p, t) < 0 \\ 0, & \text{if } W(p, t) \geqslant 0 \end{cases} \right].$$

Each vector can be considered as a matrix with a single row or as a matrix with a single column. Markings and weighted sets of transitions will always be considered as matrices with a single column, while weighted sets of places (to be defined shortly) will always be considered as matrices with a single row (although they may be shown in tables as columns).

A positive weighted set of transitions x has *concession* in a marking m iff $m \geqslant W^- * x$. When x has concession it may fire. If x fires, a new marking $m' = m + W * x$ is reached. m' is said to be *directly reachable* from m. *Reachability* is the reflexive, symmetric and transitive closure of direct reachability. From the firing rule it immediately follows that, if a marking m' is reachable from another marking m, there exists a weighted set of transitions x, such that $m' = m + W * x$.

A *weighted set of places* is a function in $[P \to \mathbb{Z}]$.

Theorem 1 ([3]). *Let v be a weighted set of places. If $v * W = 0$, then $v * m' = v * m$ for all markings m' and m, where m' is reachable from m. v is then said to be an invariant.*

Proof.

$$\begin{aligned} v * m' &= v * (m + W * x) && (m' \text{ reachable from } m) \\ &= v * m + v * (W * x) && (\text{distributivity}) \\ &= v * m + (v * W) * x && (\text{associativity}) \\ &= v * m && (\text{assumption}). \end{aligned}$$

Each linear combination of invariants is itself an invariant. Thus there is normally infinitely many invariants.

Example 1. To illustrate the use of the invariant-method we finish this section by analysing the place/transition-net in Fig. 2. It can be interpreted as a model of the well known system, consisting of n processes, $n>0$, which may read and write in a shared memory. Several processes may be reading concurrently, but when a process is writing, no other process can be reading or writing. No priority is assumed between the read and write operations. Each process can be in five different states: LP (local processing, where the shared memory is not used), WR (waiting to read), WW (waiting to write), R (reading), and W (writing). The place S (synchronization) enforces the mutual exclusion of writers. Intuitively tokens on LP, WR, WW, R and W represent processes, while tokens on S represent the state of the shared memory.

From the incidence-matrix in Fig. 1 we find three invariants shown as the columns (i1), (i2) and (i3).

From each of the invariants (i1), (i2) and (i3) we shall construct, by means of Theorem 1, an equation satisfied for all markings reachable from the initial marking. From now on we shall not distinguish between an invariant and its corresponding equation.

From

$$\text{(i1)} \qquad m(\mathrm{LP})+m(\mathrm{WR})+m(\mathrm{WW})+m(\mathrm{R})+m(\mathrm{W})=n$$

we conclude that the number of processes is invariant.

From

$$\text{(i2)} \qquad m(\mathrm{R})+n\ m(\mathrm{W})+m(\mathrm{S})=n$$

we conclude that when a process is 'writing', no other process can be 'reading' or 'writing'. The number of 'reading' processes is between zero and n. Moreover, if no processes are 'reading' or 'writing', $m(\mathrm{S})=n$. Thus t_3 has concession if at least one process is 'waiting to read' and t_4 has concession if at least one process is 'waiting to write'.

From

$$\text{(i3)} \qquad m(\mathrm{LP})+m(\mathrm{WR})+m(\mathrm{WW})=(n-1)m(\mathrm{W})+m(\mathrm{S})$$

(which is a linear combination of (i1) and (i2)) we conclude that when no process is 'writing', $m(\mathrm{WR})\leq m(\mathrm{S})$. Thus t_3 has concession if at least one process is 'waiting to read'.

Analysis 1. The place/transition-net in Fig. 2 cannot deadlock (reach a marking where no transition has concession).

Proof. If $m(\mathrm{LP})+m(\mathrm{R})+m(\mathrm{W})>0$ it follows from the net that t_1, t_2, t_5 or t_6 has concession.

If $m(\mathrm{LP})+m(\mathrm{R})+m(\mathrm{W})=0$ it follows from (i1) and (i2) that $m(\mathrm{WR})+m(\mathrm{WW})=n$, $m(\mathrm{S})=n$, and thus t_3 or t_4 have concession.

3. Informal introduction to coloured Petri nets

In the readers/writers system, treated in the previous section, it was not necessary to distinguish between different tokens at the same place. Often the situation is more complex. As an example consider the standard synchronization problem consisting of five philosophers who alternately think and eat. To eat, a philosopher needs two forks, but unfortunately there are only five forks on the circular table and each philosopher is only allowed to use the two forks nearest to him (see Fig. 3). Obviously two neighbours cannot eat at the same time.

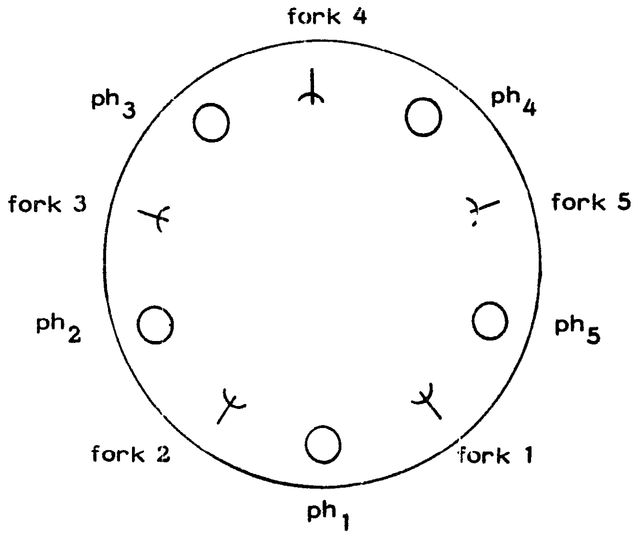

Fig. 3. Five dining philosophers.

The philosopher system can be described by a place/transition-net. Its graphical representation is shown in Fig. 4 ('th', 'e' and 'ff' are short for 'think', 'eat' and 'free forks', respectively). The incidence-matrix, initial marking and 10 invariants are shown in Fig. 5. For the moment ignore the dashed lines.

From invariants (i1)–(i5) we conclude that each philosopher is either thinking or eating, but not both. From invariants (i6)–(i10) we conclude that no philosopher can be eating at the same time as one of his neighbours.

Analysis 2. The place/transition-net in Fig. 4 cannot deadlock.

Proof. Assume that m is reachable from the initial marking. Then m satisfies (i1)–(i10).

If $m(\mathrm{e}_i)=1$ for some $i\in 1,\ldots,5$, transition t_i has concession.

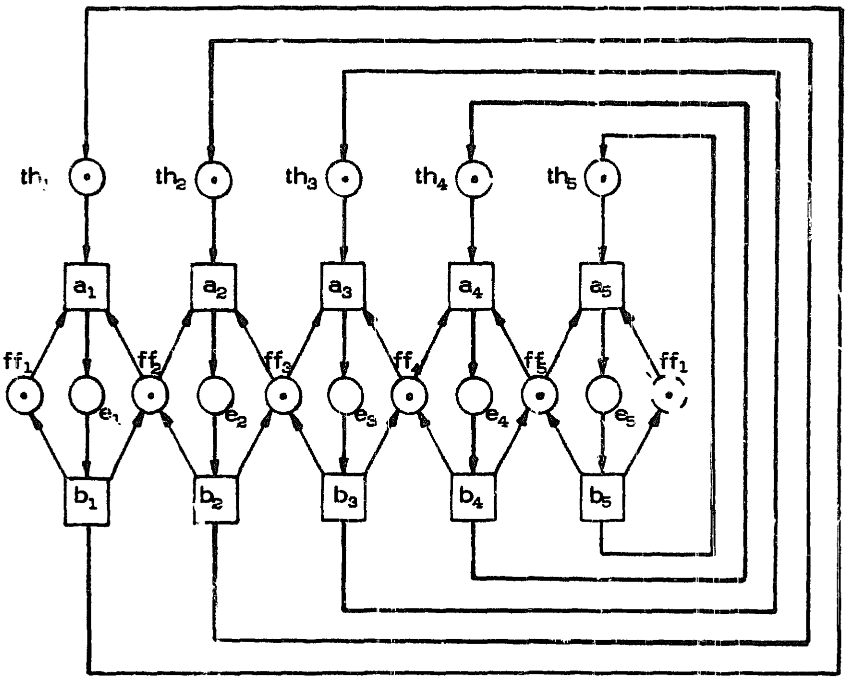

Fig. 4. Place/transition-net describing the philosopher system. For convenience the place ff_1 has been drawn twice. It has only one token.

	a_1	a_2	a_3	a_4	a_5	b_1	b_2	b_3	b_4	b_5	m_0	Invariants									
												i1	i2	i3	i4	i5	i6	i7	i8	i9	i10
th_1	-1					1					1	1									
th_2		-1					1				1		1								
th_3			-1					1			1			1							
th_4				-1					1		1				1						
th_5					-1					1	1					1					
e_1	1					-1						1					1	1			
e_2		1					-1						1					1	1		
e_3			1					-1						1					1	1	
e_4				1					-1						1					1	1
e_5					1					-1						1	1				1
ff_1	-1				-1	1				1	1						1				
ff_2	-1	-1				1	1				1							1			
ff_3		-1	-1				1	1			1								1		
ff_4			-1	-1				1	1		1									1	
ff_5				-1	-1				1	1	1										1

Fig. 5. Incidence-matrix, initial marking and 10 invariants for the place/transition-net in Fig. 4.

If $m(e_i) = 0$ for all $i \in 1, \ldots, 5$, it follows from (i1)–(i10) that

$$m(\mathrm{th}_i) = 1 \quad \text{for all } i \in 1, \ldots, 5,$$

$$m(\mathrm{f}_i) = 1 \quad \text{for all } i \in 1, \ldots, 5$$

but then a_i has concession for all $i \in 1, \ldots, 5$.

During the previous analysis of the philosopher system we constructed a large net, and found many invariants. In practical system work this is not just annoying, but it also puts rather narrow limits on the complexity of the systems which can be handled.

In the readers/writers system the size of the net was kept small by allowing tokens, representing different processes, to share the same subnet. It is tempting to use the same trick for the philosopher system. By a folding (see [5]) of the place/transition-net in Fig. 4 we obtain the net shown in Fig. 6, but unfortunately this is not a correct description of the philosopher system. In Fig. 6 each philosopher uses two forks, but he is allowed to select them among all free forks, not just the two nearest to him. Thus two neighbours can eat at the same time.

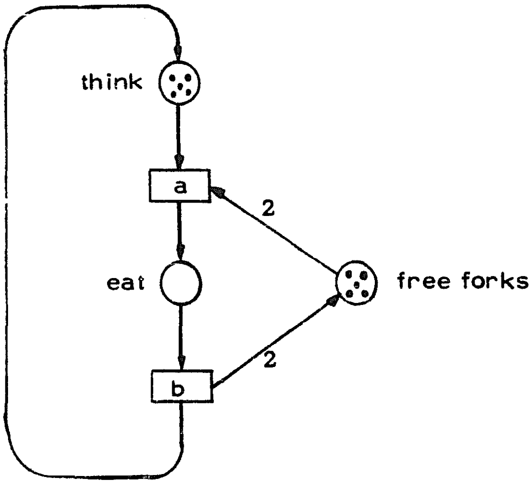

Fig. 6. The philosopher net after a folding where some places and some transitions are unified. Unfortunately, this simple net is not a correct description of the philosopher system.

Our aim is to obtain a net of the same size as Fig. 6, but a net which correctly describes the philosopher system in the sense that each philosopher can only use the two forks nearest to him. This can be done by being able to distinguish between the tokens representing the individual philosophers and also between the tokens representing the individual forks.

Our first step will be to replace the five places $\mathrm{th}_1, \mathrm{th}_2, \ldots, \mathrm{th}_5$ by a single place 'think', which can carry up to five tokens. To distinguish between these tokens, which represent different philosophers, we attach to 'think' a set of colours $PH = \{\mathrm{ph}_1, \mathrm{ph}_2, \ldots, \mathrm{ph}_5\}$, and we demand that all tokens on 'think' must be labelled by an element of PH. Markings of 'think' are functions in $[PH \to \mathbb{N}]$. They are represented as formal sums over PH. As an example $m(\text{think}) = \text{ph}_1 + \text{ph}_3 + \text{ph}_4$ represents that philosophers 1, 3 and 4 are thinking while philosophers 2 and 5 are not.

Analogously the places $e_1, e_2, \ldots, e_5$ are replaced by a single place 'eat' with PH as the set of possible colours, and the places $\text{ff}_1, \text{ff}_2, \ldots, \text{ff}_5$ are replaced by a single place 'free forks' with $F = \{f_1, f_2, \ldots, f_5\}$ as the set of possible colours.

At this stage of development each transition a_i from Fig. 4 has the form shown in Fig. 7, where the formal sums ph_i and $f_i + f_{i\oplus 1}$ at the arcs indicate that by a firing of a_i, the token removed from 'think' and the token added to 'eat' must have colour ph_i, while the two tokens removed from 'free forks' must have colours f_i and $f_{i\oplus 1}$ respectively.

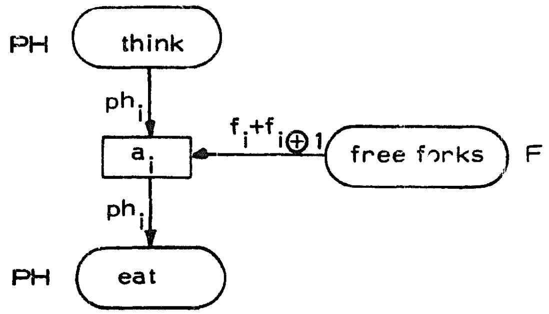

Fig. 7. Part of the philosopher net after a folding where some places are unified.

The next step will be to replace the five transitions $a_1, a_2, \ldots, a_5$ by a single transition 'take forks', which may fire in five different ways corresponding to the five philosophers. To distinguish between these different ways of firings we attach to the transition 'take forks' the set of colours, PH, representing the individual philosophers. We then get the subnet in Fig. 8, where ID, LEFT and RIGHT are functions from the set of colours PH attached to 'take forks' into the set of colours attached to its input/output places: 'think', 'eat' and 'free forks'. The functions indicate that a firing of 'take forks', with colour $v \in \text{PH}$, removes a token with colour $\text{ID}(v) \in \text{PH}$ from 'think', adds a token with colour $\text{ID}(v) \in \text{PH}$ to 'eat', and removes two tokens from 'free forks' with colours $\text{LEFT}(v) \in F$ and $\text{RIGHT}(v) \in F$ respectively. ID is the

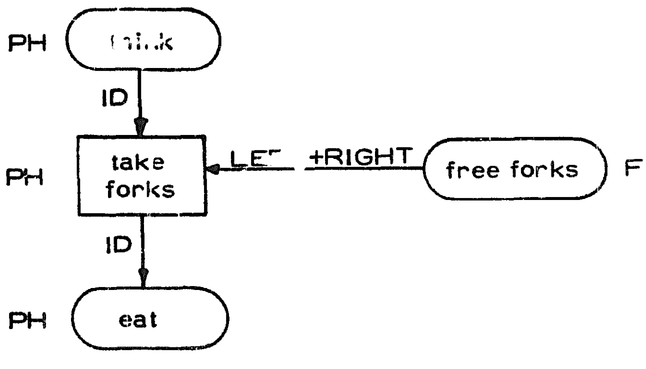

Fig. 8. Part of the philosopher net after a folding where some places and some transitions are unified.

identity function on PH. LEFT and RIGHT map each philosopher-colour into the colour of its left and right fork respectively.

Analogously we replace the transitions $b_1, b_2, \ldots, b_5$ by a single transition 'put down forks' with PH as the set of possible firing colours. We then get the coloured Petri net in Fig. 9, where by convention all unlabelled arcs represent the identity function of the set of colours attached to its transition.

Initially $m_0(\text{think}) = \sum \text{PH}$, $m_0(\text{eat}) = 0$ and $m_0(\text{free forks}) = \sum F$, where for an arbitrary set of colours A we define $\sum A = \sum_{a \in A} a$.

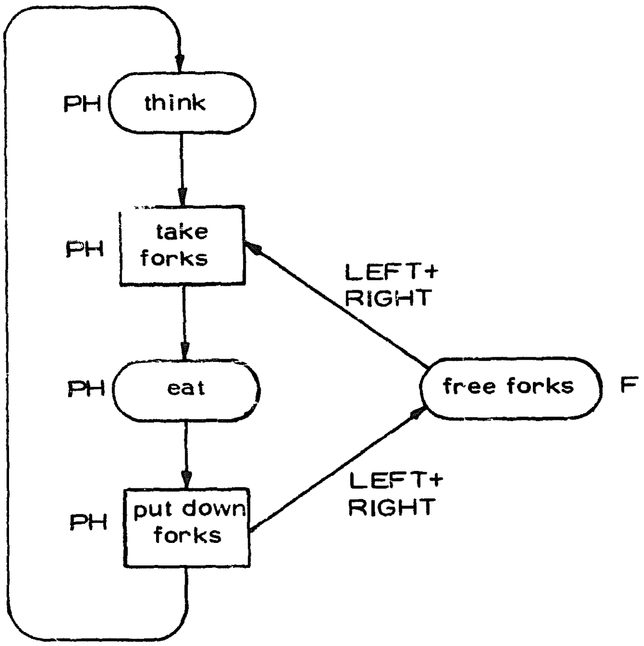

Fig. 9. Coloured Petri net describing the philosopher system.

3.1. Generalization of the invariant-method

In the previous part of this section we have seen how to obtain a coloured Petri net from a place/transition-net by a folding. Each place (transition) in the coloured Petri net replaces a group of places (transitions) in the place/transition-net. In the incidence-matrix (Fig. 5) these groups of places and transitions are indicated by dashed lines. The dashed lines divide the incidence matrix into six submatrices, each describing the tokens removed or added at a single place p (in the coloured Petri net) by firing a single transition t (in the coloured Petri net). Let $C(p)$ and $C(t)$ be the sets of colours attached to p and t respectively. The submatrix corresponding to p and t contains a row for each element in $C(p)$ and a column for each element in $C(t)$. Thus it uniquely defines a linear function in $[[C(t) \to \mathbb{Z}] \to [C(p) \to \mathbb{Z}]]$. Substituting each submatrix in Fig. 5 by the corresponding function we obtain the matrix shown in Fig. 10. We observe that it is the incidence-matrix for the coloured Petri net in Fig. 9. For the moment ignore the two invariants.

		take forks (t_1)	put down forks (t_2)	m_0	Invariants	
					i1′	i2′
		PH	PH		U_1=PH	U_2=F
think (p_1)	PH	-ID	ID	ΣPH	ID	
eat (p_2)	PH	ID	-ID		ID	LEFT +RIGHT
free forks (p_3)	F	-LEFT -RIGHT	LEFT +RIGHT	ΣF		ID

Fig. 10. Incidence-matrix for the coloured Petri net in Fig. 9.

Next consider the invariants (i6)–(i10) in Fig. 5. Each invariant (weighted set of places) is in our method considered as a matrix with a single row. Thus it would be more correct to draw (i6)–(i10) as shown in Fig. 11.

Each of the invariants (i6)–(i10) is a special instance of a common scheme, and we want to combine them into a single invariant containing colours. To distinguish between the original five invariants we need a set U containing five different colours and we choose $U=F$.

The dashed lines divide Fig. 11 into three submatrices. The submatrix corresponding to a place p uniquely defines a linear function in $[[C(p)\to \mathbb{Z}]\to[U\to\mathbb{Z}]]$. Substituting each submatrix by the corresponding function we obtain a matrix containing a single row, with the three elements shown as (i2′) in Fig. 10. In a similar way we can obtain (i1′) with $U=$ PH from the invariants (i1)–(i5).

	th_1	th_2	th_3	th_4	th_5	e_1	e_2	e_3	e_4	e_5	ff_1	ff_2	ff_3	ff_4	ff_5
i6						1				1	1				
i7						1	1					1			
i8							1	1					1		
i9								1	1					1	
i10									1	1					1

Fig. 11. Invariants (i6)–(i10) from Fig. 5 shown as rows (instead of columns).

Rename the places and transitions in the coloured Petri net as shown in the parentheses in Fig. 10. Let $(W_{ij})_{1\le i\le 3,1\le j\le 2}$ be the six submatrices from Fig. 5 and $(v_i)_{1\le i\le 3}$ the three submatrices from Fig. 11.

From the definitions in Section 2 it follows that (i6)–(i10) are invariants iff

$$\forall j\in 1,2\left[\sum_{i=1}^{3} v_i * W_{ij}=0\right].$$

Matrix multiplication is equivalent to composition of the corresponding linear functions. Thus we can replace $v_i * W_{ij}$ by $v_i' \circ W_{ij}'$, where a primed symbol denotes the linear function determined by the submatrix denoted by the corresponding unprimed symbol.

It then follows that (i6)–(i10) are invariants iff the corresponding $(i2') = (v_1', v_2', v_3')$ satisfies

$$(*) \qquad \forall j = 1, 2 \left[\sum_{i=1}^{3} v_i' \circ W_{ij}' = O_j \right],$$

where O_j is the zero function in $[[C(t_j) \to \mathbb{Z}] \to [U \to \mathbb{Z}]]$.

In the next section we shall define an invariant for the coloured Petri net to be a set of functions satisfying (*). Thus (i2′) is an invariant and we shall prove (as a generalization of Theorem 1) that this implies that the function in $[U \to \mathbb{Z}]$ defined by

$$\sum_{i=1}^{3} v_i'(m(p_i))$$

is the same for all markings m reachable from the initial marking.

4. Formal definition of coloured Petri nets

In this section we define coloured Petri nets and show how the invariant-method of Lautenbach can be generalized to coloured Petri nets.

Let A be a nonempty set and let $\mathbb{D}$ be $\mathbb{N}$ or $\mathbb{Z}$. By $[A \to \mathbb{D}]_f$ we denote the set of functions $g \in [A \to \mathbb{D}]$, where the support $\{a \in A \mid g(a) \neq 0\}$ is finite. For finite A we have $[A \to \mathbb{D}]_f = [A \to \mathbb{D}]$.

A *coloured Petri net* is a 5-tuple $\text{CPN} = (P, T, C, W, m_0)$ (fixed for the rest of this section, except for examples), where

(1) P is a set of *places*,

(2) T is a set of *transitions*,

(3) $P \cap T = \emptyset$, $P \cup T \neq \emptyset$,

(4) C is the *colour-function* defined from $P \cup T$ into nonempty sets,

(5) W is the *incidence-function* defined on $P \times T$ such that $W(p, t) \in [C(t) \to [C(p) \to \mathbb{Z}]_f]$ for all $(p, t) \in P \times T$,

(6) m_0, the *initial marking*, is a function defined on P, such that $m(p) \in [C(p) \to \mathbb{N}]_f$ for all $p \in P$.

A *marking of CPN* is a function m defined on P, such that $m(p) \in [C(p) \to \mathbb{N}]_f$ for all $p \in P$. Let p be a place and t a transition. Elements of $C(p)$ and $C(t)$ are called *colours*. p is an *input place* (*output place*) for t iff $W(p, t)(c')(c'') < 0$ ($W(p, t)(c')(c'') > 0$) for at least one pair of colours $c' \in C(t)$ and $c'' \in C(p)$. Note that in contrast to place/transition-nets a place may be both input place and output place for the same transition.

For the rest of this section we assume that the sets of places and transitions are finite and of the form $P = \{p_1, p_2, \ldots, p_n\}$ and $T = \{t_1, t_2, \ldots, t_m\}$ where $n, m \in \mathbb{N}$. As for place/transition-nets the incidence-function W can be considered as an *incidence-matrix* and the net can be represented as a directed graph.

Let A and B be nonempty sets. Each function $f \in [A \to [B \to \mathbb{Z}]_f]$ has a unique linear extension in $[[A \to \mathbb{Z}]_f \to [B \to \mathbb{Z}]_f]$. The extended function will also be denoted by f and it is defined to satisfy $f(g)(b) = \sum_{a \in A} g(a)(f(a)(b))$ for all $g \in [A \to \mathbb{Z}]_f$ and $b \in B$.

Using functions with only finite supports, excludes markings with an infinite number of tokens on a single place, and it guarantees convergence of the summation used to define linear extension.

To formalize the firing rule we need a few definitions: A *weighted set of transitions* is a function x defined on T, such that $x(t) \in [C(t) \to \mathbb{Z}]_f$ for all $t \in T$. It is *positive* iff $x(t)(c) \geq 0$ for all pairs $t \in T$ and $c \in C(t)$ and $x(t)(c) > 0$ for at least one pair $t \in T$ and $c \in C(t)$. We next generalize matrix-multiplication substituting each product by a function composition or a function application. Let $a = (a_{ij})_{1 \leq i \leq r, 1 \leq j \leq s}$ and $b = (b_{jk})_{1 \leq j \leq s, 1 \leq k \leq t}$ be two matrices and define $a \circledast b = (c_{ik})_{1 \leq i \leq r, 1 \leq k \leq t}$ by

$$c_{ik} = \sum_{j=1}^{s} a_{ij} b_{jk} \quad \text{for all } i \in 1, \ldots, r \text{ and all } k \in 1, \ldots, t,$$

where the juxtaposition $a_{ij}b_{jk}$ means function composition or function application. We shall only use this generalized operation on matrices where the elements fit together in the sense that the function compositions/applications and sums are possible.

Two markings m and m' are in the relation $m \geq m'$ iff $\forall p \in P$, $\forall c \in C(p)$ $[m(p)(c) \geq m'(p)(c)]$. W^- is a matrix constructed from W by

$$W^-(p,t)(c')(c'') = \begin{cases} -W(p,t)(c')(c''), & \text{if } W(p,t)(c')(c'') < 0, \\ 0, & \text{if } W(p,t)(c')(c'') \geq 0 \end{cases}$$

for all $(p, t) \in P \times T$, all $c' \in C(t)$ and all $c'' \in C(p)$.

Having made these definitions, *concession*, *firing* and (*direct*) *reachability* are defined exactly as for place/transition-nets (except that '$*$' is replaced by '$\circledast$'). As for place/transition-nets it follows from the firing rule that if a marking m' is reachable from another marking m, there exists a weighted set of transitions x, such that $m' = m + W \circledast x$.

Let U be a nonempty set. A *weighted set of places* (with respect to U) is a function v defined on P, such that $v(p) \in [C(p) \to [U \to \mathbb{Z}]_f]$ for all $p \in P$. For a motivation of this definition see the last part of Section 3.

Theorem 2. *Let v be a weighted set of places (with respect to U) and $O = (O_j)_{1 \leq j \leq m}$ a matrix of zero functions $O_j \in [[C(t_j) \to \mathbb{Z}]_f \to [U \to \mathbb{Z}]_f]$. If $v \circledast W = O$, then $v \circledast m' = v \circledast m$ for all markings m' and m, where m' is reachable from m. v is then said to be an invariant.*

Proof. Replace '$*$' by '$\circledast$' in the proof of Theorem 1. Distributivity follows from linearity of the functions in v. Associativity follows from associativity of functional composition.

As for place/transition-nets each linear combination of invariants is itself an invariant. Moreover if v is an invariant (with respect to U_1) and w is a function in $[U_1 \to [U_2 \to \mathbb{Z}]_f]$, then $w \circ v$ is an invariant (with respect to U_2).

To sum up, Fig. 12 gives the functionality for the functions defined in this section.

	Domain	Range
Incidence-matrix	P x T	$W(p,t) \in [C(t) \to [C(p) \to \mathbb{Z}]_f]$
Marking	P	$m(p) \in [C(p) \to \mathbb{N}]_f$
Weighted set of transitions (firing)	T	$x(t) \in [C(t) \to \mathbb{Z}]_f$
Weighted set of places (invariant)	P	$v(p) \in [C(p) \to [U \to \mathbb{Z}]_f]$

Fig. 12. Functionality for the functions defined in connection with coloured Petri nets.

Example 2. Next we analyse the coloured Petri net, Fig. 9, describing the philosopher system. Markings are represented as formal sums.

From the incidence-matrix in Fig. 10 we find the two invariants (i1′) and (i2′).

From

(i1′) $\qquad m(\text{think}) + m(\text{eat}) = \sum \text{PH}$

we conclude that each philosopher is either thinking or eating, but not both.

From

(i2′) $\qquad \text{LEFT}(m(\text{eat})) + \text{RIGHT}(m(\text{eat})) + m(\text{free forks}) = \sum F$

we conclude that no philosopher can be eating at the same time as one of his neighbours.

Analysis 3. The coloured Petri net of Fig. 9 cannot deadlock.

Proof. Assume that m is reachable from the initial marking. Then m satisfies (i1′) and (i2′).

If $m(\text{eat}) \neq 0$, 'put down forks' has concession.

If $m(\text{eat}) = 0$, it follows from (i1′) and (i2′) that

$$m(\text{think}) = \sum \text{PH}, \qquad m(\text{free forks}) = \sum F$$

and then 'take forks' has concession (for all colours in PH).

4.1. *Coloured Petri nets versus place/transition-nets*

A coloured Petri net can be transformed to a place/transition-net. This is done by replacing each place p with a set of places $C(p)$ (one for each kind of tokens p may hold) and replacing each transition t with a set of transitions $C(t)$ (one for each way in which t may fire). The relationship between the new places and transitions are determined by the corresponding elements in the matrix determined by the function $W(p, t)$.

In Section 3 we showed a transformation in the opposite direction. There we constructed a coloured Petri net from a place/transition-net. However, the constructed net is not unique. In fact given a place/transition-net, each partition of the places together with each partition of the transitions determine a coloured Petri net. As the two extremes we obtain either a coloured Petri net with the same number of places and transitions as the place/transition-net or a coloured Petri net with only one place and one transition. In the first case each place and each transition has attached a set of colours with only one element. In the second case the single place (transition) has a colour for each place (transition) in the place/transition-net.

Moreover, as shown in Section 3 each ordered list of invariants for the place/transition-net determines an invariant for the constructed coloured Petri net.

It is thus important to choose the right abstraction-level for places, transitions and invariants in coloured Petri nets. In terms of mathematics this is equivalent to the use of functions, which are determined by simple matrices where the different colours are treated in a systematic way.

From the discussion above it follows that place/transition-nets and coloured Petri nets are equivalent with respect to descriptive power (in the sense formally defined in [2]). Equivalence with respect to descriptive power, means that the two formalisms in principle can be used to describe the same class of systems. It tells nothing about the usefulness or succinctness of the respective descriptions.

It should be mentioned that our invariant-method at present is non-constructive in the sense that it gives no algorithm to construct invariants (without transforming the coloured Petri net to the corresponding place/transition-net and then constructing invariants from the expanded incidence-matrix). It will be a subject for future research to investigate to which degree the methods for solution of linear equations apply when multiplication of integers is replaced by composition of functions. In [9] we define a set of transformation rules, which can be used to simplify the incidence-matrix of a coloured Petri net, without changing the set of invariants.

Fortunately it seems often to be the case that a number of potential invariants can be found from the properties we expect the net to fulfil. It is then easy, using our method, to check whether they really are invariants. If this is the case new invariants can be constructed from them by means of addition, scalar multiplication and functional composition. An example of this will be given in Section 5.

5. Network of databases

As a more complicated example of the use of coloured Petri nets we consider the following system from [1].

A set of database managers, $\mathrm{DBM} = \{d_1, d_2, \ldots, d_n\}$, $n > 0$, communicate with each other. Each manager can make an update to his own database. At the same time he must send a message to each of the other managers thereby informing them about the update. Having sent this set of messages, the sending manager waits until all other managers have received his message, performed an update and sent an acknowledgment. When all acknowledgments are present the sending manager returns to the inactive. At that time (but not before) another manager may perform an update and send messages.

Each manager can be in three states: 'inactive', 'waiting' (for acknowledgments) and 'performing' (an update on request of another manager). The managers communicate via a fixed set of message buffers, $\mathrm{MB} = \{\langle s, r\rangle \mid s, r \in \mathrm{DBM} \wedge s \neq r\}$, where s represents the sender and r represents the receiver. Each message buffer may be in four different states: 'unused', 'sent', 'received' and 'acknowledged'. The system can be described by the coloured Petri net in Fig. 13.

E is a set containing only a single element ε. In formal sums we shall often write n instead of $n\varepsilon$, where $n \in \mathbb{N}$. Intuitively ε represents tokens without a colour. For any set of colours C we define $\mathrm{ABS} \in [C \to [E \to \mathbb{Z}]_f]$ and $\mathrm{ID} \in [C \to [C \to \mathbb{Z}]_f]$ by

$$\forall c \in C \quad [\mathrm{ABS}(c) = \varepsilon \wedge \mathrm{ID}(c) = c].$$

To be rigorous ABS and ID should be equipped by an index stating their domain. Intuitively (the linear extension of) ABS counts the number of tokens in its argument ignoring their colour. Thus it plays a similar role as the value-concept in [1, 9], but in our formalism it is fully integrated in the method and has no special status. $\mathrm{ABS}(x)$ will often be written as $|x|$. As an example $\mathrm{ABS}(3u - v + 2w) = |3u - v + 2w| = 4\varepsilon = 4$ for $u, v, w \in C$.

The functions REC and MINE are defined by

$$\forall \langle s, r\rangle \in \mathrm{MB} \quad [\mathrm{REC}(\langle s, r\rangle) = r],$$

$$\forall s \in \mathrm{DBM} \quad \left[\mathrm{MINE}(s) = \sum_{r \neq s} \langle s, r\rangle\right].$$

In the initial marking $m_0(\text{inactive}) = \sum \mathrm{DBM}$, $m_0(\text{unused}) = \sum \mathrm{MB}$ and $m_0(\text{exclusion}) = 1$. All other places are unmarked.

The incidence-matrix is shown in Fig. 14. For the moment ignore invariants (i6) and (i7).

From

(i1) $$m(\text{inactive}) + m(\text{waiting}) + m(\text{performing}) = \sum \mathrm{DBM}$$

we conclude that each database manager is in exactly one of its three states.

From

(i2) $$m(\text{unused}) + m(\text{sent}) + m(\text{received}) + m(\text{acknowledged}) = \sum \mathrm{MB}$$

we conclude that each message buffer is in exactly one of its four states.

From

(i3) $$|m(\text{waiting})| + m(\text{exclusion}) = 1$$

we conclude that at most one manager can be 'waiting'.

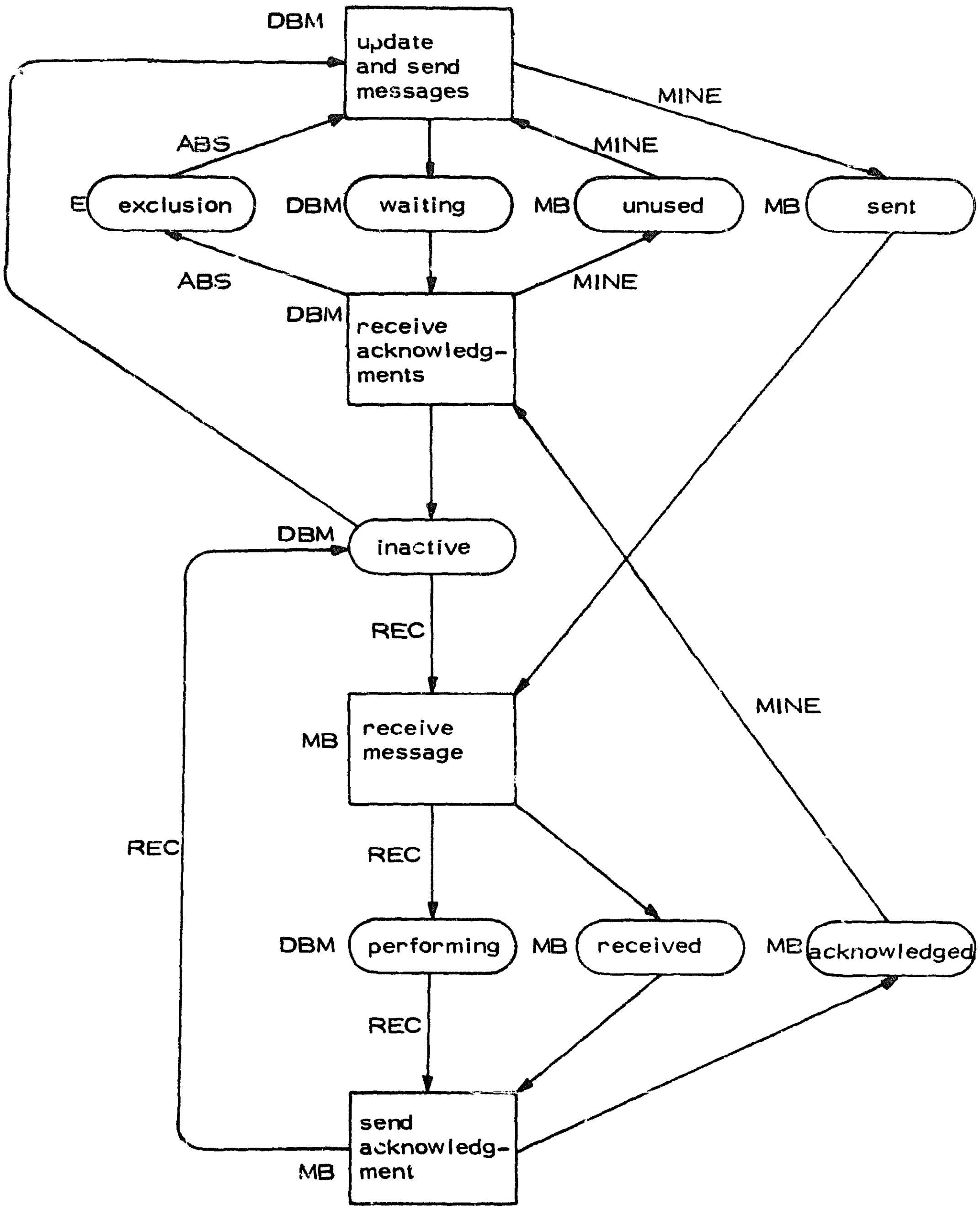

Fig. 13. Coloured Petri net describing a network of databases with a simple communication discipline.

From

(i4) $\qquad m(\text{performing}) = \text{REC}(m(\text{received}))$

we conclude that a manager is 'performing' iff there is a message buffer addressed to him on 'received'.

From

(i5) $\qquad \text{MINE}(m(\text{waiting})) = m(\text{sent}) + m(\text{received}) + m(\text{acknowledged})$

		update and send messages	receive acknowledgments	receive message	send acknowledgment	m_0	Invariants						
							i1	i2	i3	i4	i5	i6	i7
												REC(i5)−i4	(n−1)i3−ABS(i6)
		DBM	DBM	MB	MB		DBM	MB	E	DBM	MB	DBM	E
inactive	DBM	−ID	ID	−REC	REC	ΣDBM	ID						
waiting	DBM	ID	−ID				ID		ABS		MINE	(n−1)ID	
performing	DBM			REC	−REC		ID			ID		−ID	ABS
exclusion	E	−ABS	ABS			1			ID				(n−1)ID
unused	MB	−MINE	MINE			ΣMB		ID					
sent	MB	MINE		−ID				ID			−ID	−REC	ABS
received	MB			ID	−ID			ID		−REC	−ID		
acknowledged	MB		−MINE		ID			ID			−ID	−REC	ABS

Fig. 14. Incidence-matrix for the coloured Petri net in Fig. 13.

we conclude that when a manager is 'waiting' all his message buffers are either 'sent', 'received' or 'acknowledged' (and thus none of them are 'unused'). Moreover, when he is not 'waiting' none of his message buffers are 'sent', 'received' or 'acknowledged' (and thus they are all 'unused').

5.1. *Completing a marking*

Analysis 4. Let m be a marking reachable from the initial marking with $m(\text{performing}) = u_1 + u_2 + u_3$ (where u_1, u_2 and u_3 are different elements of DBM). Then $m(\text{received}) = \langle q, u_1\rangle + \langle q, u_2\rangle + \langle q, u_3\rangle$ for some $q \in \text{DBM}$ and $q \neq u_i$ for all $i \in 1, \ldots, 3$.

Proof. From (i4) we conclude that $m(\text{received}) = \langle q_1, u_1\rangle + \langle q_2, u_2\rangle + \langle q_3, u_3\rangle$ for some $q_i \in \text{DBM}$ and $q_i \neq u_i$ for all $i \in 1, \ldots, 3$.

From (i5) we conclude that $q_i \leq m(\text{waiting})$ for all $i \in 1, \ldots, 3$ and then it follows from (i3) that $q_1 = q_2 = q_3$.

The corresponding proof in [1] uses one page of rather complicated equations and moreover part of the predicate/transition-net is unfolded to a complicated place/transition-net.

5.2. *Constructing a complicated invariant from simpler ones*

In [1] an invariant is constructed through the places 'performing', 'exclusion', 'sent' and 'acknowledged'. By our invariants (i3) and (i4) there is a simple relationship between 'waiting' and 'exclusion' and between 'performing' and 'received'. Thus the above invariant in [1] is similar to our (i5). If, however, for some reason we want to construct an invariant through exactly the same four places as [1], this can be done in two steps as shown by invariants (i6) and (i7) in Fig. 14. As indicated, (i6) is constructed from (i5) and (i4) by means of the function REC and subtraction. Then (i7) is constructed from (i3) and (i6) by means of scalar multiplication, the function ABS and subtraction (n is the number of managers).

It should be added that, in our opinion, (i5) is more interesting than (i7) since it allows us to deduce more information about the colours of the involved tokens.

Analysis 5. The coloured Petri net in Fig. 13 cannot deadlock.

Proof. Assume that a marking m is reachable from the initial marking.

If *at least one manager* $d \in \text{DBM}$ *is 'waiting' in* m, it follows from (i5) that his message buffers are either 'sent', 'received' or 'acknowledged'.

If at least one buffer $\langle d, r\rangle \in \text{MB}$ is 'received' it follows from (i4) that r is 'performing' and then 'send acknowledgment' has concession (with colour r).

If at least one buffer $\langle d, r\rangle \in \text{MB}$ is 'sent' it follows from (i3) and $d \neq r$ that r cannot be 'waiting'. If r is 'performing' we conclude from (i4) that there is a buffer $\langle e, r\rangle \in \text{MB}$, which is 'received' and from (i5) e is 'waiting' but then $e = d$ from (i3).

We then have that $\langle d, r\rangle$ is both 'sent' and 'received' in contradiction to (i2). Thus it follows from (i1) that r must be 'inactive', and then 'receive message' has concession (with colour r).

If all d's buffers are 'acknowledged', 'receive acknowledgments' has concession (with colour d).

If *at least one manager* $d \in DBM$ *is 'performing' in* m, it follows from (i4) that there is a buffer $\langle s, d\rangle$, which is 'received' and thus 'send acknowledgment' has concession (with colour d).

If *no manager is 'waiting' or 'performing' in* m it follows from (i1) that all managers are 'inactive', from (i3) that 'exclusion' is marked and from (i2) and (i5) that all message buffers are 'unused'. Thus 'update and send messages' has concession (with any colour in DBM).

Acknowledgment

The development of the generalized invariant-method was strongly promoted by many discussions with Mogens Nielsen and Erik Meineche Schmidt. Useful comments were also made by Hartmann Genrich, Kurt Lautenbach, Brian Mayoh and Antoni Mazurkiewicz.

References

[1] H.J. Genrich and K. Lautenbach, The analysis of distributed systems by means of predicate/transition-nets, in: G. Kahn, Ed., *Semantics of Concurrent Computation*, Lecture Notes in Computer Science **70** (Springer, Berlin, 1979) 123–146.

[2] K. Jensen, A method to compare the descriptive power of different types of Petri nets, in: P. Dembiński, Ed., *Mathematical Foundations of Computer Science 1980*, Lecture Notes in Computer Science **88** (Springer, Berlin, 1980) 348–361.

[3] K. Lautenbach, Liveness in Petri nets, Interner Bericht ISF-75-02.1, GMD Bonn (1979).

[4] J.L. Peterson, Petri nets, *Comput. Surveys* **9** (3) (1977), 223–252.

[5] C.A. Petri, Concepts of net theory, *Summer School on Mathematical Foundations of Computer Science*, High Tatras 1973, Math. Inst. Slovak Academy of Science (1973) 137–146.

[6] C.A. Petri, Interpretations of net theory, Interner Bericht ISF-75-07, GMD Bonn (1975).

[7] M. Schiffers and H. Wedde, Analyzing program solutions of coordination problems by CP-nets, in: J. Winkowski, Ed., *Mathematical Foundations of Computer Science 1978*, Lecture Notes in Computer Science **64** (Springer, Berlin, 1978) 462–473.

[8] H.J. Genrich and K. Lautenbach, System modelling with high-level Petri nets, *Theoret. Comput. Sci.* **13** (1981) 109–136.

[9] K. Jensen, How to find invariants for coloured Petri nets, DAIMI PB-120, Aarhus University (1980).