On the Semantics of Concurrency:
Partial Orders and Transition Systems

G. Boudol & I. Castellani
INRIA Sophia-Antipolis
06560-VALBONNE FRANCE

Abstract.

We introduce an algebra of labelled event structures whose operations are sequential composition, sum, and parallel composition. A transition relation is defined on these objects, where at each step a process performs a labelled poset. It is claimed that the bisimulation relative to such transition systems brings out a clean distinction between concurrency and sequential non-determinism.

1. Introduction.

This paper may be seen as proposing a tentative synthesis of various approaches to the semantics of concurrency. Milner's work on *calculi of processes* ([17,18,19]) provides our main source of inspiration. Let us recall the main features of such calculi (cf [1]): first there is a *syntax* which describes *abstract programs* as terms of an algebra; second there are *behavioural rules* according to which each term may perform some *actions* and become another term in doing so. This brings in a notion of *labelled transitions* denoted

$$prog \xrightarrow{act} prog'$$

Finally a *semantic equality* is defined by means of the well-known notion of bisimulation [21,18,3]. This gives the scheme of the following technical material.

Plotkin has advocated in [23] that labelled transition systems determined by structural operational rules provide a fairly natural setting to describe the operational semantics of programming languages. This is even more true with regard to parallel programming where one wants to program non-terminating processes, which may communicate during the computations: here functions from input to output can no longer be used as the semantical model. A symptom of this need of a more discriminating model is that a process is sometimes thought of as giving rise to a whole domain of computations rather than interpreted as a point in a domain; this point of view is exemplified by Winskel's work [20,32,34].

We shall entirely adopt Milner's standpoint [18,19] according to which any abstract notion of process must be based firmly upon operational semantics. As a matter of fact, one often uses informal behavioural arguments in order to decide whether some processes should or should not be distinguished. For instance (taken from [5]) one can "prove"

$$(a|(b+c)) + (a|b) + ((a+c)|b) = (a|(b+c)) + ((a+c)|b)$$

(we use here a CCS-like notation) by saying that if the left-hand side performs a concurrently to the b of $(a|b)$, then the right-hand side is able to do the same thing by choosing a in $((a+c)|b)$, and so on, so that no behaviour distinguishes the two terms. We shall give here a precise meaning to such a proof, by means of bisimulations.

Bisimulations on transition systems provide a powerful concept (see [1,3]), but many authors argue ([4,6,28], to mention but a few) that this yields an inadequate description of concurrency; specifically what is questioned is Milner's expansion theorem [17,14], expressing a simulation of concurrency by sequential non-determinism. Roughly speaking $(a \mid b) = ab + ba$, thus the parallel composition operator can be eliminated (from finite terms), whence it is not primitive. As a contribution to the theory of "true concurrency", our paper aims at solving

♡ *concurrency* $\neq$ *sequentiality* + *non-determinism*

More precisely our thesis is that this can be solved while still dealing with bisimulations on transition systems.

Evolving from Petri's ideas [22], there is another way to approach the semantics of concurrency; following this way one thinks of sequentiality as *causality*, that is as prescribing an ordering on events. Dually, two events are concurrent if they are not causally related. Thus here a computation is a *partially ordered set of events* rather than a mere sequence. This is by now a widely held point of view; it appears in the early work of Mazurkiewicz on traces [15], which have been related to posets and algebraic structures (monoids) in [16] and [29]. Another generalization of words was proposed by Winkowski [30,31]. Grabowski sets up in [12] a theory of "partial words" that are labelled posets, what Pratt and Gischer call *pomsets* ([26,10,27], see also [29]). These are also the configurations of Winskel's *(labelled) event structures*, which are posets enriched with a notion of conflict [32,33,34] – a kind of object that Montanari & *al.* also deal with [4,7,8]. Fairly close is the notion of process suggested by Petri [22], which is a partial unfolding of a net into an occurrence net (cf [9,11]). Reisig studies in [28] what can or cannot be distinguished according to various notions of computations. By the way, we must point out the fact that almost all the works we have just mentioned model more or less explicitely a process as a "language", that is a set of pomsets; this entails the linearity of sequential and parallel composition, that is their distributivity over the sum interpreted as set theoretic union. Roughly speaking, $(a|(b+c)) = (a|b) + (a|c)$ and $a\,;(b+c) = a\,;b + a\,;c$, a kind of property that does not hold in Milner's calculi of processes.

Let us now introduce our contribution: first of all, in order to solve ♡ we must start with a formalism in which one can talk about sequentiality, non-determinism and concurrency as distinct notions; this is why we adopt Winskel's (labelled) event structures which are built upon the exclusive relations of causal ordering, conflict and concurrency. Each of these relations gives rise to a way of constructing event structures: one simply juxtaposes two such structures and then sets the relation between their events. These operations are *sequential composition*, *sum*, and *parallel composition*; they provide us with a syntax for finite event structures (in this paper we shall treat neither infinite structures nor communication; to get some ideas about these subjects see the full version of the paper [2]).

Here comes the main idea. We have already mentioned that an event structure determines a set of computations, what Winskel calls configurations. Then, defining "what remains of the structure" after such a computation we get a notion of labelled transition: here the action (= the computation) is a finite pomset and the reached state (= what remains...) is another event structure. The point is that we generalize what usually is "over the arrow"; a similar idea may be found in [5,8] and it seems that it could be applied to Petri nets where computations are processes (in the technical sense of [9,11]). As a matter of fact, we also extend Milner's idea ([18]) that actions should be elements of a commutative monoid (a similar notion is Winskel's synchronization algebra [33,34]): here we get elements of a "dioid", see below.

We also give a structural operational semantics (in Plotkin's style [23,24]) for our "abstract programs", and then show an exact correspondence between the semantical and syntactical notions of transition. Next we define our semantic equality, in the same way as Milner defines his strong congruence, and give an axiomatization for it. We claim that this notion of equality solves ♡.

Note: almost all the proofs are omitted; more details may be found in [2].

2. Algebra of Labelled Event Structures.

As previously announced, our first concern is in labelled event structures. For some technical reasons that will become clear later, our definition is a slight variation of Winskel's one. At some points we shall assume knowledge of the work of Nielsen, Plotkin, and Winskel [20] which shows how to derive (labelled) event structures from some kind of (labelled) Petri nets; thus we shall feel free to use standard concepts of net theory (cf [9]) when dealing with such derived event structures.

2.1 *Labelled Event Structures and Terms.*

Let as usual $\{0,1\}^*$ be the set of words over the alphabet $\{0,1\}$. The concatenation of two words u and v is denoted uv, whereas the product of two languages L and L' is

$$LL' = \{uv/u \in L \ \&\ v \in L'\}$$

DEFINITION. *Let A be a non-empty set. An A-labelled event structure (A-LES for short) is a structure $(E, \leq, \#, \lambda)$ where*

(i) $E \subseteq \{0,1\}^*$ is the set of events,

(ii) $\leq$ is a partial order on E, the causality relation,

(iii) $\# \subseteq E \times E - (\leq \cup \geq)$ is the symmetric conflict relation,

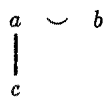
(iv) $\lambda: E \to A$ is the labelling function.

Note that we do not require Winskel's axiom of conflict heredity. Two events in E are *concurrent* if they are neither comparable nor in conflict, that is

$$\smile \ =_{\text{def}} \ E \times E - (\leq \cup \geq \cup \ \#)$$

This is a symmetric irreflexive relation. Note that by definition $\leq \cup \geq$, $\#$, and $\smile$ set a partition upon $E \times E$.

We shall always draw structures up to isomorphism, that is omitting the name of events; in the figures the order $\leq$ increases downwards and only one of the remaining relations is explicitely shown. For instance



is a structure with three events e, e' and e'' respectively labelled a, b and c such that e causes e'', e and e' are concurrent and e' and e'' are in conflict. In what follows we let $a, b, c, \ldots$ range over A.

We use $\mathcal{L}(A)^\infty$ for the set of A-labelled event structures and $\mathcal{L}(A)$ for the set of finite ones. *In this paper we shall only take finite structures into consideration* (a more general study may be found in [2]). This set is naturally supplied with an algebraic structure: let V be one of $\leq$, $\smile$, $\#$ and S_0, S_1 be A-LES's; then $S_0(V)S_1$ is the structure we get by juxtaposing S_0 and S_1 and setting the V relation between the events of S_0 and S_1. When V is $\leq$ this is called *sequential composition* of S_0 and S_1 and denoted $S_0 \,;\, S_1$, whereas if V is $\smile$ this is the *parallel composition* $S_0 \parallel S_1$ and in the case $V = \#$ this is the *sum* $S_0 + S_1$. The formal definition is the following: assuming

$$S_i = (E_i, \leq_i, \#_i, \lambda_i) \qquad \text{for} \quad i \in \{0,1\}$$

one defines $S_0(V)S_1$ to be $(E, \leq, \#, \lambda)$ where

$$\begin{cases} E = E_0 \uplus E_1 & \text{i.e. } E = \{0\}E_0 \cup \{1\}E_1 \\ ix \leq jy & \Leftrightarrow \ i = j \text{ and } x \leq_i y \text{ or } V = \leq,\ i = 0 \text{ and } j = 1 \\ ix \# jy & \Leftrightarrow \ i = j \text{ and } x \#_i y \text{ or } V = \# \text{ and } i \neq j \\ \lambda(ix) = \lambda_i(x) & \end{cases}$$

These operations are naturally defined up to isomorphism. That is, denoting $P \rightleftharpoons Q$ the relation "P and Q are isomorphic",

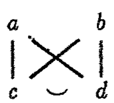
$$P \rightleftharpoons P' \text{ and } Q \rightleftharpoons Q' \quad \Rightarrow \quad \begin{cases} P \,;Q \rightleftharpoons P' \,; Q' \\ P + Q \rightleftharpoons P' + Q' \\ P \parallel Q \rightleftharpoons P' \parallel Q' \end{cases}$$

Thus $\mathcal{L}(A)/\rightleftharpoons$ inherits the algebraic structure.

All that means is that we have a *syntax* to denote finite A-LES's. This abstract syntax is the set $T(A)$ of terms built according to the following rules:
(i) $\mathbb{1}$ is a term and every *atom* $a \in A$ is a term,
(ii) if p and q are terms then so are $(p\,; q)$, $(p \parallel q)$ and $(p + q)$.
Let $J(p)$ be the labelled event structure denoted by the term p, defined as follows:

$$\begin{aligned} J(\mathbb{1}) &= (\emptyset, \emptyset, \emptyset, \emptyset) \quad \text{(the empty structure)} \\ J(a) &= (\{\varepsilon\}, =, \emptyset, \alpha) \quad \text{with } \alpha(\varepsilon) = a \\ J(p\,; q) &= \big(J(p)\,; J(q)\big) \\ J(p \parallel q) &= \big(J(p) \parallel J(q)\big) \\ J(p + q) &= \big(J(p) + J(q)\big) \end{aligned}$$

The symbol $\mathbb{1}$ will be used also for the empty structure and its isomorphism class. Let us see a few examples: the term $(a + b)\,; (c \parallel d)$ denotes the structure



This and the simpler term $(a + b)\,; c$ show why we cannot assume Winskel's axiom of *conflict heredity* [32]. The term $(a \parallel b) + c$ is interpreted as

$$a \# c \# b$$

(where $a \smile b$, and there is no non-trivial causal dependency) and is an example of "symmetric confusion" (see [9,20]).

In the next section we shall characterize both the set of structures which are interpretations of terms up to isomorphism and the *interpretation equality*

$$p =_J q \quad \Leftrightarrow_{\text{def}} \quad J(p) \rightleftharpoons J(q)$$

2.2 *Characterization.*

One may remark that in $\mathcal{L}(A)^\infty/\rightleftharpoons$ the three operations previously defined are associative and have $\mathbb{1}$ as neutral element; moreover the sum and parallel composition are commutative. This suggests the following definition:

DEFINITION. *A trioid is an algebra* $(\mathrm{T}, ;, \|, +, \mathbb{1})$ *satisfying the axioms*

(i) $(\mathrm{T}, ;, \mathbb{1})$ *is a monoid:*

A0: $(p\,;(q\,;r)) = ((p\,;q)\,;r)$

U0: $(p\,;\mathbb{1}) = p = (\mathbb{1}\,;p)$

(ii) $(\mathrm{T}, \|, \mathbb{1})$ *is a commutative monoid:*

A1: $(p \,\|\, (q \,\|\, r)) = ((p \,\|\, q) \,\|\, r)$

U1: $(p \,\|\, \mathbb{1}) = p = (\mathbb{1} \,\|\, p)$

C1: $(p \,\|\, q) = (q \,\|\, p)$

(iii) $(\mathrm{T}, +, \mathbb{1})$ *is a commutative monoid:*

A2: $(p + (q + r)) = ((p + q) + r)$

U2: $(p + \mathbb{1}) = p = (\mathbb{1} + p)$

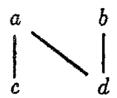
C2: $(p + q) = (q + p)$

Let Θ be the equational theory whose axioms are A0 to A2, U0 to U2, C1 and C2, and let $=_\Theta$ be the congruence on $T(A)$ generated by these equations. Then we have an obvious *soundness* property:

$$p =_\Theta q \quad \Rightarrow \quad p =_J q$$

We now wish to check whether a converse *completeness* property holds. First we shall see that not all finite labelled event structures are interpretations of terms. As a matter of fact the structure

N



(without conflict) is known to be the typical one that cannot be expressed by means of sequential and parallel composition, cf [10,12,27]. We thus want to find a class of A-LES's which does not contain N. In order to define this class and state our characterization result we need to introduce some notations. Let $R \subseteq E \times E$ be a relation on a set E.

(i) $R^\varepsilon = R \cup R^{-1} \cup R^0$ is the reflexive and symmetric closure of R, what we shall call the *R-comparability* relation.

(ii) $\ddagger(R) = (E \times E) - R^\varepsilon$ is the symmetric, irreflexive R-incomparability relation.

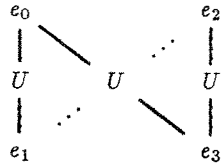
(iii) $\sim_R = (R \cup R^{-1})^*$ is the equivalence generated by R whose classes are the connected components with respect to the R-comparability relation.

For instance the comparability relations determined by # and $\smile$ are simply their reflexive closure, whereas the $\leq$-comparability is $\leq \cup \geq$ what we denote $\Diamond$. In order to avoid many useless repetitions we shall name each of the relations $\leq$, #, $\smile$ a *connective* of a given structure S.

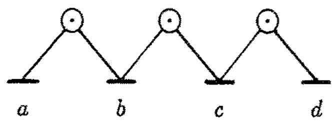
The first property we shall require is N-*freeness*; an A-LES S is N-free if it satisfies

$$\text{N-}\mathit{freeness}\quad \begin{cases} \text{for } U \text{ a connective of } S \\ \text{if } e_0\; U\; e_1 \text{ and } e_0\; \ddagger(U)\; e_2 \\ \text{if } e_2\; U\; e_3 \text{ and } e_1\; \ddagger(U)\; e_3 \\ \text{then } e_0\; U\; e_3 \;\Rightarrow\; e_2\; U\; e_1 \end{cases}$$

This property, which is obviously preserved by isomorphism, may be drawn



This typically precludes a structure such as $a \# b \# c \# d$ (where $a \smile c$, $b \smile d$, and $a \smile d$) which is derived (see [20]) from the Petri net



N-freeness is also related to Petri's notion of *K-density* [22], see [13,25].

N-freeness is not enough by itself to characterize the class of A-LES's denoted by terms. Here we need another requisite which we may call the *triangle property*: a structure S satisfies this property if it does not contain a configuration

$$\triangle \qquad e \diamondsuit e' \# e'' \smile e$$

This precludes the typical situation of "asymmetric confusion" (cf [9,20]).

In fact the "behavioural" properties of N-freeness and triangle may be combined in a single one – which is less readable but somehow more natural when looking for a property preserved by the operations.

LEMMA. *An A-labelled event structure S satisfies* N*-freeness and the triangle property if and only if it satisfies the property*

$$\mathbf{X} \quad \begin{cases} \textit{for } U \textit{ and } V \textit{ among } \leq, \#, \smile \textit{ with } U \neq V \\ \textit{if } e_0\ U^{\varepsilon}\ e_1 \textit{ and } e_0 \ddagger(U)\ e_2 \\ \textit{if } e_2\ U^{\varepsilon}\ e_3 \textit{ and } e_1 \ddagger(U)\ e_3 \\ \textit{then } e_0\ V\ e_3 \Rightarrow \{e_0, e_1\} \times \{e_2, e_3\} \subseteq V \end{cases}$$

In the course of the proof (see [2]) we use the fact that N-freeness implies

$$\mathbf{N}' \quad \begin{cases} \textit{for } U \textit{ among } \diamondsuit, \#, \smile \\ \textit{if } e_0\ U\ e_1 \textit{ and } e_0 \ddagger(U)\ e_2 \\ \textit{if } e_2\ U\ e_3 \textit{ and } e_1 \ddagger(U)\ e_3 \\ \textit{then } e_0\ U\ e_3 \Rightarrow e_2\ U\ e_1 \end{cases}$$

This fact will be also used later. We can finally define the intended class of structures as follows:

DEFINITION. *The set $\mathcal{X}(A)$ is the set of finite A-LES's satisfying the* **X** *property.*

The set of structures $\mathcal{X}(A)$ is a generalization of Grabowski-Gischer's class of *N-free pomsets* [12,10]. Clearly the **X** property is *hereditary*; this means that if

$$S' \subseteq S \quad \Leftrightarrow_{\text{def}} \quad \begin{cases} S = (E, \leq, \#, \lambda) \quad \text{and} \quad \exists F \subseteq E \\ S' = S \lceil F = (F\ ,\ \leq \cap (F \times F)\ ,\ \# \cap (F \times F)\ ,\ \lambda \lceil F\ ) \end{cases}$$

then $S' \subseteq S \;\&\; S \in \mathcal{X}(A) \;\Rightarrow\; S' \in \mathcal{X}(A)$.

We can now state the announced result, which generalizes Grabowski-Gischer's one.

THEOREM 1. *The structure* $(\mathcal{X}(A)/\rightleftharpoons, ;, \|, +, \mathbb{1})$ *is the free trioid generated by* A. *One especially has*

(i) $S \in \mathcal{X}(A) \quad \Leftrightarrow \quad \exists p \in T(A)\ J(p) \rightleftharpoons S$

(ii) $p =_J q \quad \Leftrightarrow \quad p =_\Theta q$

The complete proof is rather long, involving some straightforward parts. Here we only sketch it; more details may be found in [2]. One has to prove that $\mathcal{X}(A)/\rightleftharpoons$ is a trioid isomorphic to $T(A)/=_\Theta$. We have already seen that the algebra $\mathcal{L}(A)/\rightleftharpoons$ is a model of the theory Θ. Thus the first thing to see is that the operations preserve the **X** property; an immediate consequence will be that $\mathcal{X}(A)/\rightleftharpoons$ is a trioid which contains the interpretation of every term.

LEMMA 1. *If* $S_0, S_1 \in \mathcal{X}(A)$ *then* $S_0\,;S_1$, $S_0 + S_1$ *and* $S_0 \| S_1$ *are in* $\mathcal{X}(A)$

The proof proceeds by case inspection ∎

Next one has to show that each element of $\mathcal{X}(A)/\rightleftharpoons$ is denoted by a term of $T(A)$, univocally up to $=_\Theta$. As usual this completeness property lies upon the existence of *normal forms* for terms. These can be described as follows: let $\mathcal{N}(A) = \{\mathbb{1}\} \cup \mathcal{W}(A)$ where $\mathcal{W}(A)$ is the least set of terms built according to the rules

(i) every atom $a \in A$ is in $\mathcal{W}(A)$ and has no *head operator*,

(ii) if $p \in \mathcal{W}(A)$ does not have ; (resp. $\|$, $+$) as head operator and if $q \in \mathcal{W}(A)$ then $(p\,;q)$ (resp, $(p \| q)$, $(p+q)$) is in $\mathcal{W}(A)$ and has ; (resp. $\|$, $+$) as head operator.

One gets normal forms by cancelling the unit and using associativity to shift arguments to the right.

PROPOSITION. *Let* Γ *be the theory whose axioms are* A0 *to* A2 *and* U0 *to* U2, *and* Υ *be the theory consisting of* A0 *to* A2, C1 *and* C2. *Then*

(i) for each term $p \in T(A)$ *there exists a normal form* $t \in \mathcal{N}(A)$ *such that* $p =_\Gamma t$,

(ii) for two normal forms $t, t' \in \mathcal{N}(A) \quad t =_\Theta t' \ \Leftrightarrow\ t =_\Upsilon t'$

This is a standard result. The proof is omitted.

The crux of the characterization theorem's proof is the following property: for every finite non-empty non-atomic labelled event structure satisfying the **X** property, the set of events is connected for exactly one of the connectives $\leq$, $\smile$, # (in fact this is a purely graph-theoretical result); this relation gives the head operator of the term which denotes the structure. The existence of such a connective comes from the triangle property, whereas uniqueness comes from N-freeness (or more accurately from $\mathbf{N}'$).

LEMMA 2. *Let* $S = (E, \leq, \#, \lambda)$ *be an A-LES in* $\mathcal{X}(A)$.

(i) there exists a connective U *of* S *for which* E *is connected, that is* $\#(E/\sim_U) = 1$;

(ii) moreover if $\#(E) > 1$ *then* E *is not connected for the* U*-incomparability relation* $\ddagger(U)$, *and thus is not connected for any of the other connectives.*

PROOF. We first show that there is one such relation U, for each $S \in \mathcal{X}(A)$. Suppose not, and let C be a *maximal* (w.r.t. inclusion) subset of E connected for some connective. From our assumption $C \neq E$, so let $e \in E - C$. Then e is connected in the same way ($\diamondsuit$, # or $\smile$) with all the elements of C, otherwise E would contain a triangle. But then $\{e\} \cup C$ is, for some connective, a connected subset of E which strictly contains C.

Now to prove the second point let us assume that E is connected for both U and $\ddagger(U)$ for U among $\diamondsuit$ (since E $\leq$-connected $\Leftrightarrow$ E $\diamondsuit$-connected), # and $\smile$. Let F be a *minimal* (w.r.t. inclusion) subset of E which is both U and $\ddagger(U)$ connected and such that $\#(F) > 1$. Then $\#(F) > 2$ since one cannot build a two element structure which is connected for two exclusive relations. So let

$e_3 \in F$; since $F - \{e_3\}$ is not connected for both U and $\ddagger(U)$, let us assume for instance that $F - \{e_3\}$ is not connected for U, that is

$$(F - \{e_3\})/\sim_U = \{F_1, \ldots, F_m\} \quad \text{with } m > 1$$

Then

$$\exists i\ (1 \leq i \leq m)\ \exists e \in F_i \quad e_3\ \ddagger(U)\ e$$

otherwise F could not be $\ddagger(U)$-connected. Similarly

$$\forall i\ (1 \leq i \leq m)\ \exists e \in F_i \quad e_3\ U\ e$$

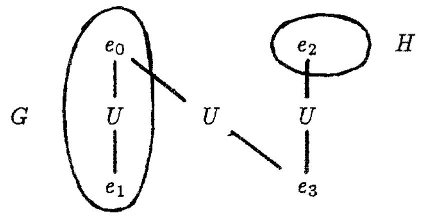
So let G be an F_i such that $\exists e \in F_i \quad e_3\ \ddagger(U)\ e$ and H be $\bigcup_{i \neq j} F_j$. Since

$$\exists e \in G\ e_3\ \ddagger(U)\ e \quad \text{and} \quad \exists e' \in G\ e_3\ U\ e' \quad \text{and} \quad G \text{ is } U-\text{connected}$$

one has

$$\exists e_0 \in G\ \exists e_1 \in G \quad e_3\ U\ e_0 \text{ and } e_0\ U\ e_1 \text{ and } e_1 \ddagger(U)\ e_3$$

If we choose an $e_2 \in H$ such that $e_3\ U\ e_2$ we may figure the situation as



By definition of G and H, $e_0\ \ddagger(U)\ e_2$ and $e_1\ \ddagger(U)\ e_2$, but this contradicts the $\mathbf{N}'$ property, which is a consequence of the $\mathbf{X}$ property.
The proof is the same when $F - \{e_3\}$ is not $\ddagger(U)$-connected ■
We can now prove

$$\forall S \in \mathcal{X}(A)\ \exists t \in \mathcal{N}(A) \quad \mathcal{J}(t) \rightleftharpoons S$$

by induction on the size $\#(E)$ of S (in fact the induction hypothesis states that the head operator of the term t corresponds to the unique connective, if it exists, for which E is connected).

If $\#(E) < 2$ then this is trivial: t is either $\mathbb{1}$ or an atom (given by the labelling function). Otherwise by the previous lemma there exists a connective U for which E is connected and not $\ddagger(U)$-connected. Let

$$\{C_1, \ldots, C_m\} = E/\sim_{\ddagger(U)}$$

Then $1 < m \leq \#(E)$. From the definition of the C_ℓ's it cannot be the case that $e \ddagger(U) e'$ for some $e \in C_i$ and $e' \in C_j$ $(i \neq j)$. Suppose now that U is $\leq$ (the other cases where U is # or $\smile$ are similar, and even simpler). Let us see that if $e < e'$ for some $e \in C_i$ and $e' \in C_j$ then for all $e'' \in C_j \quad e < e''$ whence $C_i \times C_j \subseteq <$. Otherwise there would be e_0 and e_1 in C_j such that $e_0 < e < e_1$, thus $e_0 < e_1$, and $e_0 \# e_1$ or $e_0 \smile e_1$, which is a contradiction. Thus we may assume that $\{C_1, \ldots, C_m\}$ is enumerated in such a way that $e_1 < \cdots < e_m$ for some $e_i \in C_i$. For all i $(1 \leq i \leq m)$ $S\lceil C_i \in \mathcal{X}(A)$ since the $\mathbf{X}$ property is hereditary. Thus by induction hypothesis there are terms $t_1, \ldots, t_m$ of $\mathcal{W}(A)$ (whose head operators are not ;) such that

$$\forall i\ (1 \leq i \leq m) \quad \mathcal{J}(t_i) \rightleftharpoons S\lceil C_i$$

Then

$$S \rightleftharpoons \mathcal{J}((t_1\,;(\ldots;t_m)\ldots))$$

To conclude the proof of the theorem we must show

$$t, t' \in \mathcal{N}(A) \quad \Rightarrow \quad \mathcal{J}(t) \rightleftharpoons \mathcal{J}(t') \Leftrightarrow t =_{\Upsilon} t'$$

The proof of this last point is omitted ■

3. Operational semantics.

3.1 *Transitions on labelled event structures.*

The interpretation equality $=_\Theta$ is too discriminating; from a behavioural point of view we would like to identify the terms $p+p$ and p – in the introduction we have seen another example. Thus we cannot consider equality of event structures to be the equality of their domains of configurations (see [20]). Nevertheless, the equality we look for is based upon a notion of computation which would be Winskel's notion of finite configuration if we had assumed the axiom of conflict heredity. Note that computations are deterministic: choices (or conflicts) are resolved while a "program" computes. Computations bear some analogy with processes of Petri nets ([9,11]) or more accurately with Reisig's abstract processes [28].

DEFINITION. *Given an A-labelled event structure $S = (E, \leq, \#, \lambda)$ a computation of S is a structure $S\lceil F$ where*

(i) F is a finite subset of E,
(ii) $S\lceil F$ is conflict-free: $e \in F$ & $e' \in F \Rightarrow \neg(e \# e')$
(iii) $S\lceil F$ is closed under non-conflicting causes:
$e \in F$ & $e' \leq e$ & $e' \notin F \Rightarrow \exists e'' \in F \quad e'' \# e'$

Note that we only allow finite computations, thus we cannot deal with *fairness*; an idea could be that fair computations are the – possibly infinite, but satisfying an axiom of "finite causes" – *maximal* computations, w.r.t. the ordering $\subseteq$.

We shall name *action* an isomorphism class of computations. In this paper we restrict our attention to A-LES's of $\mathcal{X}(A)$. The computations of such structures are rather special: they are finite conflict-free (*elementary* in Winskel's terminology) A-LES's satisfying the **X** property. We denote by $\mathcal{P}(A)$ the set of these computations and by $\mathcal{D}(A) = \mathcal{P}(A)/\rightleftharpoons$ the set of actions they determine. In fact $\mathcal{D}(A)$ is exactly the set of what Pratt and Gischer [10,27] call finite *N-free pomsets*. From a theorem of Grabowski-Gischer $\mathcal{D}(A)$ is the free "*dioid*" (Grabowski calls it "double monoid"), which is the same as a trioid but without sum. All that means is that actions are denoted by terms built without sum, up to the equational theory Δ whose axioms are A0, A1, U0, U1 and C1. The set of these "deterministic" terms will be denoted $D(A)$.

For instance, making a confusion between terms and the structure they denote, $(a\,;c)$ and $(b\,;c)$ are computations of $((a+b)\,;c)$, while $((a\,;b) \parallel c)$ is a computation of $(a\,;(b+d) \parallel c\,;e)$. For $F \subseteq E$ let

$$\#(F) = \{e / \exists e' \in F \ e' \# e\}$$

From a computation $P = S\lceil F$ of S we build a structure called the *residual of S by P* which is

$$(S/P) =_{\text{def}} S\lceil(E - (F \cup \#(F)))$$

This structure is "what remains of S after removing P while resolving the conflicts". Clearly $S \in \mathcal{X}(A)$ implies $(S/P) \in \mathcal{X}(A)$. We are now ready to introduce the main definition which brings a structure of transition system on event structures. Let us recall the terminology: a *(labelled) transition system* $\Sigma = (Q, Act, T)$ is a structure where

(i) Q is the set of *states*,
(ii) Act is the set of *actions*,
(iii) $T \subseteq Q \times Act \times Q$ is the *transition relation*. $p \xrightarrow[T]{\alpha} p'$ will denote $(p, \alpha, p') \in T$.

DEFINITION. *The transition relation η between A-labelled event structures is given by*
$S \xrightarrow[\eta]{P} S' \quad \Leftrightarrow_{\text{def}} \quad$ *P is a computation of S and $S' = (S/P)$.*

Here one can see some analogy with the construction h_1 `before` h_2 `gives` h of Degano and Montanari ([7]) if one reads it $h \xrightarrow{h_1} h_2$.

For instance, still using terms in place of the structures, we have

$$(a\,;b) \xrightarrow[\eta]{a} b \qquad (a \parallel b) \xrightarrow[\eta]{a} b \qquad (a \parallel b) \xrightarrow[\eta]{(a \parallel b)} \mathbb{1}$$

$$((a+b)\,;c) \xrightarrow[\eta]{(a\,;c)} \mathbb{1} \qquad ((a+b) \parallel c) \xrightarrow[\eta]{(a \parallel c)} \mathbb{1}$$

One may remark that from the definition a computation of a LES S cannot introduce causal dependencies which would not be already present in S. For instance $(a\,;b)$ is not a computation of $(a \parallel b)$. We could say that in our behavioural semantics

$$\textit{causality} \Rightarrow \textit{temporal ordering}$$

for we have

$$S \xrightarrow[\eta]{P\,;Q} S' \Rightarrow \exists S'' \quad S \xrightarrow[\eta]{P} S'' \xrightarrow[\eta]{Q} S'$$

But the converse is false (consider $(a \parallel b) \xrightarrow[\eta]{a} b \xrightarrow[\eta]{b} \mathbb{1}$). Thus our semantics makes a strong distinction between *sequence of transitions* and *"transitions of a sequence"* – compare with the CCS "action" $a.p$.

One may also note that the behavioural interpretation of parallel composition is not interleaving, but contains it. This is due to the fact that an A-LES may always perform the empty computation; we may interpret $\mathbb{1}$ as "skip" – when regarded as a computation – or "termination" – for instance in a transition $S \xrightarrow[\eta]{P} \mathbb{1}$. Our semantics of parallel composition is a generalization of the MEIJE"asynchronous" operator [1] – related to Milner's synchronous product [18] and to the notion of "step" transition of Petri nets [32,28].

3.2 *Transitions on terms.*

Since we are interested in labelled event structures denoted by terms of $T(A)$ an obvious question is: is there any *syntactic* notion of transition which reflects the semantic one? In fact the (positive) answer is rather simple; let ρ be the least subset of $T(A) \times D(A) \times T(A)$ satisfying the following clauses or *rules*

R0: $\vdash\ p \xrightarrow{\mathbb{1}} p$

R1: $a \in A \ \vdash\ a \xrightarrow{a} \mathbb{1}$

R2: $p \xrightarrow{u} p' \ \vdash\ (p\,;q) \xrightarrow{u} (p'\,;q)$

R3: $p \xrightarrow{u} p' =_\Theta \mathbb{1}\ ,\ q \xrightarrow{v} q' \ \vdash\ (p\,;q) \xrightarrow{(u\,;v)} q'$

R4: $p \xrightarrow{u} p'\ ,\ q \xrightarrow{v} q' \ \vdash\ (p \parallel q) \xrightarrow{(u \parallel v)} (p' \parallel q')$

R5: $p \xrightarrow{u} p'\ \&\ u \neq_\Theta \mathbb{1} \ \vdash\ (p+q) \xrightarrow{u} p'$

R6: $q \xrightarrow{v} q'\ \&\ v \neq_\Theta \mathbb{1} \ \vdash\ (p+q) \xrightarrow{v} q'$

(note that $r =_\Theta \mathbb{1}$ can be proved or disproved using only the axioms U0 to U2). Since ρ is the *least* relation satisfying the given clauses, a transition $p \xrightarrow[\rho]{u} p'$ cannot hold unless it has a *proof* or construction according to these rules. For instance we have

$$\text{R3}:\dfrac{\text{R5}:\dfrac{\text{R1}:\dfrac{}{a \xrightarrow[\rho]{a} \mathbb{1}}}{(a+b) \xrightarrow[\rho]{a} \mathbb{1}} \qquad \text{R1}:\dfrac{}{c \xrightarrow[\rho]{c} \mathbb{1}}}{((a+b)\,;c) \xrightarrow[\rho]{(a\,;c)} \mathbb{1}}$$

Now we state the *adequation* result making the correspondence between transitions on terms and transitions on event structures.

THEOREM 2.

(i) $p \xrightarrow[\rho]{u} q \quad \Rightarrow \quad \exists U\ \mathcal{J}(u) \rightleftharpoons U \quad \exists Q\ \mathcal{J}(q) \rightleftharpoons Q \quad \mathcal{J}(p) \xrightarrow[\eta]{U} Q$

(ii) $\mathcal{J}(p) \xrightarrow[\eta]{U} Q \quad \Rightarrow \quad \exists u\ \mathcal{J}(u) \rightleftharpoons U \quad \exists q\ \mathcal{J}(q) \rightleftharpoons Q \quad p \xrightarrow[\rho]{u} q$

The proof lies upon an analysis of $S \xrightarrow[\eta]{P} S'$ when S is $(S_0;S_1)$, (S_0+S_1) or $(S_0\|S_1)$ for $S_0 \neq \mathbb{1} \neq S_1$; here one meets a translation of the rules R2 to R6 ■

4. Semantics.

4.1 *Equipollence.*

Relative to any transition system $\Sigma = (Q, Act, T)$ one may define the well-known Park and Milner notion of bisimulation [21,18]. Here we adapt Brookes and Rounds terminology (see [3,1]):

a relation $R \subseteq Q \times Q$ is

(i) *invariant* with respect to T if and only if it satisfies

$$p\,R\,q \quad \text{and} \quad p \xrightarrow[T]{\alpha} p' \quad \Rightarrow \quad \exists q' \quad p'\,R\,q' \quad \text{and} \quad q \xrightarrow[T]{\alpha} q'$$

(ii) a *bisimulation* (w.r.t. T) if it is a symmetric invariant relation,

(iii) an *equisimulation* if it is a bisimulation and also an equivalence.

The invariance property is usually drawn

$$\begin{array}{ccccc} p & \text{—} & R & \text{—} & q \\ \vdots & & & & \vdots \\ \alpha & & & & \alpha \\ \downarrow & & & & \downarrow \\ p' & \cdots & R & \cdots & q' \end{array}$$

The following fact is standard:

LEMMA. *Given a transition system Σ let us define*

$$p \asymp_T q \quad \Leftrightarrow_{\text{def}} \quad \exists R \ \ \textit{bisimulation} \ \ p\,R\,q$$

Then $\asymp_T$ is an equisimulation and it is the coarsest one.

The only point to check is that the composition of invariant relations is itself invariant ■
We shall call this equisimulation the *equipollence* with respect to T and sometimes use the alternative notation $p \asymp q\ (T)$ instead of $p \asymp_T q$.

We are in fact interested in transitions labelled by *actions*, that is classes of structures or terms. Let us ambiguously denote $[\![P]\!]$ and $[\![p]\!]$ the isomorphism class of the A-LES P and the $=_\Theta$-class of the term p. Then we define the transition relations $\bar{\eta}$ and $\bar{\rho}$

$$S \xrightarrow[\bar{\eta}]{[\![P]\!]} S' \quad \Leftrightarrow_{\text{def}} \quad \exists Q \rightleftharpoons P \quad S \xrightarrow[\eta]{Q} S'$$

$$p \xrightarrow[\bar{\rho}]{[\![u]\!]} p' \quad \Leftrightarrow_{\text{def}} \quad \exists v =_\Theta u \quad p \xrightarrow[\rho]{v} p'$$

We can show that there is an exact correspondence between the "syntactic" and "semantic" equipollences:

$$p \asymp q\ (\bar{\rho}) \quad \Leftrightarrow \quad \mathcal{J}(p) \asymp \mathcal{J}(q)\ (\bar{\eta})$$

(see [2] for a proof). Since isomorphism of A-LES's is an equisimulation a consequence is that

$$p =_\Theta q \quad \Rightarrow \quad p \asymp q\ (\bar{\rho})$$

Moreover these equipollences are also congruences with respect to the algebraic structure, that is compatible with the operations ;, + and $\|$.

The equipollence $\asymp(\bar{\rho})$ is what we regard as defining the *semantic equality* of terms. Thus we just use $\asymp$ to denote it. For instance the three terms $(a \,\|\, b)$, $(a\,;b)+(b\,;a)$ and $(a\,;b)+(a\,\|\,b)+(b\,;a)$ are pairwise distinct with respect to $\asymp$ since the first cannot perform the action $(a\,;b)$ whereas the second cannot perform $(a \,\|\, b)$. Another example is

$$(a\,;b \,\|\, c) \not\asymp (a \,\|\, c)\,;b + a\,;(b \,\|\, c)$$

4.2 *Axiomatization.*

In this section we aim to set up a "proof theory" of $\asymp$. It should be clear that $\bar{\eta}$-equipollence of elements of $\mathcal{P}(A)$ is exactly $\rightleftharpoons$, for

$$P \in \mathcal{P}(A) \ \Rightarrow\ \left(P \xrightarrow[\eta]{Q} \mathbb{1} \ \Leftrightarrow\ Q = P\right)$$

Thus any intended axiomatization essentially states properties of the sum. As a matter of fact there is a standard way to solve the problem, by means of *sumforms* as Hennessy and Milner have shown in [14] what we will briefly recall now. For any set *Act* of actions let $K(Act)$ be the set of terms built according to the following rules:

(i) $\mathbb{1}$ is a term,
(ii) for every $\alpha \in Act$ if p is a term then $\alpha \bullet p$ is a term,
(iii) if p and q are terms then so is $(p+q)$.

Let Ψ be the theory whose axioms are A2, U2, C2, and

I: $(p+p)=p$

and μ the least transition relation on $K(Act)$ given by the rules

R0′: $\vdash\ \alpha \bullet p \xrightarrow{\alpha} p$

R5′: $p \xrightarrow{\alpha} p' \ \vdash\ (p+q) \xrightarrow{\alpha} p'$

R6′: $q \xrightarrow{\alpha} q' \ \vdash\ (p+q) \xrightarrow{\alpha} q'$

Then the Hennessy-Milner theorem roughly states

THEOREM. *Any state of a finite acyclic transition system on Act is denoted by a term of $K(Act)$. For such terms*

$$p \asymp_\mu q \quad \Leftrightarrow \quad p =_\Psi q$$

From this result, we just have to find a suitable translation from $T(A)$ to $K(Act)$ (that is an expansion of terms into finite acyclic transition systems) in order to solve our axiomatization problem. A first step is to extend our set of terms to $T'(A)$ which is built as $T(A)$ but with the additional formation rule:

(iii) if $\alpha \in D'(A)$ and $p \in T'(A)$ then $(\alpha \bullet p) \in T'(A)$.

where $D'(A)$ is the set of terms built from A using ; and $\|$ (without $\mathbb{1}$). We also extend the transition relation ρ to ρ' with the supplementary rule R0′ and adopt the previous convention for the meaning of $\bar{\rho}'$. Axiom A2 allows us to use an ambiguous notation $\sum_i p_i$ for a (finite) sum of terms. Then our axiomatization is as follows: let Φ be the (heterogeneous) theory whose axioms are those of Θ plus I and (omitting some parentheses)

$$\text{B1}: \quad a \bullet \mathbb{1} = a \text{ for } a \in A$$

$$\text{B2}: \quad (\sum_i \alpha_i \bullet p_i)\ ;q = \sum_i ((\alpha_i \bullet p_i)\ ; q)$$

$$\text{B3}: \quad (\alpha \bullet \mathbb{1})\ ; (\sum_j \beta_j \bullet q_j) = \alpha \bullet (\sum_j \beta_j \bullet q_j) + \sum_j (\alpha\ ; \beta_j) \bullet q_j$$

$$\text{B4}: \quad (\alpha \bullet (\sum_i \alpha_i \bullet p_i))\ ; q = \alpha \bullet (\sum_i (\alpha_i \bullet p_i)\ ; q)$$

$$\text{B5}: \quad (\sum_i \alpha_i \bullet p_i \parallel \sum_j \beta_j \bullet q_j) = \sum_i \alpha_i \bullet (p_i \parallel \sum_j \beta_j \bullet q_j) + \sum_{i,j} ((\alpha_i \parallel \beta_j) \bullet (p_i \parallel q_j)) + \sum_j \beta_j \bullet (\sum_i \alpha_i \bullet p_i \parallel q_j)$$

THEOREM 3. *The congruence of algebra $=_\Phi$ generated by Φ is invariant with respect to $\bar{\rho}'$. Moreover for each $p \in T(A)$ there exists an $r \in K(D'(A))$ such that $p =_\Phi r$. Therefore for $p, q \in T(A) \quad p \asymp q\ (\bar{\rho}) \quad \Leftrightarrow \quad p =_\Phi q$*

The first statement, which implies *soundness*, namely $p =_\Phi q \ \Rightarrow\ p \asymp q\ (\bar{\rho}')$, can be shown by a straightforward case inspection. For the second one, we can prove by induction on $p \in T'(A)$ that such a term is convertible by means of the given equations into a "*normal form*", which is here either $\mathbb{1}$ or a term $\sum_i \alpha_i \bullet p_i$ where each p_i is again a "normal form". A consequence is *completeness*: $p \asymp q\ (\bar{\rho}) \ \Rightarrow\ p =_\Phi q$ (note that Φ contains the equality theory Δ for the actions, which is needed to apply Hennessy-Milner theorem) ∎

One could have the idea that this result expresses a reduction of concurrency to sequential non-determinism; however this is not quite right, since actions are posets irreducibly involving parallelism. So the expansion theorem is not so bad. From a semantical point of view, the technique we used is still unsatisfactory since it gives no indication of how one could describe the equipollence classes of A-LES's. Nevertheless our purpose is achieved: we can prove semantic equalities of terms, such as the distributivity properties

$$\text{for each } p \not\asymp \mathbb{1} \text{ and } q \not\asymp \mathbb{1} \qquad (p+q)\,;r \asymp p\,;r + q\,;r$$

We can also prove

$$(a \parallel (b+c)) + (a \parallel b) + ((a+c) \parallel b) \asymp (a \parallel (b+c)) + ((a+c) \parallel b)$$

or other absorption phenomena (r is absorbed by p if $p + r \asymp p$, cf [5]). This example can be arbitrarily complicated (see [5]), so that the existence of a finite axiomatization without extending the syntax or introducing an absorption preorder is doubtful. Note: it can be proved that our equipollence is weaker than the notion of distributed bisimulation of [5]. M. Hennessy has found an example which proves that it is strictly weaker; namely

$$(a \parallel b+c) + a\,;(b+c) + (a \parallel b) + (a \parallel c) \quad \asymp \quad a\,;(b+c) + (a \parallel b) + (a \parallel c)$$

but these two equipollent terms are not d-bisimilar.

References

[1] G. BOUDOL, *Notes on Algebraic Calculi of Processes*, in Logics and Models of Concurrent Systems (K. Apt, Ed.) NATO ASI Series F13, Springer-Verlag (1985) 261-303.

[2] G. BOUDOL, I. CASTELLANI, *Concurrency and Communication*, full version of this paper, in preparation (1986).

[3] S. BROOKES, W.C. ROUNDS, *Behavioural Equivalence Relations Induced by Programming Logics*, ICALP 83, Lecture Notes in Comput. Sci. 154 (1983) 97-108.

[4] I. CASTELLANI, P. FRANCESCHI, U. MONTANARI, *Labelled Event Structures: a Model for Observable Concurrency*, in Formal Description of Programming Concepts 2 (D. Bjørner, Ed.), North-Holland (1983) 383-400.

[5] I. CASTELLANI, M. HENNESSY, *Distributed Bisimulations*, to be published (1985).

[6] Ph. DARONDEAU, L. KOTT, *On the Observational Semantics of Fair Parallelism*, ICALP 83, Lecture Notes in Comput. Sci. 154 (1983) 147-159.

[7] P. DEGANO, U. MONTANARI, *Distributed Systems, Partial Orderings of Events and Event Structures*, in Control Flow and Data Flow: Concepts of Distributed Programming (M. Broy, Ed.), NATO ASI Series F14, Springer-Verlag (1985) 7-106.

[8] P. DEGANO, R. DE NICOLA, U. MONTANARI, *Partial Ordering Derivations for CCS*, FCT 85, Lecture Notes in Comput. Sci. 199 (1985) 520-533.

[9] H.J. GENRICH, E. STANKIEWICZ-WIECHNO, *A Dictionary of Some Basic Notions of Net Theory*, in Net Theory and Applications (W. Brauer, Ed.) Lecture Notes in Comput. Sci. 84 (1980) 519-531.

[10] J.L. GISCHER, *Partial Orders and the Axiomatic Theory of Shuffle*, Ph. D. Thesis Stanford University (1984).

[11] U. GOLTZ, W. REISIG, *The Non-sequential Behaviour of Petri Nets*, Information and Control 57 (1983) 125-147.

[12] J. GRABOWSKI, *On Partial Languages*, Fundamentae Informaticae IV.2 (1981) 427-498.

[13] P.A. GRILLET, *Maximal Chains and Antichains*, Fund. Math. 65 (1969) 157-167.

[14] M. HENNESSY, R. MILNER, *Algebraic Laws for Nondeterminism and Concurrency*, JACM 32 (1985) 137-161.

[15] A. MAZURKIEWICZ, *Concurrent Program Schemes and their Interpretations*, Aarhus Workshop on Verification of Parallel Programs, Daimi PB-78, Aarhus University (1977).

[16] A. MAZURKIEWICZ, *Traces, Histories, Graphs: Instances of a Process Monoid*, MFCS 84, Lecture Notes in Comput. Sci. 176 (1984) 115-133.

[17] R. MILNER, *A Calculus of Communicating Systems*, Lecture Notes in Comput. Sci. 92 (1980).

[18] R. MILNER, *Calculi for Synchrony and Asynchrony*, Theoret. Comput. Sci. 25 (1983) 267-310.

[19] R. MILNER, *Lectures on a Calculus for Communicating Systems*, Seminar on Concurrency, Lecture Notes in Comput. Sci. 197 (1985) 197-220.

[20] M. NIELSEN, G. PLOTKIN, G. WINSKEL, *Petri Nets, Event Structures and Domains*, Theoret. Comput. Sci. 13 (1981) 85-108.

[21] D. PARK, *Concurrency and Automata on Infinite Sequences*, 5^{th} GI Conf. ,Lecture Notes in Comput. Sci. 104 (1981) 167-183.

[22] C. A. PETRI, *Non-sequential Processes*, GMD-ISF Rep. 77-05 (1977).

[23] G. PLOTKIN, *A Structural Approach to Operational Semantics*, Daimi FN-19, Aarhus University (1981).

[24] G. PLOTKIN, *An Operational Semantics for CSP*, in Formal Description of Programming Concepts 2 (D. Bjørner, Ed.), North-Holland (1983) 199-225.

[25] H. PLÜNNECKE, *K-Density, N-Density and Finiteness Properties*, in Advances in Petri Nets 84 (G. Rozenberg, Ed.) Lecture Notes in Comput. Sci. 188 (1984) 392-412.

[26] V. R. PRATT, *On the Composition of Processes*, 9^{th} POPL (1982) 213-223.

[27] V. R. PRATT, *The Pomset Model of Parallel Processes: Unifying the Temporal and the Spatial*, Seminar on Concurrency, Lecture Notes in Comput. Sci. 197 (1985) 180-196.

[28] W. REISIG, *On the Semantics of Petri Nets*, in Formal Models in Programming (G. Chroust, E. J. Neuhold, Eds.), North-Holland (1985) 347-372.

[29] M. W. SHIELDS, *Concurrent Machines*, The Computer Journal 28 (1985) 449-465.

[30] J. WINKOWSKI, *Algebras of Partial Sequences*, FCT 77, Lecture Notes in Comput. Sci. 56 (1977) 187-196.

[31] J. WINKOWSKI, *Behaviours of Concurrent Systems*, Theoret. Comput. Sci. 12 (1980) 39-60.

[32] G. WINSKEL, *Events in Computation*, Ph. D. Thesis, Edinburgh University (1980).

[33] G. WINSKEL, *Event Structure Semantics for CCS and Related Languages*, Daimi PB-159, Aarhus University (1983).

[34] G. WINSKEL, *Categories of Models for Concurrency*, Seminar on Concurrency, Lecture Notes in Comput. Sci. 197 (1985) 246-267.