



Petri nets, algebras and morphisms

Glynn Winskel

Technical reports published by the University of Cambridge
Computer Laboratory are freely available via the Internet:

<http://www.cl.cam.ac.uk/TechReports/>

ISSN 1476-2986

PETRI NETS, ALGEBRAS AND MORPHISMS

by

Glynn Winskel

University of Cambridge,
Computer Laboratory,
Corn Exchange Street,
Cambridge CB2 3QG.

Abstract

It is shown how a category of Petri nets can be viewed as a subcategory of two sorted algebras over multisets. This casts Petri nets in a familiar framework and provides a useful idea of morphism on nets different from the conventional definition—the morphisms here respect the behaviour of nets. The categorical constructions which result provide a useful way to synthesise nets and reason about nets in terms of their components; for example various forms of parallel composition of Petri nets arise naturally from the product in the category. This abstract setting makes plain a useful functor from the category of Petri nets to a category of spaces of invariants and provides insight into the generalisations of the basic definition of Petri nets—for instance the coloured and higher level nets of Kurt Jensen arise through a simple modification of the sorts of the algebras underlying nets. Further, it provides a smooth formal relation with other models of concurrency such as Milner's Calculus of Communicating Systems (CCS) and Hoare's Communicating Sequential Processes (CSP).

Introduction.

The purposes of this paper are threefold. Firstly there is a lot of interest in how to combine Petri nets to make reasoning with them simpler and more structured. An approach to this is offered here. Secondly there are many different kinds of Petri nets around and it is not always clear how they relate to each other. Thirdly it is not often clear how Petri net models of processes relate to other models like the interleaving models of Milner's CCS and Hoare's CSP. This paper demonstrates, I believe, that by casting Petri nets in a more abstract algebraic framework their features and relations to other models can be appreciated better.

The graphical representation of Petri nets has been a mixed blessing. For small examples a graphical representation has undeniable, immediate appeal. For larger examples graphical representations are hard to comprehend. This is despite some success in finding abbreviated ways to describe Petri nets such as the predicate-transition nets of Genrich and Lautenbach and the coloured nets of Jensen. A graphical notation can sometimes obscure the more abstract treatment necessary to advance our understanding. This happened with flow diagrams where Floyd's rules were fairly complicated compared with Hoare's rules for while-programs, and it seems to have happened with Petri nets too. What is simple graphically may be awfully difficult logically when it comes to reasoning about the behaviour of

programs or systems. Worse still, constructions that are meaningful on graphs may fail to make sense in terms of the behaviour they are intended to convey. I believe one can see an example of this in the old definition of morphism on Petri nets, given in [Br], which does not preserve the dynamic behaviour of nets.

It is commonly accepted that we require ways to combine Petri nets and to structure and direct our reasoning. The work of Hoare on CSP [H] and Milner on CCS [M] and earlier Campbell, Lauer and Habermann on path expressions [CL] has thrown light on useful combinators for parallel processes. So has the work of the Polish school (notably Mazurkiewicz [Maz]) and Hungarian school (notably [GKR]) on ways of combining processes modelled as sets of traces. It seems sensible to incorporate these ideas into the theory of Petri nets.

Of course, there are many ways to attempt this. One is that of Boudol, Roucairol and de Simone [BRS] which essentially translates every finite Petri net into a Meije process, and thus inherits compositionality from Meije, a descendent of the CCS and CSP family of languages. In a sense their approach implements Petri nets as Meije processes. The approach in this paper is different. It is founded on the view that Petri nets are a fundamental mathematical model of computation, like finite and infinite state machines, say, but in which the concurrency structure is given explicitly. As such, the theory of Petri nets should be developed to the point where it is easy to model and reason about a wide range of languages for parallel processes, including CCS, CSP and Meije. So, in this paper the combinators on nets are not derived from any other calculus but rather are consequences of their mathematical structure.

To establish the correct mathematical structure of nets we must look beyond their graphical representation which can be deceptive. Here we advocate the view that Petri nets are special kinds of algebras, and so are objects of a well-known mathematical nature. As algebras they support a notion of homomorphism on which we base the definition of morphism between Petri nets. The morphisms on Petri nets proposed are significantly different from the morphisms defined in [Br], and do preserve the dynamic behaviour of nets. Extended in this way Petri nets form a category. One pay-off is that now several combinators arise naturally as categorical constructions. In the category the product is a construction which takes two nets and introduces events of synchronisation between them, and the coproduct (in fact in a subcategory) is a construction which is a form of nondeterministic disjunction of nets, like Milner's sum. Incidentally, our use of category theory will be light; good references are [AM] and [Mac].

There are several important consequences of this mathematical set-up. Each categorical construction comes equipped with morphisms which relate it to its components. These furnish proof methods to reason about the construction in terms of its components. Another consequence is the way in which invariants in the domain of a morphism of nets are inherited from invariants of the codomain, giving a contravariant functor from the category of Petri nets to the category of spaces of invariants. The approach generalises to higher level nets like those introduced by Jensen. Fortunately a similar approach works

for a variety of different models of computation. In them too familiar constructions, like parallel compositions, turn out to be significant categorically. In many cases models, Petri nets among them, can be related by a pair of functors forming a coreflection, a special kind of adjunction, between the two associated categories. Because of the way in which coreflections preserve categorical constructions, they form a bridge translating between the different models, and in fact many models can be embedded in the model of Petri nets in this way. Consequently, it can be seen how semantics expressed in terms of one model translates to semantics in terms of another. This is an extra benefit to a more abstract approach to Petri nets than is usual. It will only be sketched in this paper and the reader is referred to [W4] for more details.

1. Petri nets.

Petri nets model processes in terms of how the occurrence of events incur changes in local states called conditions. This is expressed by a causal dependency (or flow) relation between sets of events and conditions, and it is this structure which determines the dynamic behaviour of nets once the causal dependency relation is given its natural interpretation. The most well-known definition of Petri nets has the following form. We refer the reader to the appendix for a detailed treatment of multisets, though for the moment we do not require much.

Definition. A *Petri net* is a 4-tuple (B, E, F, M_0) where

B is a non-null set of conditions,

E is a disjoint set of events,

F is a multiset of $(B \times E) \cup (E \times B)$, called the *causal dependency relation*,

M_0 is a non-null multiset of conditions, called the *initial marking*,

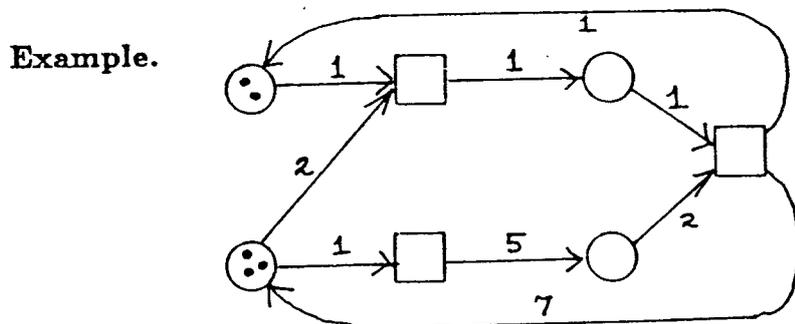
which satisfy the restrictions:

- (i) $\forall e \in E \exists b \in B. F_{b,e} > 0$ and $\forall e \in E \exists b \in B. F_{e,b} > 0$ and
- (ii) $M_b \neq 0$ or $(\exists e \in E. F_{e,b} \neq 0)$ or $(\exists e \in E. F_{b,e} \neq 0)$.

Thus we insist that each event causally depends on at least one condition and has at least one condition which is causally dependent on it. For technical reasons which will become clear in section 3 it is convenient to also insist that safe nets have no *isolated* conditions *i.e.* that a condition is either marked initially or the pre or post condition of some event. This restriction is no handicap because, according to the dynamic behaviour of nets, an isolated condition can never hold.

Such nets are often called place-transition nets though here we do not attribute a capacity to each condition (or place). Also like [J1,2], though unlike [R], we do not allow markings with infinite multiplicity. This does not seem to be a limitation in practice, though the work here could be generalised to such kinds of markings.

Nets have a well-known graphical representation in which events are represented as boxes and conditions as circles with directed arcs between them, weighted by positive integers, to represent the flow relation. The initial marking is represented by placing “tokens” to the appropriate multiplicity on each weighted condition.



By convention we understand an arc which carries no label to stand for an arc with weight 1. Sometimes we mark a condition by an integer, e.g. $\textcircled{2}$, to represent its multiplicity.

We have yet to formalise the well-known “token game” on Petri nets through which they are equipped with a dynamic behaviour. This we postpone till we have presented another view of nets which casts them in more traditional algebraic framework. It is useful to regard a Petri net as a 2-sorted algebra on multisets. This view underlies the techniques for finding invariants of nets by linear algebra [Pe, Br, R]. We shall use some basic definitions and notation of multisets, and later of vectors and modules, which is introduced in the appendix. Because we want to deal with infinite Petri nets, and not just finite nets, we must take some trouble over operations on multisets and vectors which become a little more complicated when over infinite bases. This is tackled in detail in the appendix, though a casual reader can understand the main ideas without much reference to it.

2. Nets viewed as algebras.

It is useful, both notationally and conceptually, to regard a Petri net as a 2-sorted algebra on multisets. It provides notation for describing the dynamic behaviour of nets (the “token game”) and prompts us in the direction of a useful definition of morphisms on Petri nets.

From classical mathematics we are familiar with algebras over sets, whether they are single-sorted like groups, rings or fields, or two-sorted like vector spaces or modules. It is noteworthy that nets can be viewed in this traditional setting and that when we do familiar constructions on nets reappear as well-known algebraic constructions.

Proposition. A Petri net (B, E, F, M_0) determines a 2-sorted algebra over multisets: It has sorts μB and μE , with operations a constant, $M_0 \in \mu B$ and two unary operations $\bullet(\) : E \rightarrow_{\mu} B$, with matrix $(F_{b,e})_{b \in B, e \in E}$, and $(\)^{\bullet} : E \rightarrow_{\mu} B$, with matrix $(F_{e,b})_{b \in B, e \in E}$.

Conversely, a 2-sorted algebra over multisets with sorts μB and μE , with a constant $M_0 \in \mu B$ and two unary operations $\bullet(\), (\)^\bullet : E \rightarrow_\mu B$, which satisfies

$$(i) \quad M_0 \neq \underline{0} \text{ and } (\bullet A = \underline{0} \text{ or } A^\bullet = \underline{0}) \Rightarrow A = \underline{0} \text{ and}$$

(ii) $\forall b \in B. M_b \neq 0 \text{ or } (\exists e \in E. (e^\bullet)_b \neq 0) \text{ or } (\exists e \in E. (\bullet e)_b \neq 0$
determines a Petri net (B, E, F, M_0) by taking

$$F_{b,e} = (\bullet e)_b \quad \text{and} \quad F_{e,b} = (e^\bullet)_b.$$

This gives a 1-1 correspondence between Petri nets and 2-sorted algebras over multisets which satisfy (i) and (ii).

Remark. This view of Petri nets, and more generally of predicate transition nets, as algebras was advocated by W. Reisig in [R1]. In future it will sometimes be convenient to describe a Petri net as a structure $(B, E, \bullet(\), (\)^\bullet, M_0)$.

Of course it is possible to view many structures as algebras. What is not always so clear is the use of doing so. Our first piece of evidence that it is useful comes from the fact that homomorphisms on Petri nets preserve the dynamic behaviour of nets.

3. The dynamic behaviour of nets.

As is well-known, states of a net are represented as *markings* which are simply multisets of conditions. You can think of a condition as a resource and its multiplicity as the amount of the resource. As an event occurs it consumes certain resources and produces others. What and how much is specified by the relation F . Continuing this interpretation, if there are enough resources then more than one event can occur concurrently, and it is even allowed that an event can occur to a certain multiplicity. This leads us to the following account of the "token game" on Petri nets—it differs from some others in that we do not play the token game by firing only one event at a time but allow instead transitions in which finite multisets of events fire.

Let $N = (B, E, F, M_0)$ be a Petri net.

A *marking* M is a multiset of conditions, i.e. $M \in \mu B$.

Let M, M' be markings. Let A be a finite multiset of events. Define

$$M \xrightarrow{A} M' \Leftrightarrow \bullet A \leq M \ \& \ M' = M - \bullet A + A^\bullet.$$

This gives the *transition relation* between markings. When we wish to stress the net N in which the transition $M \xrightarrow{A} M'$ occurs we write

$$N : M \xrightarrow{A} M'.$$

A *reachable marking* of N is a marking M for which

$$M_0 \xrightarrow{A_0} M_1 \xrightarrow{A_1} \dots \xrightarrow{A_{n-1}} M_n = M$$

for some markings and finite multisets of events.

The reason for only allowing finite multisets of events to occur as transitions is in order that the occurrence of an event only depends on a finite set of event occurrences, and so exclude such processes as lead to the paradoxes of Zeno. It has many technical advantages too, especially when relating Petri nets to other models, though this will not be so evident from the work here.

Now we make precise the sense in which homomorphisms of Petri nets (or strictly speaking their associated algebras) preserve their dynamic behaviour. Let's spell out what it means to be a homomorphism between 2-sorted algebras of the type associated with nets. Recall a homomorphism of 2-sorted algebras over a suitable category consists of a pair of sort-respecting morphisms of the category which preserve the operations of the algebras.

Definition. Let $N = (B, E, F, M)$ and $N' = (B', E', F', M')$ be nets. A homomorphism from N to N' is a pair of multirelations (η, β) with $\eta : E \rightarrow_{\mu} E'$ and $\beta : B \rightarrow_{\mu} B'$ such that

$$\beta M = M' \ \& \ \forall A \in \mu E. \ \bullet(\eta A) = \beta(\bullet A) \ \& \ (\eta A)^{\bullet} = \beta(A^{\bullet}).$$

Say a homomorphism is *finitary* when ηe is a finite multiset for all events e .

You can see a homomorphism of nets preserves initial markings and the environments of events.

Theorem. Let $(\eta, \beta) : N \rightarrow N'$ be a finitary homomorphism of Petri nets. Then β preserves the initial marking and if $M \xrightarrow{A} M'$ in N then $\beta M \xrightarrow{\eta A} \beta M'$ in N' .

Proof. Directly from the definition we see that β preserves the initial marking. Assume $N : M \xrightarrow{A} M'$. As the homomorphism is finitary we see ηA is a finite multiset. Also $\bullet A \leq M$, so $\bullet \eta A = \beta(\bullet A) \leq \beta M$, and

$$M' = (M - \bullet A) + A^{\bullet}.$$

Now applying β ,

$$\begin{aligned} \beta M' &= \beta((M - \bullet A) + A^{\bullet}) \\ &= \beta M - \beta(\bullet A) + \beta(A^{\bullet}) && \text{by linearity} \\ &= \beta M - \bullet(\eta A) + (\eta A)^{\bullet} && \text{by the defn. of h'morphism.} \end{aligned}$$

But these facts express that $N' : \beta M \xrightarrow{\eta A} \beta M'$. ■

Corollary. *Finitary homomorphisms preserve reachable markings i.e. if M is a reachable marking of a net N and (η, β) is a finitary homomorphism from N to N' then βM is a reachable marking of N' .*

Proof. By repeated application of the theorem above. ■

If (η, β) is a homomorphism from N to N' , as a computation

$$M_0 \xrightarrow{A_0} M_1 \xrightarrow{A_1} \dots \xrightarrow{A_{n-1}} M_n \xrightarrow{A_n} \dots$$

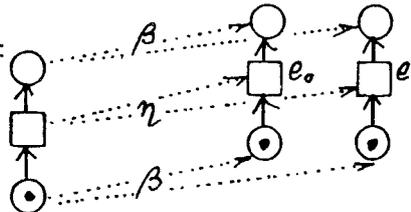
is traced-out in N so the computation

$$\beta M_0 \xrightarrow{\eta A_0} \beta M_1 \xrightarrow{\eta A_1} \dots \xrightarrow{\eta A_{n-1}} \beta M_n \xrightarrow{\eta A_n} \dots$$

is traced-out in N' .

Thus finitary homomorphisms preserve the behaviour of nets. They do this in a local way by expressing how the occurrence of an event in one net induces the a finite multiset of occurrences in the other and the holding of a condition in one net induces a multiplicity of condition holdings in the other. Still, this is not to say that such homomorphisms are the natural morphisms to take on Petri nets from all points of view. Here is one example which one can argue runs counter to intuitions about the nature of events in Petri nets.

Example. A finitary homomorphism:



There is a problem with the interpretation of the homomorphism in this example. The occurrence of a single event in the domain of the homomorphism induces the simultaneous or coincident occurrence of two events e_0 and e_1 in its range. This goes against a view of Net theory, expressed by Petri, that events which are coincident are the same event. It seems that morphisms should be homomorphisms which *preserve events*, in the sense that η should be a partial function, thus forbidding the example above. Our argument is based, in addition, on the kinds of homomorphisms that arise naturally from familiar constructions on Petri nets and the fact that by making the suggested restriction we obtain many useful categorical constructions and a smooth relationship between Petri nets and other models of parallel computation.

For emphasis:

Definition. A morphism on Petri nets $N \rightarrow N'$ is a homomorphism $(\eta, \beta) : N \rightarrow N'$, on the nets viewed as algebras, in which η is a partial function (recall we identify partial functions with their linear extensions to multirelations) *i.e.* the matrix of η satisfies

$$\eta_{e,e'} \leq 1 \quad \text{and} \\ \eta_{e,e'} = 1 \ \& \ \eta_{e,e''} = 1 \Rightarrow e' = e''$$

for events e , e' and e'' . Say a morphism (η, β) of nets is *synchronous* when η is a total function on events.

Remark. This definition of morphism generalises those in [W3] for safe nets and in [GR] from causal nets to Petri nets.

Because morphisms on Petri nets are finitary homomorphisms it is obvious that

Theorem. *Let $(\eta, \beta) : N \rightarrow N'$ be a morphism of Petri nets. Then β preserves the initial marking and if $M \xrightarrow{A} M'$ in N then $\beta M \xrightarrow{\eta A} \beta M'$ in N' .*

It is not obvious straightaway that Petri nets with finitary homomorphisms and Petri nets with morphisms form categories because the composition of multirelations might yield an ∞ -multirelation. However because we insist that nets do not have isolated conditions, it turns out that composition of finitary homomorphisms is a finitary homomorphism, and consequently the composition of morphisms on Petri nets always exists.

Proposition. *Petri nets with finitary homomorphisms form a category in which the composition of two finitary homomorphisms $(\eta_0, \beta_0) : N_0 \rightarrow N_1$ and $(\eta_1, \beta_1) : N_1 \rightarrow N_2$ is $(\eta_1 \eta_0, \beta_1 \beta_0) : N_0 \rightarrow N_2$ and the identity morphism for a net N has the form $(1_E, 1_B)$ where 1_E and 1_B are the identities on the spaces of event-multisets and condition-multisets respectively.*

Petri nets with morphisms, and Petri nets with synchronous morphisms form subcategories of the category of nets with finitary homomorphisms.

Proof. We only check that finitary homomorphisms are closed under composition. It is easy to verify the other properties required of a category.

Let $(\eta_0, \beta_0) : N_0 \rightarrow N_1$ and $(\eta_1, \beta_1) : N_1 \rightarrow N_2$ be finitary homomorphisms between nets N_i for $i = 1, 2, 3$ with conditions, events and initial markings B_i, E_i, M_i . Clearly $\eta_1 \eta_0(e)$ is a finite multiset over E_2 and $\beta_1 \beta_0 : B_0 \rightarrow_{\mu}^{\infty} B_2$. By the properties of homomorphisms we obtain

$$\begin{aligned}\beta_1 \beta_0 M_0 &= \beta_1 M_1 = M_2, \\ \beta_1 \beta_0(\bullet e) &= \beta_1(\bullet \eta_0 e) = \bullet(\eta_1 \eta_0 e), \\ \beta_1 \beta_0(e \bullet) &= (\eta_1 \eta_0 e) \bullet.\end{aligned}$$

As each condition b of N_0 is not isolated, for c a condition of N_2 , either

$$\begin{aligned}(\beta_1 \beta_0)_{c,b} &\leq (M_2)_c < \infty \quad \text{or} \\ (\beta_1 \beta_0)_{c,b} &\leq (\bullet(\eta_1 \eta_0 e))_c < \infty \quad \text{or} \\ (\beta_1 \beta_0)_{c,b} &\leq ((\eta_1 \eta_0 e) \bullet)_c < \infty.\end{aligned}$$

Thus in any case $(\beta_1 \beta_0)_{c,b} \neq \infty$. Hence $\beta_1 \beta_0 : B_0 \rightarrow_{\mu} B_2$ making the composition $(\eta_1 \eta_0, \beta_1 \beta_0) : N_0 \rightarrow N_2$ a finitary homomorphism of nets.

As the composition of partial functions on events gives a partial function on events, and the composition of total functions is total, it is now easy to see that nets with net-morphisms, and synchronous morphisms, form subcategories. ■

This result has significance now we turn to consider some constructions on nets and the role of morphisms in their definition and characterisation.

4. Some constructions on Petri nets.

Perhaps the most interesting construction is a generalisation of the product-machine construction from Automata theory. A restricted form of it was presented in the early work of Lauer and Campbell when they were giving a Petri net semantics to path expressions in [LC]. This construction arises naturally when modelling concurrent processes, like those associated with CCS or CSP programs, which synchronise at certain events. Precisely which events depends on their nature and this is generally captured in the net model by adding extra structure in the form of labels attached to the events. The construction we give allows arbitrary synchronisations—unwanted synchronisations can be removed using an operation of restriction which we present later.

The product of Petri nets:

Let $N_0 = (B_0, E_0, F_0, M_0)$ and $N_1 = (B_1, E_1, F_1, M_1)$ be nets. Disjoint copies of the two nets N_0 and N_1 are juxtaposed and extra events of synchronisation of the form (e_0, e_1) are adjoined, for e_0 an event of N_0 and e_1 an event of N_1 ; an extra event (e_0, e_1) has as preconditions and postconditions those of its components in the obvious way, which we shall make precise in a moment. It is useful to think of the copies of the original events, those which are not synchronised with any companion event of the the other process as having the form $(e_0, *)$ in the copy of N_0 and the form $(*, e_1)$ in the copy of N_1 . Then the events of the product have the form

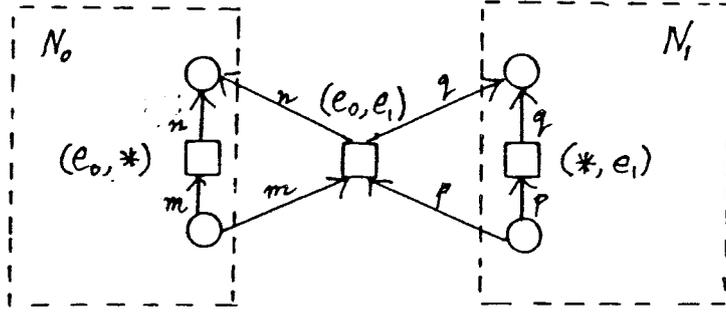
$$E = \{(e_0, *) \mid e_0 \in E_0\} \cup \{(e_1, *) \mid e_1 \in E_1\} \cup \{(e_0, e_1) \mid e_0 \in E_0 \ \& \ e_1 \in E_1\}.$$

There is an obvious partial function from the events of the product to the events of a component. Define $\pi_0 : E \rightarrow E_0$ by $\pi_0(e_0, e_1) = e_0$ —this will be undefined if $e_0 = *$. Define π_1 similarly. To be more precise about the conditions we can assume that they have the form $B = B_0 \uplus B_1$ the disjoint union of B_0 and B_1 . Define ρ_0 to be the opposite relation to the injection $\rho_0^{op} : B_0 \rightarrow B$. This projects conditions in the product back to the component. Define ρ_1 similarly. Take $\rho_0^{op} M_0 + \rho_1^{op} M_1$ as the initial marking of the product. Now we can define the flow relation F in the product by

$$F_{b,e} = \begin{cases} F_0 \rho_0 b, \pi_0 e & \text{if } \rho_0 b \text{ and } \pi_0 e \text{ are defined,} \\ F_1 \rho_1 b, \pi_1 e & \text{if } \rho_1 b \text{ and } \pi_1 e \text{ are defined,} \\ 0 & \text{otherwise} \end{cases}$$

$$F_{e,b} = \begin{cases} F_0 \pi_0 e, \rho_0 b & \text{if } \pi_0 e \text{ and } \rho_0 b \text{ are defined,} \\ F_1 \pi_1 e, \rho_1 b & \text{if } \pi_1 e \text{ and } \rho_1 b \text{ are defined,} \\ 0 & \text{otherwise.} \end{cases}$$

The product of N_0 and N_1 :



Write $N_0 \times N_1$ for the product of the nets N_0 and N_1 .

To understand the construction we must understand the behaviour of the product of two nets in terms of the behaviour of the original nets. For this we need to project the behaviour of the product net to the behaviour of a component net. There are two parts to such a projection, the event part π_i , and the condition part ρ_i . The projection (π_i, ρ_i) from $N_0 \times N_1$ to N_i is a morphism of nets.

Now with the help of the projections we can describe the behaviour of $N_0 \times N_1$.

Theorem. *The behaviour of a product of nets $N_0 \times N_1$ is related to the behaviour of its components N_0 and N_1 by*

$$N_0 \times N_1 : M \xrightarrow{A} M' \quad \text{iff} \quad (N_0 : \rho_0 M \xrightarrow{\pi_0 A} \rho_0 M' \ \& \ N_1 : \rho_1 M \xrightarrow{\pi_1 A} \rho_1 M').$$

A marking M is reachable in $N_0 \times N_1$ iff $\rho_0 M$ is reachable in N_0 and $\rho_1 M$ is reachable in N_1 .

Proof. Let the net N_0 have multirelations $\bullet(\)_0$ and $(\)_0^\bullet$, and N_1 have $\bullet(\)_1$ and $(\)_1^\bullet$. It is easy to see that the projection $(\pi_i, \rho_i) : N_0 \times N_1 \rightarrow N_i$, for $i = 0, 1$, is a morphism i.e. it preserves initial markings and

$$\begin{aligned} \rho_i(\bullet A) &= \bullet(\pi_i A)_i, \\ \rho_i(A^\bullet) &= (\pi_i A)_i^\bullet \end{aligned}$$

for a multiset of events A of the product.

Clearly

$$\begin{aligned} M \leq M' &\Leftrightarrow \rho_0 M \leq \rho_0 M' \ \& \ \rho_1 M \leq \rho_1 M' \quad \text{and} \\ M = M' &\Leftrightarrow \rho_0 M = \rho_0 M' \ \& \ \rho_1 M = \rho_1 M'. \end{aligned}$$

By definition, $M \xrightarrow{A} M'$ in the product iff

$$\bullet A \leq M \ \& \ M' = M - \bullet A + A^\bullet.$$

Now

$$\begin{aligned} \bullet A \leq M &\Leftrightarrow \rho_0(\bullet A) \leq \rho_0 M \ \& \ \rho_1(\bullet A) \leq \rho_1 M \\ &\Leftrightarrow \bullet(\pi_0 A)_0 \leq \rho_0 M \ \& \ \bullet(\pi_1 A)_1 \leq \rho_1 M \end{aligned}$$

as the projections are morphisms. Also

$$M' = M - \bullet A + A^\bullet \Leftrightarrow \rho_0 M' = \rho_0(M - \bullet A + A^\bullet) \ \& \ \rho_1 M' = \rho_1(M - \bullet A + A^\bullet).$$

However

$$\begin{aligned} \rho_0(M - \bullet A + A^\bullet) &= \rho_0 M - \rho_0(\bullet A) + \rho_0(A^\bullet) \\ &= \rho_0 M - \bullet(\pi_0 A)_0 + (\pi_0 A)_0^\bullet \end{aligned}$$

by linearity and the fact that (π_0, ρ_0) is a morphism. Similarly

$$\rho_1(M - \bullet A + A^\bullet) = \rho_1 M - \bullet(\pi_1 A)_1 + (\pi_1 A)_1^\bullet.$$

It follows directly that

$$N_0 \times N_1 : M \xrightarrow{A} M' \quad \text{iff} \quad (N_0 : \rho_0 M \xrightarrow{\pi_0 A} \rho_0 M' \ \& \ N_1 : \rho_1 M \xrightarrow{\pi_1 A} \rho_1 M').$$

Because projections are morphisms and so preserve initial markings, repeated application of this result ensures a marking M is reachable in the product iff $\rho_0 M$ and $\rho_1 M$ are reachable in the components. ■

Intuitively the behaviour of the product is precisely that allowed when we project into the components. The pair of maps (π_0, ρ_0) specifies how the dynamic behaviour of the product of nets, $N_0 \times N_1$, projects to the dynamic behaviour in the component N_0 . The pair (π_1, ρ_1) plays the same role but for the component N_1 . They are essential in describing the behaviour of the product of nets. The proposition above could be turned into a proof rule enabling us to reason about a product of nets in terms of its components.

The name “product” of nets is well-chosen because it is in fact the product in the category of nets with our definition of morphism. Recall the definition of product in a category. A product of two objects N_0 and N_1 consists of an object $N_0 \times N_1$ with projection morphisms $\Pi_0 : N_0 \times N_1 \rightarrow N_0$ and $\Pi_1 : N_0 \times N_1 \rightarrow N_1$ which satisfy the universal property that given any pair of morphisms $f_0 : N \rightarrow N_0$ and $f_1 : N \rightarrow N_1$ there is a *unique* morphism $[f_0, f_1] : N \rightarrow N_0 \times N_1$ such that $f_0 = \Pi_0 \circ [f_0, f_1]$ and $f_1 = \Pi_1 \circ [f_0, f_1]$.

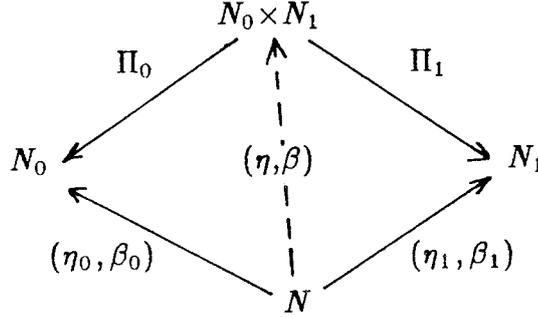
The proof of this characterisation of the product of nets can be seen to rely on the nature of products in two more elementary categories, the category of sets with partial functions, to deal with the event part of the morphisms between nets, and the category of sets with multirelations, to deal with the condition part. The product of two sets E_0 and E_1 in the category of sets with partial functions has the form

$$\{(e_0, *) \mid e_0 \in E_0\} \cup \{(e_1, *) \mid e_1 \in E_1\} \cup \{(e_0, e_1) \mid e_0 \in E_0 \ \& \ e_1 \in E_1\}$$

with the partial functions π_0, π_1 as projections onto the components. The product of two sets B_0 and B_1 in the category of sets with multirelations has the form $B_0 \uplus B_1$ with projections the multirelations ρ_0 and ρ_1 , opposite to the obvious injections.

Theorem. *The product $N_0 \times N_1$, with morphisms (π_0, ρ_0) and (π_1, ρ_1) , is a product in the category of Petri nets.*

Proof. We use the notation introduced in the definition of product. Clearly the product of safe nets is safe. The projections were shown to be morphisms in the previous theorem.



Consider the above diagram in the category of Petri nets morphisms on nets. Take

$$\eta(e) = (\eta_0(e), \eta_1(e))$$

for e an event of N , with the understanding that undefined is represented by $*$. Take

$$\beta = \rho_0^{op} \beta_0 + \rho_1^{op} \beta_1.$$

Then (η, β) is a morphism which makes the diagram commute. The partial function η is uniquely determined by η_0 and η_1 . The β is the unique multirelation such that $\rho_0 \beta = \beta_0$ and $\rho_1 \beta = \beta_1$. Thus the product of nets with projections is a categorical product in the category of nets with net morphisms. ■

The fact that parallel composition is so closely related to a product adds mathematical substance to the intuition that parallelism is a form of orthogonality.

The product construction corresponds to a very liberal form of parallel composition of nets in which arbitrary synchronisations are allowed. Obviously in general some synchronisations are possible and others are not. The operation of restriction allows only certain events to occur. It can be modelled simply by “deleting” the forbidden events from the net and then removing all the conditions which become isolated.

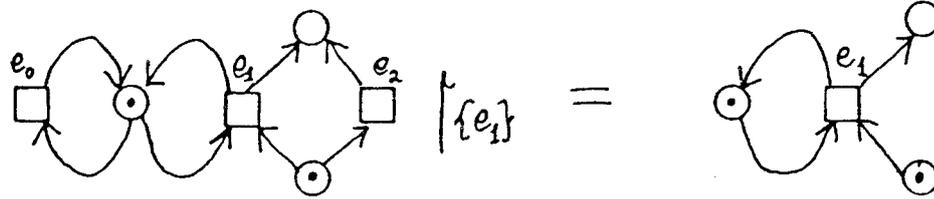
Restriction:

Let $N = (B, E, F, M_0)$ be a net. Let $E' \subseteq E$. Define the restriction of N to E' to be $N[E'] = (B', E', F', M_0)$ where

$$B' = \{b \in B \mid M_{0b} \neq 0 \text{ or } \exists e \in E'. F_{b,e} \neq 0 \text{ or } F_{e,b} \neq 0\},$$

the remaining nonisolated conditions, and F' is F restricted to $(B' \times E') \cup (E' \times B')$ i.e. $F'_{b,e} = F_{b,e}$ and $F'_{e,b} = F_{e,b}$ for $e \in E'$ and $b \in B'$.

Example. Here is a net with its restriction to a subset of events



The behaviour of a net restricted to a set of events is a restriction of the behaviour of the original net.

Proposition. Let $N = (B, E, F, M_0)$ be a net. Let $E' \subseteq E$. Let M and M' be markings of N . Then

$$N[E' : M \xrightarrow{A} M' \Leftrightarrow N : M \xrightarrow{A} M' \ \& \ A \in \mu E']$$

The product and restriction constructions are useful for modelling as nets a wide range of parallel compositions in the literature (see [W1,2,3]). Then it is generally necessary to have some extra labelling structure on the events in the net. The two propositions characterising the behaviour of the product and restriction in terms of their component nets reduce reasoning about a parallel composition to reasoning about its components and the synchronisation discipline.

Synchronous product:

Another important construction can be derived from the product construction with restriction, that of *synchronous product*. It is the restriction of the product of two nets to events of the form (e_0, e_1) where both e_0 and e_1 must be proper, non- $*$ events. Thus there is a tight synchronisation between the components of a synchronous product; in order to occur within a synchronous product every event of one component must synchronise with an event from the other.

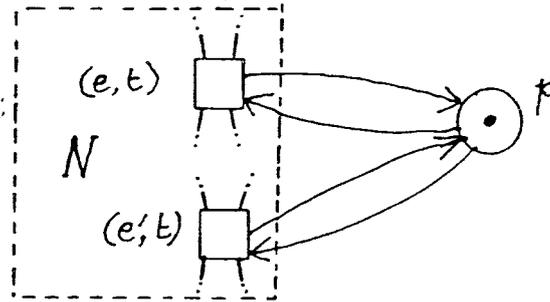
Let $N_0 = (B_0, E_0, F_0, M_0)$ and $N_1 = (B_1, E_1, F_1, M_1)$ be nets. Define their *synchronous product* $N_0 \otimes N_1$ to be the restriction $N_0 \times N_1 | (E_0 \times E_1)$. There are obvious projections got by restricting the projections of the product.

Example. One can represent a ticking clock as the following simple net, call it Ω :



Given an arbitrary net N it is a simple matter to serialise, or interleave, its event occurrences; just synchronise them one at a time with the ticks of the clock. This amounts to forming the synchronous product $N \otimes \Omega$ of N with Ω .

The synchronous product of a net with Ω :



It is easy to check that the synchronous product $N \otimes \Omega$ does serialise the event occurrences of N .

Proposition. M is a reachable marking of $N \otimes \Omega$ and $M \xrightarrow{A} M'$ in $N \otimes \Omega$ iff $M - p$ is a reachable marking of N and $\exists e. A = (e, t) \ \& \ N : (M - p) \xrightarrow{e} (M' - p)$.

Proof. Use the properties of restriction and product. ■

Again the synchronous product has a categorical characterisation; it is the categorical product in the subcategory of nets with synchronous morphisms, where the event part of morphisms corresponds to a total function.

Theorem. The synchronous product $N_0 \otimes N_1$, with the restrictions of the projections is a product in the subcategory of nets in which all the morphisms are synchronous.

Proof. Like that for product. ■

On the coproduct of nets:

I am not sure of the most useful sum construction on Petri nets in general. This is partly because there is not always a coproduct in the category of all nets, as the following counterexample shows. Recall coproduct is the dual notion to that of product got by reversing the arrows.

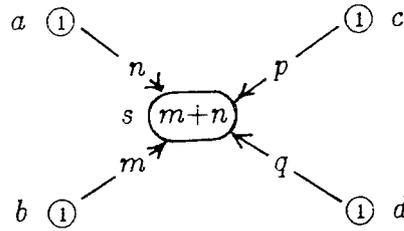
Proposition. The category of Petri nets does not have coproducts in general.

Proof. Let N_0 be the net consisting of just the conditions a and b both in its initial marking. Let N_1 be an isomorphic net with conditions c and d , both marked. We show N_0 and N_1 do not have a coproduct.

Assume they did. Then any marked condition of the coproduct is represented by an element of the set

$$S = \{(ma + nb, pc + qd) \in \mu B_0 \times \mu B_1 \mid 0 < m + n = p + q\}$$

where the pair $(ma + nb, pc + qd)$ represents a condition s marked with multiplicity $m + n = p + q$ related to conditions of the components by these multirelations:



The set S is closed under scalar multiplication and addition given by

$$m(\lambda_0, \lambda_1) = (m\lambda_0, m\lambda_1)$$

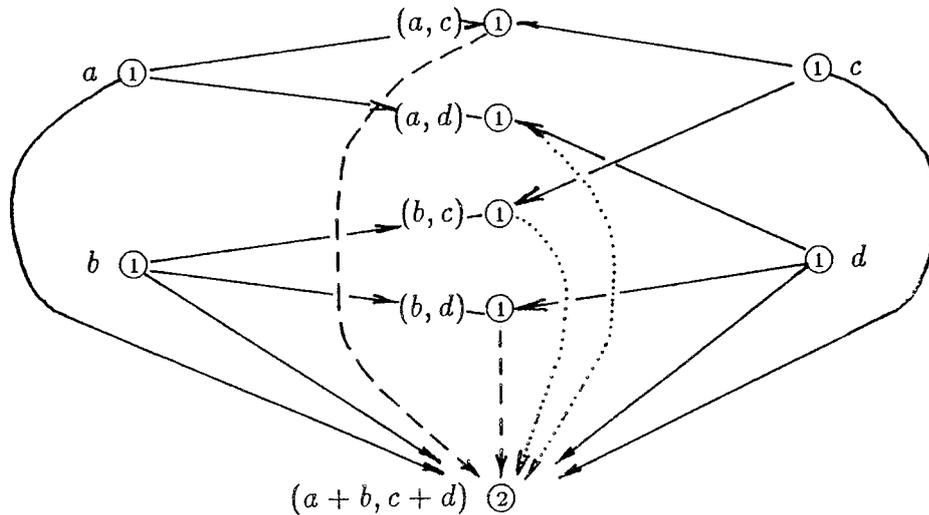
$$(\lambda_0, \lambda_1) + (\nu_0, \nu_1) = (\lambda_0 + \nu_0, \lambda_1 + \nu_1).$$

From the characterising property of coproduct the marked conditions of the coproduct must be represented as a subset $B \subseteq S$ such that any $s \in S$ can be written as a unique linear combination of elements of B . Thus certainly B must contain all the pairs (a, c) , (a, d) , (b, c) , (b, d) because they are all irreducible—expressible as only one linear combination. However

$$(a + b, c + d) = (a, c) + (b, d) = (a, d) + (b, c),$$

so $(a + b, c + d)$ is not uniquely expressible as a linear combination of elements of B . This contradicts the existence of a coproduct for N_0 and N_1 .

Illustration of the counterexample—the two forms of dotted line represent the two different ways to express $(a + b, c + d)$:



■

Despite this negative result, there are coproducts in the more restricted category of safe nets as we shall see in the next section.

The category of Petri nets can be made to work to construct recursively defined nets though we shall not describe the details here. Such nets can be defined in the standard way one builds-up sets by inductive definitions (see [Ac]); one must however take a little care to ensure that the operations on nets are monotonic with respect to the ordering of coordinatewise inclusion on nets, but this is not hard (see [Stu], [GM] though the latter is unnecessarily complicated because they work with equivalence classes of nets). Alternatively, recursion can be handled in a categorical setting using the notion of ω -limits of chains of *net-embeddings* and ω -continuous functors (though at present I have only done this for safe nets).

Projections on nets are examples of a more general notion of morphism between nets. Note how natural is the additional requirement we have imposed on the event part of homomorphisms. Note we do not want morphisms to “preserve conditions” in the sense that β should be a partial function; to do so would rule out the injections used to characterise the behaviour of our sum construction on safe nets in next section.

We remark that the categorical constructions seem to be rather uninteresting in the broader category of Petri nets with finitary homomorphisms. For example the coproduct does not always exist—for the same reasons it does not in the smaller categories—and the product is given simply by the disjoint juxtaposition of nets.

One important consequence of the constructions being categorical is that each comes accompanied by a characterisation to within isomorphism. This means that we need not worry about the details of the concrete and *ad hoc* construction we chose to build-up our product, synchronous product and sum of safe nets. But more important perhaps is the use, which we shall describe briefly later, to translate between different models including Petri nets.

5. Safe nets.

Now we define an important subclass of Petri nets—the safe nets. Some of the results only apply to this subclass. In particular, properties of safe nets can be described with reasonable convenience using just the notation of sets, without using multisets and multirelations (as was done in [W3] when I didn’t know the simpler definition for Petri nets in general).

Definition. Say a Petri net $N = (B, E, F, M_0)$ is safe iff

$$F \leq \underline{1} \quad \text{and} \quad M \leq \underline{1}$$

for all reachable markings M . For safe nets we can write xFy instead of $F_{x,y} = 1$.

For safe nets a condition only holds or fails to hold and an event either occurs or does not occur; they do not happen with multiplicities. For these nets the term “condition” is

consistent with its more usual use where it is imagined to assert a state of affairs which either holds or does not hold. In fact, often people go to the extent of using different terms, like “place” and “transition”, for the conditions and events of the general nets. (I’m not convinced the distinction is worthwhile.)

The behaviour of safe nets is particularly simple and can be expressed just with sets, without the use of multisets. Recall we identify sets with those multisets in which the multiplicity is 1 at greatest.

Proposition. Let $N = (B, E, F, M_0)$ be a safe net.

Let A be a set of events. Then $\bullet A$ and A^\bullet are sets too.

Any reachable marking is a set.

Let M be a reachable marking. Let M' be a marking of N and A a finite multiset of its events. Then $M \xrightarrow{A} M'$ iff A is a set and

$$(\forall e \in A. \bullet e \subseteq M) \ \& \ (\forall e, e' \in A. \bullet e \cap \bullet e' = \emptyset) \ \& \ M' = (M \setminus \bullet A) \cup A^\bullet.$$

For a safe net N , an event e is said to have *concession* at a reachable marking M if $\bullet e \subseteq M$. If two events e and e' have concession at a reachable marking M and share a common precondition, so $\bullet e \cap \bullet e' \neq \emptyset$, the events e, e' are said to be in *conflict* at M because if one occurs at M then the other does not. On the other hand, if $M \xrightarrow{A} M'$ the events in A are said to occur *concurrently*.

Although when working with safe nets it is fairly easy to use only the notation of set theory, a little care is needed in translating between multiset notation and set notation. In this section we need to distinguish notationally between the usual set theoretic application of a relation to a set and multirelation application.

Notation. Let β be a relation from X to Y . Let $Z \subseteq X$. Define the *image* of the Z under the relation β to be the set

$$\beta^{\llcorner} Z = \{y \mid \exists z \in Z. z\beta y\}.$$

Recall we use β^{op} for the opposite relation to β . It is useful to observe that if $\beta : X \rightarrow Y$ is a relation such that $\beta^{op} : Y \rightarrow X$ is a partial function then the multirelation application $\beta(Z)$ of β , regarded as a multirelation, to X , regarded as a multiset, is equal to the set image $\beta^{\llcorner} X$.

When nets are safe, just as their behaviour can be described using sets and relations instead of multisets and multirelations, so can morphisms be characterised in a more elementary manner.

Proposition. Let $N_0 = (B_0, E_0, F_0, M_0)$ and $N_1 = (B_1, E_1, F_1, M_1)$ be safe nets. A pair (η, β) is a morphism $N_0 \rightarrow N_1$ iff η is a partial function from E_0 to E_1 , and β is a relation between B_0 and B_1 such that:

- (i) β^{op} restricts to a total function $M_1 \rightarrow M_0$,
- (ii) If $\eta(e_0) = e_1$ then

$$\begin{aligned} \beta^{op} &\text{ restricts to a total function } \bullet e_1 \rightarrow \bullet e_0 \quad \text{and} \\ \beta^{op} &\text{ restricts to a total function } e_1 \bullet \rightarrow e_0 \bullet, \end{aligned}$$

- (iii) If $\eta(e_0)$ is undefined then $\beta^{\bullet\bullet} e_0 = \emptyset$ and $\beta^{\bullet} e_0 \bullet = \emptyset$.

Proof.

“if”: A pair (η, β) , where η is a partial function on events and β is a relation between conditions, which satisfies (i), (ii) and (iii) above, does form a morphism on nets because:

(i) above ensures β regarded as a multirelation preserves the initial marking—that each condition in M_1 is the image of a unique condition in M_0 ensures βM_0 is a set with $\beta M_0 = M_1$ so they are equal as multisets,

(ii) ensures that if ηe is defined then the multisets $\bullet(\eta e)$ and $\beta(\bullet e)$ are sets and are equal, and similarly that $(\eta e) \bullet = \beta(e \bullet)$, while

(iii) ensures that if ηe is undefined then the multisets $\bullet(\eta e)$ and $\beta(\bullet e)$ are both equal to the null multiset $\underline{0}$, and that $(\eta e) \bullet = \beta(e \bullet) = \underline{0}$.

Thus by linearity (η, β) does indeed form a morphism.

“only if”: Now suppose $(\eta, \beta) : N_0 \rightarrow N_1$ is a morphism on safe nets. By definition η is a partial function on events while β is a relation by the following simple argument. Suppose $\beta_{c,b} > 0$. Recall we assume that the condition b is not isolated, so either b is in the initial marking M_0 or is a pre or post condition of some event e . Accordingly c is in the initial marking M_1 or is the pre or post condition of an event ηe in N_1 . Consider one case, when $b \in \bullet e$. Because (η, β) is a morphism $\beta(\bullet e) = \bullet(\eta e)$. Hence

$$\beta_{c,b} \leq (\beta(\bullet e))_c = (\bullet \eta e)_c = 1$$

as $\bullet \eta e$ is a set. Thus β is a relation.

The remaining properties (i), (ii) and (iii) above express that (η, β) is a homomorphism and take account of multiplicities. ■

Note, it is only because we insist there are no isolated conditions that multirelations on safe nets can be represented as relations.

We showed that morphisms on Petri nets preserved their dynamic behaviour. For safe nets, in the language of sets this becomes:

Proposition. Let $(\eta, \beta) : N_0 \rightarrow N_1$ be a morphism between safe nets N_0 and N_1 . Then M is a reachable marking of N_0 and $N_0 : M \xrightarrow{A} M'$ implies $\beta^{\omega}M$ is a reachable marking of N_1 and $N_1 : \beta^{\omega}M \xrightarrow{\eta^{\omega}A} \beta^{\omega}M'$.

Proof. Suppose M is a reachable marking of N_0 and $N_0 : M \xrightarrow{A} M'$. Then we know the multiset βM is a reachable marking of N_1 and $\beta M \xrightarrow{\eta A} \beta M'$ in N_1 . However, just because βM is reachable in a safe net, it is a set and thus $\beta M = \beta^{\omega}M$. Because $\beta M \xrightarrow{\eta A} \beta M'$ in a safe net N_1 , ηA must be a set. Hence $\eta A = \eta^{\omega}A$. ■

Thus a morphism $(\eta, \beta) : N_0 \rightarrow N_1$ between safe nets expresses how the occurrence of an event e of N_0 induces either a single or no occurrence of an event in N_1 , and how a condition holding in N_0 induces the holding of a set of conditions in N_1 .

The product of safe nets is a safe net, and remains as the categorical product in the subcategory of safe nets. As before its behaviour is expressed in terms of the behaviour of the components. However the statement is slightly different when using set instead of multiset notation.

Proposition. Let $N_0 \times N_1$, $\Pi_0 = (\pi_0, \rho_0)$ and $\Pi_1 = (\pi_1, \rho_1)$ be a product of safe nets N_0, N_1 . Then M is a reachable marking of $N_0 \times N_1$ and $M \xrightarrow{A} M'$ iff

$$\begin{aligned} &\rho_0^{\omega}M \text{ is a reachable marking of } N_0 \text{ and} \\ &\rho_0^{\omega}M \xrightarrow{\pi_0^{\omega}A} \rho_0^{\omega}M' \ \& \ \forall e, e' \in A \forall e_0 \in E_0. e\pi_0e_0 \ \& \ e'\pi_0e_0 \Rightarrow e = e' \text{ and} \\ &\rho_1^{\omega}M \text{ is a reachable marking of } N_1 \text{ and} \\ &\rho_1^{\omega}M \xrightarrow{\pi_1^{\omega}A} \rho_1^{\omega}M' \ \& \ \forall e, e' \in A \forall e_1 \in E_1. e\pi_1e_1 \ \& \ e'\pi_1e_1 \Rightarrow e = e'. \end{aligned}$$

Proof. The proof follows from the more general theorem of the last section. For the “only if” direction of the proof, use the fact that *e.g.* $\forall e, e' \in A \forall e_0 \in E_0. e\pi_0e_0 \ \& \ e'\pi_0e_0 \Rightarrow e = e'$ implies $\pi_0^{\omega}A = \pi_0A$. ■

Theorem. The product of safe nets is safe and is a product in the category of safe nets.

Proof. By the theorem above the product of safe nets is safe, and the categorical properties follow from the corresponding theorem of the last section. ■

The operation of restriction clearly preserves safeness. Consequently, the construction of the synchronous product of the last section always produces a safe net from safe components.

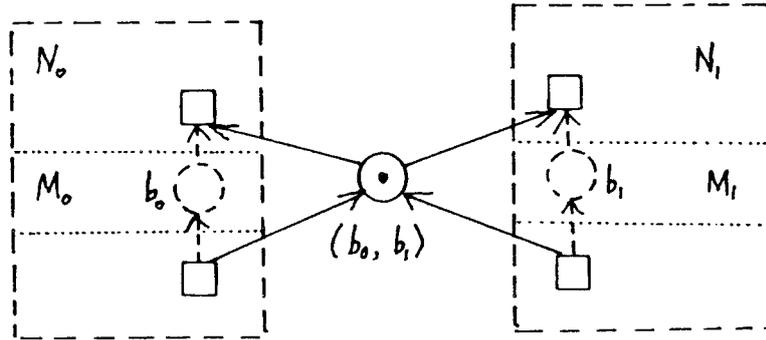
Theorem. The synchronous product of safe nets is safe and with its projections is the product in the category of safe nets with synchronous morphisms.

Proof. Obvious ■

Unlike the larger category of all nets the subcategory of safe nets does have a coproduct.

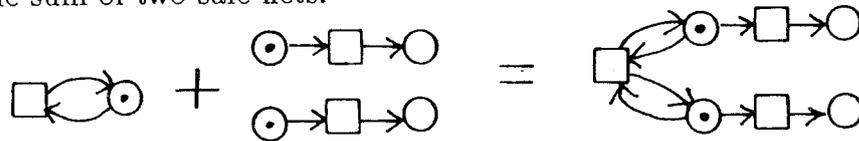
Let $N_0 = (B_0, E_0, F_0, M_0)$ and $N_1 = (B_1, E_1, F_1, M_1)$ be safe nets. The two nets N_0 and N_1 are laid side by side and then a little surgery is performed on their initial markings. For each pair of conditions b_0 in the initial marking of N_0 and b_1 in the initial marking of N_1 a new condition (b_0, b_1) is created and made to have the same pre and post events as b_0 and b_1 together. The conditions in the original initial markings are removed and replaced by a new initial marking consisting of these newly created conditions.

The sum of two nets:



Notice a condition in the initial marking of one component is generally represented by more than one condition in the initial marking of the sum.

Example. The sum of two safe nets:



The set of events of the sum E is the disjoint union $E_0 \uplus E_1$ of the events of the components. There are the obvious injections $in_0 : E_0 \rightarrow E$ and $in_1 : E_1 \rightarrow E$ on events. The initial marking of the sum can be represented by

$$M = M_0 \times M_1,$$

and its set of conditions by

$$B = \{(b_0, *) \mid b_0 \in B_0 \setminus M_0\} \cup \{(*, b_1) \mid b_1 \in B_1 \setminus M_1\} \cup M.$$

Then there are the obvious injection relations ι_0 and ι_1 where

$$\begin{aligned} b_0 \iota_0 b &\Leftrightarrow \exists b_1 \in B_1 \cup \{*\}. b = (b_0, b_1), \\ b_1 \iota_1 b &\Leftrightarrow \exists b_0 \in B_0 \cup \{*\}. b = (b_0, b_1). \end{aligned}$$

Thus the injection relations on conditions are opposite to the obvious partial functions taking a condition in B to its first or second component. Together the injections on

events and the injections on conditions provide injection morphisms $I_0 = (in_0, \iota_0)$ and $I_1 = (in_1, \iota_1)$ from the component nets to their sum. Using the injections we can express the behaviour of the sum in terms of the behaviour of its components (using multiset notation).

Theorem. *Let $N_0 + N_1$ be the sum of safe nets with injections $I_0 = (in_0, \iota_0)$ and $I_1 = (in_1, \iota_1)$. Then X is a reachable marking of $N_0 + N_1$ and $X \xrightarrow{A} X'$ iff*

\exists reachable marking X_0, A_0, X'_0 .

$$N_0 : X_0 \xrightarrow{A_0} X'_0 \quad \& \quad A = in_0 A_0 \quad \& \quad X = \iota_0 X_0 \quad \& \quad X' = \iota_0 X'_0$$

or

\exists reachable marking X_1, A_1, X'_1 .

$$N_1 : X_1 \xrightarrow{A_1} X'_1 \quad \& \quad A = in_1 A_1 \quad \& \quad X = \iota_1 X_1 \quad \& \quad X' = \iota_1 X'_1.$$

Proof. Let $N_0 = (B_0, E_0, F_0, M_0)$ and $N_1 = (B_1, E_1, F_1, M_1)$ be safe nets. It is obvious that the injections are morphisms $I_k = (in_k, \iota_k) : N_k \rightarrow N_0 + N_1$ for $k = 0, 1$. From this the “if” part of the proof follows directly.

To show the converse (“only if”), we first show the following fact:

If X_0 is a reachable marking of N_0 and $N_0 + N_1 : \iota_0 X_0 \xrightarrow{A} X'$ then either

$$A = in_0 A_0 \quad \& \quad X' = \iota_0 X'_0 \quad \& \quad N_0 : X_0 \xrightarrow{A_0} X'_0 \tag{1}$$

for some subset A_0 of events and marking X'_0 of N_0 or

$$A = in_1 A_1 \quad \& \quad X_0 = M_0 \quad \& \quad X' = \iota_1 X'_1 \quad \& \quad N_1 : M_1 \xrightarrow{A_1} X'_1 \tag{2}$$

for some subset of events A_1 and marking X'_1 of N_1 .

To show this assume X_0 is a reachable marking of N_0 and $N_0 + N_1 : \iota_0 X_0 \xrightarrow{A} X'$. There are two cases to consider: when A contains the image of an event of N_1 and when it does not.

Suppose first that A contains the event $in_1 e_1$ for some $e_1 \in E_1$. Then, as A is conflict-free, A must have the form $A = in_1 A_1$ for some events $A_1 \subseteq E_1$. As in particular e_1 has concession at $\iota_0 X_0$ we see

$$M_0 \times \bullet e_1 = \bullet in_1 e_1 \subseteq \iota_0 X_0.$$

Hence $M_0 \subseteq X_0$. Because N_0 is safe $M_0 = X_0$ —otherwise a repeat of the behaviour which led to the reachable marking X_0 will cause the conditions in $X_0 \setminus M_0$ to hold with multiplicity greater than 1. Thus $\iota_0 X_0 = M_0 \times M_1$, the initial marking of the sum. Take $X'_1 = M_1 - \bullet A_1 + A_1 \bullet$. Then by linearity, and as the injection (in_1, ι_1) is a morphism, we obtain

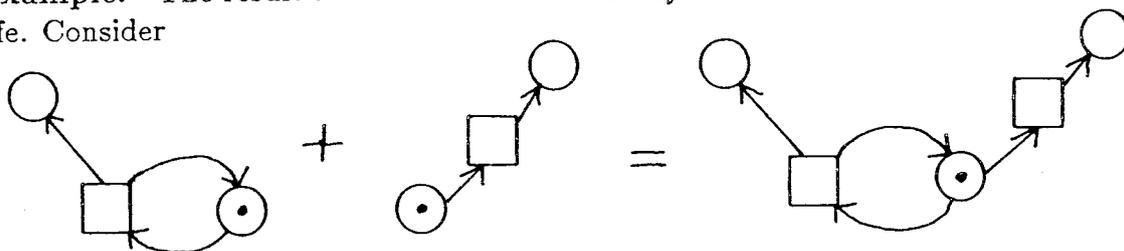
$$X' = M_0 \times M_1 - \bullet A + A \bullet = \iota_1 (M_1 - \bullet A_1 + A_1 \bullet) = \iota_1 X'_1.$$

Thus in this case (2) holds.

Now suppose $A \cap in_1 E = \emptyset$. Then A has the form $A = in_0 A_0$ for some $A_0 \subseteq E_0$. Take $X'_0 = X_0 - \bullet A_0 + A_0 \bullet$. Then using the fact that (in_0, ι_0) is a morphism and the linearity of ι_0 one obtains $X' = \iota_0 X'_0$. Hence in this case we satisfy (1) above.

The analogous result holds for N_1 in place of N_0 . Using these two results we argue by induction on the number of transitions to the reachable marking X of $N_0 + N_1$, to complete the proof of the theorem. ■

Example. The result above does not necessarily hold for the sum of nets which are not safe. Consider

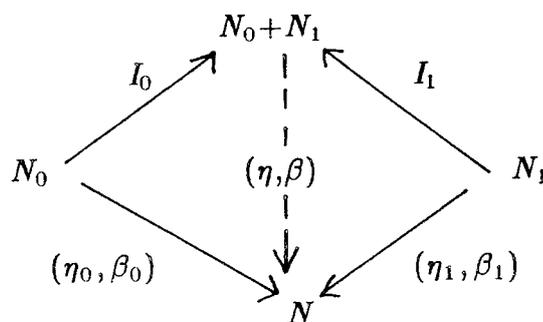


Those familiar with Milner's work may be a little bothered by our definition of sum. For the $+$ of CCS and SCCS once a component has been selected nondeterministically the choice is stuck to, which is not true in general for our sum—consider the example above. However our construction will agree with Milner's on those safe nets for which $\forall b \in M_0 \nexists e. eFb$ i.e. no event leads into the initial marking. If one were to systematically give a net semantics to the languages like CCS, SCCS and CSP, all the nets constructed would be safe and satisfy this property.

This time the sum construction is the coproduct in the category of safe Petri nets.

Theorem. The sum $N_0 + N_1$ with injections I_0 and I_1 is a coproduct in the category of safe Petri nets with morphisms on nets and also in the category of safe nets with synchronous morphisms.

Proof. By the above theorem the sum of safe nets is safe and as we observed in its proof the injections are morphisms. We use the notation introduced in the definition of sum.



Consider the above diagram in the category of safe Petri nets with morphisms on nets. Assume $N = (B, E, F, M)$.

Take

$$\eta(e) = \begin{cases} \eta_0(e_0) & \text{if } e = in_0(e_0), \\ \eta_1(e_0) & \text{if } e = in_1(e_0) \end{cases}$$

for e an event of $N_0 + N_1$. Note η is the unique function such that $\eta in_0 = \eta_0$ and $\eta in_1 = \eta_1$.

Take β to be the relation between conditions of $N_0 + N_1$ and conditions of N given by

$$b\beta c \Leftrightarrow (\forall b_0. b_0 \iota_0 b \Rightarrow b_0 \beta_0 c) \ \& \ (\forall b_1. b_1 \iota_1 b \Rightarrow b_1 \beta_1 c),$$

where b is a condition of $N_0 + N_1$, and c is condition of N . Then considering considering the three different kinds of condition in the sum, the multirelation composition $\beta \iota_0$ is a relation with $\beta \iota_0 = \beta_0$. For the same reasons, $\beta \iota_1 = \beta_1$. Indeed, again considering the nature of conditions in the sum, β is the unique relation such that $\beta \iota_0 = \beta_0$ and $\beta \iota_1 = \beta_1$.

Hence, using the properties of morphisms, we see

$$\beta(M_0 \times M_1) = \beta(\iota_0 M_0) = \beta_0(M_0) = M,$$

the initial marking of N , and that for an event $e = in_0 e_0$ of the sum

$$\beta(\bullet e) = \beta(\bullet in_0 e_0) = \beta(\iota_0 \bullet e_0) = \beta_0(\bullet e_0) = \bullet(\eta_0 e_0) = \bullet(\eta e).$$

In the same manner we can show $\beta(\bullet e) = \bullet(\eta e)$ and $\beta(e \bullet) = (\eta e) \bullet$ for any event e in the sum.

Thus (η, β) is a morphism, and by our earlier remarks it is the unique morphism which makes the diagram commute. This shows that the sum with injections is a coproduct in the category of safe nets with net morphisms. A similar proof goes through in the subcategory with synchronous morphisms; simply note the injections are synchronous and that in this case η_0 and η_1 will be total so η will be total too. ■

6. A look at coloured nets.

Kurt Jensen introduced coloured nets in [J1] and higher level nets in [J2]. The only difference is that higher level nets are a little more general in that their incidence relation is split into a positive and negative part so they can handle side conditions. The two kinds of net are so similar we shall call both coloured nets. Like predicate transition nets before them they were designed as an abbreviated form of Petri net description in a sense we shall make precise here. The relation nets of [R] are a special kind of coloured net with an extra capacity function associated with the places. Here we see how, formally at least, coloured nets are an obvious generalisation of Petri nets.

The idea of coloured nets is best explained through the use of products of spaces μC , an idea familiar from products of vector spaces and of modules.

Definition. Let $C(p)$ be a set for each $p \in P$. Define the product of multisets

$$\prod_{p \in P} C(p) = \mu\{(p, c) \mid p \in P \ \& \ c \in C(p)\}.$$

Proposition. The set $\prod_{p \in P} C(p)$ is in 1-1 correspondence with the set

$$F = \{f : P \rightarrow \bigcup_{p \in P} \mu C(p) \mid f(p) \in \mu C(p)\}$$

under the maps θ, ϕ given by

$$\theta : \prod_{p \in P} C(p) \rightarrow F \text{ where } ((\theta g)p)_c = g_{p,c}$$

and

$$\phi : F \rightarrow \prod_{p \in P} C(p) \text{ where } (\phi f)_{p,c} = (fp)_c.$$

Thus the product of spaces $\prod_{p \in P} C(p)$ which was defined to be the space of multisets of the set $\{(p, c) \mid c \in C(p)\}$ can be identified with the set consisting of P -tuples of multisets of colours.

It is useful to describe coloured nets as being built out of *places* and *transitions* rather than conditions and events because they have a higher level nature, standing for sets of conditions and sets of events, respectively. In a coloured net each place is associated with a set of *colours*. You can think of each such colour of a place as standing for a condition of the form used in Petri nets, so a place stands for a set of conditions, one for each of its colours. Thus naturally, in a coloured net instead of a marking associating each condition with a non-negative integer each place is associated with a multiset of colours. In coloured nets, you can, if you like, think of the tokens as being coloured. A transition too is associated with a set of colours. It really stands for a set of events one for each of its colours. Thinking

of it this way it is natural to allow a transition to fire with various multiplicities for each of its colours *i.e.* to allow it to fire with value a multiset of its colours. Then in analogy with Petri nets, when a transition fires in such a way it consumes a certain number of tokens of various colours at various places and similarly produces a distribution of tokens of various colours at various places.

Thus coloured nets are like Petri nets but with the difference that now we must account for the fact that places stand for sets of conditions and transitions stand for sets of events.

Definition. A *coloured net* is a structure $(P, T, C, \bullet(\), (\)^\bullet, M_0)$ where

- P is a non-null set of *places*,
- T is a disjoint set of *transitions*,
- C is a *colour function* associating each place p with a non-null set $C(p)$ and each transition t with a non-null set $C(t)$,
- $\bullet(\), (\)^\bullet : \{(t, c) \mid t \in T \ \& \ c \in C(t)\} \rightarrow_\mu \{(p, c) \mid p \in P \ \& \ c \in C(p)\}$
- and
- $M_0 \in \prod_{p \in P} C(p)$, the *initial marking*

which satisfy the restrictions:

- (i) $M_0 \neq \underline{0}$ and $(\bullet A = \underline{0} \text{ or } A^\bullet = \underline{0}) \Rightarrow A = \underline{0}$ and
- (ii) $M_b \neq 0$ or $(\exists e \in E. F_{e,b} \neq 0)$ or $(\exists e \in E. F_{b,e} \neq 0)$.

Now this is not quite the way that K. Jensen defined coloured or high level nets. Some differences are trivial, like the fact that we insist the initial marking is non-null and that there are no isolated conditions. The main difference in presentation is that Jensen describes the multirelations $\bullet(\), (\)^\bullet$ by means of the matrices

$$\begin{aligned} I_{p,t}^- &: C(t) \rightarrow_\mu C(p), \\ I_{p,t}^+ &: C(t) \rightarrow_\mu C(p) \end{aligned}$$

on the p and t coordinates which clearly determine and are determined by $\bullet(\)$ and $()^\bullet$ by linearity.

It is now a simple matter to define the behaviour of coloured nets. (We use the identification mentioned above.) Just like Petri nets we define

$$M \xrightarrow{A} M' \Leftrightarrow \bullet A \leq M \ \& \ M' = M - \bullet A + A^\bullet$$

where $M, M' \in \prod_{p \in P} C(p)$ are markings and $A \in \prod_{t \in T} C(t)$ is a finite multiset, as firing value.

We said coloured nets were an abbreviated way of describing Petri nets, and it is easy to see how because a coloured net is so closely associated with a 2-sorted algebra over multisets.

Proposition. A coloured net $(P, T, C, \bullet(\), (\)^\bullet, M_0)$ determines a Petri net with conditions $B = \{(p, c) \mid p \in P \ \& \ c \in C(p)\}$, events $E = \{(t, c) \mid t \in T \ \& \ c \in C(t)\}$, initial marking M_0 and multirelations $\bullet(\), (\)^\bullet : E \rightarrow_\mu B$.

Thus a coloured net can be viewed as arising from a Petri net simply by regrouping the elements of the bases of the space of multisets of conditions and events, and it is a simple matter to recover the underlying Petri net by going back to the bases. Of course many different coloured nets have the same underlying Petri net because there are many different ways in which the bases can be regrouped.

What should we take as the definition of morphism on coloured nets? It is desirable that a morphism between coloured nets should induce a morphism between the underlying Petri nets. A very general candidate is the following.

Definition. (Tentative) A morphism between coloured nets is a morphism between their underlying Petri nets.

However it is not so clear whether or not morphisms should respect the extra colour structure C on coloured nets. I leave this open—my intuition about colours is not sharp. The above definition would be appropriate if coloured nets were no more than representations of Petri nets

I have not looked very closely at the many other generalisations of Petri nets; maybe many of their definitions too are obtained as slight variants of that of the original Petri nets, got by varying the sorts of the associated algebra.

7. Net invariants.

The use of the technique of invariants to obtain properties of nets was discovered by Kurt Lautenbach. Here we examine the sense in which finitary homomorphisms and morphisms preserve invariants of nets. Recall the definition of condition invariant of a net (called an S -invariant in [R]). We add some further restrictions to the usual definition in order to cope with infinite nets, so we can make sense of invariants which form infinite matrices.

Definition. Let $N = (B, E, F, M_0)$ be a Petri net. A condition invariant of N is a matrix $I : B \rightarrow_\nu \mathbf{1}$ such that

- (i) $I(M_0)$ is defined and $I(\bullet e)$ and $I(e^\bullet)$ are defined for all events e , and
- (ii) $I(M)$ is defined and $I(M) = I(M_0)$ for every reachable marking M .

Write $\text{Inv}N$ for the set of invariants of N .

Note that the condition (i) is trivially true and, in (ii), $I(M)$ is always defined for finite Petri nets.

Invariants can be characterised in a more local way when the Petri net satisfies the restriction that every event can occur sometime, as expressed in the following proposition. Its proof is easy and well-known, see *e.g.* [R].

Proposition. *Assume N is a net in which every event can occur, i.e. for all events e there is some reachable marking M for which $\bullet e \leq M$. Then $I \in \text{Inv}N$ iff $I(M_0)$ is defined and $I(\bullet e)$ and $I(e\bullet)$ are both defined and equal for all events e .*

Also well-known, and easy to show, is the fact that invariants form a \mathbf{Z} -module. Recall a \mathbf{Z} -module M is an Abelian group with composition $+$ and identity 0 , together with an operation, called *scalar product*, $\mathbf{Z} \times M \rightarrow M$, which satisfies

- (i) $n(u + v) = nu + nv$,
- (ii) $(m + n)v = mv + nv$,
- (iii) $(mn)v = m(nv)$,
- (iv) $1v = v$

for all $m, n \in \mathbf{Z}$, $u, v \in M$.

Recall too that a *morphism* between \mathbf{Z} -modules M and N is a function $\alpha : M \rightarrow N$ which is *linear* in the sense that $\alpha(nv) = n(\alpha v)$ and $\alpha(u+v) = \alpha u + \alpha v$ for all $u, v \in M$ and $n \in \mathbf{Z}$. Note \mathbf{Z} -modules correspond to Abelian groups and their morphisms to homomorphisms on Abelian groups.

Proposition. *Let N be a net. Then $\text{Inv}N$ form a \mathbf{Z} -module under matrix addition and scalar multiplication.*

Let us see what the relation is between the categories of Petri nets with finitary homomorphisms and morphisms and the category of \mathbf{Z} -modules. Assume $(\eta, \beta) : N_0 \rightarrow N_1$ be a morphism of nets. The natural way to form the image of an invariant I of N_0 would seem to be by taking $(\beta(I^{op}))^{op}$. However it is easy to produce examples where the image $(\beta(I^{op}))^{op}$ of an invariant I of N_0 is not an invariant of N_1 . Invariants are not preserved in the direction of homomorphisms but rather in the opposite direction by the *dual map* β^* given by $\beta^*(I) = I\beta$. The image $\beta^*(I)$ of an invariant I of N_1 is an invariant of N_0 , and this is true not just for morphisms but for finitary homomorphisms as well. Consequently there is a contravariant functor from the category of nets with net finitary homomorphisms to the category of \mathbf{Z} -modules, which cuts down to a functor from the category of nets with net morphisms. (It is *contravariant* because it switches the direction of the arrows.)

Lemma. *Let $(\eta, \beta) : N_0 \rightarrow N_1$ be a finitary homomorphism of nets. Then $I \in \text{Inv}N_1$ implies $I\beta$ is defined and $I\beta \in \text{Inv}N_0$.*

Proof. Let $N_0 = (B_0, E_0, F_0, M_0)$ and $N_1 = (B_1, E_1, F_1, M_1)$. Suppose $I \in \text{Inv}N_1$.

We must first show that $I\beta$ is defined. Let $b \in B_0$. Either

$$(M_0)_b > 0 \text{ or } (\bullet e)_b > 0 \text{ or } (e\bullet)_b > 0.$$

If $(M_0)_b > 0$ then

$$\{y \mid I_y \cdot \beta_{y,b} \neq 0\} \subseteq \{y \mid I_y \cdot (M_1)_y \neq 0\}$$

which is finite as $I(M_1)$ is defined. If $(\bullet e)_b > 0$ then

$$\{y \mid I_y \cdot \beta_{y,b} \neq 0\} \subseteq \{y \mid I_y \cdot (\bullet \eta e)_y \neq 0\}$$

which is finite as ηe is a finite multiset. The remaining case $(e^\bullet)_b > 0$ is similar, so in all cases $\{y \mid I_y \cdot \beta_{y,b} \neq 0\}$ is finite. Consequently $I\beta$ is defined and equals the finite sum $\sum_{y \in B_1} I_y \cdot \beta_{y,b}$.

We need to establish that $(I\beta)M_0$, $(I\beta)(\bullet e)$ and $(I\beta)(e^\bullet)$ are defined for all $e \in E_0$. In fact let M be any reachable marking of N_0 , not just M_0 . Then we see that

$$\{(x, y) \mid I_y \cdot \beta_{y,x} \cdot M_x \neq 0\} \subseteq \{(x, y) \mid I_y \cdot (\beta M)_y \neq 0 \ \& \ \beta_{y,x} \cdot M_x \neq 0\}$$

which is finite as $\{y \mid I_y \cdot (\beta M)_y\}$ is finite (because $I(\beta M)$ is defined) and the sets $\{x \mid \beta_{y,x} \cdot M_x \neq 0\}$ are finite for all $y \in B_1$. Therefore $(I\beta)M$ is defined and equals the finite sum $\sum_{(x,y) \in B_0 \times B_1} I_y \cdot \beta_{y,x} \cdot M_x \neq 0$. Similarly $(I\beta)(\bullet e)$ and $(I\beta)(e^\bullet)$ are defined for all $e \in E_0$.

Let M be a reachable marking of N_0 . Then $(I\beta)M$ is defined and equals $I(\beta M) = I(M_1) = I(\beta M_0) = (I\beta)(M_0)$. Thus $I\beta$ is an invariant of N_0 . ■

Remark. Note how simple the proof is when the Petri nets are finite; then all the verification of definedness is unnecessary.

Theorem. *There is a contravariant functor from the category of Petri nets with finitary homomorphisms to the category of \mathbf{Z} -modules with linear maps; on objects it acts as $N \mapsto \text{Inv}N$, and it takes a finitary homomorphism $(\eta, \beta) : N_0 \rightarrow N_1$ on nets to the linear map $\beta^* : \text{Inv}N_1 \rightarrow \text{Inv}N_0$ on \mathbf{Z} -modules given by $\beta^*(I) = I\beta$.*

Proof. Because of the lemma above, the proof is now a simple matter of checking the functor laws hold. Clearly if $(1_E, 1_B)$ is the identity homomorphism on a net N with conditions B then $(1_B)^* : \text{Inv}N \rightarrow \text{Inv}N$ is the identity on $\text{Inv}N$. And, if $(\eta_0, \beta_0) : N_0 \rightarrow N_1$ and $(\eta_1, \beta_1) : N_1 \rightarrow N_2$ are finitary homomorphisms on nets N_0, N_1, N_2 then $(\beta_1 \beta_0)^* = \beta_0^* \beta_1^* : \text{Inv}N_2 \rightarrow \text{Inv}N_0$. ■

This shows the general relationship between nets and their spaces of invariants. However much more can be done with the interplay between homomorphisms and invariants. For example, it is easy to show that the space of invariants of the product of two nets is just the product space of the spaces of invariants of the two nets, and that $I \in \text{Inv}N_0 + N_1$ iff $I\iota_0 \in \text{Inv}N_0$ and $I\iota_1 \in \text{Inv}N_1$ where ι_0 and ι_1 are the condition parts of the injection functions for the coproduct of nets. Following this kind of idea, Mogens Nielsen and I have produced a little calculus for building up invariants of larger nets using constructions like product, synchronous product, restriction, sum, and a loop construct not mentioned here, in terms of their component nets [NW]. The calculus and its proof of completeness make essential use of morphisms and homomorphisms on Petri nets.

8. Formalising the relation of Petri nets with other models.

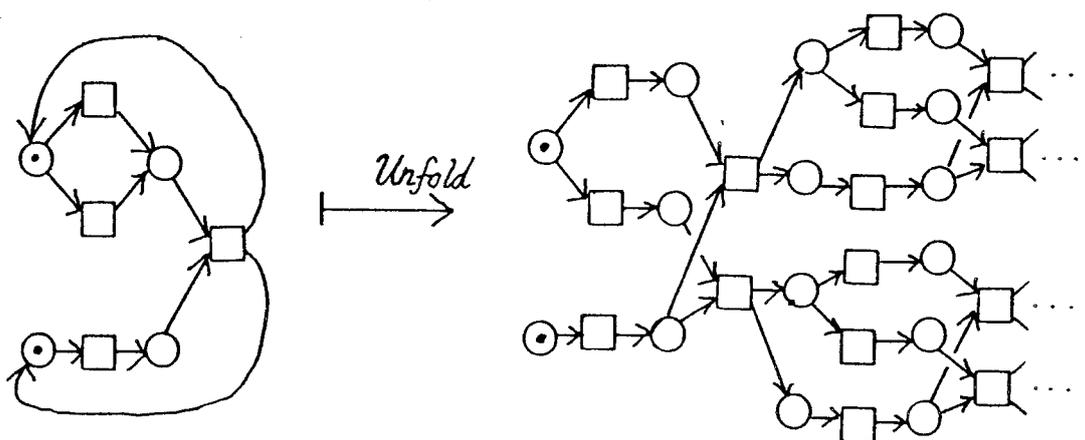
We indicate briefly how other models can be embedded in the category of nets. More details can be found in [W4].

Many other models of computation, occurrence nets, event structures, synchronisation trees and transition systems, can be made into categories. In them too parallel compositions are obtained by restricting the product, and the sum of processes will be modelled as a coproduct. Often the categories can be related by coreflections, pairs of functors in a kind of “embedding” adjunction, passing back and forth, so that the categorical constructions are preserved as well. I present one example, the relation between safe Petri nets and occurrence nets, to give the idea.

I use “occurrence net” in the sense in which it was initially used in [NPW1, NPW, W]; its later use in [Br] to mean a more restricted class of net, what were formerly called “causal nets”, is unfortunate.

Nets are rather complex objects with an intricate behaviour which so far has been expressed in a dynamic way. We would like to know when two nets have essentially the same behaviour and [NPW1, NPW, W] proposed a “static” representation of their behaviour as a certain kind of net, a net of condition and event occurrences. This generalised the familiar unfolding of a state-transition system to a tree. The theorems we shall mention only work for the the class of safe nets, though something similar should go through for nets in general. The occurrence net we associate with a safe net will be built-up essentially by unfolding the net to its occurrences. This unfolding is a canonical representative of the behaviour of the original net. Rather than give any formal definitions the main ideas can be seen in the following example which illustrates a safe net together with its occurrence net unfolding—see the references above for details.

The occurrence net unfolding of a safe net:



Think of the unfolding operation as taking a model of a computation as a Petri net to a model as an occurrence net. Occurrence nets have product and sum constructions given

by the categorical product and coproduct. Clearly we would like products and sums to be preserved and this almost follows from an adjunction relation between them. Occurrence nets form a subcategory of safe Petri nets so there is an inclusion functor from occurrence nets to safe nets. It has the unfolding operation, extended to a functor, as its right adjoint, and the fact that unfolding twice is the same as unfolding once makes the adjunction a coreflection. Because right adjoints preserve products we know for abstract reasons (once we've shown unfolding really is a right adjoint) that unfolding preserves product, and with labelled nets this gives us that parallel compositions are preserved. Right adjoints preserve limits like products but not necessarily colimits like coproduct and it is easy to find examples where the unfolding of the sum of two nets is not isomorphic to the sum of their unfoldings (such examples were in mind when we discussed the sum of Petri nets in section 4). However for a wide subclass of nets sum is preserved too.

The same general scheme is true for other models as well. For example, there is an interleaving, or serialising, functor from nets to a category of trees, which can be obtained as is to be expected from the synchronous product $(- \otimes \Omega)$. The product in the category of trees is that expected from Milner's expansion theorem and the coreflection provides a bridge between the Petri net model of computation and the interleaving models of Robin Milner, C.A.R. Hoare and others [M1, M2, H, HBR]. The paper [W2] spells out the structure of the appropriate categories of trees and transition systems.

9. Conclusion.

A case has been made for a new concept of morphism on Petri nets. The new definition supports a compositional approach to describe and reason about nets, it ties in nicely with the view of nets as algebras which underpins the use of linear algebra in net theory, and provides a formal translation with other models. Here there are some loose ends to tidy up such as the categorical relation between general Petri nets and occurrence nets. Then there is the relationship between invariants of compound nets and those of their components, studied in [NW]. The way is set to analyse the way in which properties are preserved by finitary homomorphisms, morphisms, more restricted kinds of morphism, or their opposite morphisms in the dual category as is the case for invariants. More speculatively, the general view proposed here may offer some new leads to future directions in net theory, perhaps by choosing some radically different structures for the sorts in formulation of another kind of net as a form of algebra, for example to model probabilistic computation. There are certainly some connections with the work of Main and Benson [MB], though it is not clear how fruitful they are.

I think the tangible results here stand up rather well against the old definition of morphism in [Br]; the definition there does not even respect the behaviour of nets. It also generalises the definition in [W3] and the morphisms of processes in [GR]. Of course no one could quarrel with the uses proposed for the old net morphisms. What is far from clear is how the definition there meets any nontrivial formal requirement. I do not claim the morphisms on Petri nets advocated here do everything one might wish of morphisms.

They do not for instance enable you to collapse a closed subset to a single compound event, one task proposed for morphisms in [Br]. Maybe the definition here can be extended to do this too—I don't know. A complete treatment would carefully relate our definition to the old definition in [Br]. Perhaps someone more committed to the old definition would like to try.

Acknowledgements

I am grateful for lengthy discussions with Mogens Nielsen. Talks with Ursula Goltz and Wolfgang Reisig were helpful.

Appendix: Vectors, matrices, multisets and multirelations.

Vectors and multisets

We first define *vectors* of integers and operations on them.

Let X be a set. A *vector* over X is a function from X to \mathbf{Z} , the positive and negative integers. Write f_x for $f(x)$, the x -component of f . Write νX for the set of vectors over X . Call νX the *space of vectors* over X , and X its *basis*. A vector is *finite* if all but finitely many components are 0.

We use $\mathbf{1}$ to represent a set with a single element; so vectors over $\mathbf{1}$ are isomorphic to \mathbf{Z} .

A *multiset* over a set X is a vector over X in which all the components are nonnegative, and so is a vector $f : X \rightarrow \mathbf{N}$, associating a natural number, possibly zero, with each $x \in X$. Write μX for the set of multisets over X . Call μX the *space of multisets* over X and X its *basis*.

Let $n \in \mathbf{N}$. Define \underline{n} of X to be the multiset $\underline{n} : x \mapsto n$ for all $x \in X$. In particular, the *null multiset* $\underline{0}$ of X is the function $\underline{0} : x \mapsto 0$ for any $x \in X$.

Let $x \in X$. Define the *singleton* multiset \hat{x} to be the function

$$\hat{x} : y \mapsto \begin{cases} 1 & \text{if } x = y \\ 0 & \text{otherwise.} \end{cases}$$

Say a multiset is a singleton if it has this form. It is convenient to write x for \hat{x} .

By convention, we shall identify subsets of X with those multisets of $f \in \mu X$ such that $f \leq \underline{1}$.

Operations on vectors and multisets

Useful operations and relations on vectors are induced pointwise by operations and relations on integers. These generally restrict to give operations and relations on multisets.

Let $f, g \in \nu X$. Define

$$\begin{aligned} (f + g)_x &= f_x + g_x, \\ (f - g)_x &= f_x - g_x \end{aligned}$$

for $x \in X$. Define

$$f \leq g \Leftrightarrow \forall x \in X. f_x \leq g_x.$$

Clearly multisets are closed under $+$ but not necessarily under $-$. Of course, if $g \leq f$, for two multisets f and g , then their difference $f - g$ is a multiset.

Let $n \in \mathbf{Z}$ and $f \in \nu X$. Define the *scalar multiplication* nf to be the vector given by $(nf)_x = nf_x$ for $x \in X$.

Let f, g be vectors over X . Define their *inner product* $f \cdot g$ to be

$$f \cdot g = \sum_{x \in X} f_x \cdot g_x$$

when the set $\{x \in X \mid f_x \cdot g_x \neq 0\}$ is finite, and to be undefined otherwise.

Matrices and multirelations

Let X and Y be sets. A \mathbf{Z} -matrix from X to Y is a vector $\alpha : Y \times X \rightarrow \mathbf{Z}$ which associates an integer, $\alpha_{y,x}$ with each pair (y, x) , $y \in Y, x \in X$. We write $\alpha : X \rightarrow_\nu Y$, and sometimes $(\alpha_{x,y})_{x \in X, y \in Y}$, to mean α is a \mathbf{Z} -matrix from X to Y . Because matrices are vectors we can *e.g.* form sums and scalar products of matrices.

A *multirelation* from X to Y is a matrix α from X to Y in which all entries $\alpha_{x,y}$ are nonnegative. So a multirelation from X to Y is a function $\alpha : X \times Y \rightarrow \mathbf{N}$. We write $\alpha : X \rightarrow_\mu Y$ to mean α is a multirelation from X to Y .

By convention, we shall identify the *relations* between a set X and a set Y with those multirelations $\theta : X \rightarrow_\mu Y$ for which $\theta_{y,x} \leq 1$. In particular, we shall identify *functions* and *partial functions* with their extensions to multirelations. We shall use standard notation for relations and functions *e.g.* writing xRy when x and y are in relation R .

Given a matrix $\theta : X \rightarrow_\nu Y$ it is sometimes useful to consider a matrix $\theta^{op} : Y \rightarrow_\nu X$ in the opposite direction specified as the matrix $(\theta_{x,y}^{op})_{x \in X, y \in Y}$ which is the *transpose* of θ , so $\theta_{x,y}^{op} = \theta_{y,x}$ for all $x \in X, y \in Y$. Clearly, if $\theta : X \rightarrow_\mu Y$, a multirelation, then so is θ^{op} a multirelation $\theta^{op} : Y \rightarrow_\mu X$. For a relation R the notation R^{op} represents the converse or opposite relation $xR^{op}y \Leftrightarrow_{def} yRx$.

Infinite sums of nonnegative integers

The definition of inner product illustrates a problem we have to face because we do not insist vectors and multisets are only over finite sets. We quickly run into infinite sums of integers. This obliges us to consider infinite sums of integers and how to deal with the fact that such sums do not converge in general. Fortunately for sums of nonnegative integers, at least, the treatment of nonconvergence is simple. We extend the nonnegative integers \mathbf{N} by the new element ∞ , so ∞ represents nonconvergence. Write

$$\mathbf{N}^\infty = \mathbf{N} \cup \{\infty\}.$$

Extend addition and multiplication on integers to the element ∞ by defining

$$\infty + n = n + \infty = \infty,$$

for all $n \in \mathbf{N}^\infty$, and

$$\infty \cdot n = n \cdot \infty = \infty,$$

for all $n \in \mathbf{N}^\infty \setminus \{0\}$, but where

$$\infty \cdot 0 = 0 \cdot \infty = 0.$$

More precisely the extended operations $+$ and \cdot are the smallest operations which behave like addition and multiplication on the integers and satisfy the above laws involving ∞ .

Now we can define sums of arbitrary subsets of \mathbf{N}^∞ in the following way. Let $\{f_i \mid i \in I\}$ be an *indexed set* of \mathbf{N}^∞ . Say such an indexed set is *finite* precisely in the case where each $f_i \neq \infty$, for $i \in I$, and the set $\{i \in I \mid f_i \neq 0\}$ is finite. In the case where $\{f_i \mid i \in I\}$ is finite in this sense define $\Sigma_{i \in I} f_i$ to be the usual sum and otherwise to be ∞ . This notation generalises that for finite sums of integers. It is easy to check that rules hold for generalised sums, such as *partition associativity*, a name used in [AM1] to mean if $\{I_j \mid j \in J\}$ is a partition of the set I then

$$\Sigma_{i \in I} f_i = \Sigma_{j \in J} (\Sigma_{i \in I_j} f_i).$$

Notice this rule, and other natural rules like distributivity of multiplication over sum, do *not* hold for infinite sums of positive *and* negative integers \mathbf{Z} in general; this is why we choose a different approach for sums of infinite subsets of \mathbf{Z} .

∞ -Multisets and ∞ -multirelations

We generalise multisets and multirelations so they can take the value ∞ . Let X be a set. A ∞ -*multiset* over X is a function $f : X \rightarrow \mathbf{N}^\infty$, which associates f_x , a nonnegative integer or ∞ , with each $x \in X$. Let $\mu^\infty X$ denote the set of ∞ -multisets over X . Let X and Y be sets. A ∞ -*multirelation* from X to Y is a \mathbf{N}^∞ -matrix $\alpha : Y \times X \rightarrow \mathbf{N}^\infty$. We write $\alpha : X \rightarrow_\mu^\infty Y$ to mean α is a ∞ -multirelation from X to Y .

Through the introduction of ∞ we can avoid niggling considerations like whether the application of a multirelation to a multiset exists or not.

Multirelation composition and application

Let $f \in \mu^\infty X$. Let $\alpha : X \rightarrow_\mu^\infty Y$. Define the *application* of α to f to be the ∞ -multiset $\alpha f \in \mu^\infty Y$ which satisfies

$$(\alpha f)_y = \Sigma_{x \in X} \alpha_{y,x} \cdot f_x$$

where the sums may be infinite.

Let $\alpha : X \rightarrow_\mu^\infty Y$ and $\beta : Y \rightarrow_\mu^\infty Z$. Define their *composition* $\beta \circ \alpha$ (often written as just $\beta\alpha$) to be the matrix $\beta \circ \alpha : X \rightarrow_\mu^\infty Z$ given by

$$(\beta \circ \alpha)_{z,x} = \Sigma_{y \in Y} \beta_{z,y} \cdot \alpha_{y,x},$$

where the again the sums may be infinite. Notice that multirelation application is a special case of composition if we make the natural identification of multisets $\mu^\infty X$ with multirelations $\mathbf{1} \rightarrow_\mu^\infty X$.

Note the composition of multirelations $\alpha : X \rightarrow_\mu Y$ and $\beta : Y \rightarrow_\mu Z$ need not give a multirelation $\beta\alpha : X \rightarrow_\mu Z$, and specifically the application of a multirelation $\beta : Y \rightarrow_\mu Z$ to a multiset $\alpha \in \mu Y$ need *not* yield a multiset $\beta\alpha \in \mu Z$. This is easily seen in the following example. Let f be the multiset over $B = \{b_0, b_1, \dots, b_n, \dots\}$ given by $f_b = 1$ for all $b \in B$, and let $\beta : B \rightarrow_\mu \{c\}$ be the multirelation given by $\beta_{c,b} = 1$ for all $b \in B$. Then βf is the multiset over $\{c\}$ with $(\beta f)_c = \infty \notin \mu\{c\}$. Of course such situations cannot occur for finite multirelations and multisets.

Multirelation composition and application are *linear* in the sense that

$$\begin{aligned}\beta(n \cdot \alpha) &= n \cdot (\beta\alpha) \quad \text{and} \\ \beta(\Sigma_{i \in I} \alpha(i)) &= \Sigma_{i \in I} \beta\alpha(i)\end{aligned}$$

where $\alpha, \alpha(i) : X \rightarrow_\mu^\infty Y$, for $i \in I$, $n \in \mathbf{N}^\infty$ and $\beta : Y \rightarrow_\mu^\infty Z$.

In fact, ∞ -multirelations $X \rightarrow_\mu^\infty Y$ are in 1-1 correspondence with linear functions $\mu^\infty X \rightarrow \mu^\infty Y$; such a linear function $\theta : \mu^\infty X \rightarrow \mu^\infty Y$ determines and is determined by the ∞ -multirelation with components $\alpha_{y,x} = (\theta x)_y$, for $x \in X, y \in Y$. Note however, it is not the case that an arbitrary multirelation $\alpha : X \rightarrow_\mu Y$, with no ∞ -components, gives a map $\mu X \rightarrow \mu Y$.

Notice that as a consequence of linearity if $\alpha : X \rightarrow_\mu Y$ and $f - g \in \mu X$ then $\alpha(f - g) = \alpha f - \alpha g$: take $h = f - g$ then $h + g = f$ so, by linearity, $\alpha(h + g) = \alpha h + \alpha g = \alpha f$ so $\alpha h = \alpha f - \alpha g$.

We identify sets and relations with special kinds of multisets and multirelations, though be aware that the multiset application of a relation R to a set X does not always yield a set because more than one element of X may have the same image under R , and similarly that the multirelation composition of relations does not always yield a relation for essentially the same reason.

Infinite sums of vectors

In our treatment of invariants of infinite nets we shall use infinite sums of vectors. The treatment of convergence and nonconvergence of sums of infinite sets of integers in \mathbf{Z} is considerably more subtle than that of integers in \mathbf{N} ; whether such a sum converges or not and to what value can depend on how it is grouped, and partition associativity is lost. This means that were we to extend \mathbf{Z} by ∞ and define the matrix operations correspondingly we would lose many pleasant properties such as associativity of matrix composition. A way to preserve such properties as partition associativity and distribution of multiplication over sums is to take indexed sum of positive and negative integers to be a partial operation, only defined when the sum is finite.

Let $\{f_i \mid i \in I\}$ be an indexed set of integers in \mathbf{Z} . The sum $\sum_{i \in I} f_i$ is only defined when the indexed set $\{f_i \mid i \in I\}$ is finite, when it is the usual sum; otherwise the sum is undefined.

Thus, in general, indexed sum is a partial operation. This possibility of a result being undefined affects vector and matrix operations too. For instance we can define a partial sum operation on an indexed sets of vectors $\{f(i) \mid i \in I\}$ over X by taking

$$(\sum_{i \in I} f(i))_x = \sum_{i \in I} f(i)_x$$

provided each sum $\sum_{i \in I} f(i)_x$ is finite, and taking it as undefined otherwise.

Matrix composition and application

Let $f \in \nu X$. Let $\alpha : X \rightarrow_\nu Y$. Define the application of α to f to be the vector $\alpha f \in \nu Y$ which satisfies

$$(\alpha f)_y = \sum_{x \in X} \alpha_{y,x} \cdot f_x$$

for all $y \in Y$, provided each indexed sum of integers $\{\alpha_{y,x} \cdot f_x \mid x \in X\}$ is finite; otherwise take αf to be undefined.

Let $\alpha : X \rightarrow_\nu Y$ and $\beta : Y \rightarrow_\nu Z$. Define their composition $\beta \circ \alpha$ (often written as just $\beta\alpha$) to be the matrix $\beta \circ \alpha : X \rightarrow_\nu Z$ given by

$$(\beta \circ \alpha)_{z,x} = \sum_{y \in Y} \beta_{z,y} \cdot \alpha_{y,x},$$

for $x \in X, z \in Z$, provided each such sum is defined; otherwise take the matrix composition to be undefined. Again application is a special case of composition once we identify νX with $\mathbf{1} \rightarrow_\nu X$.

Let $\gamma : W \rightarrow_\nu X, \beta : X \rightarrow_\nu Y, \alpha : Y \rightarrow_\nu Z$. There are unfortunately examples where the composition $\alpha(\beta\gamma)$ is defined and yet $(\alpha\beta)\gamma$ is not, and *vice versa*, so in this sense associativity is still lost. However in the case where $\{(x, y) \mid \alpha_{z,y} \cdot \beta_{y,x} \cdot \gamma_{x,w} \neq 0\}$ is finite for all $w \in W, z \in Z$ we do have $\alpha(\beta\gamma)$ and $(\alpha\beta)\gamma$ are both defined and equal.

Matrix composition and application are *linear* in the sense that

$$\begin{aligned} \beta(n \cdot \alpha) &= n \cdot (\beta\alpha) \quad \text{and} \\ \beta(\sum_{i \in I} \alpha(i)) &= \sum_{i \in I} \beta\alpha(i), \end{aligned}$$

with one side being defined iff the other is, where $\alpha, \alpha(i) : X \rightarrow_\nu Y$, for $i \in I$ a finite indexing set, $n \in \mathbf{Z}$ and $\beta : Y \rightarrow_\nu Z$.

The fact that operations can be partial has caused us some trouble. Fortunately in almost all of our treatment of Petri nets we are able to avoid partial operations and the value ∞ ; the extra structure present in Petri nets will define subspaces on which all the operations we shall consider will be defined for vectors, and never yield the value ∞ for multisets.

References

- [Ac] Aczel, P., An introduction to inductive definitions. In the handbook of Mathematical Logic, Ed. Barwise, J., North-Holland (1983).
- [AM] Arbib, M.A., and Manes, E.G., Arrows, Structures and Functors, The categorical imperative. Academic Press (1975).
- [AM1] Arbib, M.A., and Manes, E.G., Formal semantics of programming languages. Final version forthcoming, preliminary version (1982).
- [Br] Brauer, W. (Ed.), Net Theory and Applications, Springer-Verlag Lecture Notes in Comp. Sci., vol.84 (1980).
- [BRS] Boudol, G., Roucairol, G., and de Simone, R., Petri nets and algebraic calculi of processes. Submitted to Advances in Petri nets (1985).
- [CH] Campbell, R. H., and Habermann, A. N., The Specification of Process Synchronisation by Path Expressions. Springer-Verlag Lecture Notes in Comp. Sc. Vol.16 (1974).
- [GKR] Györy, G., Knuth, E., and Romai, L., Grammatical projections. Working paper of Comp. and Automation Institute, Hungarian Academy of Sciences (1979).
- [GR] Goltz, U. and Reisig, W., Processes of Place/Transition Nets. Icalp 83 and to appear in Information and Control.
- [GM] Goltz, U. and Mycroft, A., On the relationship of CCS and Petri nets. ICALP 84 Springer-Verlag Lecture Notes in Comp. Sc. Vol.172 (1984).
- [H] Hoare, C.A.R., Communicating sequential processes. Comm. ACM 21 (1978).
- [HBR] Hoare, C.A.R., Brookes, S.D., and Roscoe, A.W., A Theory of Communicating Processes, Technical Report PRG-16, Programming Research Group, University of Oxford (1981); in JACM (1984).
- [J1] Jensen, K., Coloured Petri nets and the invariant method. DAIMI PB-104, Report of Comp. Sc. Dept., University of Aarhus, Denmark (1979).
- [J2] Jensen, K., High level Petri nets. DAIMI PB-151, Report of Comp. Sc. Dept., University of Aarhus, Denmark (1982).
- [LC] Lauer, P. E. and Campbell, R. H., Formal semantics for a class of high-level primitives for coordinating concurrent processes. Acta Informatica 5 pp.297-332 (1974).
- [Mac] MacLane, S., Categories for the Working Mathematician. Graduate Texts in Mathematics, Springer (1971).

[Maz] Mazurkiewicz, A., Concurrent program schemes and their interpretations. Report PB-78 of the Computer Sc. Dept., University of Aarhus, Denmark (1977).

[MB] Main, M.G., and Benson, D.B., Functional behaviour of nondeterministic programs. In the proceedings of FCT 83, Springer-Verlag Lecture Notes in Comp. Sc. (1983).

[M1] Milner, R., A Calculus of Communicating Systems. Springer Lecture Notes in Comp. Sc. vol. 92 (1980).

[M2] Milner, R., Calculi for synchrony and asynchrony. Theoretical Computer Science, pp.267-310 (1983).

[NW] Nielsen, M., Winskel, G., Compositional reasoning about invariants in Petri nets. In preparation.

[NPW1] Nielsen, M., Plotkin, G., Winskel, G., Petri nets, Event structures and Domains. Proc. Conf. on the Semantics of Concurrent Computation, Evian, France, Springer-Verlag Lecture Notes in Comp. Sc. 70 (1979).

[NPW] Nielsen, M., Plotkin, G., Winskel, G., Petri nets, Event structures and Domains, part 1. Theoretical Computer Science, vol. 13 (1981).

[Pe] Peterson, J. L., Petri Net Theory and the Modelling of Systems. Prentice-Hall (1981).

[R] Reisig, W., Petri nets. Springer Lecture Notes in Comp. Sc. (1984).

[R1] Reisig, W., Predicate/transition nets as abstract data types. Unpublished note, Dept. of Comp. Sc., University of Aachen, W.Germany (1984).

[Stu] Student projects for the course "Models for Concurrency", given by M. Nielsen and G. Winskel at the Computer Science Dept., University of Aarhus, Denmark (1980).

[W] Winskel, G., Events in Computation. Ph.D. thesis, University of Edinburgh (1980).

[W1] Winskel, G., Event structure semantics of CCS and related languages, Springer-Verlag Lecture Notes in Comp. Sc. 140 and as a report of the Computer Sc. Dept., University of Aarhus, Denmark (1982).

[W2] Winskel, G., Synchronisation trees. In Theoretical Computer Science, May 1985.

[W3] Winskel, G., A New Definition of Morphism on Petri Nets. Springer Lecture Notes in Comp Sc, vol. 166 (1984).

[W4] Winskel, G., Categories of Models for Concurrency. In the proceedings of the workshop on the semantics of concurrency, Carnegie-Mellon University, Pittsburgh, Springer Lecture Notes in Computer Science 197 (July 1984), and appears as a report of the Computer Laboratory, University of Cambridge (1984).