Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

F.W. Vaandrager

A simple definition for parallel composition of prime event structures

Computer Science/Department of Software Technology Report CS-R8903 March

The Centre for Mathematics and Computer Science is a research institute of the Stichting Mathematisch Centrum, which was founded on February 11, 1946, as a nonprofit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands Organization for the Advancement of Research (N.W.O.).

A Simple Definition for Parallel Composition of Prime Event Structures

Frits W. Vaandrager
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

A simple, non-inductive construction is presented for parallel composition of prime event structures with binary conflict. It is shown that the construction determines the same operation as the categorical construction of WINSKEL [6] by proving that it is a product in the category of prime event structures with binary conflict.

1985 Mathematics Subject Classification: 68Q05, 68Q55.
1980 Mathematics Subject Classification: 68B10.
1982 CR Categories: F.1.1, F.3.3.
Key Words & Phrases: theory of concurrency, prime event structures, binary conflict, conflict event structures, parallel composition, categories of event structures, product, stable families, configurations.
Note: Partial support received from the European Communities under ESPRIT project no. 432, An Integrated Formal Approach to Industrial Software Development (METEOR).

1. INTRODUCTION

Prime event structures were introduced by NIELSEN, PLOTKIN & WINSKEL [4]. They form a simple and intuitive model of concurrent processes, on the one hand related to Petri nets and on the other hand related to finitary prime algebraic domains. However, many people think that prime event structures have one severe drawback: some operations, especially parallel composition, are difficult to define. WINSKEL, for instance, used this argument when he motivated the introduction of *stable event structures* ([6], p. 336). Stable event structures are certainly very interesting, but they are more complicated than prime event structures. The representation of domains by means of prime event structures is more direct. Furthermore stable event structures can be related to Petri nets only in an indirect manner.

One of the main applications of event structures is that they can be used to give semantics to CCS-like languages. As it turns out prime event structures with *binary conflict* (also called *conflict event structures (CES's)* by DEGANO, DE NICOLA & MONTANARI [2]), are appropriate for this purpose in principle. All CCS-like operations are easily definable on CES's except for parallel composition. If someone wants to compute the parallel composition of two CES's in the way proposed by WINSKEL, (s)he has to determine the families of configurations associated to these event structures, next turn the configurations into stable event structures, next compute the parallel composition of these stable event structures, then translate the result back to a family of configurations, and as a last step turn the configurations into a prime event structure again.

A number of authors have looked for more direct definitions of parallel composition. LOOGEN & GOLTZ [3] present a very complicated definition in the setting of TCSP. In [2], DEGANO, DE NICOLA & MONTANARI actually work out the inductive definition which WINSKEL did not give to 'avoid a messy inductive naming of events' ([5], p. 564). Presumably the definition in [2] is less messy than

Report CS-R8903
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

WINSKEL thought it would become, but it is certainly not simple. Furthermore one can argue that this definition, due to its inductive nature, is too 'operational'. Finally, we mention the solution of BOUDOL & CASTELLANI [1]. Like WINSKEL, they introduce a more general form of event structures, mainly because of trouble with parallel composition. But whereas the ingredients of a stable event structure bear no resemblance to the ingredients of a CES, the *flow event structures* of BOUDOL & CASTELLANI just have the same ingredients. The only difference with CES's is that certain restrictions on the ordering and conflict relations have been dropped. It is for instance no longer required that the ordering relation is a partial order. Every CES is a flow event structure and conversely, every flow event structure can be turned in a standard way into a CES. Advantages of the approach of BOUDOL & CASTELLANI are that the definitions of various operations become rather simple and that there is a close connection with Petri nets. A disadvantage is that it is difficult to have a good intuition about the behaviour of these structures: the relation with the domain of configurations is not so simple.

Summarising, one can say that many people have been intrigued by the question how to define parallel composition on CES's, but that a satisfactory solution has not been found. This brings us to the topic of this note, which is a simple, non-inductive definition of parallel composition on CES's. In length it is comparable to the length of the definition in [6] of parallel composition on stable event structures. What the construction essentially does is that it characterises the finite configurations of the product of two CES's and then builds a CES out of that by taking the primes.

In proving that our construction gives in fact the same operation as the existing constructions, we profit from the categorical setup of WINSKEL [6]. In [6] it is shown that parallel composition of CES's is a product in certain category $\underline{\mathbf{P}}^{\#}$. Since products are determined uniquely up to isomorphism, it is sufficient to show that our construction gives a product in this category. A proof of this fact is presented in section 3 of this note. The advantages of this way of proving correctness are that (1) it makes the paper self-contained in a simple way, and (2) it illustrates that CES's have an existence of their own and that one does not have to move to any other model of concurrency if one wants to prove something about them.

WINSKEL [7] generalised the result of section 3 to stable families of configurations using the characterisation of configurations of the product of stable families which he gave in [5, 6]. A sketch of WINSKEL's proof is included in section 4. This section leans heavily on the results of [6]; the observation that it generalises our result follows from the coreflection between the categories of prime event structures and stable families. Section 4 is included because it can give some additional insight to those readers who are acquainted with WINSKEL's work. No attempt has been made to make this part of the paper self-contained.

We would like to make a few remarks about the specific type of prime event structures for which we give our construction. Firstly, we work with prime event structures with *binary* conflict. However, as the result of section 4 shows, a generalisation to general prime event structures is trivial. Most readers will be more familiar with event structures with binary conflict. Moreover, the most important application of our construction will be to give semantics to CCS-like languages where conflict is always binary. A second choice that we made was to pay no attention to the issue of *labels*. Often event structures have as an additional ingredient a labelling function $l:E \rightarrow A$ which associates to each event a label in some action alphabet A. However, WINSKEL [5, 6] has convincingly shown that it is possible to separate the issue of labels from other important questions. Using the notion of *synchronisation algebra*, the construction of this paper generates a whole family of parallel composition operators for labelled event structures. This family contains the parallel combinators of CCS, CSP, MEIJE and ACP.

The introduction of stable event structures and flow event structures has been motivated with the argument that parallel composition is difficult to define on prime event structures. Stable event structures and flow event structures are very interesting and worth studying. However, this paper shows that one has to motivate them in a different way. One argument for stable event structures and flow event structures could be that in many cases they provide a better model of physical reality, in the sense that an entity which we would like to consider as a single event (like for instance the printing of a certain file) can indeed be modelled as a single event in this type of event structures. In prime event structures on the contrary one obtains a multitude of events for every possible history leading to such a 'real' event.

ACKNOWLEDGEMENTS
I would like to thank Gérard Boudol for encouraging me to write this note, and Glynn Winskel for allowing me to include his generalisation to stable families of my result.

2. CES's and Parallel Composition

2.1. DEFINITION. A *prime event structure with binary conflict* or *conflict event structure (CES)* is a triple $(E, \leqslant, \#)$, where

- E is a set of *events*;
- $\leqslant \subseteq E \times E$ is a partial order satisfying the principle of *finite causes*:

  $\{e' \in E \mid e' \leqslant e\}$ is finite for $e \in E$;

- $\# \subseteq E \times E$ is an irreflexive, symmetric relation (the *conflict relation*) satisfying the principle of *conflict heredity*:

  $e \# e' \leqslant e'' \Rightarrow e \# e''$;

As usual we write $e' < e$ for $e' \leqslant e$ & $e' \neq e$, $\geqslant$ for $\leqslant^{-1}$, and $>$ for $<^{-1}$. We write $\lceil e \rceil$ for the set $\{e' \in E \mid e' \leqslant e\}$. The components of a CES $\mathbf{E}$ will be denoted by respectively $E_{\mathbf{E}}$, $\leqslant_{\mathbf{E}}$ and $\#_{\mathbf{E}}$. The derived relations will be denoted $<_{\mathbf{E}}$, $\geqslant_{\mathbf{E}}$, $>_{\mathbf{E}}$. We also write $\lceil\ \rceil_{\mathbf{E}}$.

2.2. Graphical representation. In the graphical representation the partial order relation is indicated by arrows. We draw no arrow between events if, by means of the transitive and reflexive closure of the arrows, it can be deduced that they are ordered. The conflict relation is denoted by means of dotted lines. We draw no dotted line between events if conflict can be deduced by means of the principle of conflict heredity.

2.3. DEFINITION. Let $\mathbf{E}$ be a CES and let X be a subset of $E_{\mathbf{E}}$. We say that X is *left-closed* if:

$$e' \leqslant_{\mathbf{E}} e \in X \quad \Rightarrow \quad e' \in X.$$

X is *conflict-free* if X does not contain a pair of events which are in conflict, so if $\#_{\mathbf{E}} \cap (X \times X) = \varnothing$. A *configuration* of $\mathbf{E}$ is a left-closed, conflict-free subset of $E_{\mathbf{E}}$. By $\mathcal{C}(\mathbf{E})$ we denote the set of finite configurations of $\mathbf{E}$.

2.4. Notation. In this note we use partial functions. We indicate that θ is a partial function from E_0 to E_1 by writing $\theta: E_0 \rightarrow_* E_1$. Then it may not be the case that $\theta(e)$ is defined and sometimes we use $*$ to represent undefined, so $\theta(e) = *$ should be seen as a notation expressing that $\theta(e)$ is undefined. We adopt the convention that predicates are always defined (they either hold or do not hold) and are strict in the sense that they only hold if all their arguments are defined. The image of a set under a partial function is represented by:

$$\theta(X) = \{\theta(e) \mid e \in X \text{ and } \theta(e) \text{ is defined}\}.$$

2.5. DEFINITION (parallel composition). Let $\mathbf{E}_i = (E_i, \leqslant_i, \#_i)$ $(i=0,1)$ be CES's. Let the set of *pre-events* be defined as

$$E_0 \times_* E_1 = \{(e,*) \mid e \in E_0\} \cup \{(*,e) \mid e \in E_1\} \cup \{(e,e') \mid e \in E_0,\ e' \in E_1\}.$$

Let projections $\pi_i : E_0 \times_* E_1 \to_* E_i$ be given by $\pi_i((e_0,e_1)) = e_i$ for $i=0,1$. Call a subset X of $E_0 \times_* E_1$ a *preconfiguration* iff:

i) $\pi_0(X) \in \mathcal{C}(\mathbf{E}_0)$ and $\pi_1(X) \in \mathcal{C}(\mathbf{E}_1)$;

ii) $\leqslant_X$, the transitive closure of $\preccurlyeq \cap (X \times X)$, is a partial order, where $\preccurlyeq \subseteq E_0 \times_* E_1$ is defined by:

$$e \preccurlyeq e' \quad \Leftrightarrow \quad \pi_0(e) \leqslant_0 \pi_0(e') \text{ or } \pi_1(e) \leqslant_1 \pi_1(e').$$

If X moreover has a unique maximal element w.r.t. $\leqslant_X$, then X is called a *complete prime*. The *product* or *parallel composition* $\mathbf{E}_0 \times \mathbf{E}_1$ is the structure $(E, \leqslant, \#)$ with:

$$E = \{X \mid X \text{ is a complete prime}\},$$

$$X \leqslant Y \Leftrightarrow X \subseteq Y,$$

$$X \# Y \Leftrightarrow X \cup Y \text{ is not a preconfiguration.}$$

Let us try to give some intuition for this definition. The basic idea of the construction is that we first characterise (what essentially are) the finite configurations of the product event structure. If $\mathbf{E}_0$ and $\mathbf{E}_1$ execute in parallel, then events of $\mathbf{E}_0$ and $\mathbf{E}_1$ can either occur in isolation (this corresponds to pre-events of the form $(e,*)$ resp. $(*,e)$), or an event of $\mathbf{E}_0$ can synchronise with an event of $\mathbf{E}_1$ (in which case we have a pre-event of the form (e,e')). Now at any moment during execution of $\mathbf{E}_0 \times \mathbf{E}_1$ a set of pre-events has occurred. The notion of preconfiguration gives a characterisation of these sets. Condition (i) says that if we project a preconfiguration onto one of the two components, the result must denote a state (finite configuration) of this component. Condition (ii) says that the events of the component may occur only once and that both components must agree on the causal relations between events in the parallel composition: causal loops are not allowed. Once we have characterised the finite configurations of the product event structure, there exists a standard procedure to turn this into a CES. There is a natural choice of events *viz.* the complete primes. A complete prime corresponds to some pre-event together with the history which tells how this pre-event occurred.

2.6. PROPOSITION. *Let* $\mathbf{E}_i = (E_i, \leqslant_i, \#_i)$ $(i=0,1)$ *be CES's and let* $\mathbf{E} = (E, \leqslant, \#)$ *be the product of* $\mathbf{E}_0$ *and* $\mathbf{E}_1$. *Then* $\mathbf{E}$ *is a CES.*

PROOF. From the definition of product it is immediate that $\leqslant$ is a p.o. satisfying the principle of finite causes. We show that $\mathbf{E}$ satisfies the principle of conflict heredity. Let $X,Y,Z \in E$ with $X \# Y \leqslant Z$. Suppose that $\neg(X \# Z)$. This means that $X \cup Z$ is a preconfiguration. We will derive a contradiction by showing that $X \cup Y$ is a preconfiguration too. $\pi_0(X \cup Y)$ is finite and conflict-free since $\pi_0(X \cup Y) \subseteq \pi_0(X \cup Z) \in \mathcal{C}(\mathbf{E}_0)$. Further $\pi_0(X \cup Y)$ is left-closed since both $\pi_0(X)$ and $\pi_0(Y)$ are left-closed. Hence $\pi_0(X \cup Y) \in \mathcal{C}(\mathbf{E}_0)$. In the same way one can show that $\pi_1(X \cup Y) \in \mathcal{C}(\mathbf{E}_1)$. $\leqslant_{X \cup Y}$ is a p.o. because $\leqslant_{X \cup Z}$ is a p.o. Hence $X \cup Y$ is a preconfiguration and we have a contradiction. □

2.7. *Example.* As an example of our construction we consider the product of the following two CES's.

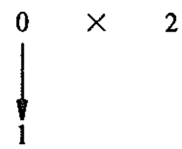

FIGURE 1.

In figure 2 below, the pre-events associated to this pair of CES's are depicted. With arrows the relation $\preccurlyeq$ is indicated. We did not draw the arrow from a pre-event to itself; note that the relation $\preccurlyeq$ is always reflexive.

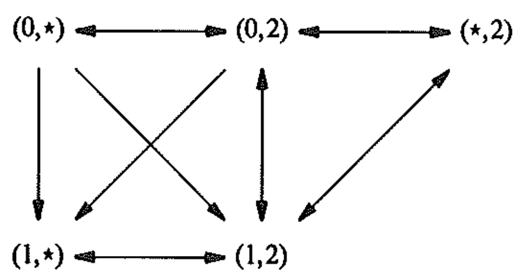

FIGURE 2.

Now the preconfigurations are given in figure 3. Here an arrow denotes extension of a preconfiguration with a single element. The complete primes are the preconfigurations with exactly one incoming arrow.

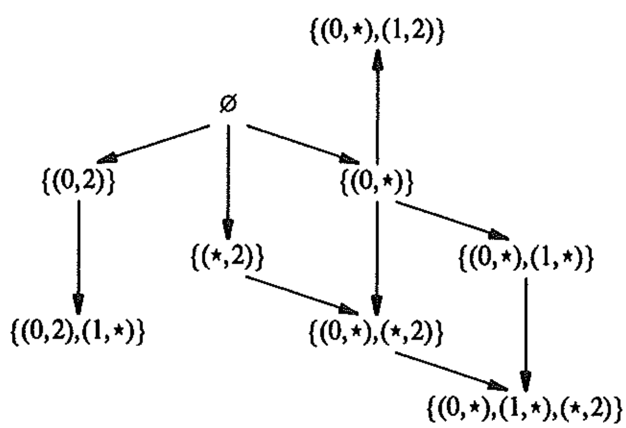

FIGURE 3.

It is trivial to go from figure 3 to figure 4 which depicts the product event structure: conflict is introduced between those complete primes for which there is no preconfiguration which is a superset of both; next all the preconfigurations which are not complete primes are dropped including the associated arrows.

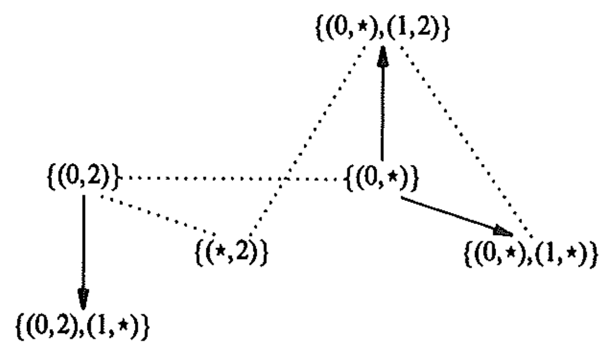

FIGURE 4.

2.8. LEMMA. *Let* $\mathbf{E}_i=(E_i, \leqslant_i, \#_i)$ $(i=0,1)$ *be CES's and let* $\mathbf{E}=(E, \leqslant, \#)$ *be the product of* $\mathbf{E}_0$ *and* $\mathbf{E}_1$. *Let* $max: E \rightarrow E_0 \times_* E_1$ *be the function that associates to each complete prime X its unique maximal element w.r.t.* $\leqslant_X$. *Let X be a preconfiguration of* $E_0 \times_* E_1$.
i) *If* $e \in X$ *then* $Y = \{e' \in X \mid e' \leqslant_X e\} \in E$ *and* $max(Y)=e$,
ii) *If* $Y \subseteq X$ *and* $Y \in E$ *then* $Y = \{e' \in X \mid e' \leqslant_X max(Y)\}$.
PROOF.
(i) It is not hard to see that $\leqslant_Y = \leqslant_X \cap (Y \times Y)$. Hence $\leqslant_Y$ is a p.o. and Y has e as a unique maximal element w.r.t. $\leqslant_Y$. Since $\pi_0(X)$ is finite and conflict-free, $\pi_0(Y)$ is finite and conflict-free too. Let $f \in Y$ and $g \in E_0$ with $g <_0 \pi_0(f)$. In order to prove that $\pi_0(Y)$ is left-closed, we must prove $g \in \pi_0(Y)$. Since $\pi_0(f) \in \pi_0(X)$ and $\pi_0(X)$ is left-closed, $g \in \pi_0(X)$. Hence, there exists an $f' \in X$ such that $\pi_0(f')=g$. Because $\pi_0(f') <_0 \pi_0(f)$, $f' \leqslant_X f$. Thus $f' \in Y$ and $g \in \pi_0(Y)$. We have now shown that $\pi_0(Y) \in \mathcal{C}(\mathbf{E}_0)$. Analogously one can prove that $\pi_1(Y) \in \mathcal{C}(\mathbf{E}_1)$.
(ii) Let $e' \in Y$. Then $e' \leqslant_Y max(Y)$. Since $Y \subseteq X$ we have $\leqslant_Y \subseteq \leqslant_X \cap (Y \times Y)$. Hence $e' \leqslant_X max(Y)$. Now conversely, let $e' \in X$ with $e' \leqslant_X max(Y)$. We must prove $e' \in Y$. There exists an $n > 0$ and $e_0, \ldots, e_n \in X$ with:

$$e' = e_0 \lessdot \cdots \lessdot e_n = max(Y).$$

Suppose $e' \notin Y$. Then $e_i \notin Y$ and $e_{i+1} \in Y$ for some $i < n$. By definition of $\lessdot$, either $\pi_0(e_i) \leqslant_0 \pi_0(e_{i+1})$ or $\pi_1(e_i) \leqslant_1 \pi_1(e_{i+1})$. It cannot be that $\pi_0(e_i)=\pi_0(e_{i+1})$ or $\pi_1(e_i)=\pi_1(e_{i+1})$ because this would imply $e_i \lessdot e_{i+1}$ and $e_{i+1} \lessdot e_i$ even though $e_i \neq e_{i+1}$ which is in contradiction with the fact that $\leqslant_X$ is a p.o. It is also not possible that $\pi_0(e_i) <_0 \pi_0(e_{i+1})$. Since $\pi_0(e_{i+1}) \in \pi_0(Y)$ and $\pi_0(Y) \in \mathcal{C}(\mathbf{E}_0)$ (and thus left-closed) this would imply $\pi_0(e_i) \in Y$. Consequently there would be an f in Y with $\pi_0(f)=\pi_0(e_i)$. But this is again in contradiction with the fact that $\leqslant_X$ is a p.o. For the same reason it is not possible that $\pi_1(e_i) <_1 \pi_1(e_{i+1})$. □

2.9. PROPOSITION. *Let* $\mathbf{E}_i=(E_i, \leqslant_i, \#_i)$ $(i=0,1)$ *be CES's and let* $\mathcal{C}_{pre}$ *be the set of preconfigurations of* $E_0 \times_* E_1$. *Let* $\mathbf{E}=(E, \leqslant, \#)$ *be the product of* $\mathbf{E}_0$ *and* $\mathbf{E}_1$. *Let for* $X \in \mathcal{C}_{pre}$, $f(X)$ *be given by:*

$$f(X) = \{\{e' \in X \mid e' \leqslant_X e\} \mid e \in X\}$$

Then $f(X) \in \mathcal{C}(\mathbf{E})$ *and* $f: \mathcal{C}_{pre} \rightarrow \mathcal{C}(\mathbf{E})$ *is a bijection.*
PROOF. Routine and left to the reader. □

3. Categorical Characterisation

3.1. Definition. A *morphism* between CES's $(E_0, \leqslant_0, \#_0)$ and $(E_1, \leqslant_1, \#_1)$ is a partial function $\theta: E_0 \rightarrow_* E_1$ such that:

$$\forall e \in E_0\ \forall f \in E_1 \,.\, f <_1 \theta(e) \;\Rightarrow\; \exists e' \in E_0 \,.\, e' <_0 e \text{ and } \theta(e')=f \quad \& \tag{a}$$

$$\forall e,e' \in E_0 \,.\, \theta(e) \#_1 \theta(e') \text{ or } \theta(e)=\theta(e') \;\Rightarrow\; e \#_0 e' \text{ or } e=e'. \tag{b}$$

The next proposition shows that the above notion of morphism is in fact the same as the one used by Winskel [6] (see proposition 3.4.7).

3.2. Proposition. *Let $\mathbf{E}_i=(E_i, \leqslant_i, \#_i)$ $(i=0,1)$ be CES's and let $\theta: E_0 \rightarrow_* E_1$ be a partial function. Then θ is a morphism iff*

$$\forall e \in E_0 \,.\, \theta(e) \textit{ is defined} \;\Rightarrow\; \lceil \theta(e) \rceil_{\mathbf{E}_1} \subseteq \theta(\lceil e \rceil_{\mathbf{E}_0}) \;\; \& \tag{i}$$

$$\forall e,e' \in E_0 \,.\, \theta(e) \#_1 \theta(e') \textit{ or } \theta(e)=\theta(e') \;\Rightarrow\; e \#_0 e' \textit{ or } e=e'. \tag{ii}$$

Proof.
"$\Rightarrow$" Suppose θ is a morphism. Then condition (ii) is trivially satisfied. In order to see that condition (i) holds, let $e \in E_0$ with $\theta(e)$ defined. Then:

$$\lceil \theta(e) \rceil_{\mathbf{E}_1} = \{f \mid f <_1 \theta(e)\} \cup \{\theta(e)\} \subseteq \{\theta(e') \mid e' <_0 e\} \cup \{\theta(e)\} = \theta(\lceil e \rceil_{\mathbf{E}_0}).$$

"$\Leftarrow$" Suppose θ satisfies conditions (i) and (ii). Condition (b) is clearly satisfied. We prove condition (a). Let $e \in E_0$, $f \in E_1$ with $f <_1 \theta(e)$. Then $f \in \lceil \theta(e) \rceil_{\mathbf{E}_1}$. Hence $f \in \theta(\lceil e \rceil_{\mathbf{E}_0})$ and there exists an $e' \in E_0$ such that $e' <_0 e$ and $\theta(e')=f$. □

3.3. Proposition. *CES's with morphisms of event structures form a category with the usual composition of partial functions as composition and the identity functions on events as identity morphisms.*
Proof. Straightforward. □

We write $\underline{\mathbf{P}}^{\#}$ for the category of CES's with morphisms of CES's. At first sight, the notion of morphism in $\underline{\mathbf{P}}^{\#}$ looks complicated and not intuitive. The next proposition however supports the view that in fact these morphisms are fundamental and also rather natural.

3.4. Proposition. *Let $\mathbf{E}_i=(E_i, \leqslant_i, \#_i)$ $(i=0,1)$ be CES's, let $\theta: E_0 \rightarrow_* E_1$ be a morphism and let X be a configuration of $\mathbf{E}_0$. Then $\theta(X)$ is a configuration of $\mathbf{E}_1$.*
Proof. $\theta(X)$ is conflict-free because if for some $e,e' \in X$ we would have $\theta(e) \#_1 \theta(e')$, this would imply $e \#_0 e'$ which is a contradiction since X is conflict-free. Let $e \in X$ and $f \in E_1$ with $f <_1 \theta(e)$. Then there exists an $e' \in E_0$ with $e' <_0 e$ and $\theta(e')=f$. Since X is left-closed we have $e' \in X$. Hence $f \in \theta(X)$ and thus $\theta(X)$ is left-closed. □

3.5. Proposition. *Let $\mathbf{E}_i=(E_i, \leqslant_i, \#_i)$ $(i=0,1)$ be CES's and let $\mathbf{E}=(E, \leqslant, \#)$ be the product of $\mathbf{E}_0$ and $\mathbf{E}_1$. Define the partial functions $\phi_i: E \rightarrow_* E_i$ $(i=0,1)$ by:*

$$\phi_i(X) = \pi_i(max(X)).$$

Then ϕ_0 and ϕ_1 are morphisms.
Proof. For reasons of symmetry it is enough to prove that ϕ_0 is a morphism, i.e. that condition (a) and (b) hold in 3.1.
(a) Let $X \in E$ and $f \in E_0$ with $f <_0 \phi_0(X)$. X is a preconfiguration so $\pi_0(X) \in \mathcal{C}(\mathbf{E}_0)$. Since $\phi_0(X) \in \pi_0(X)$ and $\pi_0(X)$ is left-closed, $f \in \pi_0(X)$. So there exists a $g \in X$ with $\pi_0(g)=f$. By lemma

2.8(i) $Y = \{g' \in X \mid g' \leqslant_X g\} \in E$ and $max(Y) = g$. Clearly $Y < X$. Furthermore $\phi_0(Y) = f$.
(b) Let $X, Y \in E$ with $\phi_0(X) \#_0 \phi_0(Y)$. Since both $\phi_0(X)$ and $\phi_0(Y)$ are in $\pi_0(X \cup Y)$, $\pi_0(X \cup Y) \notin \mathcal{C}(\mathbf{E}_0)$. Hence $X \cup Y$ is not a preconfiguration and thus $X \# Y$. Now suppose $X, Y \in E$ with $X \neq Y$ and $\phi_0(X) = \phi_0(Y)$. We show that $X \# Y$ by proving that $X \cup Y$ is not a preconfiguration. W.l.o.g. we may assume that there is some $f \in X - Y$. If $max(X) \neq max(Y)$ we are ready: $X \cup Y$ cannot be a preconfiguration since $max(X) \preccurlyeq max(Y)$ and $max(Y) \preccurlyeq max(X)$. So let us assume that $max(X) = max(Y) = e$. Since $f \leqslant_X e$, there is an $n > 0$ and there are $e_0, \ldots, e_n \in X$ such that:

$$f = e_0 \preccurlyeq \cdots \preccurlyeq e_n = e.$$

Now $f \notin Y$ and $e \in Y$. Hence, for some $i < n$, $e_i \notin Y$ and $e_{i+1} \in Y$.
If $\pi_0(e_i) = \pi_0(e_{i+1})$ then we have a contradiction since then $e_i \preccurlyeq e_{i+1}$ and $e_{i+1} \preccurlyeq e_i$ so $\leqslant_X$ cannot be a p.o. In the same way we can derive a contradiction if $\pi_1(e_i) = \pi_1(e_{i+1})$.
Assume $\pi_0(e_i) <_0 \pi_0(e_{i+1})$. Since $\pi_0(Y) \in \mathcal{C}(\mathbf{E}_0)$ is left-closed there is a $g \in Y$ with $\pi_0(g) = \pi_0(e_i)$. But then $g \preccurlyeq e_i$ as well as $e_i \preccurlyeq g$. This means that $\leqslant_{X \cup Y}$ is not a p.o. and hence $X \cup Y$ is not a preconfiguration. The case $\pi_1(e_i) <_1 \pi_1(e_{i+1})$ is similar. □

3.6. Theorem. *Let $\mathbf{E}_i = (E_i, \leqslant_i, \#_i)$ $(i = 0,1)$ be CES's. The product $\mathbf{E}_0 \times \mathbf{E}_1$, with projections ϕ_0 and ϕ_1 as in proposition 3.5, is a product in the category $\underline{\mathbf{P}}^{\#}$.*

Proof. By proposition 2.6 $\mathbf{E}_0 \times \mathbf{E}_1$ is a CES, which we shall assume is $\mathbf{E} = (E, \leqslant, \#)$. By proposition 3.5 ϕ_0 and ϕ_1 are morphisms. Assume $\theta_0 : E' \to_* E_0$ and $\theta_1 : E' \to_* E_1$ are morphisms from a CES $\mathbf{E}' = (E', \leqslant', \#')$. In order to have a product, we require that there is a unique morphism $\theta : E' \to_* E$ making the following diagram commute:

$$\begin{tikzcd}
& \mathbf{E}_0 \times \mathbf{E}_1 \arrow[dl, "\phi_0"'] \arrow[dr, "\phi_1"] & \\
\mathbf{E}_0 & & \mathbf{E}_1 \\
& \mathbf{E}' \arrow[ul, "\theta_0"] \arrow[ur, "\theta_1"'] \arrow[uu, dotted, "\theta"'] &
\end{tikzcd}$$

Figure 5.

For the definition of morphism θ we need two auxiliary functions. The partial function $\theta_{pre} : E' \to_* E_0 \times_* E_1$ is given by:

$$\theta_{pre}(e) = \begin{cases} * & \text{if } \theta_0(e) = * \text{ and } \theta_1(e) = * \\ (\theta_0(e), \theta_1(e)) & \text{otherwise} \end{cases}$$

Claim 1. *Let $e, e' \in E'$ with $\neg(e \#' e')$ and $\theta_{pre}(e) \prec \theta_{pre}(e')$. Then $e <' e'$.*
Proof. By definition of $\preccurlyeq$ either $\theta_0(e) \leqslant_0 \theta_0(e')$ or $\theta_1(e) \leqslant_1 \theta_1(e')$. Since θ_0 is a morphism $\theta_0(e) = \theta_0(e')$ implies $e \#' e'$ or $e = e'$, which is a contradiction. In the same way one can prove that $\theta_1(e) \neq \theta_1(e')$. Now suppose $\theta_0(e) <_0 \theta_0(e')$. Because θ_0 is a morphism there is an $e'' \in E'$ such that $e'' <' e'$ and $\theta_0(e'') = \theta_0(e)$. Again using that θ_0 is a morphism we obtain that either $e'' \#' e$ or $e'' = e$. Since $\neg(e \#' e')$ and $\mathbf{E}'$ satisfies conflict heredity, it is not possible that $e'' \#' e$. Hence $e'' = e$ and thus $e <' e'$. Similarly we can derive that $\theta_1(e) <_1 \theta_1(e')$ implies $e <' e'$. □

Next we define another auxiliary partial function $\Theta : E' \to_* Pow(E_0 \times_* E_1)$ by:

$$\Theta(e) = \begin{cases} * & \text{if } \theta_{pre}(e) = * \\ \theta_{pre}(\lceil e \rceil_{\mathbf{E}'}) & \text{otherwise} \end{cases}$$

CLAIM 2. *Let $e \in E'$ with $\Theta(e)$ defined. Then $\Theta(e)$ is a preconfiguration of $E_0 \times_* E_1$.*
PROOF. Since

$$\pi_0(\Theta(e)) = \pi_0(\{(\theta_0(e'), \theta_1(e')) \mid e' \leqslant' e\}) = \theta_0(\lceil e \rceil_{\mathbf{E}'})$$

and $\lceil e \rceil_{\mathbf{E}'} \in \mathcal{C}(\mathbf{E}')$ and since by proposition 3.4 finite configurations are preserved under morphisms, we have $\pi_0(\Theta(e)) = \theta_0(\lceil e \rceil_{\mathbf{E}'}) \in \mathcal{C}(\mathbf{E}_0)$. Similarly one can show that $\pi_1(\Theta(e)) \in \mathcal{C}(\mathbf{E}_1)$.
Next we prove that $\leqslant_{\Theta(e)}$ is a p.o. Suppose it is not. Then $\Theta(e)$ must have a causal loop, i.e. for some $n > 0$ and $e_0, \ldots, e_n \in \lceil e \rceil_{\mathbf{E}'}$:

$$\theta_{pre}(e_0) \prec \theta_{pre}(e_1) \prec \cdots \prec \theta_{pre}(e_n) \prec \theta_{pre}(e_0).$$

But now application of claim 1, combined with the fact that $\leqslant'$ is a p.o., gives a contradiction. □

We are now in a position to define the morphism θ:

$$\theta(e) = \begin{cases} * & \text{if } \theta_{pre}(e) = * \\ \{f \in \Theta(e) \mid f \leqslant_{\Theta(e)} \theta_{pre}(e)\} & \text{otherwise} \end{cases}$$

CLAIM 3. *θ is a morphism from E' to E.*
PROOF. From claim 2 and lemma 2.8(i) we can conclude directly that θ is a partial function from E' to E. Let $e' \in E'$ and $X \in E$ with $X \subset \theta(e)$. In order to prove that θ satisfies condition (a) of 3.1 we have to find an $e' \in E'$ such that $e' <' e$ and $\theta(e') = X$. From claim 2 and lemma 2.8(ii) we conclude:

$$X = \{f \in \Theta(e) \mid f \leqslant_{\Theta(e)} max(X)\}.$$

Using that $X \subset \theta(e)$ we find that there exists an $e' <' e$ such that $\theta_{pre}(e') = max(X)$. Clearly $\theta(e') = X$.
Next we prove that θ satisfies condition (b) of 3.1. Let $e, e' \in E'$. We distinguish between two cases:
(1) $e \neq e'$ and $\theta(e) = \theta(e')$. We prove that $e \ \#' \ e'$. W.l.o.g. we may assume that there exists an $f \in \lceil e \rceil_{\mathbf{E}'} - \lceil e' \rceil_{\mathbf{E}'}$. Since $\theta_{pre}(f) \in \theta(e)$ and $\theta(e) = \theta(e')$ there exists an $f' \in \lceil e' \rceil_{\mathbf{E}'}$ with $\theta_{pre}(f) = \theta_{pre}(f')$. This means that $\theta_0(f) = \theta_0(f')$ or $\theta_1(f) = \theta_1(f')$. Because θ_0 and θ_1 are morphisms this implies $f \ \#' \ f'$. Applying conflict heredity twice gives $e \ \#' \ e'$.
(2) $\theta(e) \ \# \ \theta(e')$. We prove $e \ \#' \ e'$. Write $\theta(e) = X$ and $\theta(e') = Y$. So $X \cup Y$ is not a preconfiguration of $E_0 \times_* E_1$. We distinguish between three cases:

2.1. $\pi_0(X \cup Y) \notin \mathcal{C}(\mathbf{E}_0)$. $\pi_0(X \cup Y)$ is left-closed because $\pi_0(X)$ and $\pi_0(Y)$ are left-closed. So it must be that $\pi_0(X \cup Y)$ is not conflict-free. Thus we can find $f, f' \in \pi_0(X \cup Y)$ with $f \ \#_0 \ f'$. W.l.o.g. we may assume that there exist $g, g' \in E'$ with $g \leqslant' e$ and $g' \leqslant' e'$ satisfying $\theta_0(g) = f$ and $\theta_0(g') = f'$. Now use that θ_0 is a morphism to obtain $g \ \#' \ g'$. Applying conflict heredity twice gives $e \ \#' \ e'$.
2.2. $\pi_1(X \cup Y) \notin \mathcal{C}(\mathbf{E}_1)$. This case is similar to case 2.1.
2.3. $\leqslant_{X \cup Y}$ is not a p.o. Then for some $n > 1$ and $e_0, \ldots, e_n \in \lceil e \rceil_{\mathbf{E}'} \cup \lceil e' \rceil_{\mathbf{E}'}$ we have $e_0 = e_n$ and

$$\theta_{pre}(e_0) \prec \theta_{pre}(e_1) \prec \cdots \prec \theta_{pre}(e_n).$$

If, for some $i < n$, $e_i \ \#' \ e_{i+1}$ we are done: conflict heredity gives $e \ \#' \ e'$ in that case. Now this situation occurs since otherwise we can derive a contradiction using claim 1.

This finishes the proof of claim 3. □

It is obvious that θ makes the diagram of figure 5 commute. Suppose that $\theta' : E' \to_* E$ is also a morphism that makes the diagram commute. We prove $\theta = \theta'$. The requirement that the diagram should commute immediately gives:

$$\theta'(e) = * \iff \theta_{pre}(e) = * \iff \theta(e) = *, \tag{1}$$

$$\theta'(e) \text{ is defined} \quad \Rightarrow \quad max(\theta'(e)) = \theta_{pre}(e). \tag{2}$$

Now let $e \in E'$ with $\theta'(e)$ defined. We show $\theta'(e) \subseteq \Theta(e)$. Let $f \in \theta'(e)$. If $f = \theta_{pre}(e)$ then clearly $f \in \Theta(e)$ so assume $f \neq \theta_{pre}(e)$. By lemma 2.8(i) $Y = \{f' \in \theta'(e) \mid f' \leqslant_{\theta'(e)} f\} \in E$ and $max(Y) = f$. Since $Y \subset \theta'(e)$ and θ' is a morphism there exists an $e' \in E'$ with $e' <' e$ and $\theta'(e') = Y$. But then:

$$f = max(Y) = max(\theta'(e')) = \theta_{pre}(e') \in \Theta(e).$$

Now apply claim 2 and lemma 2.8(ii) to obtain: $\theta'(e) = \{e' \in \Theta(e) \mid e' \leqslant_{\Theta(e)} \theta_{pre}(e)\} = \theta(e)$. This concludes the proof of theorem 3.6. □

4. Generalisation to Stable Families

In this section we present a generalisation of theorem 3.6 to stable families of configurations. This result was kindly made available to us by Winskel [7]. First we recall some basic definitions.

4.1. Definition. Let $\mathbf{F}$ be a family of subsets of a set E. Say a subset X of $\mathbf{F}$ is *compatible*, and write $X\uparrow$, iff $\exists y \in \mathbf{F}\ \forall x \in X.\ x \subseteq y$. Write $X\uparrow^{fin}$ when every finite subset of X is compatible.

4.2. Definition. Let $\mathbf{F}$ be a family of subsets of a set E. Say $\mathbf{F}$ is a *stable family of configurations on* E when it satisfies:

(i) *finite-completeness*: $\forall X \subseteq \mathbf{F}.\ X\uparrow^{fin} \Rightarrow \bigcup X \in \mathbf{F}$,
(ii) *stability*: $\forall X \subseteq \mathbf{F}.\ X \neq \varnothing\ \&\ X\uparrow \Rightarrow \bigcap X \in \mathbf{F}$,
(iii) *finiteness*: $\forall x \in \mathbf{F}\ \forall e \in x\ \exists y \in \mathbf{F}.\ (y \text{ is finite } \&\ e \in y\ \&\ y \subseteq x)$,
(iv) *coincidence-freeness*: $\forall x \in \mathbf{F}\ \forall e,e' \in x.\ e \neq e' \Rightarrow \exists y \in \mathbf{F}.\ y \subseteq x\ \&\ (e \in y \Leftrightarrow e' \notin y)$.

4.3. Definition. Let $\mathbf{F}$ be a stable family of configurations on a set E. Let x be a configuration. For $e,e' \in x$ define

$$e' \leqslant_x e \quad \Leftrightarrow \quad \forall y \in \mathbf{F}.\ e \in y\ \&\ y \subseteq x \Rightarrow e' \in y.$$

When $e \in x$ define

$$\lceil e \rceil_x = \bigcap \{y \in \mathbf{F} \mid e \in y\ \&\ y \subseteq x\}.$$

4.4. Proposition. *Let x be a configuration of a stable family* $\mathbf{F}$. *Then $\leqslant_x$ is a partial order and $\lceil e \rceil_x$ is a configuration such that*

$$\lceil e \rceil_x = \{e' \in x \mid e' \leqslant_x e\}.$$

Moreover the configurations $y \subseteq x$ are exactly the left-closed subsets of $\leqslant_x$.
Proof. See [6], proposition 1.2.6. □

Recall the definition of configurations of the product of stable families from [6] (there stated as theorem 2.5.10).

4.5. Definition. Let $\mathbf{F}_0, \mathbf{F}_1$ be stable families of configurations on E_0, E_1. Then $x \in \mathbf{F}_0 \times \mathbf{F}_1$ iff $x \subseteq E_0 \times_* E_1$ and

(a) $\pi_0(x) \in \mathbf{F}_0$ and $\pi_1(x) \in \mathbf{F}_1$;
(b) $\forall e,e' \in x.\ \pi_0(e) = \pi_0(e')$ or $\pi_1(e) = \pi_1(e') \Rightarrow e = e'$;
(c) $\forall e,e' \in x.\ e \neq e' \Rightarrow \exists y \subseteq x.\ \pi_0(y) \in \mathbf{F}_0\ \&\ \pi_1(y) \in \mathbf{F}_1\ \&\ (e \in y \Leftrightarrow e' \notin y)$;
(d) $\forall e \in x\ \exists y \subseteq x.\ \pi_0(y) \in \mathbf{F}_0$ and $\pi_1(y) \in \mathbf{F}_1\ \&\ e \in y\ \&\ |y| < \infty$.

4.6. PROPOSITION (Winskel). *Let* $\mathbf{F}_0,\mathbf{F}_1$ *be stable families of configurations on* E_0,E_1 *and let* $x \subseteq E_0 \times_* E_1$ *be finite. Then* x *is a finite configuration in* $\mathbf{F}_0 \times \mathbf{F}_1$ *iff*

i) $\pi_0(x) \in \mathbf{F}_0$ *and* $\pi_1(x) \in \mathbf{F}_1$*;*

ii) $\preccurlyeq$*, the transitive closure of* $\preccurlyeq_1$*, is a partial order, where*

$$e \preccurlyeq_1 e' \quad \Leftrightarrow \quad \pi_0(e) \leqslant_{\pi_0(x)} \pi_0(e') \text{ or } \pi_1(e) \leqslant_{\pi_1(x)} \pi_1(e').$$

PROOF.

"$\Leftarrow$" We verify (a)-(d). Part (a) is immediate. Part (b) trivially follows from the fact that $\preccurlyeq$ is a partial order. In order to verify part (c) we need the following claim.

CLAIM. *Let* $e \in x$ *and* $[e] =_{def} \{e' \in x \mid e' \preccurlyeq e\}$. *Then* $\pi_0([e]) \in \mathbf{F}_0$ *and* $\pi_1([e]) \in \mathbf{F}_1$.

PROOF. By symmetry we only have to show $\pi_0([e]) \in \mathbf{F}_0$. Let $f \in \pi_0([e])$ and $f' \in \pi_0(x)$ such that $f' \leqslant_{\pi_0(x)} f$. Then there are $g,g' \in x$ and $g \preccurlyeq e$ with $\pi_0(g)=f$ and $\pi_0(g')=f'$. But this implies $g' \preccurlyeq g$ and, since $\preccurlyeq$ is a partial order, also $g' \preccurlyeq e$. Thus $g' \in [e]$ and hence $f'=\pi_0(g') \in \pi_0([e])$. Now we have shown that $\pi_0([e])$ is a $\leqslant_{\pi_0(x)}$-left-closed subset of configuration $\pi_0(x)$. Application of proposition 4.4 gives that therefore $\pi_0([e])$ is itself a configuration. □

Now we can verify part (c). Suppose $e,e' \in x$ with $e \neq e'$. Assume $\neg(e \preccurlyeq e')$. In this case take $y=[e']$. By the above claim $y \subseteq x$, $\pi_0(y) \in \mathbf{F}_0$, $\pi_1(y) \in \mathbf{F}_1$, $e \notin y$ and $e' \in y$. If, on the other hand, $e \preccurlyeq e'$ then take $y=[e]$. We then have $y \subseteq x$, $\pi_0(y) \in \mathbf{F}_0$, $\pi_1(y) \in \mathbf{F}_1$, $e \in y$ and $e' \notin y$.

Part (d) is immediate, since set x is finite.

"$\Rightarrow$" Suppose x is a finite configuration of $\mathbf{F}_0 \times \mathbf{F}_1$. We require that $\preccurlyeq$ is a p.o. Relation $\preccurlyeq$ is transitive by definition and reflexive because $\preccurlyeq_1$ is so. Thus we only need to show antisymmetry. However we observe:

$$e \preccurlyeq e' \quad \Rightarrow \quad e \leqslant_x e' \tag{*}$$

So any nontrivial loop in $\preccurlyeq$ would induce one in the partial order $\leqslant_x$, which is absurd and hence $\preccurlyeq$ is a partial order. To show (*) it suffices to show:

$$\pi_i(e) \leqslant_{\pi_i(x)} \pi_i(e') \quad \Rightarrow \quad e \leqslant_x e' \quad \text{for } i=0,1.$$

By symmetry we need only do this for $i=0$.

Suppose $\pi_0(e) \leqslant_{\pi_0(x)} \pi_0(e')$. We require $e \leqslant_x e'$, i.e. $\forall y \in \mathbf{F}_0.\ e' \in y \ \&\ y \subseteq x \Rightarrow e \in y$. Let $y \in \mathbf{F}_0$ with $e' \in y \subseteq x$. Then:

$$\pi_0(e) \leqslant_{\pi_0(x)} \pi_0(e') \in \pi_0(y) \subseteq \pi_0(x)$$

where $\pi_0(y)$ is a subconfiguration of the configuration $\pi_0(x)$. Thus by proposition 4.4 $\pi_0(e) \in \pi_0(y)$. Thus there is some $e'' \in y$ such that $\pi_0(e'')=\pi_0(e)$. But x is a configuration, so $e=e'' \in y$. □

Theorem 3.6 now follows from the coreflection between the categories of prime event structures and stable families as established in [6].

REFERENCES

[1] G. BOUDOL & I. CASTELLANI (1988): *Permutation of transitions: an event structure semantics for CCS and SCCS.* Rapports de Recherche 798, INRIA, Centre Sophia Antipolis, to appear in: Proceedings REX School/Workshop on Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency, Noordwijkerhout, (J.W. de Bakker, W.-P. de Roever & G. Rozenberg, eds.), LNCS.

[2] P. DEGANO, R. DE NICOLA & U. MONTANARI (1988): *On the consistency of 'truly concurrent' operational and denotational semantics.* In: Proceedings Symposium on Logic in Computer Science (LICS), Edinburgh, pp. 133-141.

[3] R. LOOGEN & U. GOLTZ (1987): *A non-interleaving semantic model for nondeterministic concurrent processes.* Aachener Informatik-Berichte 87-15, RWTH Aachen.

[4] M. NIELSEN, G.D. PLOTKIN & G. WINSKEL (1981): *Petri nets, event structures and domains, part I.* Theoretical Computer Science 13(1), pp. 85-108.

[5] G. WINSKEL (1982): *Event structure semantics for CCS and related languages.* In: Proceedings ICALP 82, Aarhus (M. Nielsen & E.M. Schmidt, eds.), LNCS 140, Springer-Verlag, pp. 561-576, see also DAIMI report PB-159, Computer Science Department, Aarhus University, 1983.

[6] G. WINSKEL (1987): *Event structures.* In: Petri Nets: Applications and Relationships to Other Models of Concurrency, Advances in Petri Nets 1986, Part II; Proceedings of an Advanced Course, Bad Honnef, September 1986 (W. Brauer, W. Reisig & G. Rozenberg, eds.), LNCS 255, Springer-Verlag, pp. 325-392.

[7] G. WINSKEL (February 1989): *Personal communication.*