Fundamental Study

Denotational semantics in the cpo and metric approach

Christel Baier and Mila E. Majster-Cederbaum
Fakultät für Mathematik und Informatik, Universität Mannheim, 68131 Mannheim, Germany

Communicated by M. Nivat
Received May 1993
Revised February 1994

Abstract


We investigate the implications of choosing a cpo-framework resp. a complete metric space framework for defining denotational semantics of languages that allow for recursion/iteration, communication and concurrency. We first establish a general framework for the cpo and the metric approach. The existence and uniqueness of meaning functions is studied. In the metric case the existence and uniqueness of a meaning function can be established under some reasonable assumptions. In the cpo-case we obtain the existence of a least meaning function. From these theorems consistency results can be concluded. In the second part we study the impact of the choice between cpo and metric for semantics based on event-structures and for semantics based on pomset classes.

Contents

1. Introduction .............................................................. 172
2. Denotational semantics ............................................. 174
  2.1. The language \( \mathcal{L}_1 \) ..................................... 174
  2.2. Interpretations of symbol-algebras ......................... 176
3. \( \Sigma \)-algebras in the metric approach ......................... 179
  3.1. Symbol-algebras with guardedness conditions ........... 179
  3.2. \( \ell \)-complete-metric-spaces .......................... 181
4. \( \Sigma \)-algebras in the cpo approach ........................... 184

Correspondence to: C. Baier, Fakultät für Mathematik und Informatik, Universität Mannheim, 68131 Mannheim, Germany

0304-3975/94/$07.00 © 1994 Elsevier Science B.V. All rights reserved
SSDI 0304-3975(94)00046-L
1. Introduction

Various methods to define the semantics of languages that allow for parallelism, communication and synchronisation have been proposed in the last years. They can be classified by several criteria as e.g. operational versus denotational versus axiomatic methods, interleaving versus true parallelism approaches, branching time versus linear time models, choice of mathematical discipline to assist the handling of recursion and the solution of domain equations.

Three approaches to the handling of recursion in a denotational context and to the solution of domain equations can be distinguished: one uses complete partial orders and Tarski’s fixed point theorem, the second uses complete metric spaces and Banach's fixed point theorem and the last works with nonstandard set theory. In the literature these methods are used as follows: A fixed language $\mathcal{L}$ with recursion or repetition together with a semantic domain $M$ and a cpo (resp. metric) structure on $M$. Semantic operators on $\mathcal{L}$ are defined and shown to have the desired properties (continuous resp. nondistance increasing). For each recursive or repetitive program $P$ an associated operator $\Omega_P$ is defined and shown explicitly to have the desired property (continuous resp. contracting). The meaning of $P$ in $M$ is then defined as the least resp. unique fixed point of $\Omega_P$.

We exhibit here a very general framework for dealing with denotational semantics, i.e. $\Sigma$-algebras. $\Sigma$ consists of a set of operator symbols with associated arity. A $\Sigma$-algebra consists of a set $M$ together with an operator $\omega_M$ for each operator symbol $\omega\in\Sigma$. A special case of a $\Sigma$-algebra is the word algebra $\mathcal{L}(\Sigma, \text{Idf})$ which is built from identifiers $\text{Idf}$ and operator symbols $\Sigma$. Elements of $\mathcal{L}(\Sigma, \text{Idf})$ are called statements. A process is a pair $\langle \sigma, s \rangle$ where $s$ is a statement and $\sigma$ a declaration mapping variables to statements (by this recursion is introduced). Given an arbitrary $\Sigma$-algebra $M$ a meaning function for processes is a mapping $Me$ from processes to $M$ which is required to satisfy the following conditions:

(I) $Me$ is a homomorphism.

(II) $Me(\langle \sigma, x \rangle) = Me(\langle \sigma, \sigma(x) \rangle)$ for each $x \in \text{Idf}$ (recursion condition).
The first condition means that $Me$ should be compositional, the second says that, given a declaration $\sigma$, the meaning of $x$ is determined by $\sigma(x)$.

In Theorem 2.12 it is shown that $Me$ satisfies (I) and (II) iff, for each declaration $\sigma$, the function $f_\sigma: \mathcal{L} \to M$ given by $f_\sigma(s) = Me(\langle \sigma, s \rangle)$ is a fixed point of some operator $\phi^M[\sigma]$. Please note that this result does not make use of any additional structure on $M$. Adding metric to $M$ yields the existence of a unique meaning function satisfying (I) and (II) (Theorem 3.10). Adding a cpo structure on $M$ yields the existence of a least meaning function satisfying (I) and (II) (Theorem 4.4). Given two $\Sigma$-algebras $M$ and $N$ endowed with metric and meaning functions $Me^M$ resp. $Me^N$ then for each homomorphism $f: M \to N$ of $\Sigma$-algebras

$$f \circ Me^M = Me^N$$

which can be interpreted as a consistency result (Theorem 3.11). In the case of a cpo structure on $M$ and $N$ the homomorphism $f$ must be cpo-continuous and preserve the bottom-element in order to guarantee the analogous result (Theorem 4.8).

Recently attempts are made to derive some partial order structure from a metric space with the intention to transform a metric space semantics into a partial order semantics. We show that there cannot be a general way of transforming semantics in such a way.

(1) We show that we may use event structures endowed with metric in order to give meaning to a CCS-like language including a concatenation operator $;$. The attempt to model the same language using event structures with cpo structure fails, as the semantic operator for $;$ is not continuous (Section 5.5).

(2) We show that pomset classes endowed with metric can be used to model a simple CCS-style language without synchronisation whereas this is not possible in a cpo-setting (Section 8).

Whereas the metric approach suffers from the drawbacks that unguarded recursion cannot be handled and that is not possible to give an input/output meaning to sequential programs using metric (compare [6, 7]), the two later results indicate situations in favour of a metric setting.

The paper is organized in 8 sections. Section 2 introduces $\Sigma$-algebras, interpretations of $\Sigma$-algebras and abstract meaning functions. A general condition for the existence of a meaning function with recursion condition is given. Section 3 introduces $\mathcal{C}$-complete-metric-spaces. Given a language $\mathcal{L}(\mathcal{G}, Idf)$ over a symbol algebra $\mathcal{G}$ with guardedness conditions we show that any $\mathcal{C}$-complete-metric-space $M$ can serve as a semantic domain for $\mathcal{L}(\mathcal{G}, Idf)$ and that there is a unique meaning function from processes to $M$ satisfying (I) and (II). This meaning function is obtained automatically. The consistency of the meaning functions of any two homomorphic $\mathcal{C}$-complete-metric-spaces is shown. Section 4 introduces $\Sigma$-cpo’s. We show that any $\Sigma$-cpo can be used as a semantic domain for a language $\mathcal{L}(\Sigma, Idf)$ and that there is always a least meaning function satisfying (I) and (II). A weaker consistency result is presented in the cpo case. In Section 5 we discuss the issue metric versus cpo in the case of event structures. Various classes of event structures have been used in the past to provide
a true concurrency meaning to concurrent programs. The class of finitely approximable prime event structures enriched with metric has been used by [3] and [1] to provide a semantics to TCSP (with guarded recursion), yielding an $C$-complete-metric-space. It is easy to see that the CCS-parallel-operator and the CCS-choice operator can be treated alike. On the other hand the class of prime event structures with cpo structure has been used by Winskel [14,16] to model CCS, yielding a $\Sigma_{\text{tcp}}$-cpo. The metric space semantics and the cpo-semantics for CCS with guarded recursion coincide. However, we show that it is possible to define a semantic operator for sequential composition (\(\cdot\)) using metric which is not possible using cpo. In Section 6 we give a brief account of pomsets as proposed by [12] which have been used in [2] to provide a true concurrency, linear time semantics to a language with sequential operator, parallelism and choice. For Section 7 we need to extend this operator set to cover the operators in $\Sigma$, which is obtained from CCS by substituting the CCS-parallel operator by a parallel operator $|$ without synchronisation, yielding an $C_{\text{hcp}}$-complete-metric-space. In Section 7 we construct a homomorphism from the $C_{\text{hcp}}$-complete-metric-space of finitely approximable prime event structures to the $C_{\text{hcp}}$-complete-metric-space of pomset classes. Section 8 shows that it is not possible to work with cpo instead of metric in the case of pomset classes when modelling a language including choice and recursion.

2. Denotational semantics

2.1. The language $\mathcal{P}(\Sigma, \text{Idf})$

In this section we define a very general language with abstract operator symbols and recursion which is introduced by declarations.

Definition 2.1. A symbol-algebra is a pair $\Sigma=\langle Op, \cdot \rangle$ consisting of a set $Op$ of operator symbols and a function $\cdot: Op \to \mathbb{N}_0$ which assigns the arity $|\cdot|$ to each operator symbol $\cdot$. Operator symbols of arity 0 are called constant symbols. $\text{Const}(\Sigma)$ denotes the set of constant symbols.

If $\Sigma=\langle Op, \cdot \rangle$ is a symbol-algebra and $\text{Idf}$ is a set of identifiers then the associated language $\mathcal{L}(\Sigma, \text{Idf})$ is given by the production system

$$s ::= a \mid x \mid \omega(s_1, \ldots, s_n),$$

where $a \in \text{Const}(\Sigma), x \in \text{Idf}$ and $\omega \in \text{Op}, |\omega|=n \geq 1, s_1, \ldots, s_n \in \mathcal{L}(\Sigma, \text{Idf})$. The elements of $\mathcal{L}(\Sigma, \text{Idf})$ are called statements over $(\Sigma, \text{Idf})$.

Let $\mathcal{L}(\Sigma, \text{Idf})$ be the set of all primitive statements, i.e. the set of all statements $s \in \mathcal{L}(\Sigma, \text{Idf})$ which do not contain any occurrence of an identifier $x \in \text{Idf}$. The statements $s \in \mathcal{L}(\Sigma, \text{Idf})$ are given by the production system $s ::= a \mid \omega(s_1, \ldots, s_n)$ where $a \in \text{Const}(\Sigma), \omega \in \text{Op}, |\omega|=n \geq 1, s_1, \ldots, s_n \in \mathcal{L}(\Sigma, \text{Idf})$. 


In $\mathcal{L}(\Sigma, \text{Idf})$ recursion can be introduced by declarations, i.e. functions which assign a statement to each identifier. A process over $(\Sigma, \text{Idf})$ is a pair $\langle \sigma, s \rangle$ consisting of a declaration $\sigma$ and a statement $s$. If $P = \langle \sigma, s \rangle$ is a process then the behaviour of $P$ is given by $s$ where each occurrence of a variable $x$ in $s$ is interpreted as a recursive call of the procedure $\sigma(x)$.

**Notation 2.2.** $\mathcal{D}(\Sigma, \text{Idf})$ denotes the set of declarations, i.e. the set of functions $\sigma : \text{Idf} \rightarrow \mathcal{L}(\Sigma, \text{Idf})$. $\mathcal{P}(\Sigma, \text{Idf})$ denotes the set of processes over $(\Sigma, \text{Idf})$, i.e. the set of pairs $\langle \sigma, s \rangle$ where $\sigma \in \mathcal{D}(\Sigma, \text{Idf})$ and $s \in \mathcal{L}(\Sigma, \text{Idf})$.

**Example 2.3.** The language CCS [8–10] without recursion is associated with the symbol-algebra $\Sigma_{\text{CCS}}$ which consists of the following operator symbols.

- the constant symbol $\text{nil}$,
- for each action $a \in \text{Act}$ an operator symbol $\gamma_a$ of the arity 1, $\gamma_a(s) = a \cdot s$, modelling prefixing,
- the binary operator symbol $\|$, modelling nondeterminism,
- the binary operator symbol $\parallel$, modelling parallelism with possible communication on complementary actions,
- for each $L \subseteq \text{Act} \setminus \{\tau\}$ an operator symbol $\rho_L$, $\rho_L(s) = s \setminus L$, of the arity 1 for modelling restriction on actions $\notin L \cup \overline{L}$,
- for each relabelling function $\lambda : \text{Act} \rightarrow \text{Act}$ an operator symbol $\ell_\lambda$ of the arity 1, $\ell_\lambda(s) = s \setminus \overline{\lambda}$, which is used for renaming the actions.

Here $\text{Act}$ is a nonempty set of actions which contains an internal action denoted by $\tau$. We assume a function $(\cdot) : \text{Act} \rightarrow \text{Act}, a \mapsto \overline{a}$, such that $\overline{\tau} = \tau$ and $\overline{a} = a$ for each action $a \in \text{Act}$. If $L \subseteq \text{Act}$ then $\overline{L} = \{\overline{a} : a \in L\}$. A relabelling function is a function $\lambda : \text{Act} \rightarrow \text{Act}$ with $\overline{\lambda(\tau)} = \tau$ and $\overline{\lambda(a)} = \overline{\lambda(a)}$.

The process $\langle \sigma, s \rangle$ corresponds to Milners CCS-process

$$s[\text{fix}(x = \sigma(x))/x : x \in \text{Idf}],$$

where $s[\tau_x/x : x \in \text{Idf}]$ means the statement which arises from $s$ by substituting each occurrence of the identifier $x \in \text{Idf}$ by the statement $\tau_x$.

**Example 2.4.** In our examples in Sections 5 and 6 we consider the language $\mathcal{D}(\Sigma_1, \text{Idf})$ where $\Sigma_1$ coincides with $\Sigma_{\text{CCS}}$ except the parallelism operator: In $\Sigma_1$ the CCS-parallel operator $\|$ is substituted by the parallel operator $\parallel$ without synchronisation or communication.

**Example 2.5.** The language $\mathcal{L}_0 = \mathcal{L}_0(\Sigma_0, \text{Idf})$ which is considered e.g. in [2] is given by the symbol-algebra $\Sigma_0$ which contains

- a nonempty set $\text{Act}$ of actions as the constant symbols,
- binary operator symbols $\|$, $\parallel$ and $;$ for modelling nondeterminism, parallelism (without synchronisation or communication) resp. sequential execution.
Definition 2.6. If \( \Sigma = (\text{Op}_\Sigma, |\cdot|) \) is a symbol-algebra then a \( \Sigma \)-algebra is a pair \((M, \text{Op}_M)\) consisting of a set \( M \) and a set \( \text{Op}_M \) of operators

\[
\text{Op}_M = \{ \omega_M : \omega \in \text{Op} \}
\]
such that \( \omega_M : M^n \to M \) is a function where \( |\omega| = n \). (In the case \( n = 0, \omega_M \in M \).)

In the following we often omit the operator set \( \text{Op}_M \) of a \( \Sigma \)-algebra and we shortly write \( M \) instead of \( (M, \text{Op}_M) \).

If \( M \) and \( N \) are \( \Sigma \)-algebras then a function \( f : M \to N \) is called a homomorphism iff

\[
f(\omega_M(\xi_1, \ldots, \xi_n)) = \omega_N(f(\xi_1), \ldots, f(\xi_n))
\]
for each \( \omega \in \text{Op}, |\omega| = n \).

Example 2.7. Identifying the constant symbol \( \sigma \in \text{Const}(\Sigma) \) with the constant \( a \in \mathcal{L}(\Sigma, \text{Id}) \) and the operator symbols \( \omega \in \text{Op}, |\omega| = n \geq 1 \), with the \( n \)-ary operator

\[
\omega : \mathcal{L}(\Sigma, \text{Id})^n \to \mathcal{L}(\Sigma, \text{Id}), \quad (s_1, \ldots, s_n) \mapsto \omega(s_1, \ldots, s_n),
\]
we get that \( \mathcal{L}(\Sigma, \text{Id}) \) together with \( \text{Op} \) is a \( \Sigma \)-algebra.

Remark 2.8. If \( A \) is a nonempty set and \( M \) is a \( \Sigma \)-algebra then the space \( A \to M \) of all functions \( h : A \to M \) is also a \( \Sigma \)-algebra where we define the semantic operators as follows:

1. For each constant symbol \( a \in \text{Const}(\Sigma) \) the function \( a_{A \to M} : A \to M \) maps each \( \xi \in A \) to \( a_{M} \).
2. For each operator symbol \( \omega \in \text{Op}, |\omega| = n \geq 1 \), the operator

\[
\omega_{A \to M} : (A \to M)^n \to (A \to M)
\]
is given by

\[
\omega_{A \to M}(h_1, \ldots, h_n)(\xi) = \omega_M(h_1(\xi), \ldots, h_n(\xi)).
\]

In the following we use this construction of a \( \Sigma \)-algebra with \( A = \mathcal{D}(\Sigma, \text{Id}) \) or \( A = \mathcal{D} \) a subset of \( \mathcal{D}(\Sigma, \text{Id}) \) or \( A = \mathcal{D}(\Sigma, \text{Id}) \to M \).

Our aim is to define a compositional semantics \( M_e : \mathcal{D}(\Sigma, \text{Id}) \to M \) which satisfies the recursion condition

\[
M_e(\langle \sigma, x \rangle) = M_e(\langle \sigma, \sigma(x) \rangle)
\]
for each identifier \( x \in \text{Id} \). The recursion condition corresponds to our intuition of recursion. “Compositional” means that there exist operators on \( M \) corresponding to
the operator symbols of $\Sigma$ such that $M$ is a $\Sigma$-algebra and such that $Me(\langle \sigma, a \rangle) = a_M$ for each $a \in \text{Const}(\Sigma)$ and

$$Me(\langle \sigma, o(s_1, \ldots, s_n) \rangle) = \omega_M(\langle \sigma, s_1 \rangle, \ldots, \langle \sigma, s_n \rangle)$$

for each operator symbol $\omega \in Op$, $|\omega| = n \geq 1$. This can be formulated as follows: The associated function

$$F: \mathcal{L}(\Sigma, Idf) \rightarrow (\mathcal{D}(\Sigma, Idf) \rightarrow M),$$

where $F(s): \mathcal{D}(\Sigma, Idf) \rightarrow M$ is given by $F(s)(\sigma) = Me(\langle \sigma, s \rangle)$, is a homomorphism from the $\Sigma$-algebra $\mathcal{L}(\Sigma, Idf)$ to the $\Sigma$-algebra $\mathcal{D}(\Sigma, Idf) \rightarrow M$. The recursion condition is equivalent to

$$F(x)(\sigma) = F(x)(\sigma)x \forall x \in Idf.$$ 

On the other hand, given such a function

$$F: \mathcal{L}(\Sigma, Idf) \rightarrow (\mathcal{D}(\Sigma, Idf) \rightarrow M),$$

we get a compositional meaning function $Me: \mathcal{D}(\Sigma, Idf) \rightarrow M$ when we define

$$Me(\langle \sigma, s \rangle) = F(s)(\sigma).$$

The recursion condition is true for $Me$ if and only if $F$ satisfies the corresponding recursion condition.

**Remark 2.9.** If we drop the recursion condition then for each $\Sigma$-algebra $M \neq \emptyset$ there always exists a homomorphism $F: \mathcal{L}(\Sigma, Idf) \rightarrow (\mathcal{D}(\Sigma, Idf) \rightarrow M)$ (and then a compositional meaning function with semantic domain $M$).

In the following two remarks we will see that given an arbitrary $\Sigma$-algebra it is possible that there does not exist any homomorphism $F: \mathcal{L}(\Sigma, Idf) \rightarrow (\mathcal{D}(\Sigma, Idf) \rightarrow M)$ with the recursion condition and on the other side it is possible that there is more than one solution.

**Remark 2.10.** In the case $M = \mathcal{L}(\Sigma, Idf)$ where $\Sigma$ contains at least one operator symbol $\omega$ of the arity $n \geq 1$ there does not exist a homomorphism $F$ with the recursion condition: If we assume that there exists such a homomorphism $F$ with the recursion condition we consider a declaration $\sigma$ with $\sigma(x) = \omega(x, \ldots, x)$ for some identifier $x \in Idf$. Then the statements $F(x)(\sigma)$ and $\omega(F(x)(\sigma), \ldots, F(x)(\sigma))$ must be the same. This is always wrong.

**Remark 2.11.** It is possible that there exist more than one homomorphism with the recursion condition. Let $\Sigma = (Op, | \cdot |)$ where $Op = \{a, b, \omega\}$ with $|a| = |b| = 0, |\omega| = 1$. We
consider the $\Sigma$-algebra $M = \{0, 1\}$ where $a_M = 0$, $b_M = 1$ and $\omega_M(\xi) = \xi$, $\xi = 0, 1$. Then

$$F_1, F_2 : \mathcal{L}(\Sigma, Idf) \rightarrow (\mathcal{D}(\Sigma, Idf) \rightarrow M)$$

defined by $F_i(x)(\sigma) = 0$ if $\sigma(x) = \psi^k(a)$ for some $k \geq 0$, $F_i(x)(\sigma) = 1$ if $\sigma(x) = \omega^k(b)$ for some $k \geq 0$, $F_2(x)(\sigma) = 0$ (respectively $F_2(x)(\sigma) = 1$) otherwise, are homomorphisms with recursion condition.

A general characterisation of possible meaning functions is given in the following theorem. In this theorem we allow for subsets $\mathcal{D}$ of declarations in order to be able to model also restrictions on recursion, as guardedness.

**Theorem 2.12.** Let $\Sigma = (Op, |::|)$ be a symbol-algebra, $\mathcal{D}$ a subset of $\mathcal{D}(\Sigma, Idf)$. $M$ a $\Sigma$-algebra and

$$F : \mathcal{L}(\Sigma, Idf) \rightarrow (\mathcal{D} \rightarrow M)$$

a function. Then we have: $F$ is a homomorphism with the recursion condition on $\mathcal{D}$ if and only if for each declaration $\sigma \in \mathcal{D}$ the function

$$f_\sigma : \mathcal{L}(\Sigma, Idf) \rightarrow M, \quad f_\sigma(s) = F(s)(\sigma).$$

is a fixed point of $\Phi^M[\sigma] : (\mathcal{L}(\Sigma, Idf) \rightarrow M) \rightarrow (\mathcal{L}(\Sigma, Idf) \rightarrow M)$ where $\Phi^M[\sigma]$ is defined by structural induction:

1. $\Phi^M[\sigma](f)(a) = a_M \forall a \in \text{Const}(\Sigma)$.
2. $\Phi^M[\sigma](f)(x) = f(\sigma(x)) \forall x \in Idf$.
3. $\Phi^M[\sigma](f)(\omega(s_1, \ldots, s_n)) = \omega_M(\Phi^M[\sigma](f)(s_1), \ldots, \Phi^M[\sigma](f)(s_n))$ for all $s_1, \ldots, s_n \in \mathcal{L}(\Sigma, Idf)$, $\omega \in Op$, $|\omega| = n \geq 1$.

**Proof.** Let $\mathcal{L} = \mathcal{L}(\Sigma, Idf)$, $\text{Const} = \text{Const}(\Sigma)$.

$F$ is a homomorphism with the recursion condition if and only if

1. For each $a \in \text{Const}$ the function $F(a) : \mathcal{D} \rightarrow M$ agrees with the function $\mathcal{D} \rightarrow M$.
2. $F(x)(\sigma) = F(\sigma(x))(\sigma)$ for each $\sigma \in \mathcal{D}$ and $x \in Idf$.
3. $F(\omega(s_1, \ldots, s_n)) = \omega_M(F(s_1), \ldots, F(s_n))$ for all $\omega \in Op$, $|\omega| = n \geq 1$, $s_1, \ldots, s_n \in \mathcal{L}$.

Conditions (i), (ii) and (iii) are equivalent to the following conditions (I), (II) and (III):

1. $f_\sigma(a) = a_M$ for each declaration $\sigma \in \mathcal{D}$ and each $a \in \text{Const}$.
2. $f_\sigma(x) = f_\sigma(\sigma(x))$ for each $\sigma \in \mathcal{D}$ and $x \in Idf$.
3. $f_\sigma(\omega(s_1, \ldots, s_n)) = \omega_M(f_\sigma(s_1), \ldots, f_\sigma(s_n))$ for all $\sigma \in \mathcal{D}$, $s_1, \ldots, s_n \in \mathcal{L}$, $\omega \in Op$, $|\omega| = n \geq 1$.

Conditions (I), (II) and (III) are equivalent to the condition that for each $\sigma \in \mathcal{D}$ the function $f_\sigma$ is a fixed point of $\Phi^M[\sigma]$.

**Theorem 2.13.** Let $\Sigma = (Op, |::|)$ be a symbol-algebra, $\mathcal{D}$ a subset of $\mathcal{D}(\Sigma, Idf)$, $M$ and $N$ two $\Sigma$-algebras and $f : M \rightarrow N$ a homomorphism from $M$ to $N$. If $F : \mathcal{L}(\Sigma, Idf)\rightarrow$
Denotational semantics

($\mathcal{D} \to M$) is a homomorphism with the recursion condition then \( f \circ M^\mathcal{D} : \mathcal{P}_0(\Sigma, \text{Idf}) \to \mathcal{N} \) is a meaning function which satisfies the recursion condition.

Here \( \mathcal{P}_0(\Sigma, \text{Idf}) \to \mathcal{N} \) is the set of processes \( \langle \sigma, s \rangle \) with \( \sigma \in \mathcal{D} \) and \( M^\mathcal{D} \) is the meaning function with range \( \mathcal{M} \) induced by \( F \), i.e. \( M^\mathcal{D}(\langle \sigma, s \rangle) = F(s)(\sigma) \).

**Proof.** By Theorem 2.12 we have to show that for fixed declaration \( \sigma \in \mathcal{D} \) the function

\[
g_\sigma : \mathcal{P}(\Sigma, \text{Idf}) \to \mathcal{N}, \quad g_\sigma(s) = f(M^\mathcal{D}(\langle \sigma, s \rangle))
\]

is a fixed point of \( \Phi^\mathcal{N}[\sigma] \).

1. If \( a \in \text{Const}(\Sigma) \) then \( M^\mathcal{D}(\langle \sigma, a \rangle) = a_M \). Since \( f \) is a homomorphism we have \( f(a_M) = a_N = \Phi^\mathcal{N}[\sigma](g_\sigma)(a) \).

2. If \( x \in \text{Idf} \) then \( M^\mathcal{D}(\langle \sigma, x \rangle) = M^\mathcal{D}(\langle \sigma, \sigma(x) \rangle) \) and therefore

\[
g_\sigma(x) = f(M^\mathcal{D}(\langle \sigma, x \rangle)) = f(M^\mathcal{D}(\langle \sigma, \sigma(x) \rangle)) = g_\sigma(\sigma(x)) = \Phi^\mathcal{N}[\sigma](g_\sigma)(x).
\]

3. If \( \omega \in \text{Op}, |\omega| \geq 1 \), \( s_1, \ldots, s_n \in \mathcal{P}(\Sigma, \text{Idf}) \) then

\[
g_\sigma(\omega(s_1, \ldots, s_n)) = f(M^\mathcal{D}(\langle \sigma, \omega(s_1, \ldots, s_n) \rangle))
\]

\[
= f(\omega M(M^\mathcal{D}(\langle \sigma, s_1 \rangle), \ldots, M^\mathcal{D}(\langle \sigma, s_n \rangle)))
\]

\[
= \omega_N(f(M^\mathcal{D}(\langle \sigma, s_1 \rangle)), \ldots, f(M^\mathcal{D}(\langle \sigma, s_n \rangle)))
\]

\[
= \omega_N(g_\sigma(s_1), \ldots, g_\sigma(s_n)) = \Phi^\mathcal{N}[\sigma](g_\sigma)(\omega(s_1, \ldots, s_n)).
\]

3. \( \Sigma \)-algebras in the metric approach

In this section we introduce the concept of an \( \mathcal{O} \)-complete-metric-space. \( \mathcal{O} \)-complete-metric-spaces constitute an abstraction of all those properties that are necessary to define semantics on the basis of complete metric spaces and Banach’s fixed point theorem. Given any \( \mathcal{O} \)-complete-metric-space \( M \) there is by Theorem 3.10 an automatic way to obtain a meaning function \( M^\mathcal{O} \) with range \( M \). This meaning function is unique, i.e. there is no other meaning function with range \( M \) that is compositional and satisfies the recursion condition. Hence providing a semantics using the metric approach really means constructing a suitable \( \mathcal{O} \)-complete-metric-space. In addition, if we consider two \( \mathcal{O} \)-complete-metric-spaces \( M \) and \( N \) for which we can establish a homomorphism \( f : M \to N \) then Theorem 3.11 ensures that \( f \circ M^\mathcal{O} = M^N \) (cf. [5]).

3.1. Symbol-algebras with guardedness conditions

Imposing a metric on a \( \Sigma \)-algebra \( M \) is one way to ensure the existence of a meaning function with range \( M \). The aim is to apply Banach’s fixed point theorem to \( \Phi^\mathcal{N}[\sigma] \)
thus ensuring the existence of a unique fixed point \( f_\sigma \) and then apply Theorem 2.12 to define \( F \). For this purpose we have to exclude unguarded recursion as it leads to a noncontracting operator \( \phi^M[\sigma] \).

**Definition 3.1.** A symbol-algebra with guardedness conditions is a triple \( \mathcal{C} = (\text{Op}, \cdot \cdot \cdot, \deg) \) such that \( \Sigma = (\text{Op}, \cdot \cdot \cdot) \) is a symbol-algebra and \( \deg : \text{Op} \to \mathbb{N}_0 \) is a function with \( 0 \leq \deg(\omega) \leq |\omega| \) for all \( \omega \in \text{Op} \). \( \deg(\omega) \) is called the degree of guardedness of \( \omega \).

**Statements over (C, Idf).** Statements over \( (\mathcal{C}, \text{Idf}) \) are statements over \( (\Sigma, \text{Idf}) \). \( \mathcal{L}(\mathcal{C}, \text{Idf}) = \mathcal{L}(\Sigma, \text{Idf}) \) denotes the set of statements over \( (\mathcal{C}, \text{Idf}) \). We also write \( \text{Const}(\mathcal{C}) \) instead of \( \text{Const}(\Sigma) \).

In the language \( \mathcal{L}(\mathcal{C}, \text{Idf}) \) of a symbol-algebra with guardedness conditions we are able to define guarded statements: Each constant symbol is a guarded statement. If the degree of guardedness of an operator symbol \( \omega \) is \( k \) where \( 1 \leq k \leq |\omega| \) then \( \omega \) ensures guardedness in its last \( k \) arguments, i.e. if \( s_1, \ldots, s_k \) are arbitrary statements and \( \xi_1, \ldots, \xi_{n-k} \) are guarded statements then \( \omega(s_1, \ldots, s_k) \) is a guarded statement.

\( \mathcal{L}^g(\mathcal{C}, \text{Idf}) \) denotes the set of all guarded statements, i.e. the set of statements \( \xi \in \mathcal{L}(\mathcal{C}, \text{Idf}) \) which are given by

\[
\xi := a | \omega(\xi_1, \ldots, \xi_{n-k}, s_1, \ldots, s_k),
\]

where \( s_1, \ldots, s_k \in \mathcal{L}(\mathcal{C}, \text{Idf}) \), \( \xi_1, \ldots, \xi_{n-k} \in \mathcal{L}^g(\mathcal{C}, \text{Idf}) \), \( a \in \text{Const}(\mathcal{C}) \), \( \omega \in \text{Op} \), \( |\omega| = n \geq 1 \), \( \deg(\omega) = k \).

**Definition 3.2.** A declaration over \( (\mathcal{C}, \text{Idf}) \) is a function \( \sigma : \text{Idf} \to \mathcal{L}^g(\mathcal{C}, \text{Idf}) \). \( \mathcal{D}(\mathcal{C}, \text{Idf}) \) denotes the set of declarations over \( (\mathcal{C}, \text{Idf}) \).

A process over \( (\mathcal{C}, \text{Idf}) \) is a pair \( \langle \sigma, s \rangle \) consisting of a declaration \( \sigma \) over \( (\mathcal{C}, \text{Idf}) \) and a statement \( s \) over \( (\mathcal{C}, \text{Idf}) \). \( \mathcal{P}(\mathcal{C}, \text{Idf}) \) is the set of all processes over \( (\mathcal{C}, \text{Idf}) \).

**Example 3.3.** The guardedness conditions in the symbol-algebra \( \Sigma_{\text{CCS}} \) as defined in Example 2.3 are given by: The prefixing operator symbols \( \gamma_a \) are guarded operator symbols, i.e.

\[
\deg(\gamma_a) = |\gamma_a| = 1 \quad \forall a \in \text{Act}.
\]

The other operator symbols have 0 as the degree of guardedness, i.e.

\[
\deg(\cdot) = \deg(\cdot) = \deg(\rho_1) = \deg(\cdot) = 0.
\]

Hence a statement \( s \in \mathcal{L}(\Sigma_{\text{CCS}}, \text{Idf}) \) is guarded if and only if each occurrence of an identifier \( x \) in \( s \) is in the scope of a prefixing operator symbol. This is equivalent to Milners definition of guarded CCS-terms without recursion (see [10]). In the following \( \mathcal{C}_{\text{CCS}} = (\Sigma_{\text{CCS}}, \deg) \) denotes the symbol-algebra with guardedness conditions as defined here.
In the same way we define guardedness conditions for the symbol-algebra $\Sigma_1$ of Example 2.4: $\mathcal{O}_1 = (\Sigma_1, \text{deg})$ coincides with $\mathcal{O}_{\text{CCS}}$ where the degree of guardedness of the parallel operator $\parallel$ is 0.

**Example 3.4.** Guardedness of statements of the language $\mathcal{L}_0$ (see Example 2.5) in the sense of [2] can be obtained when we define

$$\text{deg}(+) = \text{deg}(\parallel) = 0, \quad \text{deg}(:) = 1.$$ 

A statement $s \in \mathcal{L}_0$ is guarded if and only if each occurrence of an identifier $x$ in $s$ is contained in the second argument of a sequential composition, i.e. there exists a subterm $s_1; s_2$ of $s$ such that the occurrence of $x$ is in $s_2$. We define $\mathcal{O}_0$ to be the symbol-algebra with guardedness conditions consisting of the symbol-algebra $\Sigma_0$ and the guardedness conditions $\text{deg}(+) = \text{deg}(\parallel) = 0, \text{deg}(:) = 1$.

### 3.2. $\mathcal{O}$-complete-metric-spaces

**Definition 3.5.** Let $\mathcal{O} = (\Sigma, \text{deg})$, $\Sigma = (\mathcal{O}_p, |\cdot|)$, be a symbol-algebra with guardedness-conditions. An $\mathcal{O}$-complete-metric-space (shortly $\mathcal{O}$-cms) is a $\mathcal{O}$-algebra $M$ such that:

- There exists a metric $\delta = \delta_M$ on $M$, $0 \leq \delta \leq 1$, such that $(M, \delta)$ is a complete metric space.
- For each operator symbol $o \in \mathcal{O}_p$ with $|o| = n \geq 1$, $\text{deg}(o) = k$ the associated operator $o_M : M^n \to M$ is nondistance increasing and contracting in its last $k$ arguments:

$$\delta(o_M(\bar{\xi}), o_M(\bar{\xi}')) \leq \max \left\{ \max_{1 \leq i \leq n-k} \delta(\xi_i, \xi_i'), \frac{1}{2} \cdot \max_{n-k+1 \leq j \leq n} \delta(\xi_j, \xi'_j) \right\}$$

for all $\bar{\xi} = (\xi_1, \ldots, \xi_n)$, $\bar{\xi}' = (\xi_1', \ldots, \xi_n') \in M^n$.

Now we show that for each symbol-algebra $\mathcal{O}$ with guardedness-conditions and for each $\mathcal{O}$-cms $M$ there exists a unique homomorphism

$$F^M : \mathcal{O}(\mathcal{O}, Idf) \to (\mathcal{O}(\mathcal{O}, Idf) \to M)$$

which satisfies the recursion condition. By Theorem 2.12 we have to show that $\Phi^M[\sigma]$ has a unique fixed point. To do so we show that $\Phi^M[\sigma]$ is a contracting function on the complete metric space $\mathcal{O}(\mathcal{O}, Idf) \to M$. Then we apply Banach’s fixed point theorem. Here the metric on $\mathcal{O}(\mathcal{O}, Idf) \to M$ is defined by

$$\delta(f_1, f_2) = \sup \{ \delta(f_1(s), f_2(s)) : s \in \mathcal{O}(\mathcal{O}, Idf) \}.$$ 

**Lemma 3.6.** Let $\mathcal{O} = (\mathcal{O}_p, |\cdot|, \text{deg})$ be a symbol-algebra with guardedness conditions. Let $\sigma$ be a declaration over $(\mathcal{O}, Idf)$ and $M$ an $\mathcal{O}$-cms. Then $\Phi^M[\sigma]$ is a contracting
self-mapping of the complete metric space \( (\mathcal{L}(\emptyset, Idf) \to M) \), more precisely

\[
\delta(\Phi[\sigma](f_1), \Phi[\sigma](f_2)) \leq \frac{1}{2} \delta(f_1, f_2) \quad \forall f_1, f_2: \mathcal{L}(\emptyset, Idf) \to M.
\]

**Proof.** Let \( \Phi = \Phi[\sigma] \), \( \mathcal{L} = \mathcal{L}(\emptyset, Idf) \), \( \mathcal{L}^g = \mathcal{L}(\emptyset, Idf^g) \), \( \text{Const} = \text{Const}(\emptyset) \).

First we show that for all statements \( \sigma \in \mathcal{L}^g \) and \( f_1, f_2: \mathcal{L} \to M \):

\[
\delta(\Phi(f_1)\sigma, \Phi(f_2)\sigma) \leq \sup \{ \delta(f_1(\zeta), f_2(\zeta)) : \zeta \in \mathcal{L}^g \}.
\]

Second, we will see that for all \( f_1, f_2: \mathcal{L} \to M \) and \( \zeta \in \mathcal{L}^g \):

\[
\delta(f_1(\zeta), f_2(\zeta)) \leq \frac{1}{2} \delta(f_1, f_2).
\]

**ad 1.** By structural induction on the syntax of \( \sigma \in \mathcal{L}^g \).

*Basis of induction:* If \( \sigma = a \in \text{Const} \) then \( \Phi(f_1)(a) = a_M = f_i(a) \) and \( a \in \mathcal{L}^g \).

If \( \sigma = x \in Idf \) then \( \Phi(f_i)(x) = f_i(\sigma(x)) \) and \( \sigma(x) \in \mathcal{L}^g \).

*Induction step:* \( \sigma = \omega(s_1, \ldots, s_n) \) where \( s_1, \ldots, s_n \in \mathcal{L} \) and \( \omega \in \emptyset \).

Then \( \Phi(f_i)(\sigma) = \omega_M(\Phi(f_i)(s_1), \ldots, \Phi(f_i)(s_n)), i = 1, 2 \), and

\[
\delta(\Phi(f_1)(\sigma), \Phi(f_2)(\sigma)) \\
= \delta(\Phi(f_1)(\omega(s_1, \ldots, s_n)), \Phi(f_2)(\omega(s_1, \ldots, s_n))) \\
= \delta(\omega_M(\Phi(f_1)(s_1), \ldots, \Phi(f_1)(s_n)), \omega_M(\Phi(f_2)(s_1), \ldots, \Phi(f_2)(s_n))) \\
\leq \max \{ \delta(\Phi(f_1)(s_i), \Phi(f_2)(s_i)) : 1 \leq i \leq n \} \quad \text{(by Definition 3.5)} \\
\leq \sup \{ \delta(f_1(\zeta), f_2(\zeta)) : \zeta \in \mathcal{L}^g \} \quad \text{(by induction hypothesis)}
\]

**ad 2.** By structural induction on the syntax of \( \zeta \in \mathcal{L}^g \).

*Basis of induction:* If \( \zeta = a \in \text{Const} \) then \( f_i(a) = a_M \) and \( \delta(f_1(a), f_2(a)) = 0 \leq \frac{1}{2} \delta(f_1, f_2) \).

*Induction step:* \( \zeta = \omega(s_1, \ldots, s_{n-k}, s_{n-k+1}, \ldots, s_k) \) where \( \omega \in \mathcal{L} \), \( |\omega| = n \geq 1 \), \( \deg(\omega) = k \), \( \zeta_1, \ldots, \zeta_{n-k} \in \mathcal{L}^g \) and \( s_1, \ldots, s_k \in \mathcal{L} \). Then

\[
f_i(\zeta) = \omega_M(f(\zeta_1), \ldots, f(\zeta_{n-k})), f(s_1), \ldots, f(s_k)), \quad i = 1, 2.
\]

and by induction hypothesis:

\[
\delta(f_1(\zeta), f_2(\zeta)) \leq \max \left\{ \max_{1 \leq i \leq n-k} \delta(f_1(\zeta_i), f_2(\zeta_i)), \frac{1}{2} \max_{1 \leq j \leq k} \delta(f_1(s_j), f_2(s_j)) \right\} \\
\leq \frac{1}{2} \delta(f_1, f_2). \quad \square
\]

Since \( \mathcal{L}(\emptyset, Idf) \to M \) is a complete metric space and since \( \Phi[\sigma] \) is a contracting self-mapping of \( \mathcal{L}(\emptyset, Idf) \to M \) (Lemma 3.6) \( \Phi^M[\sigma] \) has exactly one fixed point.

**Definition 3.7.** Let \( \sigma \) be declaration over \( (\emptyset, Idf) \) and let \( M \) be an \( \emptyset \)-cms. The unique fixed point of \( \Phi^M[\sigma] \) is denoted by \( f^M_\sigma \).
Lemma 3.8. Let $\mathcal{O}$ be a symbol-algebra with guardedness conditions and let $\sigma$ be a declaration over $(\mathcal{O}, Idf)$. If $M$ is an $\mathcal{O}$-cms then $f^M_\sigma$ is a homomorphism $\mathcal{L}(\mathcal{O}, Idf) \to M$ with

$$f^M_\sigma(x) = f^M_\sigma(\sigma(x)) \quad \forall x \in Idf.$$

Proof. It is easy to see that $f^M_\sigma$ is a homomorphism. If $x \in Idf$ then

$$f^M_\sigma(x) = \Phi^M[\sigma](f^M_\sigma)(x) = f^M_\sigma(\sigma(x)).$$

Theorem 3.9. Let $\mathcal{O} = (Op, \cdot, deg)$ be a symbol-algebra with guardedness conditions. For each $\mathcal{O}$-cms $M$ there exists a unique homomorphism

$$F^M: \mathcal{L}(\mathcal{O}, Idf) \to (\mathcal{O}, Idf) \to M$$

with $F^M(x)(\sigma) = F^M(\sigma(x))$ for all $x \in Idf$ and $\sigma \in \mathcal{O}(\mathcal{O}, Idf)$.

Proof. Let $\mathcal{L} = \mathcal{L}(\mathcal{O}, Idf)$, $\mathcal{D} = \mathcal{D}(\mathcal{O}, Idf)$.

Existence: The function $F^M$ is defined by $F^M(s): \mathcal{D} \to M$, $F^M(s)(\sigma) = f^M_\sigma(s)$ for all statements $s \in \mathcal{L}$, $\sigma \in \mathcal{D}$. Since $f^M_\sigma$ is a fixed point of $\Phi^M[\sigma]$ we get by Theorem 2.12 that $F^M$ is a homomorphism with the recursion condition.

Uniqueness: Let $\sigma$ be a fixed declaration over $(\mathcal{O}, Idf)$ and $F: \mathcal{L} \to (\mathcal{O}, Idf)$ a homomorphism with the recursion condition. Let $f$ be given by

$$f: \mathcal{L} \to M, \quad f(s) = F(s)(\sigma).$$

By Theorem 2.12 $f$ is fixed point of $\Phi^M[\sigma]$. Since $\Phi^M[\sigma]$ is a contracting self-mapping of the complete metric space $\mathcal{L} \to M$, $\Phi^M[\sigma]$ has exactly one fixed point (Banach’s fixed point theorem). We conclude that $f = f^M_\sigma$ for each declaration $\sigma \in \mathcal{D}$ and therefore $F = F^M$.

Theorem 3.10. Let $\mathcal{O} = (Op, \cdot, deg)$ be a symbol-algebra with guardedness conditions and let $M$ be an $\mathcal{O}$-cms. Then

$$M^M: \mathcal{D}(\mathcal{O}, Idf) \to M, \quad M^M(\langle \sigma, s \rangle) = F^M(s)(\sigma)$$

is the unique meaning function which satisfies the following conditions:

1. $M^M(\langle \sigma, a \rangle) = a_M$ where $a \in \text{Const}(\mathcal{O})$.
2. $M^M(\langle \sigma, x \rangle) = M^M(\langle \sigma, \sigma(x) \rangle)$ for all $x \in Idf$.
3. $M^M(\langle \sigma, \omega(s_1, \ldots, s_n) \rangle) = \omega_M(M^M(\langle \sigma, s_1 \rangle), \ldots, M^M(\langle \sigma, s_n \rangle))$ for all statements $s_1, \ldots, s_n \in \mathcal{L}(\mathcal{O}, Idf)$ and $\omega \in Op$, $|\omega| = n \geq 1$.

In addition we have that if $s$ is a primitive statement then $M^M(\langle \sigma, s \rangle)$ is independent of the declaration $\sigma$.

Proof. Follows immediately by Theorem 3.9.
Because of Theorem 3.10 we may write \( Me^M(s) \) instead of \( Me^M(\langle \sigma, s \rangle) \) for each primitive statement \( s \). By the uniqueness of a meaning function with the recursion condition (Theorem 3.10) we get the following consistency result.

**Theorem 3.11.** Let \( \mathcal{C} = (\mathbb{O}, \cdot, \deg) \) be a symbol-algebra with guardedness conditions and let \( M \) and \( N \) be two \( \mathcal{C} \)-cns. If \( f: M \to N \) is a homomorphism from \( M \) to \( N \) then the meaning functions \( Me^M \) and \( Me^N \) are consistent with respect to \( f \), i.e.,

\[
 f \circ Me^M = Me^N.
\]

**Proof.** By Theorem 2.13 we have that \( f \circ Me^M \) is a meaning function which satisfies the recursion condition. By the uniqueness of the meaning function in Theorem 3.10 we get \( f \circ Me^M = Me^N \). \( \square \)

**Remark 3.12.** Note that in Theorem 3.11 the homomorphism \( f \) need not be continuous.

### 4. \( \Sigma \)-algebras in the cpo approach

In this section we interpret the cpo approach in our algebraic context. We show that the meaning function can be defined as the least homomorphism

\[
 F: \mathcal{L}(\Sigma, Idf) \to (\mathcal{O}(\mathcal{L}, Idf) \to M)
\]

with the recursion condition. In contrast to the metric case it is possible that there exist other compositional meaning functions with the recursion condition. Hence we cannot guarantee the consistency of the meaning function of homomorphic \( \Sigma \)-cpo’s.

**Definition 4.1.** Let \( \Sigma = (\mathbb{O}, \cdot, \cdot) \) be a symbol-algebra. A \( \Sigma \)-cpo is a \( \Sigma \)-algebra \( D \) which satisfies the following condition: There exists a partial order \( \preceq_D \) on \( D \) such that \( (D, \preceq) \) is a complete partial order (cpo) and such that for each operator symbol \( \omega \in \mathbb{O}, |\omega| = n \geq 1 \), the associated operator \( \omega_D: D^n \to D \) is continuous with respect to \( \preceq \).

**Remark 4.2.** If \( D \) is a \( \Sigma \)-cpo then also the \( \Sigma \)-algebra \( A \to D \) where \( A \) is an arbitrary set is a \( \Sigma \)-cpo where the partial order \( \preceq \) on \( A \to D \) is given by

\[
 f_1 \preceq f_2 \iff f_1(\sigma) \preceq_D f_2(\sigma) \quad \forall \sigma \in \mathcal{L}(\Sigma, Idf)
\]

In particular the space \( \mathcal{L}(\Sigma, Idf) \to D \) and also the space \( \mathcal{L}(\Sigma, Idf') \to (\mathcal{L}(\Sigma, Idf') \to D) \) are \( \Sigma \)-cpo’s.
Theorem 4.3. Let $\Sigma = (O_p, \cdot \cdot \cdot)$ be a symbol-algebra and $D$ a $\Sigma$-cpo. Then there exists a least homomorphism

$$F^D : \mathcal{L}(\Sigma, Idf) \rightarrow (\mathcal{D}(\Sigma, Idf) \rightarrow D)$$

which satisfies the recursion condition $F^D(x)(\sigma) = F^D(\sigma(x))(\sigma)$ for all $x \in Idf$ and $\sigma \in \mathcal{D}(\Sigma, Idf)$.

Proof. It is easy to see that for fixed declaration $\sigma$ the function $\Phi^D[\sigma]$ is cpo-continuous. By Tarski's fixed point theorem there exists a least point $f_\sigma^D$ of $\Phi^D[\sigma]$. Then we define

$$F^D(s) : \mathcal{D}(\Sigma, Idf) \rightarrow D, \quad F^D(s)(\sigma) = f_\sigma^D(s).$$

By Theorem 2.12 $F^D$ is a homomorphism which satisfies the recursion condition. Now we have to show that $F^D$ is the least homomorphism which satisfies the recursion condition. Let $F$ be a homomorphism with the recursion condition and $\sigma \in \mathcal{D}(\Sigma, Idf)$. We have to show that $F^D(\sigma) \subseteq F(\sigma)$ for all $s \in \mathcal{L}(\Sigma, Idf)$ and $\sigma \in \mathcal{D}(\Sigma, Idf)$.

Let $\sigma$ be a fixed declaration. We define

$$f : \mathcal{L}(\Sigma, Idf) \rightarrow D, \quad f(s) = F(s)(\sigma).$$

Since $F$ is a homomorphism with the recursion condition $f$ is fixed point of $\Phi^D[\sigma]$ (Theorem 2.12). Therefore $f_\sigma^D \subseteq f$. We conclude that $F^D(\sigma) \subseteq F(\sigma)$ for all $s \in \mathcal{L}(\Sigma, Idf)$.

Theorem 4.4. Let $\Sigma = (O_p, \cdot \cdot \cdot)$ be a symbol-algebra and $D$ a $\Sigma$-cpo. Then

$$Me^D : \mathcal{P}(\Sigma, Idf) \rightarrow D, \quad Me^D(\langle \sigma, s \rangle) = F^D(s)(\sigma)$$

is the least meaning function $G : \mathcal{P}(\Sigma, Idf) \rightarrow D$ which satisfies the following conditions:

1. $G(\langle \sigma, a \rangle) = a_\sigma$ where $a \in \text{Const}(\Sigma)$.
2. $G(\langle \sigma, x \rangle) = G(\langle \sigma, a(x) \rangle)$ for all $x \in Idf$.
3. $G(\langle \sigma, o(s_1, \ldots, s_n) \rangle) = o_\sigma(G(\langle \sigma, s_1 \rangle), \ldots, G(\langle \sigma, s_n \rangle))$ for all statements $s_1, \ldots, s_n \in \mathcal{L}(\Sigma, Idf)$ and $\sigma \in \mathcal{D}(\Sigma, Idf)$ and $|\sigma| = n \geq 1$.

Proof. It is easy to see that $Me^D$ satisfies the conditions (1)–(3). Let $G$ be a meaning function $\mathcal{P}(\Sigma, Idf) \rightarrow D$ for which the conditions (1)–(3) are true. We have to show that $Me^D \subseteq G$.

Let $F : \mathcal{L}(\Sigma, Idf) \rightarrow (\mathcal{P}(\Sigma, Idf) \rightarrow D)$ be given by $F(s)(\sigma) = G(\langle \sigma, s \rangle)$. Then $F$ is a homomorphism with the recursion condition. By Theorem 4.3 we conclude $F^D \subseteq F$.

Then

$$Me^D(\langle \sigma, s \rangle) = F^D(s)(\sigma) \subseteq F(s)(\sigma) = G(\langle \sigma, s \rangle)$$

for all $s \in \mathcal{L}(\Sigma, Idf)$ and $\sigma \in \mathcal{D}(\Sigma, Idf)$. Therefore $Me^D \subseteq G$. $\square$
Remark 4.5. If \( D \) is a \( \Sigma \)-cpo then the meaning \( Me^D(\langle \sigma, x \rangle) \) of a recursive program \( \langle \sigma, x \rangle \in \mathcal{P}(\Sigma, \{x\}) \) is the supremum of its finite approximations, i.e. \( \xi_0 \sqsubseteq \xi_1 \sqsubseteq \xi_2 \sqsubseteq \ldots \) and

\[
Me^D(\langle \sigma, x \rangle) = \bigsqcup_{i=0}^{\infty} \xi_i,
\]

where \( \xi_0 = \bot_D, \xi_{i+1} = \Psi[\sigma(x)](\xi_i) \). Here the functions \( \Psi[s] : D \to D \) are defined by structural induction on the syntax of \( s \in \mathcal{L}(\Sigma, \{x\}) \):

\[
\Psi[a](\xi) = a_B \quad \text{for all } a \in \text{Const}(\Sigma),
\]

\[
\Psi[x](\xi) = \xi,
\]

\[
\Psi[\ast_1(s_1, \ldots, s_n)](\xi) = \psi_D(\Psi[s_1](\xi), \ldots, \Psi[s_n](\xi)).
\]

Proof. Let \( \sigma \) be a declaration over \( \langle \Sigma, \{x\} \rangle \). We know by Tarski’s fixed point theorem that the sequence \( (f_i)_{i \geq 0} \) which is given by

\[
f_0 = \lambda x. \bot_D, \quad f_{i+1} = \Phi^D[\sigma](f_i)
\]

is monotone and the supremum \( \bigsqcup f_i \) is the least fixed point of \( \Phi^D[\sigma] \). In particular we have

\[
Me^D(\langle \sigma, x \rangle) = \bigsqcup_{i=0}^{\infty} f_i(x).
\]

Let \( f : \mathcal{L}(\Sigma, \{x\}) \to D \) be a homomorphism. Then we get by structural induction on the syntax of \( s \) that \( \Psi[s](f(x)) = f(s) \). Since for each natural number \( i \) the function \( f_i \) is a homomorphism we get with \( \sigma(x) = s \):

\[
f_{i+1}(x) = f_i(\sigma(x)) = \Psi[\sigma(x)](f_i(x)).
\]

If the sequence \( (\xi_i)_{i \geq 0} \) is defined by \( \xi_0 = \bot_D \) and \( \xi_{i+1} = \Psi[\sigma(x)](\xi_i) \), then we get by induction on \( i \) that \( \xi_i = f_i(x) \) for all \( i \geq 0 \). Therefore \( \xi_0 \sqsubseteq \xi_1 \sqsubseteq \xi_2 \sqsubseteq \ldots \) and

\[
Me^D(\langle \sigma, x \rangle) = \bigsqcup_{i=0}^{\infty} f_i(x) = \bigsqcup_{i=0}^{\infty} \xi_i.
\]

Remark 4.6. In the metric aproach we obtained a consistency result in the form that whenever \( f \) is a homomorphism between two \( \ell \)-cms then the meaning functions are consistent (Theorem 3.11). This result is wrong when we deal with \( \Sigma \)-cpo’s even if we require that \( f \) is cpo-continuous.

Let \( \Sigma = \text{Const} \cup \{a, \gamma\} \) where \( \text{Const} = \{a, h\} \), \( |a| = 1 \), \( |\gamma| = 2 \), and \( \text{Idf} = \{x\} \). We consider the following \( \Sigma \)-cpo’s:
(1) \( D = \mathbb{N}_0 \cup \{ \infty \} \) together with the operators \( a_D = 0, b_D = 1, \gamma_D \) is the addition on natural numbers which is extended on \( \infty \) in the following way:

\[
n + \infty = \infty + n = \infty \quad \forall n \in D,
\]

\( \omega_D : D \rightarrow D \) is the identical function, i.e. \( \omega_D(n) = n \) for all \( n \in D \).

(2) \( E = \mathbb{N}_0 \cup \{ \infty \} \) together with the operators \( a_E = 1, b_E = 2, \gamma_E \) is the usual multiplication on natural numbers which is extended on \( \infty \) in the following way:

\[
n \cdot \infty = \infty \cdot n = \infty \quad \forall n \in E,
\]

\( \omega_E : E \rightarrow E \) is the identical function, i.e. \( \omega_E(n) = n \) for all \( n \in E \).

The partial order \( \sqsubseteq \) on \( D \) resp. \( E \) is the usual order \( \leq \) of natural number with \( \infty \) as the top element. Then \( D \) resp. \( E \) is a cpo and the operators \( \gamma_D = +, \gamma_E = \cdot, \omega_D = \omega_E = id \) are cpo-continuous.

It is easy to see that \( f : D \rightarrow E, f(n) = 2^n \) if \( n \) is a natural number, \( f(\infty) = \infty \), is a homomorphism. Now we show that the result corresponding to the consistency result in the metric approach (Theorem 3.11) is wrong.

Let \( \mathcal{L} = \mathcal{L}(\Sigma, \text{Idf}), \mathcal{D} = \mathcal{D}(\Sigma, \text{Idf}) \). We consider the declaration \( \sigma \) where \( \sigma(x) = \omega(x) \). Let \( f^D : \mathcal{L} \rightarrow D \) resp. \( f^E : \mathcal{L} \rightarrow E \) be defined by

\[
f^D(s) = F^D(s)(\sigma), \quad f^E(s) = F^E(s)(\sigma).
\]

Since \( F^D \) resp. \( F^E \) is the least homomorphism \( \mathcal{L} \rightarrow (\mathcal{D} \rightarrow D) \) resp. \( \mathcal{L} \rightarrow (\mathcal{D} \rightarrow E) \) which satisfies the recursion condition we get that \( f^D \) resp. \( f^E \) is the least fixed point of \( \Phi^D[\sigma] \) resp. \( \Phi^E[\sigma] \). We get

(i) \( f^D(a) = 0, f^D(b) = 1 \) and \( f^D(x) = 0 \),

(ii) \( f^E(a) = 1, f^E(b) = 2 \) and \( f^E(x) = 0 \).

On the other side if the consistency result \( f \circ \text{Me}^D = \text{Me}^E \) would be true then

\[
(f \circ f^D)(s) = f(F^D(s)(\sigma)) = f(\text{Me}^D(\langle \sigma, s \rangle)) = \text{Me}^E(\langle \sigma, s \rangle) = F^E(s)(\sigma) = f^E(s)
\]

for all \( s \in \mathcal{L}(\Sigma, \text{Idf}) \). With \( s = x \) we get \( 0 = f^E(x) = f(f^D(x)) = f(0) = 1 \), a contradiction.

\textbf{Remark 4.7.} One might think that if the given homomorphism is the least homomorphism \( f : D \rightarrow E \) then \( f \circ \text{Me}^D \) equals \( \text{Me}^E \). This is also wrong: We show that in our example there exist exactly two homomorphisms \( D \rightarrow E \). First the homomorphism \( f' \) defined as above: \( f(n) = 2^n \) if \( n \in \mathbb{N}_0 \), \( f(\infty) = \infty \). Second \( f' : D \rightarrow E, f'(n) = 2^n \) if \( n \in \mathbb{N}_0 \), \( f'(\infty) = 0 \).

It is easy to see that \( f' \) is indeed a homomorphism \( D \rightarrow E \). Let \( g : D \rightarrow E \) be a homomorphism. We show by induction on \( n, n \geq 0 \) that \( g(n) = 2^n \). In the cases \( n = 0 \) or \( n = 1 \) we have \( g(0) = g(a_D) = a_E = 1 \) and \( g(1) = g(b_D) = b_E = 2 \). If \( n \) is a natural number
$\geq 2$ then by induction hypothesis
\[ g(n) = g((n - 1) + 1) = g(n - 1) \cdot 2^{n-1} = 2^n. \]

Now we have $g(n) = 2^n$ for each natural number $n$. We have to show that $g(x) = 0$ or $\infty$.
\[ g(x) = g(1 + x) = g(1) \cdot g(x) = 2 \cdot g(x). \]

Then either $g(x) = 0$ or $g(x) = \infty$. In the first case $g = f'$. In the second case $g = f$.

It is clear that $f' \subseteq f$. Therefore $f'$ is the least homomorphism $D \to E$. But then again the consistency result $f' \cdot \mathcal{M}^{D} = \mathcal{M}^{E}$ is false: As below we consider the declaration $\sigma$ with $\sigma(x) = \omega(x)$. We saw above $\mathcal{M}^{D}(\langle \sigma, x \rangle) = 0$ and $\mathcal{M}^{E}(\langle \sigma, x \rangle) = 0$:
\[ (f' \cdot \mathcal{M}^{D})(\langle \sigma, x \rangle) - f'(\mathcal{M}^{D}(\langle \sigma, x \rangle)) - f'(0) - 1 \neq 0 - \mathcal{M}^{E}(\langle \sigma, x \rangle). \]

**Theorem 4.8.** Let $\Sigma$ be a symbol-algebra and let $D,E$ be $\Sigma$ cpo's. If there exists a cpo-continuous homomorphism $f : D \to E$ such that $f(\bot_D) = \bot_E$ where $\bot_D$ resp. $\bot_E$ denotes the bottom element of $D$ resp. $E$ then $f \cdot \mathcal{M}^{D} = \mathcal{M}^{E}$.

**Proof.** First we show that for fixed declaration $\sigma$ $f \cdot f_\sigma^D = f_\sigma^E$ where $f_\sigma^D$ resp. $f_\sigma^E$ denotes the least fixed point of $\Phi^D[\sigma]$ resp. $\Phi^E[\sigma]$. By Tarski’s fixed point theorem we know that
\[ f_\sigma^D = \bigsqcup_{i=0}^{\omega} f_i, \quad f_\sigma^E = \bigsqcup_{i=0}^{\omega} g_i, \]
where $\bigsqcup_{i=0}^{\omega} h_i$ denotes the least upper bound of the (monotone) sequence $(h_i)_{i \geq 0}$ and where
\[ f_i : \mathcal{P}(\Sigma, 1D) \to D, \quad g_i : \mathcal{P}(\Sigma, 1E) \to E \]
are defined by induction on $i$:
\[ f_0(s) = \bot_D, \quad g_0(s) = \bot_E \quad \forall s \in \mathcal{P}(\Sigma, 1D), \]
\[ f_{i+1} = \Phi^D[\sigma](f_i), \quad g_{i+1} = \Phi^E[\sigma](g_i). \]

We show by induction on $i$ that $f \cdot f_i = g_i$.

**Basis of induction:** For all $s \in \mathcal{P}(\Sigma, 1D)$ we have
\[ (f \cdot f_0)(s) = f(f_0(s)) = f(\bot_D) = \bot_E = g_0(s). \]

**Induction step** $i \Rightarrow i+1$: By induction hypothesis we have $f \cdot f_i = g_i$. We show by structural induction on the syntax of $s \in \mathcal{P}(\Sigma, 1D)$ that $f(f_{i+1}(s)) = g_{i+1}(s)$.
If $s = a \in \text{Const}(\Sigma)$ then $f_{i+1}(a) = \Phi^D[\sigma](f_i)(a) = a_D$, $g_{i+1}(a) = \Phi^K[\sigma](a) = a_E$. Since $f$ is a homomorphism we have

$$f(f_{i+1}(a)) = f(a_D) = a_E = g_{i+1}(a).$$

If $s = x \in \text{Idf}$ then

$$f_{i+1}(x) = \Phi^D[\sigma](f_i)(x) = f_i(\sigma(x)),
\quad g_{i+1}(x) = \Phi^K[\sigma](g_i)(x) = g_i(\sigma(x)).$$

By induction hypothesis we get

$$f(f_{i+1}(x)) = f(f_i(\sigma(x))) = g_i(\sigma(x)) = g_{i+1}(x).$$

If $s = \omega(s_1, \ldots, s_n), \omega \in \text{Op}, |\omega| = n \geq 1$, then we have by induction hypothesis applied to the statements $s_1, \ldots, s_n$: $f(f_{i+1}(s_j)) = g_{i+1}(s_j), j = 1, \ldots, n$. By definition of $\Phi^D[\sigma]$ resp. $\Phi^K[\sigma]$ we get

$$f(f_{i+1}(s)) = f(\omega_D(f_i(s_1), \ldots , f_i(s_n)))
\quad = \omega_E(f(f_i(s_1), \ldots , f_i(s_n)))
\quad = \omega_E(g_i(s_1), \ldots , g_i(s_n)) = g_{i+1}(s).$$

Since $f$ is cpo-continuous we have

$$f(f^D_D(s)) = f\left(\bigcup_{i=0}^{\infty} f_i(s)\right) = \bigcup_{i=0}^{\infty} f(f_i(s)) = \bigcup_{i=0}^{\infty} g_i(s) = f^E_E(s)$$

for all $s \in \mathcal{P}(\Sigma, \text{Idf})$. We conclude: $f^E_E = f \circ f^D_D$.

Since $F^D_D(s)(\sigma) = f^D_D(s), F^E_E(s)(\sigma) = f^E_E(s)$ as we saw in the proof of Theorem 4.3 we get

$$f(Me^D_D(\langle \sigma, s \rangle)) = f(F^D_D(s)(\sigma)) = f(f^D_D(s))
\quad = f^E_E(s) = F^E_E(s)(\sigma) = Me^E_E(\langle \sigma, s \rangle).$$

5. Denotational event structure semantics

Event structures endowed with cpo have been proposed by [14, 16] as semantic domain for CCS. Event structures with metric have been used in [3] to model TCSP. We briefly sketch that the latter approach carries easily over to CCS and $\Sigma_0$ and then we discuss in Section 5.5 the issue cpo versus metric for event structures.

5.1. Prime event structures

A natural domain for modelling a representation for processes which formally allows to distinguish between parallelism and arbitrary choice, i.e. a noninterleaving
representation, is the class of event structures, a special kind of directed graphs, which were introduced in [11]. Each node, called event, represents an action occurrence \( a \in \text{Act} \) of a process. The edges describe causal dependency between actions. In addition, there is a relation, called conflict relation, on the nodes which contains all those pairs of events which exclude each other. Events that are neither in conflict nor causally dependent may be executed in parallel, i.e. independently. Many different types of event structures have been defined (see [15]). Operators on event structures for modelling the TCSP-operator symbols are defined in [3]. References [14,16] introduce operators on event structures corresponding to the CCS-operator symbols.

**Definition 5.1.** \( \varepsilon = (E, \leq, \neq, l) \) is called a prime event structure iff \( E \) is a set (of events), \( \leq \) is a partial order on \( E \) (i.e. \( \leq \) is a transitive, reflexive, antisymmetric relation on \( E \)), \( l: E \to \text{Act} \) is a function, called the labelling function and \( \neq \) is an irreflexive, symmetric relation on \( E \), called conflict relation, which satisfies the conflict inheritance

\[
(e_1 \leq e_2 \land e_1 \neq e_3) \Rightarrow e_3 \neq e_1
\]

and such that for each event \( e \in E \) the set \( \downarrow e = \{ e' \in E : e' \leq e \} \) is finite.

If \( \varepsilon = (E, \leq, \neq, l) \) is a prime event structure then \( \text{depth}(e) = \sup \{ \text{depth}(e) : e \in E \} \) where

\[
\text{depth}_\varepsilon(e) = \sup \{ n \in \mathbb{N}_0 : \exists e_1, \ldots, e_n \in E \text{ s.t. } e_1 < e_2 < \cdots < e_n = e \}
\]

Here \( e < e' \) means \( (e \leq e') \land (e \neq e') \).

A prime event structure \( \varepsilon \) is called finitely approximable if for each \( n \in \mathbb{N}_0 \) the set

\[
E_n[\varepsilon] = \{ e \in E : \text{depth}_\varepsilon(e) \leq n \}
\]

is finite.

In the following we write \( \text{depth}(e) \) instead of \( \text{depth}_\varepsilon(e) \) and \( E[n] \) instead of \( E_n[\varepsilon] \). Two prime event structures \( \varepsilon_1 = (E_1, \leq_1, \neq_1, l_1), i=1,2 \) are isomorphic (denoted by \( \varepsilon_1 \cong \varepsilon_2 \)) if there exists a bijective mapping \( f : E_1 \to E_2 \) such that \( l_2, f = f_1 \) and

1. \( e_1 \leq_1 e_2 \iff f(e_1) \leq_2 f(e_2) \quad \forall e_1, e_2 \in E_1. \)
2. \( e_1 \neq_1 e_2 \iff f(e_1) \neq_2 f(e_2) \quad \forall e_1, e_2 \in E_1. \)

In the following we abstract from the names of the events, i.e. we will not distinguish between isomorphic prime event structures. In the cases that the names are of importance we will speak of plain prime event structures.

**Definition 5.2.** PlainPrimeEv denotes the set of plain prime event structures (where the events belong to some fixed set). PlainFinPrimeEv denotes the subset of finitely approximable plain prime event structures.

PrimeEv denotes the set of all prime event structures (i.e. isomorphism classes of plain event structures). FinPrimeEv denotes the subset of all finitely approximable
Prime event structures can be depicted graphically by representing events as boxes (inscribed with the event label) and connecting them with their direct predecessors and successors. A conflict between two events $e_1, e_2$ is a direct conflict if no predecessor of $e_1$ is in conflict with $e_2$, no predecessor of $e_2$ is in conflict with $e_1$, and no predecessors of $e_1$ and $e_2$ are in conflict. Direct conflicts are depicted graphically by a broken line. Example: The event structure $\varepsilon=(E, \leq, \#, l)$ with $E=\{e_1, e_2, e_3\}$, $e_1 \leq e_2$, $e_1 \neq e_3$ (and $e_2 \neq e_3$), $l(e_1)=\alpha$, $l(e_2)=\beta$, $l(e_3)=\gamma$ is shown as

![Diagram of event structure]

Definition 5.3. Let $\varepsilon=(E, \leq, \#, l)$ be a prime event structure, $A \subseteq E$. $A$ is called left-closed iff each predecessor of an event $e \in A$ belongs to $A$, i.e. if $e \in A$, $e' \in E$, $e' \leq e$ then $e' \in A$. If $A$ is a left-closed subset of $E$, then the event structure $\varepsilon \upharpoonright A$ is defined by

$$\varepsilon \upharpoonright A = (A, \leq \cap A \times A, \# \cap A \times A, l \upharpoonright A).$$

It is clear that $\varepsilon \upharpoonright A$ is left-closed. The $n$-cut of an event structure $\varepsilon$ is defined by $\varepsilon[n] = \varepsilon \upharpoonright [E[n]]$.

5.2. Prime event structures as a $\Sigma_0$-, $\Sigma_{CCS}$-resp. $\Sigma_1$-algebra

Semantic operators on prime event structures for choice ($+$), prefixing, the parallel operator $|$ without synchronisation, the parallel operator $|$ with CCS-style synchronisation, restriction and relabelling have been introduced by $[3,13,14,16]$ and are to be found in the Appendix A. Here we define a semantic sequential operator.

$\varepsilon_1; \varepsilon_2$ arises from $\varepsilon_1$ when we append to each “maximal conflict-free”, left-closed and depth-finite subset $A$ of $E_1$ a copy of $\varepsilon_2$.

Definition 5.4. Let $\varepsilon=(E, \leq, \#, l)$ be a prime event structure, $A \subseteq E$. $A$ is called
- conflict-free iff $A$ does not contain conflicting events, i.e. $\neg(e \# e')$ for all $e, e' \in A$.
- maximal conflict-free iff $A$ is conflict-free and each event in $E \setminus A$ is in conflict with some event in $A$, i.e. if $e' \in E \setminus A$ then $e \# e'$ for some $e \in A$.
- depth-finite iff $\text{depth}(A) = \sup \{\text{depth}(e) : e \in A\}$ is finite. Here we take the supremum in $\mathbb{N}_0 \cup \{\infty\}$.

Definition 5.5. Let $\varepsilon_1, \varepsilon_2 \in \text{PrimeEv}$, $\varepsilon_i=(E_i, \leq_i, \#_i, l_i)$, $i=1,2$, $E_1 \cap E_2 = \emptyset$. Then

$\varepsilon_1; \varepsilon_2 = (E, \leq, \#, l)$,
where \( E = E_1 \cup \{(A,e) : A \in K, e \in E_2 \} \) and where \( K \) denotes the set of all left-closed, maximal conflict-free and depth-finite subsets of \( E_1 \). The partial order \( \leq \) on \( E \) is given by

\[
e_1 \leq e_2 \iff (e_1, e_2) \in E_1 \land e_1 \leq_1 e_2
\]

\[
\lor (e_1 \in A, e_2 = (A,e) \text{ for some } e \in E_2, A \in K)
\]

\[
\lor (e_1 = (A,e_1), e_2 = (A,e_2) \text{ for some } A \in K \land e_1 \leq_1 e_2).
\]

The conflict relation \( \# \) on \( E \) is the smallest conflict relation on \( E \) which satisfies:

1. \[e_1 \#_1 e_2 \Rightarrow e_1 \# e_2 .\]
2. \[e_1 \#_2 e_2, A \in K \Rightarrow (A,e_1) \# (A,e_2) .\]

The labelling function \( l : E \to \text{Act} \) is defined by

\[
l(e) = \begin{cases} l_1(e) : e \in E_1, \\
l_2(e') : e = (A,e') \text{ for some } A \in K, e' \in E_2. \\
\end{cases}
\]

**Example 5.6.** Let \( e_1 \) be

\[\begin{array}{c}
\chi \\
\Downarrow \\
\beta
\end{array}
\]

and let \( e_2 \) be

\[\begin{array}{c}
\chi \\
\Downarrow \\
\beta
\end{array}
\]

Then \( e_1 ; e_2 \) is given by

\[\begin{array}{c}
\chi \\
\Downarrow \\
\beta
\end{array}
\]

**Example 5.7.** Let \( e_1 \) resp. \( e_2 \) be given by

\[\begin{array}{c}
\beta \\
\Downarrow \\
\chi
\end{array}
\]

\[\begin{array}{c}
\chi \\
\Downarrow \\
\beta
\end{array}
\]
Then $e_1; e_2$ is given by

Prime event structures together with the operators $\|$, $+$ and $;$ form a $\Sigma_0$-algebra where the constant symbol $\alpha \in Act$ of $\Sigma_0$ is represented by the prime event structure which consists of a single event labelled by $\alpha$.

**Remark 5.8.** As an operator on $PrimeEv$ is not distributive over $+$. For example: Let $e_i = [\ell], \ i = 1, 2, 3$.

then

$$e_1; (e_2 + e_3) = [\ell] \neq [\ell] \to [\ell] = (e_1; e_2) + (e_1; e_3).$$

In order to be able to deal with recursion we first consider the metric approach.

5.3. Finitely approximable prime event structures as an $\mathcal{O}_{\text{CS}}$-cms resp. $\mathcal{O}_1$-cms resp. $\mathcal{O}_0$-cms

Following [3] we define the distance $d(e_1, e_2)$ of event structures $e_1, e_2$.

**Definition 5.9.** Let $d: FinPrimeEv_0 \times FinPrimeEv_0 \to [0, 1]$ be defined by

$$d(e_1, e_2) = \inf \left\{ \frac{1}{2^n} : \text{hom}[n] = e_2[n] \right\}.$$

It is shown in [3] that $(FinPrimeEv_0, d)$ is a complete ultrametric space. Since $FinPrimeEv$ is a closed subspace of $FinPrimeEv_0$ $FinPrimeEv$ is also a complete ultrametric space.

**Remark 5.10.** Applying Definition 5.9 to arbitrary prime event structures yields the problem that the distance of two distinct prime event structures may be 0. Hence $PrimeEv$ is a pseudo-metric space, but not a metric space.
\textit{FinPrimeEv} is closed under +, $\|$, $\odot$, prefixing, restriction and relabelling, i.e. if \( \varepsilon_1, \varepsilon_2 \in \text{FinPrimeEv} \) then \( \text{op}'(\varepsilon_1, \varepsilon_2) \in \text{FinPrimeEv} \) where \( \text{op} \in +, \|, \odot \) and where \( \text{op}' \) is a prefixing, restriction or relabelling operator. \( \text{FinPrimeEv} \) is closed under +, $\|$ and $\odot$. Since \( \emptyset \in \text{FinPrimeEv} \) and since the event structures which consists of a single event labelled by \( \varepsilon \in \text{Act} \) belongs to \( \text{FinPrimeEv} \) we get: \( \text{FinPrimeEv}_0 \) is a \( \Sigma_{\text{CCS}} \)-resp. \( \Sigma_1 \)-algebra, \( \text{FinPrimeEv} \) is a \( \Sigma_0 \)-algebra. In [3] the semantic operators for TCSP are shown to be nondistance increasing resp. contracting. Similarly we get the following theorem.

\textbf{Theorem 5.11.} The nondeterminism operator $+$, the parallel operators $\|$, $\odot$, the restriction and the relabelling operators and the sequence operator; as operators on \( \text{FinPrimeEv}_0 \) are nondistance increasing. The prefixing operators on \( \text{FinPrimeEv}_0 \) are contracting. In addition the sequence operator, as an operator on \( \text{FinPrimeEv} \) is contracting in its second argument. More precisely
\[
d(\varepsilon_1, \varepsilon_2, \varepsilon'_1, \varepsilon'_2) \leq \max \{ d(\varepsilon_1, \varepsilon_1'), \frac{1}{2} d(\varepsilon_2, \varepsilon_2') \},
\]
\[
d(\varepsilon, \varepsilon, \varepsilon) = \frac{1}{2} \cdot d(\varepsilon, \varepsilon')
\]
for all \( \varepsilon, \varepsilon', \varepsilon_1, \varepsilon_2 \in \text{FinPrimeEv}_0, \varepsilon_1, \varepsilon'_1 \in \text{FinPrimeEv}. \)

\textbf{Proof.} Easy verification. \( \square \)

By Theorem 5.11 we get the following theorem.

\textbf{Theorem 5.12.} \( \text{FinPrimeEv}_0 \) is an \( \mathcal{C}_{\text{CCS}} \)-cms and an \( \mathcal{C}_1 \)-cms. \( \text{FinPrimeEv} \) is an \( \mathcal{C}_0 \)-cms.

By Theorem 3.10 we get event structure meanings for the languages \( \mathcal{P}(\mathcal{C}_{\text{CCS}}, \text{Idf}) \) and \( \mathcal{P}(\mathcal{C}_0, \text{Idf}) \). In the following we write \( ev_{\text{cm}}(\langle \sigma, s \rangle) \) instead of \( M \langle \sigma, s \rangle \) where \( M = \text{FinPrimeEv}_0 \) or \( M = \text{FinPrimeEv} \). The resulting event structure semantics is a branching time since; is not distributive over $+$. 

\textbf{Remark 5.13.} The reason why we deal with nonempty event structures in the case \( \mathcal{C} = \mathcal{C}_0 \) is that the sequence operator on \( \text{FinPrimeEv}_0 \) (instead of \( \text{FinPrimeEv} \)) is not contracting in its second argument: If \( \varepsilon \neq \varepsilon' \) then \( d(\emptyset; \varepsilon, \emptyset; \varepsilon') = d(\varepsilon, \varepsilon') \).

\textbf{Example 5.14.} Let \( s \) be the following primitive statement: \( s = (z \parallel z); (\beta + \beta; (\gamma \parallel \delta)). \) Then

\[
\begin{align*}
\text{ev}_{\text{cm}}(s) &= \varepsilon_2 \rightarrow \beta \\
&\quad \text{or} \\
&\quad \varepsilon_1 \rightarrow \beta \\
&\quad \delta \\
&\quad \gamma
\end{align*}
\]
If $\sigma$ is a declaration with $\sigma(x) = (x + \beta) \cdot x$ then
\[
ev_{\text{cms}}(\langle \sigma, s \rangle) = \ldots
\]

5.4. Plain prime event structures as a $\Sigma_{\text{ccs}}$ resp. $\Sigma_1$-cpo

Winskel gave in [14] a denotational semantics for CCS using plain prime event structures. Plain prime event structures together with Winskels partial order form a $\Sigma_{\text{ccs}}$ resp. $\Sigma_1$-cpo.

Definition 5.15. (Winskel [14]). If $\varepsilon = (E, \leq, \#, I)$ and $\varepsilon'$ are plain prime event structures then
\[
\varepsilon' \sqsubseteq \varepsilon : \iff \varepsilon' = \varepsilon \upharpoonright A \quad \text{for some left-closed subset } A \text{ of } E.
\]

It is easy to see that $\sqsubseteq$ is indeed a partial order on $\text{PlainPrimeEv}$ which turns $\text{PlainPrimeEv}$ into a cpo with bottom element $\emptyset$. It should be noted that the analogous definition for $\text{PrimeEv}$ yields a preorder which is not complete (see Appendix B). For semantic purposes the fact that this is a preorder does not cause severe problems for the following reason: semantic operators on $\text{PlainPrimeEv}$ can be adapted to the case of $\text{PrimeEv}$ in such a way that the isomorphism class of a fixed point of an operator on $\text{PlainPrimeEv}$ is a fixed point of the adapted operator on $\text{PrimeEv}$. In particular, Winskel gave continuous operators for CCS on $\text{PlainPrimeEv}$ that adapted to $\text{PrimeEv}$ coincide with our operators on $\text{PrimeEv}$. $\text{PlainPrimeEv}$ is a $\Sigma_{\text{ccs}}$-cpo and $\Sigma_1$-cpo. By Theorem 4.4 there exists a least compositional meaning function
\[
ev_{\text{cpo}} : \mathcal{P}(\Sigma, \text{Idf}) \rightarrow \text{PlainPrimeEv}
\]
with recursion condition where $\Sigma = \Sigma_{\text{ccs}}$ or $\Sigma = \Sigma_1$. $\ev_{\text{cpo}}$ coincides with Winskels semantics. Still, it can be considered as a drawback of the cpo-based event structure approach that one has to work with plain prime event structures instead of isomorphism classes. The definition of operators tends to become awkward as the names of events are relevant. In the metric approach these difficulties do not arise.

5.5. Prime event structures as $\Sigma$-cpo versus $\Theta$-cms

In this section we give a brief account of the pros and contras for the $\Sigma$-cpo approach resp. the $\Theta$-cms approach in the case of event structures.
A first observation is that if \( \sigma \in \mathcal{P}(C_{\text{CCS}}, Id) \) (resp. \( \mathcal{P}(C_1, Id) \)) then the semantics coincide, i.e.
\[
ev_{\text{cpo}}(\langle \sigma, s \rangle) = \left[ \ev_{\text{cpo}}(\langle \sigma, s \rangle) \right]
\]
for all statements \( s \in \mathcal{P}(C_{\text{CCS}}, Id) = \mathcal{P}(C_1, Id) \). As \( \ev_{\text{cpo}} \) can handle unguarded recursion, e.g. \( \ev_{\text{cpo}}(\langle \sigma, s \rangle) = 0 \) where \( \sigma(x) = x \), \( \ev_{\text{cpo}} \) could be considered as an extension of \( \ev_{\text{ems}} \). Consequently for each process \((\langle \sigma, s \rangle) \in \mathcal{P}(C_{\text{CCS}}, Id) \) (resp. \( \mathcal{P}(C_1, Id) \)) the plain prime event structure \( \ev_{\text{cpo}}(\langle \sigma, s \rangle) \) is finitely approximable, i.e. the range of \( \ev_{\text{cpo}} \) is \( \text{PlainFinPrime}_{E_0} \). However, \( \text{PlainFinPrime}_{E_0} \) together with the partial order of Definition 5.15 does not constitute a complete partial order forcing us to choose \( \text{PlainPrime}_{E_0} \) as the range of \( \ev_{\text{cpo}} \).

**Example 5.16.** If \( \varepsilon_n = (E_n, 0, 0, l_n) \) where \( E_n = \{1, 2, \ldots, n\} \), \( l_n(i) = i, i = 1, 2, \ldots, n \), then the sequence \( (\varepsilon_n)_{n \geq 0} \) in \( \text{PlainFinPrime}_{E_0} \) is monotone but the supremum does not exist in \( \text{FinPrime}_{E_0} \). The isomorphism classes of \( s_1, s_2, s_3, \ldots \) can be depicted as follows:

\[
\begin{array}{c}
\varepsilon_1 \\
\varepsilon_2 \\
\varepsilon_3 \\
\vdots
\end{array}
\]

A second observation concerns the sequence operator. The sequence operator; on (plain) prime event structures is not monotone (and therefore not cpo-continuous).

Let \( \varepsilon_1 = (E_1, \leq_1, 0, l_1) \), \( \varepsilon_1' = (E_1, \leq_1', 0, l_1') \) and \( \varepsilon = (E, \leq, 0, l) \) where \( E_1 = \{e_1\} \), \( E_2 = \{e_1, e_2\} \), \( E_3 = \{e_1'\} \), \( l_1(e_1) = l_1'(e_1) = \gamma, l_2'(e_2) = \beta, l_3(e) = \gamma \) and \( e_1 \leq_1 e_2 \) in \( \varepsilon_1 \) then
\[
\left[ \varepsilon_1 \right]_s = [\varepsilon_1], \quad \left[ \varepsilon_1' \right]_s = [\varepsilon_1'], \quad \left[ \varepsilon \right]_s = [\varepsilon]
\]
and \( \varepsilon_1 \leq \varepsilon_1' \) whereas \( \varepsilon_1; \varepsilon \nleq \varepsilon_1'; \varepsilon \) since
\[
\left[ \varepsilon_1; \varepsilon \right]_s = [\varepsilon_1] \to [\varepsilon]
\]
\[
\left[ \varepsilon_1'; \varepsilon \right]_s = [\varepsilon_1'] \to [\varepsilon] \to [\varepsilon]
\]
We recall that in the metric approach we had also a problem with the sequence operator since \( ; \) as an operator on \( \text{FinPrime}_{E_0} \) is not contracting in its second argument (Remark 5.13). Since the restriction of \( ; \) on \( \text{FinPrime}_{E_0} \) is contracting in its second argument we were able to deal with \( \text{FinPrime}_{E_0} \) instead of \( \text{FinPrime}_{E_0} \) since \( \text{FinPrime}_{E_0} \) is also a complete metric space which is closed under \( +, \| \) and \( ; \) and which contains the associated event structure meaning of the constant symbols \( x \in \text{Const}(C_0) \). In the cpo approach we cannot restrict the semantic domain.
Denotational semantics

PlainPrimeEv to some subspace $D$ such that Theorem 4.4 can be applied: If we want to define a plain prime event structure semantics for $\mathcal{P}(\Sigma_0, Idf)$ by Theorem 4.4 which uses the semantic operators defined in Section 5.2 and the partial order $\sqsubseteq$ then we need a subset $D$ of PlainPrimeEv which is closed under $+, \parallel$ and $;$ and which satisfies the following three conditions:

(i) For each $a \in \text{Act}$ $D$ contains a plain prime event structure which consists of a single event labelled by $a$.

(ii) $D$ together with the restriction of $\sqsubseteq$ on $D$ is a cpo.

(iii) $;$ as an operator on $D$ is cpo-continuous.

But such a subset $D$ does not exist since if condition (i) is true and if $D$ is closed under $;$ then we may assume that the plain event structures $e_1, e_1', e$ of above belong to $D$. We saw that $e_1 \sqsubseteq e_1'$ and $e_1 \nsubseteq e_1'; e$. Therefore $;$ as an operator on $D$ is not monotone and condition (iii) is violated. Here we assume that $\text{Act}$ contains at least two elements $a \neq \beta$. One might think that for fixed declaration $\sigma \in \mathcal{D}(\Sigma_0, Idf)$ the function $\Phi[\sigma]$ has still a least fixed point $f_{\sigma}$ (and then the meaning of $(\sigma, s) \in \mathcal{P}(\Sigma_0, Idf)$ could be defined as $f_{\sigma}(s)$). In general this is not the case: Let $\sigma$ be defined by $\sigma(x) = x; a$ and $\sigma(y) = y$ for all $y \in Idf$, $y \neq x$. For each action $\beta \in \text{Act}$ let

$$f_\beta: \mathcal{L}(\Sigma_0, Idf) \to \text{PlainPrimeEv}$$

be the unique homomorphism with

$$f_\beta(y) = \emptyset \quad \text{for all } y \in Idf, \quad y \neq x,$$

$$f_\beta(x) = e \quad \text{where } [e]_e = [\beta] \to [\beta] \to [\beta] \to \ldots$$

Then $f_\beta \neq f_\gamma$ if $\beta \neq \gamma$ and each homomorphism $f_\beta$ is a fixed point of $\Phi[\sigma]$.

If $\sigma$ is a declaration in $\mathcal{D}(\Sigma_0, Idf)$ then we know by the metric approach that $\Phi[\sigma]$ has exactly one fixed point. One might think that the unique fixed $f_{\sigma}$ could be computed by iteration in the typical cpo-style, i.e. if $f_0 = \emptyset$ (the bottom element of the cpo $\mathcal{L}(\Sigma_0, Idf) \to \text{PlainPrimeEv}$) and $f_{i+1} = \Phi[\sigma](f_i)$ then $(f_i)_{i \geq 0}$ is a monotone sequence with $f_{\sigma}$ as least upper bound. This is also wrong since in general the sequence $(f_i)_{i \geq 0}$ is not monotone: Let $\sigma \in \mathcal{D}(\Sigma_0, Idf)$ with $\sigma(x) = x; a$ and $\sigma(y) = \beta; x; y$ and let $(f_i)_{i \geq 0}$ be defined as above. Then the isomorphism classes of $f_i(x)$ resp. $f_i(y)$ are given by

$$f_1(x): [\emptyset], \quad f_1(y): [\beta]$$

$$f_2(x): [\emptyset] \to [\emptyset], \quad f_2(y): [\beta] \to [\beta] \to [\beta]$$

$$f_3(x): [\emptyset] \to [\emptyset] \to [\emptyset], \quad f_3(y): [\beta] \to [\beta] \to [\beta]$$

We see that $f_2(y) \not\sqsubseteq f_3(y)$ and therefore $f_2 \not\sqsubseteq f_3$.

Hence we conclude that the sequence operator can be treated in the metric approach but not in the cpo approach.
6. Denotational pomset class semantics

Pomset classes are introduced in [12]. They are used to describe the true parallelism and linear time behaviour of processes. Reference [2] uses the metric space of pomset classes to define a noninterleaving semantics for the language $\mathcal{L}(\Sigma_0, Idf)$. Pomset classes form an $\mathcal{O}_0$-ems and the meaning function which is given by Theorem 3.10 coincides with the meaning function of [2]. Pomsets can be defined in various ways. One way is to look at a pomset as a conflict-free event structure.

**Definition 6.1.** A prime event structure $\mathcal{E}=(E, \leq, \#, l)$ is called conflict-free iff $\# = \emptyset$. Conflict-free prime event structures are also called pomsets.

$\text{Porn}_0$ denotes the class of all finitely approximable pomsets, $\text{Pom}$ denotes the subclass of nonempty pomsets.

If $\mathcal{E}=(E, \leq, \#, l)$ is a pomset (i.e. $\# = \emptyset$) we write shortly $\mathcal{E}=(E, \leq, l)$. It is easy to see that $\text{Porn}_0$ and $\text{Pom}$ are closed subspaces of $\text{FinPrimeEo}$. In particular $\text{Porn}_0$ and $\text{Pom}$ (with the subspace metric) are complete ultrametric spaces which are closed under prefixing, relabelling and the sequence operator; and under the parallel operator $\|$. In addition $\text{Porn}_0$ is closed under restriction.

**Definition 6.2.** Let $\text{Porn}_0^*$ denote the class of all closed subsets of $\text{Pom}$. The elements of $\text{Porn}_0^*$ are called pomset classes. $\text{Pom}^*$ denotes the subclass of all nonempty pomset classes.

**Definition 6.3.** The metric $d$ on $\text{Pom}$ induces the Hausdorff-metric on $\text{Porn}_0^*$ (which is also called $d$):

$$d(H_1, H_2) = \max \left\{ \sup_{p \in H_1} \inf_{q \in H_2} d(p, q), \sup_{p \in H_2} \inf_{q \in H_1} d(p, q) \right\}.$$ 

for all $H_1, H_2 \in \text{Pom}^*$ and

$$d(H, \emptyset) - d(\emptyset, H) = \begin{cases} 1 & \text{if } H \in \text{Pom}^*, \\ 0 & \text{if } H = \emptyset. \end{cases}$$

The restriction of $d$ on $\text{Pom}^*$ is also denoted by $d$.

In [4] it is shown that the Hausdorff-metric on the set of nonempty and closed subspaces of a complete ultrametric space yields a complete ultrametric space. Hence, $(\text{Porn}_0^*, d)$ and $(\text{Pom}^*, d)$ are complete ultrametric spaces. In order to model $\mathcal{L}(\Sigma_1, Idf)$ we extend the definition of [2] for $\cdot$ and $\|$ by definitions for prefixing, restriction and relabelling as follows.
**Definition 6.4.** Let $H, H_1, H_2$ be nonempty subsets of $\text{Pom}$. Then

\[
H_1 + H_2 = H_1 \cup H_2,
\]

\[
H_1 \| H_2 = \{ p_1 \| p_2 : p_1 \in H_1, p_2 \in H_2 \},
\]

\[
H_1 ; H_2 = \{ p_1 ; p_2 : p_1 \in H_1, p_2 \in H_2 \},
\]

\[
x.H = \{ x.p : p \in H \},
\]

\[
H \backslash L = \{ p \backslash L : p \in H, p \backslash L \neq \emptyset \},
\]

\[
\{ p \} \lambda = \{ p(\lambda) : p \in H \},
\]

where $x \in \text{Act}$, $L \subseteq \text{Act} \setminus \{ \tau \}$ and $\lambda : \text{Act} \rightarrow \text{Act}$ is a relabelling function.

We extend the operators on the empty set as follows:

\[
H \op \emptyset = \emptyset \op H = H \quad \text{where} \quad \op \in \{ +, \|, ; \},
\]

\[
x \emptyset = \{ \emptyset \},
\]

\[
\emptyset \backslash L = \emptyset \{ \lambda \} = \emptyset.
\]

It is easy to show that for all $H, H_1, H_2 \in \text{Pom}^*$ the sets $H_1 + H_2$, $H_1 \| H_2$, $H_1 ; H_2$, $x.H$, $H \backslash L$ and $\{ p \} \lambda$ are closed and do not contain the empty pomset, i.e. if $H_1, H_2, H \in \text{Pom}^*$ then $H_1 + H_2, H_1 \| H_2, H_1 ; H_2, x.H, H \backslash L, H \{ p \} \lambda \in \text{Pom}^*$. Since $\text{Pom}$ is closed under $+$, $\|$ and $;$, $\text{Pom}^*$ is closed under $+$, $\|$ and $;$. Reference [2] has shown that $+$, $\|$ and $;$ are nondistance increasing operators on $\text{Pom}^*$ and that $;$ is contracting in its second argument. In particular the prefixing operator as a special case of the sequence operator is contracting. It is easy to see that also the restricting and the relabelling operator on pomset classes are nondistance increasing. Hence, $\text{Pom}^*$ is an $\mathcal{C}_1$-cms where the constant symbol nil is represented by the empty pomset class $\emptyset$. $\text{Pom}^*$ is an $\mathcal{C}_0$-cms where the constant symbols $x \in \text{Act}$ are represented by $\{ x \}$.

As mentioned in [2] there are some problems with the parallel operator $\| $ with synchronisation on pomset classes. The parallel operator corresponding to the intuitive understanding of the synchronisation of CCS is not nondistance-increasing. For this reason we cannot deal with pomset classes as a $\mathcal{C}_{\text{CCS}}$-cms.

By Theorem 3.10 we get pomset class meanings for the languages $\mathcal{P}(\emptyset, Idf)$ and $\mathcal{P}(\emptyset_0, Idf)$. In the following we write $\text{pom}^*(\langle \sigma, s \rangle)$ instead of $Me^M(\langle \sigma, s \rangle)$ where $M = \text{Pom}^*$ or $M = \text{Pom}^*_0$. Since $;$ distributes over $+$ the resulting pomset class semantics $\text{pom}^*$ is a linear time semantics.

**Example 6.5.** Let $s_1 = x; \beta + \gamma \| (\delta; \gamma)$, $s_2 = x; \gamma + \delta$. Then

\[
\text{pom}^*(s_2) = \{ [\emptyset \rightarrow \emptyset], [\emptyset \rightarrow \emptyset] \}
\]
If $\sigma$ is a declaration with $\sigma(x) = x; x, \sigma(y) = x; y + \beta$ then

\[
\text{pom}^*(\langle \sigma, x \rangle) = \{ p \}, \quad \text{pom}^*(\langle \sigma, y \rangle) = \{ p_1, p_2, p_3, \ldots \} \cup \{ p \},
\]

where

\[
\begin{align*}
p &= [x] \rightarrow [x] \rightarrow [x] \rightarrow \ldots \\
p_1 &= [y] \\
p_2 &= [x] \rightarrow [y] \\
p_3 &= [x] \rightarrow [x] \rightarrow [y] \\
\vdots
\end{align*}
\]

7. A homomorphism from event structures to pom-set classes

We present a method how to transform a finitely approximable prime event structure $\varepsilon$ into a pomset class which describes the linear time behaviour of $\varepsilon$. This pomset class represents the set of possible executions of $\varepsilon$ (as a machine which performs its actions with respect to the causal dependency $\leq$ and the conflict relation $\#$) where all nondeterministic choices (represented by direct conflicts) are made before $\varepsilon$ starts its execution.

7.1. Splitting event structures into maximal components

**Definition 7.1.** Let $\varepsilon = (E, \leq, \#, l) \in \text{FinPrimeEv}_0$. A component of $\varepsilon$ is a nonempty pomset $p$ of the form $p = \varepsilon[A]$ where $A$ is a (nonempty) conflict-free, left-closed subset of $E$. A component $p = \varepsilon[A]$ of $\varepsilon$ is called maximal iff $A$ is maximal with respect to the conflict-freeness, i.e. maximal conflict-free (see Definition 5.4).

**Definition 7.2.** Let $\mathcal{X}_0(\varepsilon)$ denote the set of all maximal components of $\varepsilon$. 
**Example 7.3.** Let $\varepsilon$ be given by

$$
\begin{array}{c}
\beta \\
\downarrow \\
\gamma
\end{array}
\quad
\begin{array}{c}
\varepsilon
\end{array}
\quad
\begin{array}{c}
\beta \\
\downarrow \\
\gamma
\end{array}
$$

Then $\varepsilon$ has exactly three maximal components

$$
\begin{array}{c}
\beta \\
\downarrow \\
\gamma
\end{array},
\begin{array}{c}
\varepsilon
\end{array},
\begin{array}{c}
\beta \\
\downarrow \\
\gamma
\end{array}
$$

and

$$
\begin{array}{c}
\beta \\
\downarrow \\
\gamma
\end{array}
$$

**Example 7.4.** If $\varepsilon'$ is

$$
\begin{array}{c}
\beta \\
\downarrow \\
\gamma
\end{array}
\quad
\begin{array}{c}
\varepsilon'
\end{array}
\quad
\begin{array}{c}
\beta \\
\downarrow \\
\gamma
\end{array}
$$

then $\varepsilon'$ has only one maximal component. This is

$$
\begin{array}{c}
\varepsilon
\end{array}
$$

**Remark 7.5.** In general $X_0(\varepsilon)$ is not closed. For example: Let $\varepsilon=(E, \leq, #, l)$ with $E=\{0, 1\} \times \mathbb{N}_0$ and

$$(i, j) \leq (k, h) :\iff (i = 1 \wedge j < h) \vee (i = k \wedge j = h)$$

and where # is the smallest conflict relation on $E$ which contains #’ where #’ is given by

$$(i, j) \#'(k, h) :\iff (i = 0 \wedge j \geq 1 \wedge k = 1 \wedge h = j + 1) \vee (i = k = j = 0 \wedge h \geq 1).$$

The labelling function $l: E \rightarrow Act$ maps each event $e \in E$ to a fixed action $\varepsilon \in Act$. 

$$
\begin{array}{c}
\varepsilon
\end{array}
\quad
\begin{array}{c}
\varepsilon
\end{array}
\quad
\begin{array}{c}
\varepsilon
\end{array}
$$
Then the sets $A_m$ are maximal conflict-free and left-closed where

$$A_m = \{(1, j): 0 \leq j \leq m+1\} \cup \{(0, m+1), (0, m+2)\}.$$ 

The components $\varepsilon[A_m]$ are given by

$m = 0$:

$$\xymatrix{ 2 \ar[r] & 2 \ar[r] & 2 \ar[r] & 2 \ar[r] & \cdots }$$

$m = 1$:

$$\xymatrix{ 2 \ar[r] & 2 \ar[r] & 2 \ar[r] & 2 \ar[r] & \cdots }$$

$m = 2$:

$$\xymatrix{ 2 \ar[r] & 2 \ar[r] & 2 \ar[r] & 2 \ar[r] & \cdots }$$

The sequence $(p_m)_{m \geq 1}$ where $p_m = \varepsilon[A_m]$ converges to $\varepsilon[A]$ with $A = \{(1, j): j \in \mathbb{N}_0\}$

$$\xymatrix{ 2 \ar[r] & 2 \ar[r] & 2 \ar[r] & 2 \ar[r] & \cdots }$$

This is a component of $\varepsilon$ but not a maximal component of $\varepsilon$. $\varepsilon[A]$ is contained in the maximal component $\varepsilon[A']$ where $A' = A \cup \{(0,0)\}$.

$$\xymatrix{ 2 \ar[r] & 2 \ar[r] & 2 \ar[r] & 2 \ar[r] & \cdots }$$

Remark 7.6. If $\text{depth}(\varepsilon) < \infty$ then the event set $E$ of $\varepsilon$ is finite. Therefore, the set of left-closed and conflict-free subsets of $E$ is finite. In particular, $\mathcal{X}_0(\varepsilon)$ is finite and closed.

Theorem 7.7. We have for all $\varepsilon, \varepsilon_1, \varepsilon_2 \in \text{FinPrime} E_{\varepsilon_0}$:

(a) $\mathcal{X}_0(\varepsilon_1 + \varepsilon_2) = \mathcal{X}_0(\varepsilon_1) + \mathcal{X}_0(\varepsilon_2)$,

(b) $\mathcal{X}_0(\varepsilon_1 \| \varepsilon_2) = \mathcal{X}_0(\varepsilon_1) \| \mathcal{X}_0(\varepsilon_2)$,

(c) $\mathcal{X}_0(\varepsilon_1; \varepsilon_2) = \mathcal{X}_0(\varepsilon_1) : \mathcal{X}_0(\varepsilon_2)$,

(d) $\mathcal{X}_0(\varepsilon; \varepsilon) = \varepsilon \cdot \mathcal{X}_0(\varepsilon)$,

(e) $\mathcal{X}_0(\varepsilon; L) = \mathcal{X}_0(\varepsilon) \setminus L$,

(f) $\mathcal{X}_0(\varepsilon; \lambda) = \mathcal{X}_0(\varepsilon) \setminus \lambda$. 

Proof. (a)–(f) are easy verifications if one of the event structures is empty. (d) follows by (c). We omit the proof of (e) and (f).

Let $\varepsilon_i = (E_i, \#, l_i)$, $E_i \neq \emptyset$, $i = 1, 2$. We may assume w.l.o.g. that $E_1$ and $E_2$ are disjoint. We consider the representative $(E, \ll, \#)$ of $\varepsilon_1 \parallel \varepsilon_2$ which is defined as in the definitions for $+$, $\parallel$ resp.

In the following $\mathcal{X}(\varepsilon)$ denotes the set of all components of $\varepsilon$.

(a) First we show that $\mathcal{X}(\varepsilon_1 \parallel \varepsilon_2) = \mathcal{X}(\varepsilon_1) \cup \mathcal{X}(\varepsilon_2)$.

"$\subseteq$": Let $\varepsilon \mid A$ be a component of $\varepsilon$. Since $A$ is a conflict-free and left-closed subset of $E = E_1 \cup E_2$ and since each pair $(e_1, e_2)$ of events $e_1 \in E_1$, $e_2 \in E_2$ is in conflict, $A$ is a left-closed and conflict-free subset of $E_1$ or $E_2$. We conclude that $\varepsilon \mid A$ is a component of $\varepsilon_1$ or $\varepsilon_2$.

"$\supseteq$": Each conflict-free and left-closed subset $A$ of $E_1$ or $E_2$ (with respect to $\varepsilon_1$ resp. $\varepsilon_2$) is a conflict-free and left-closed subset of $E$.

It is easy to see that $\varepsilon \mid A$ is a maximal component of $\varepsilon$ if and only if it is a maximal component of $\varepsilon_1$ resp. $\varepsilon_2$.

(b) First we show that $\mathcal{X}(\varepsilon_1 \parallel \varepsilon_2) = \mathcal{X}(\varepsilon_1) \cup \mathcal{X}(\varepsilon_2)$.

"$\subseteq$": Let $\varepsilon \mid A$ be a component of $\varepsilon$. Since $A$ is a conflict-free and left-closed subset of $E = E_1 \cup E_2$ the sets $A_1 = A \cap E_1$ and $A_2 = A \cap E_2$ are left-closed and conflict-free (with respect to $\varepsilon_1$ resp. $\varepsilon_2$). We conclude that $\varepsilon_1 \mid A_1$ are components of $\varepsilon_1$, $i = 1, 2$. We get:

$$\varepsilon \mid A = (\varepsilon_1 \mid A_1) \parallel (\varepsilon_2 \mid A_2) \in \mathcal{X}(\varepsilon_1) \parallel \mathcal{X}(\varepsilon_2).$$

"$\supseteq$": For each pair $(A_1, A_2)$ of conflict-free and left-closed subsets of $E_1$ resp. $E_2$ (with respect to $\varepsilon_1$ resp. $\varepsilon_2$) $A_1 \cup A_2$ is a conflict-free and left-closed subset of $E$ with

$$(\varepsilon_1 \mid A_1) \parallel (\varepsilon_2 \mid A_2) = \varepsilon \mid (A_1 \cup A_2) \in \mathcal{X}(\varepsilon_1 \parallel \varepsilon_2).$$

Now we show that $\mathcal{X}_0(\varepsilon_1 \parallel \varepsilon_2) = \mathcal{X}_0(\varepsilon_1) \parallel \mathcal{X}_0(\varepsilon_2)$.

"$\subseteq$": Since there does not exist any pair $(e_1, e_2)$ of events $e_1 \in E_1$, $e_2 \in E_2$ which is in conflict it follows immediately that $A_1 \cup A_2$ is a maximal conflict-free subset of $E$ if and only if only if $A_1$ and $A_2$ are maximal conflict-free subsets of $E_1$ resp. $E_2$ (with respect to $\varepsilon_1$ resp. $\varepsilon_2$). We get that each maximal component $p$ of $\varepsilon$ is of the form $p = p_1 \parallel p_2$ where $p_1 \in \mathcal{X}_0(\varepsilon_1)$ and $p_2 \in \mathcal{X}_0(\varepsilon_2)$.

"$\supseteq$": If $p_1$ and $p_2$ are maximal components of $\varepsilon_1$ resp. $\varepsilon_2$ the parallel composition $p_1 \parallel p_2$ is a maximal component of $\varepsilon$.

(c) First we show that $\mathcal{X}_0(\varepsilon_1 \parallel \varepsilon_2) \subseteq \mathcal{X}_0(\varepsilon_1) \parallel \mathcal{X}_0(\varepsilon_2)$.

Analogously to the first part of (b) it can be shown that if $p = \varepsilon \mid A$ is a component of $\varepsilon = \varepsilon_1 \parallel \varepsilon_2$ then $p_1 = \varepsilon_1 \mid A_1$ is a component of $\varepsilon_1$ where $A_1 = A \cap E_1$, $A_2 = E \setminus A_1$ is a conflict-free subset of $\{(B, e): B \in K, e \in E_2\}$ where $K$ is the set of finite, left-closed and
maximal conflict-free subsets of $E_1$ (with respect to $e_i$). Since each pair $((B_1, e_1), (B_2, e_2)$ with $B_1, B_2 \in K$, $B_1 \neq B_2$ and $e_1, e_2 \in E_2$ is in conflict we get that

1. either $A_2 = \emptyset$,
2. or $A_2 = \{(B, e) : e \in A_2, (B, e) \text{ is in conflict with some } B \in K \}$ and a conflict-free and left-closed subset $A_2$ of $E_2$.

In case (1) we have $p = e_1 \upharpoonright A_1 \in \mathcal{K}(e_1)$, $p$ is a maximal component of $e$ if and only if $p \in \mathcal{K}_0(e_1)$ and $\text{depth}(p) = \infty$.

In case (2) we get (by the conflict-freeness of $A$) that $B \subseteq A_1$. Since $B$ is maximal conflict-free we conclude that $B = A_1$ and

$$p = (e_1 \upharpoonright A_1) ; (e_2 \upharpoonright A_2) \in \mathcal{K}_0(e_1) \cdot \mathcal{K}_0(e_2).$$

If $p$ is a maximal component of $e$ then $A_2$ is maximal with respect to the conflict-freeness in $e_2$.

Next we show that $\mathcal{K}_0(e_1 \cdot e_2) \subseteq \mathcal{K}_0(e_1) \cdot \mathcal{K}_0(e_2)$.

If $p_1$ is a maximal component of $e_i$ with $\text{depth}(p_1) = \infty$ then $p = p_1$ is a maximal component of $e_i$. If $p_i$ are maximal components of $e_i$, $i = 1, 2$, $\text{depth}(p_1) < \infty$, $p_1 = e_1 \upharpoonright A_1$, $p_2 = e_2 \upharpoonright A_2$, then $A_1 \in K$ and $A = A_1 \cup \{(A_1, e) : e \in A_2\}$ is a left-closed and maximal conflict-free subset of $E$. We get

$$p_1 \cdot p_2 = (e_1 \upharpoonright A_1) ; (e_2 \upharpoonright A_2) = e \upharpoonright A \in \mathcal{K}_0(e).$$

7.2. Compact pomset classes

In this section we show that for each finitely approximable prime event structure $\mathcal{E}$ the closure of $\mathcal{K}_0(\mathcal{E})$ is a compact subset of $\text{Pom}$. And vice versa, each compact subset of $\text{Pom}$ is the set of maximal components of some finitely approximable prime event structure. To do so we need a characterization of compact subsets of $\text{Pom}$.

**Definition 7.8.** A pomset class of finite partition is a pomset class $H \in \text{Pom}_F$ such that for each natural number $n \geq 1$ the set $H[n] = \{ p[n] : p \in H \}$.

**Lemma 7.9.** Each pomset class of finite partition is a compact subset of $\text{Pom}$.

**Proof.** Let $H$ be a pomset class of finite partition and let $(U_i)_{i \in I}$ be a family of open sets $U_i \subseteq \text{Pom}$ such that

$$H \subseteq \bigcup_{i \in I} U_i.$$  

We have to show that there exists a finite subcover $(U_{i_1}, \ldots, U_{i_n})$ of $H$. We may assume w.l.o.g. that $H \neq \emptyset$. Since each pomset $p \in H$ belongs to some open set $U_i$ we may define

$$N(p) = \min \{ N \in \mathbb{N} : \text{there exists } i \in I \text{ with } B(p, \frac{1}{2^N} \subseteq U_i) \},$$

where $B(p, r)$ denotes the open ball with center $p$ and radius $r$. 

Claim 1. There exists a natural number \( n \geq 1 \) with \( n \geq N(p) \) for each pomset \( p \in H \).

Proof. We assume that for each \( n \geq 1 \) there exists a pomset \( p_n \in H \) with \( n < N(p_n) \). By induction on \( k \) we define a subsequence \( (p_{n_k})_{k \geq 1} \) of \( (p_n)_{n \geq 1} \) such that \( p_{n_k}[k] = p_n[k] \) for \( n \in I_k \) where \( I_k \) is an infinite subset of \( \mathbb{N} \).

Let \( I_0 \) be the set of natural numbers and \( k \geq 1 \). Then we can perform the basis of induction \( (k = 1) \) and the induction step \( (k - 1 \Rightarrow k) \) simultaneously.

Since \( H[k] \) is finite and since the finite set of pomsets \( p_n[k], \forall n \in I_{k-1} \), is contained in \( H[k] \) there exists a natural number \( n_k \in I_{k-1} \) and an infinite subset \( I_k \) of \( I_{k-1} \), such that

(i) \( n_k > n_{k-1} \) (where \( n_0 = 0 \)),
(ii) \( p_{n_k}[k] = p_n[k] \) \( \forall n \in I_k \).

Since \( d(p_{n_k}, p_n) \leq 1/2^k \) and since \( d \) is an ultrametric we get \( d(p_{n_k}, p_{n_m}) \leq 1/2^k \) for all \( m \geq k \geq 1 \). We conclude that \( (p_{n_k})_{k \geq 1} \) is a Cauchy sequence in \( P \) which converges (in the complete metric space \( P \)) to some pomset \( p \). \( p \) belongs to \( H \) because \( H \) is closed.

In addition, we have

\[
d(p_{n_k}, p) = \lim_{m \to \infty} d(p_{n_k}, p_n) \leq \frac{1}{2^k} \quad \forall k \geq 1.
\]

With \( k = N(p) \) we get

\[
B(p_{n_k}, 1/2^n) \subseteq B(p_{n_k}, 1/2^k) \text{ (since } n_k \geq k) \\
= B(p, 1/2^k) \text{ (since } d(p_{n_k}, p) \leq 1/2^k) \\
= B(p, 1/2^{k+1}) \subseteq U_i \text{ (by definition of } N(p))
\]

for some \( i \in I \). We conclude that \( n_k \geq N(p_n) \). On the other hand we have \( N(p_{n_k}) > n_k \) (by choice of the sequence \( (p_{n_k})_{n \geq 1} \)). This is a contradiction.

Let \( n \) be a natural number with \( n \geq N(p) \) for each pomset \( p \) in \( H \). Since \( H[n] \) is finite there exists a finite sequence \( p_1, \ldots, p_m \) in \( H \) such that \( H[n] = \{ p_j[n]: j = 1, \ldots, m \} \). For each \( j, 1 \leq j \leq m \), there exists an index \( i_j \in I \) with

\[
B(p_j, 1/2^{i_j}) \subseteq U_{i_j}.
\]

Claim 2. \( H \subseteq U_{i_1} \cup \cdots \cup U_{i_m} \).

Proof. If \( q \in H \) then \( q[n] \in H[n] \) and therefore \( q[n] = p_j[n] \) for some \( j \in \{1, \ldots, m\} \). We get

\[
d(q, p_j) \leq \frac{1}{2^n} \leq \frac{1}{2^{N(p)}}
\]
and therefore
\[ q \in B(p, 1/2^{[N]}) \subseteq U_p. \]

Lemma 7.10. If \( H \) is a compact subset of \textit{Pom} then \( H \) is of finite partition.

\textbf{Proof.} If \( H = \emptyset \) then \( H \) is of finite partition. Now let \( H \neq \emptyset \) be a compact subset of \textit{Pom}. We assume that there exists a natural number \( N \geq 1 \) such that \( H[\{N\}] \) is infinite. Let \( H' \) be an infinite subset of \( H \) which satisfies the following conditions:

\begin{enumerate}[label=(i)]
\item \( H[\{N\}] = H'[\{N\}] \).
\item \( p[\{N\}] \neq q[\{N\}] \quad \forall p, q \in H', \ p \neq q. \)
\end{enumerate}

We define
\[ U'_p = B\left(p, \frac{1}{2^N}\right) \quad \forall p \in H'. \]

\textbf{Claim 1.} \( (U_p)_{p \in H'} \) is an open cover of \( H \).

\textbf{Proof.} Let \( q \) be a pomset in \( H \). Since \( q[\{N\}] \in H[\{N\}] = H'[\{N\}] \) there exists a pomset \( p \in H' \) with \( q[\{N\}] = p[\{N\}] \) and therefore
\[ q \in B(p, 1/2^N) \subseteq U_p. \]

\textbf{Claim 2.} For each finite subset \( H'' \) of \( H' \) there exists a pomset \( q \in H \) with
\[ q \notin \bigcup_{p \in H''} U_p. \]

\textbf{Proof.} If \( H'' \) is a finite subset of \( H' \) we can choose a pomset \( q \in H' \setminus H''. \) Since \( H' \) satisfies condition (ii) we get that
\[ q[\{N\}] \in H \setminus \left( \bigcup_{p \in H''} U_p \right). \]

We conclude that \( (U_p)_{p \in H'} \) is an infinite open cover of \( H \) which does not contain a finite subcover of \( H \). Since \( H \) is compact this is impossible. \( \square \)

By Lemmas 7.9 and 7.10 we immediately get the following theorem.

\textbf{Theorem 7.11.} A pomset class is of finite partition if and only if it is compact.

Lemma 7.12. If \( \varepsilon \in \textit{FinPrimeEv}_\emptyset \) then \( \mathcal{H}_\emptyset(\varepsilon) \) is a pomset class of finite partition.

\textbf{Proof.} Let \( \varepsilon = (E, \leq, \#) \in \textit{FinPrimeEv}_\emptyset \) and \( n \geq 1 \). Since the set \( E[\{n\}] \) is finite the power set of \( E[\{n\}] \) is also finite. Therefore, the set of all left-closed and conflict-free
subsets of $E[n]$ is finite. We get

$$\mathcal{X}_0(\varepsilon)[n] = \{p[n]: p \in \mathcal{X}_0(\varepsilon) \subseteq \{\varepsilon: A \subseteq E[n] \} \text{ left-closed and conflict-free}\}$$

is a finite set. □

**Lemma 7.13.** For each compact subset $H$ of Pom there exists a finitely approximable prime event structure $\varepsilon$ with $\mathcal{X}_0(\varepsilon) = H$. If $H \in \text{Pom}^*$ then $H = \mathcal{X}_0(\varepsilon)$ for some $\varepsilon \neq \emptyset$.

**Proof.** If $H = \emptyset$ then $H = \mathcal{X}_0(\emptyset)$. Now we assume that $H \neq \emptyset$. By Lemma 7.10 we get that $H$ is of finite partition. Since $H[n]$ is finite there exist finite families $(p^n_1, \ldots, p^n_{k_n})$ of pomsets $p^n_i = (E^n_i, \leq^n_i, l^n_i)$ with

1. $\text{depth}(p^n_i) \leq n$ and $p^n_i = q^n_i[n]$ for some $q^n_i \in H$,
2. $p^n_i \neq p^n_j$ for all $1 \leq i < j \leq k_n$,
3. $H = H^n_1 \cup H^n_2 \cup \cdots \cup H^n_{k_n}$ where $H^n_i = \{p \in H: p[n] = p^n_i\}$.

Since $p^n_i[n] = q^n_{i+1}[n] \in H[n] = \{p^n_1, \ldots, p^n_{k_n}\}$ (property (3)) there exist surjective mappings

$$\sigma^n_i: \{1, \ldots, k_n + 1\} \rightarrow \{1, \ldots, k_n\}$$

with $p^n_i[n] = p^n_{\sigma^n_i(1)}$ for $i = 1, \ldots, k_n + 1$. The mappings

$$\sigma^n_m: \{1, \ldots, k_{n+1}\} \rightarrow \{1, \ldots, k_n\}.$$

$0 \leq m \leq n$, are defined as follows:

$$\sigma^n_m(i) = \begin{cases} \sigma^n_{m+1} \circ \cdots \circ \sigma^n_1(i) & \text{if } 1 \leq m \leq n, \\ 1 & \text{if } m = 0. \end{cases}$$

We may assume w.l.o.g. that

1. $E_{i+1}^{n+1}[n] = E_{\sigma^n(i)}^{n+1} \subseteq E_{\sigma^n(i)}^n = E_{\sigma^n(i)}^n \cap E_{\sigma^n(i)}^n \times E_{\sigma^n(i)}^n$ and $l^n_{\sigma^n(i)} = l_{\sigma^n(i)}^{n+1} |_{E_{\sigma^n(i)}^n}$,
2. If $1 \leq i < j \leq k_{n+1}$ and $\mu = \max\{m: 0 \leq m \leq n, \sigma^n_{m}(i) = \sigma^n_{m}(j)\}$ then $E_{i+1}^{n+1} \cap E_{j+1}^{n+1} = E_{\sigma^n_{\mu}}^{n+1}$.

Let $e = (E, \leq, \# , l)$ be given by

$$E = \bigcup_{n \geq 1} \bigcup_{1 \leq i \leq k_n} E^n_i,$$

$$\leq = \bigcup_{n \geq 1} \bigcup_{1 \leq i \leq k_n} \leq^n_i,$$

$$l: E \rightarrow \text{Act}, \quad l(e) = l^n_i(e) \quad \text{if } e \in E^n_i$$

and $\# = \{(e_1, e_2) \in E \times E: \exists e'_1, e'_2 \in E e'_1 \leq e_1, e'_2 \leq e_2 \land (e'_1, e'_2) \in C\}$ where

$$C = \bigcup_{n \geq 0} \bigcup_{1 \leq i, j \leq k_{n+1}} (E_{i+1}^{n+1} \setminus E[n]) \times (E_{j+1}^{n+1} \setminus E[n]).$$
It is easy to see that $e \in \text{FinPrimeE}_v$ and that $E^n_i$ is left-closed and conflict-free with $e \subseteq E^n_i = p^n_i$ for all $n \geq 1$ and $1 \leq i \leq k_n$.

Let $I$ be the set of all (infinite) sequences $(j_n)_{n \geq 1}$ with $1 \leq j_n \leq k_n$ and $\sigma_n(j_{n+1}) = j_n$ for all $n \geq 1$.

Then the set of all left-closed and maximal conflict-free subsets of $E$ is

$$K = \left\{ \bigcup_{n \geq 1} E^n_{j_n} : (j_n)_{n \geq 1} \in I \right\}.$$ 

Then $X_0(e) = \{ e[A] : A \in K \}$. Now we show that $H = X_0(e)$.

(1) For each $p \in H$ and $n \geq 1$ there exists an index $j_n \in \{1, \ldots, k_n\}$ with $p[n] = p^n_{j_n}$. Then $\sigma_n(j_{n+1}) = j_n$ for all $n \geq 1$ and therefore $(j_n)_{n \geq 1} \in I$ and

$$p = e \left[ \left( \bigcup_{n \geq 1} E^n_{j_n} \right) \right] \in X_0(e).$$

(2) For each $p \in X_0(e)$ there exists a sequence $(j_n)_{n \geq 1} \in I$ such that $p = e[A]$ where

$$A = \bigcup_{n \geq 1} E^n_{j_n} \in K.$$ 

Then $p[n] = e \left[ E^n_{j_n} = p^n_{j_n} = q^n_{j_n}[n] \right]$ for all $n \geq 1$ and $p = \lim q^n_{j_n} \in H = H$. □

**Theorem 7.14.** The function $\mathcal{F} : \text{FinPrimeE}_{v_0} \to \text{Pom}^*$, $\mathcal{F}(e) = \overline{K_0(e)}$, is well defined. We have

(a) $\mathcal{F}(p) = \{ p \}$ for all $p \in \text{Pom}_0$.

(b) $\mathcal{F}(e_1 \circ_p e_2) = \mathcal{F}(e_1) \circ_p \mathcal{F}(e_2)$ for all $e_1, e_2 \in \text{FinPrimeE}_{v_0}$ and $p \in \{ +, \mid, \cdot \}$.

(c) $\mathcal{F}(\circ_p e) = \circ_p(\mathcal{F}(e))$ where $\circ_p$ is a prefixing, a relabelling or a restriction operator.

(d) If $e \neq \emptyset$ then $\mathcal{F}(e) \in \text{Pom}^*$. In particular $\mathcal{F} | \text{FinPrimeE}_{v_0} \to \text{Pom}^*$ is well defined and the image is the collection of all nonempty and compact pomset classes.

**Proof.** $\mathcal{F}$ is well defined by Lemma 7.12. (a) is an easy verification. (c) and (d) follow immediately by Lemma 7.13. We only have to show property (b).

Since $X_0(e_1 \circ_p e_2) = X_0(e_1) \circ_p X_0(e_2)$ (Theorem 7.7) and since $H_1 \circ_p H_2$ is closed whenever $H_1$ and $H_2$ are closed we get

$$\mathcal{F}(e_1 + e_2) = \mathcal{F}(e_1) + \mathcal{F}(e_2) = \mathcal{F}(e_1)[e_2],$$

$$\mathcal{F}(e_1 \parallel e_2) \subseteq \mathcal{F}(e_1) \parallel \mathcal{F}(e_2),$$

$$\mathcal{F}(e_1 : e_2) \subseteq \mathcal{F}(e_1) ; \mathcal{F}(e_2).$$

Let $\circ_p$ be one of the operators $\parallel$ or $;$ and let $p_1, p_2$ be pomsets in $\mathcal{F}(e_1)$ resp. $\mathcal{F}(e_2)$. Then there exist sequences $(p_{n,i})_{n \geq 1}$ in $X_0(e_i)$ with

$$\lim_{n \to \infty} p_{n,i} = p_i, \quad i = 1, 2.$$
Since \( d(p_1 \text{ op } p_2, p_{n,1} \text{ op } p_{n,2}) \leq \max \{d(p_1, p_{n,1}), d(p_2, p_{n,2})\} \) we have

\[
p_1 \text{ op } p_2 = \lim_{n \to \infty} p_{n,1} \text{ op } p_{n,2} \in \mathcal{K}_0(e_1 \text{ op } e_2) = \mathcal{F}(e_1 \text{ op } e_2).
\]

It follows that \( \mathcal{F}(e_1 \text{ op } \mathcal{F}(e_2)) \subseteq \mathcal{F}(e_1 \text{ op } e_2) \).

Since prefixing is a special case of the sequence operator we get

\[
\mathcal{F}(x \cdot e) = \mathcal{F}(\{ x \}; e) = \mathcal{F}(\{ x \}) ; \mathcal{F}(e) = \{ [x] \} ; \mathcal{F}(e) = x \cdot \mathcal{F}(e).
\]

In the same way of above it can be shown that \( \mathcal{F}(\text{op}(e)) = \text{op}(\mathcal{F}(e)) \) where \( \text{op} \) denotes restriction or relabelling.

**Remark 7.15.** \( \mathcal{F} \) is not continuous. We consider the following sequence \( (\varepsilon_n)_{n \geq 1} \) which converges to \( \varepsilon \):

\[
\varepsilon_n = \begin{bmatrix}
\mathbb{X} \\
\mathbb{X} \\
\mathbb{X} \\
\mathbb{X} \\
\mathbb{X} \\
\mathbb{X}
\end{bmatrix},
\]

and

\[
\varepsilon = \begin{bmatrix}
\mathbb{X} \\
\mathbb{X} \\
\mathbb{X} \\
\mathbb{X} \\
\mathbb{X} \\
\mathbb{X}
\end{bmatrix}
\]

Then \( d(\varepsilon_n, \varepsilon) = 1/2^n \) and \( \lim_{n \to \infty} \varepsilon_n = \varepsilon \). Since \( \varepsilon \) is a pomset we have \( \mathcal{K}_0(\varepsilon) = \{ \varepsilon \} \). Since

\[
\mathbb{X} \to \mathbb{X} \to \cdots \to \mathbb{X} \to \mathbb{X} \to \cdots
\]

we get \( d(\mathcal{K}_0(\varepsilon_n), \mathcal{K}_0(\varepsilon)) = 1 \).

**Theorem 7.16.** Let \( \mathcal{O} \) be one of the symbol-algebras with guardedness conditions \( \mathcal{O}_0 \) or \( \mathcal{O}_1 \). For all processes \( \langle \sigma, s \rangle, \langle \sigma', s' \rangle \in \mathcal{P}(\mathcal{O}, \text{Idf}) \) we have

\[
d(\text{pom}^*(\langle \sigma, s \rangle), \text{pom}^*(\langle \sigma', s' \rangle)) \leq d(\text{ev}_{\text{cms}}(\langle \sigma, s \rangle), \text{ev}_{\text{cms}}(\langle \sigma', s' \rangle)),
\]

\[
\text{pom}^*(\langle \sigma, s \rangle) = \mathcal{K}_0(\text{ev}_{\text{cms}}(\langle \sigma, s \rangle)).
\]

**Proof.** If \( \varepsilon = (E, \leq, \#, l) \in \text{FinPrime}E_{\text{Ev}} \) then \( \#_0 \) denotes the set of direct conflicts. \( Ev \) denotes the class of all finitely approximable prime event structures \( \varepsilon = (E, \leq, \#, l) \) such that for all events \( e_1, e_2 \in E \): If \( e_1 \#_0 e_2 \) then \( \text{depth}(e_1) = \text{depth}(e_2) \).

It is easy to see that \( Ev \) is closed under \( +, \|, \cdot, \) restriction, relabelling and prefixing and that

\[
\varepsilon \in Ev \iff \varepsilon[n] \in Ev \quad \forall n \geq 0.
\]
We show that $\mathcal{F} \mid Ev$ is weakly contracting and that for each process $\langle \sigma, s \rangle \in \mathcal{P}(C, Id_f)$ we have: $ev_{cms}(\langle \sigma, s \rangle) \in Ev$.

**Claim 1.** If $\varepsilon \in Ev$ then $\mathcal{X}_\varepsilon(\varepsilon[n]) = \mathcal{X}_\varepsilon(\varepsilon)[n]$ for all $n \geq 0$.

**Proof.** Let $K$ resp. $K'$ denote the set of all left-closed subsets of $E$ resp. $E[n]$ which are maximal conflict-free with respect to the conflict relation of $\varepsilon$ resp. $\varepsilon[n]$. It is enough to show that $K' = \{ A[n] \mid A \in K \}$.

“$\subseteq$”: If $A' \in K'$ then there exists $A \in K$ with $A' \subseteq A$. We show that $A' = A[n]$. Since for each event $e$ in $A'$ the depth of $e$ (as an event of $\varepsilon$) is at most $n$ we get $A' \subseteq A[n]$. Now, let $e$ be an event in $A[n]$. We assume that $e \notin A'$. Since $A'$ is maximal conflict-free (with respect to the conflict relation of $\varepsilon[n]$) and since $e \in A[n] \setminus A'$ there exists an event $e' \in A'$ with $e \neq e'$. It follows that $e$ and $e'$ are conflicting events which belong to $A$. This is a contradiction to the conflict-freeness of $A$.

“$\supseteq$”: If $A \in K$ then $A[n]$ is a left-closed and conflict-free subset of $E[n]$. We have to show that $A[n]$ is maximal conflict-free with respect to the conflict relation of $\varepsilon[n]$. Let $e$ be an event in $E[n] \setminus A[n]$. Then $e \notin A$. There exists an event $e_A \in A$ with $e \neq e_A$. Let $e'$ and $e'_A$ be the predecessors of $e$ resp. $e_A$ such that $e'$ and $e'_A$ are in direct conflict. Since $A$ is left-closed we have $e'_A \in A$. Since $e \notin A$ we get $\text{depth}(e') = \text{depth}(e) < \text{depth}(e) = n$. Therefore: $e'_A \in A[n]$ and $e \neq e'_A$.

**Claim 2.** $d(\mathcal{F}(e_1), \mathcal{F}(e_2)) \leq d(e_1, e_2) \forall e_1, e_2 \in Ev$.

**Proof.** Let $e_1, e_2 \in Ev$. If $e_1 = e_2$ then there is nothing to show. Otherwise there exists $n \in \mathbb{N}$ such that $d(e_1, e_2) = 1/2^n$ and $p \in \mathcal{F}(e_1)$. There exists $p' \in \mathcal{X}_\varepsilon(e_1)$ with $d(p, p') \leq 1/2^n$. Then $p[n] = p'[n] \in \mathcal{X}_\varepsilon[e_1][n]$ and (by Claim 1)

$\mathcal{X}_\varepsilon(e_1)[n] = \mathcal{X}_\varepsilon(e_1[n]) = \mathcal{X}_\varepsilon(e_2[n]) = \mathcal{X}_\varepsilon(e_2)[n]$.

Let $q \in \mathcal{X}_\varepsilon(e_2)$ with $p[n] = q[n]$. We get

$\inf_{p \in \mathcal{F}(e_1)} d(p, q) \leq d(p[n], q[n]) = 1/2^n \forall p \in \mathcal{F}(e_1)$.

Analogously

$\inf_{q \in \mathcal{F}(e_2)} d(r, q) \leq 1/2^n \forall q \in \mathcal{F}(e_2)$.

We conclude $d(\mathcal{F}(e_1), \mathcal{F}(e_2)) \leq 1/2^n$.

**Claim 3.** $ev_{cms}(\langle \sigma, s \rangle) \in Ev \forall \langle \sigma, s \rangle \in \mathcal{P}(C, Id_f)$.

**Proof.** Let $\sigma \in \mathcal{P}(C, Id_f)$. We show by induction on $n \in \mathbb{N}$ that

(i) $ev_{cms}(\langle \sigma, s \rangle)[n] \in Ev$ for each guarded statement $s \in \mathcal{L}(C, Id_f)$.

(ii) If $n \geq 1$ then $ev_{cms}(\langle \sigma, s \rangle)[n-1] \in Ev$ for each statement $s \in \mathcal{L}(C, Id_f)$. 


For simplicity we only consider the case $\emptyset = \emptyset_0$. In the case $n = 0$ there is nothing to show. Induction step $n \Rightarrow n+1$: We prove (i) and (ii) by structural induction on the syntax of $s \in \Sigma$.

**Basis of induction.** If $s = z \in \text{Act}$ then

$$ev_{cm}(\langle \sigma, z \rangle) = [z] \in Ev.$$

If $s = x \in \text{Idf}$ then we have to show that $ev_{cm}(\langle \sigma, x \rangle)[n] \in Ev$. We get (by induction hypothesis (i) applied to the guarded statement $\sigma(x)$):

$$ev_{cm}(\langle \sigma, x \rangle)[n] = ev_{cm}(\langle \sigma, \sigma(x) \rangle)[n] \in Ev.$$

**Induction step.**

*Case 1.* $s = s_1 \text{ op } s_2$ where $\text{ op } \in \{ +, \| \}$. 

(i) If $s$ is a guarded statement then $s_1$ and $s_2$ are guarded and we get by induction hypothesis (i) (applied to $s_1$ and $s_2$) $ev_{cm}(\langle \sigma, s_i \rangle)[n+1] \in Ev$, $i = 1, 2$. Since $Ev$ is closed under the operator $\text{ op }$ we get

$$ev_{cm}(\langle \sigma, s \rangle)[n+1] = (ev_{cm}(\langle \sigma, s_1 \rangle)[n+1] \text{ op } ev_{cm}(\langle \sigma, s_2 \rangle)[n+1])[n+1]$$

belongs to $Ev$.

(ii) We get by induction hypothesis (ii) (applied to $s_1$ and $s_2$): $ev_{cm}(\langle \sigma, s_1 \rangle)[n] \in Ev$ and $ev_{cm}(\langle \sigma, s_2 \rangle)[n] \in Ev$. Since $Ev$ is closed under the operator $\text{ op }$ we conclude:

$$ev_{cm}(\langle \sigma, s \rangle)[n] = ev_{cm}(\langle \sigma, s_1 \rangle)[n] \text{ op } ev_{cm}(\langle \sigma, s_2 \rangle)[n]$$

belongs to $Ev$.

*Case 2.* $s = s_1 \text{ ; } s_2$. By induction hypothesis (ii) (applied to $s_2$) we have $ev_{cm}(\langle \sigma, s_2 \rangle)[n] \in Ev$.

(i) If $s$ is guarded then $s_1$ is guarded. We get by induction hypothesis (i) applied to $s_1$:

$$ev_{cm}(\langle \sigma, s_1 \rangle)[n+1] \in Ev.$$

Since $Ev$ is closed under $;$ we get

$$e = ev_{cm}(\langle \sigma, s_1 \rangle)[n+1] \ ; ev_{cm}(\langle \sigma, s_2 \rangle)[n] \in Ev.$$

It follows (since $ev_{cm}(\langle \sigma, s_1 \rangle) \neq \emptyset$):

$$ev_{cm}(\langle \sigma, s \rangle)[n+1] = \varepsilon[n+1] \in Ev.$$

(ii) By induction hypothesis (ii) applied to $s_1$ we have $ev_{cm}(\langle \sigma, s_1 \rangle)[n] \in Ev$ and therefore

$$e - ev_{cm}(\langle \sigma, s_1 \rangle)[n] \ ; ev_{cm}(\langle \sigma, s_2 \rangle)[n] \in Ev.$$

We conclude that $ev_{cm}(\langle \sigma, s \rangle)[n] = e[n]$ belongs to $Ev$. 
Claim 4. $X_0(e)$ is closed for each $e \in Ev$.

Proof. Let $e=(E, \leq, \#, l) \in Ev$ and let $(p_n)_{n \geq 0}$ be a sequence in $X_0(e)$ which converges to some pomset $p$. We have to show that $p \in X_0(e)$.

We may assume w.l.o.g. that

$$d(p_n, p_m) \leq \frac{1}{2^n} \quad \forall m \geq n \geq 0.$$ 

There exist left-closed and maximal conflict-free subsets $A_n$ of $E$ with $p_n = e[A_n]$. Since the sets $A_n[k]$ are subsets of the finite set $E[k]$ there exists a subsequence $(p_{n_m})_{m \geq 0}$ of $(p_n)_{n \geq 0}$ such that $A_{n_m}[k] = A_{n_m}[k]$ for all $m \geq k \geq 0$. We define

$$A = \bigcup_{k \geq 0} A_{n_m}[k].$$

Then $A$ is a left-closed and conflict-free subset of $E$. Since

$$p[k] = p_{n_m}[k] = e[A_{n_m}[k]] = e[A[k]],$$

we get $p = e[A]$. Now we have to show that $A$ is maximal conflict-free. (Then $p$ is a maximal component of $e$ and belongs to $X_0(e)$.)

Let $e$ be an event in $E \setminus A$, $depth(e) = k$. Then $e \notin A_{n_m}$. Since $A_{n_m}$ is maximal conflict-free there exists an event $e_0 \in A_{n_m}$ such that $e$ and $e_0$ are in conflict. Since $e \in Ev$ the predecessors $e'$ resp. $e'_0$ of $e$ resp. $e'_0$ which are in direct conflict have the same depth. We conclude $e' \neq e'_0$ and

$$depth(e'_0) = depth(e') \leq depth(e) = k.$$ 

Since $A_{n_m}$ is left-closed $e'_0$ belongs to $A_{n_m}$. We get $e'_0 \in A_{n_m}[k] \subseteq A$. \qed

7.3. The consistency of the prime event structure and pomset class semantics

Theorem 7.17. The function $p: FinPrimeEv_0 \to \text{Pom}_0^*$, $e \mapsto X_0(e)$, is a noncontinuous homomorphism from $FinPrimeEv_0$ as an $\mathcal{C}_1$-cms to $\text{Pom}_0^*$ as an $\mathcal{C}_1$-cms. The function

$$p \mid FinPrimeEv \to \text{Pom}_*$$

is a homomorphism from $FinPrimeEv$ as an $\mathcal{C}_0$-cms to $\text{Pom}_*$ as an $\mathcal{C}_0$-cms. In particular we have

$$p \circ ev_{cms} = pom_*.$$ 

The image of $p$ is the subspace of the nonempty and compact subsets of $\text{Pom}_0$.

For each process $\langle \sigma, s \rangle \in \mathcal{P}(\emptyset, Idf)$ we have

$$pom_*(\langle \sigma, s \rangle) = X_0(ev_{cms}(\langle \sigma, s \rangle))$$

is a compact pomset class.
The restriction of \( p \) on those event structures which are the meaning of some process \( \langle \sigma, s \rangle \in \mathcal{P}(\mathcal{O}, \text{Idf}) \) is weakly contracting, i.e.
\[
d(pom^*(\langle \sigma, s \rangle), pom^*(\langle \sigma', s' \rangle)) \leq d(ev_{cms}(\langle \sigma, s \rangle), ev_{cms}(\langle \sigma', s' \rangle))
\]
for all \( \langle \sigma, s \rangle, \langle \sigma', s' \rangle \in \mathcal{P}(\mathcal{O}, \text{Idf}) \). Here \( \mathcal{O} = \mathcal{E}_{\text{CCS}} \) resp. \( \mathcal{E}_0 \).

**Proof.** Follows immediately by Theorem 7.14 and Theorem 7.16. \( \square \)

Let us assume that there is a homomorphism \( f: \text{Pom}^* \to \text{FinPrimeEv} \). Then by Theorem 3.11 \( f \circ \text{pom}^* = \text{ev}_{cms} \). This is impossible since \( \text{ev}_{cms} \) is a branching time semantics whereas \( \text{pom}^* \) is a linear time semantics for \( \mathcal{O} \). For example we regard the primitive statements \( s_1 = x; (\beta + y) \) and \( s_2 = x; (\beta + x; y) \). These statements have the same meaning in \( \text{Pom}^* \)
\[
pom^*(s_1) = \text{pom}^*(s_2) = \{ [x] \to [\beta], [x] \to [y] \}
\]
but the meanings in \( \text{FinPrimeEv} \) are different:
\[
ev_{cms}(s_1) = [x] [\beta] \neq [x] [\beta] = ev_{cms}(s_2).
\]
The reason for this difference is founded in the distributive law which holds in \( \text{Pom}^* \) but not in \( \text{FinPrimeEv}_0 \).

8. Pomset classes and partial orders

In this section we point out that the use of a partial order instead of the Hausdorff metric on pomset classes causes some problems. We consider a very simple subset of \( \text{CCS} \) where the statements are given by the production system
\[
s ::= \text{nil} \mid x \mid s_1 + s_2 \mid x; s,
\]
i.e. we consider the language \( \mathcal{P}(\Sigma, \text{Idf}) \) where \( \Sigma \) consists of the 0-ary operator symbol \( \text{nil} \), the 1-ary prefixing operator symbols and the binary operator symbol \( + \). It seems to be natural that the pomset class meaning of the language \( \mathcal{P}(\Sigma, \text{Idf}) \) is consistent with the event structure meaning in the style of the consistency result of Theorem 7.17, i.e. \( \text{pom}^*(\langle \sigma, s \rangle) = \mathcal{K}_0(ev_{cms}(\langle \sigma, s \rangle)) \) for all processes \( \langle \sigma, s \rangle \) over \( \mathcal{P}(\Sigma, \text{Idf}) \). We show that the cpo approach as proposed in Section 4 cannot be applied to define such a denotational pomset class semantics, even if we only allow guarded declarations. In the following \( \mathcal{O} \) is the restriction of \( \mathcal{E}_{\text{CCS}} \) to the operator symbols of \( \Sigma \). In addition we assume that \( \text{Idf} \) consists of a single variable \( x \).

Let \( \mathcal{K} \) denote the collection of all subsets of \( \text{Pom}_0 \). We assume that there exists a semantic domain \( D \subseteq \mathcal{K} \) and a partial order on \( D \) such that \( D \) becomes a \( \Sigma \)-cpo
where \( \emptyset \) is the associated meaning of \( \text{nil} \), the union is the semantic operator for modelling nondeterminism and the prefixing operators are defined as in Definition 6.4 and such that the meaning of a process \( \langle \sigma, s \rangle \in \mathcal{P}(\mathcal{C}, \{x\}) \) induced by Theorem 4.4 equals to the set of maximal components of its event structure meaning \( ev_{\text{em}}(\langle \sigma, s \rangle) \).

It seems to be natural that \( \emptyset \) as the meaning of \( \text{nil} \) is the bottom element of \( D \). By Remark 4.5 we know that the meaning of a recursive program \( \langle \sigma, x \rangle \) is the supremum of its finite approximations. We will see that this is impossible.

We show that there does not exist a partial order \( \subseteq \) on a suitable subset \( D \) of \( \mathcal{P} \) such that there exists a meaning function \( \text{pom}^*: \mathcal{P}(\mathcal{C}, \{x\}) \rightarrow D \) which uses the union as semantic choice operator, \( \emptyset \) as the associated meaning of \( \text{nil} \) and the prefixing operators as defined in Section 6 and which satisfies the following conditions:

- \( \emptyset \in D \) is the bottom element, i.e. \( \emptyset \subseteq H \) for all \( H \in D \).
- \( D \) is closed under prefixing and \( + \).
- the semantic choice operator \( + \) (i.e. the union) is monotone.
- \( \text{pom}^*(\langle \sigma, s \rangle) = \mathcal{X}(ev_{\text{em}}(\langle \sigma, s \rangle)) \) for all \( \langle \sigma, s \rangle \in \mathcal{P}(\mathcal{C}, \{x\}) \).
- \( \text{pom}^* \) applied to a recursive program equals the supremum of its finite approximations.

The last condition means that for each guarded statement \( s \) the sequence \( (H_n)_{n \geq 0} \) where

\[
H_0 = \emptyset, \quad H_{n+1} = \Psi[s](H_n)
\]

is monotone, i.e. \( H_0 \subseteq H_1 \subseteq H_2 \subseteq \ldots \), and the supremum of this sequence exists and equals the pomset class meaning of \( \langle \sigma, x \rangle \) where \( \sigma(x) = s \). Here \( \Psi[s]: D \rightarrow D \) is defined by structural induction:

\[
\Psi[\text{nil}](H) = \emptyset,
\Psi[x](H) = H,
\Psi[\alpha . s](H) = \alpha \cdot \Psi[s](H),
\Psi[s_1 + s_2](H) = \Psi[s_1](H) + \Psi[s_2](H).
\]

Now we show that under these conditions there exists \( I, J \in D \) with \( I \neq J \) and \( I \subseteq J \).

**Notation 8.1.** For each natural number \( n \geq 1 \) the pomset \( p_n \) is given by

\[
p_n = \frac{\overbrace{X \rightarrow X \rightarrow \cdots \rightarrow X}^n}{n}
\]

\( p \) denotes the limit of \( (p_n)_{n=1} \) i.e.

\[
p = \lim_{n \to \infty} p_n
\]

We define

\[
H_0 = \emptyset, \quad H_n = \{p_n\}, \quad H = \lim_{n \to \infty} p_n.
\]
**Lemma 8.2.** \( H_n \) is the pomset class meaning of the primitive statement \( \underbrace{\alpha \ldots \alpha}_{n} \) nil. In particular \( H_n \in D \).

(a) The sequence \( (H_n)_{n \geq 0} \) is monotone.

(b) \( H \in D \) and \( H \) is the supremum of the sequence \( (H_n)_{n \geq 0} \).

**Proof.** Let \( s = \alpha \cdot x \) and \( \sigma \) the declaration with \( \sigma(x) = s \). Since \( \text{ev}_{c_{ms}}(\langle \sigma, x \rangle) = p \) we have

\[
pom^*(\langle \sigma, x \rangle) = X_0(p) = \{ p \} = H.
\]

Since \( \Psi[s](H) = \alpha \cdot H \) and therefore \( H_{n+1} = \Psi[s](H_n) \) for all \( n \geq 0 \) we conclude that \( H_0 \subseteq H_1 \subseteq \cdots \) and that the supremum of \( (H_n)_{n \geq 0} \) exists and equals to the pomset class meaning of \( \langle \sigma, x \rangle \). We get

\[
H = \bigcup_{n=0}^{\infty} H_n.
\]

**Notations 8.3.** Let \( t \) and \( u \) be the following statements over \( (C, \{x\}) \):

\[
t = \beta \cdot x \cdot x \cdot x + x \cdot x \cdot \text{nil}, \quad u = x \cdot x \cdot (x + \text{nil}) + x \cdot x \cdot \text{nil}.
\]

The pomset classes \( I_n, J_n \) are defined as

\[
I_0 = J_0 = \emptyset, \quad I_{n+1} = \Psi[t](I_n), \quad J_{n+1} = \Psi[u](J_n).
\]

**Lemma 8.4.** For each \( n \geq 1 \) we have: \( I_n, J_n \in D \) and

(a) \( I_n = \{ p_i : 1 \leq i \leq n \} = \{ p_1, p_2, \ldots, p_{2n} \} \),

(b) \( J_n = \{ p_i : 2 \leq i \leq 2n + 1 \} = \{ p_2, p_3, \ldots, p_{2n}, p_{2n+1} \} \),

(c) \( I_n \subseteq J_n \) for all \( n \geq 1 \),

(d) \( J_n \subseteq I_{n+1} \) for all \( n \geq 1 \).

**Proof.** First we show (a) and (b). By definition of \( I_n \) and \( J_n \) we get immediately that \( I_n \) and \( J_n \) belong to \( D \). It is easy to see that for all \( K \in D \):

\[
\Psi[t](K) = \{ p_2 \} \cup \{ x \cdot x \cdot p : p \in K \},
\]

\[
\Psi[u](K) = \{ p_2, p_3 \} \cup \{ x \cdot x \cdot p : p \in K \}.
\]

Then we get the desired result by induction on \( n \).

Now we show (c) and (d): Let \( K_n = \{ p_{2i-1} : 1 \leq i \leq n \} \).

(c) First we show that \( I_n \subseteq K_n \). By Lemma 8.2(a) we have \( H_i \subseteq H_{i+1} \). Since \( + = \cup \) is monotone we get (by (a) and (b)):

\[
I_n = H_2 + H_4 + \cdots + H_{2n} \subseteq H_3 + H_5 + \cdots + H_{2n+1} = K_n.
\]

Therefore \( I_n = I_n + I_n \subseteq I_n + K_n = J_n \).
(d) By Lemma 8.2(a) we have $H_{2l-1} \subseteq H_{2l}$. By the monotony of $+ = \cup$ and by (a) and (b) we get $J_n = I_n + (H_3 + H_5 + \cdots + H_{2n+1}) \subseteq I_n + (H_4 + H_6 + \cdots + H_{2n+2}) = I_{n+1}$. □

Lemma 8.5. The pomset classes

$$I = \{ p_{2n}: n \geq 1 \} \cup \{ p \}, \quad J = \{ p_n: n \geq 2 \} \cup \{ p \}$$

belong to $D$ and we have $I \subset J$ and $J \subset I$.

Proof. It is easy to see that $I$ resp. $J$ is the set of maximal components of the event structure $ev_{cm}(\langle \sigma, x \rangle)$ resp. $ev_{cm}(\langle \sigma_n, x \rangle)$ where $\sigma_i(x) = t$ and $\sigma_n(x) = u$. Therefore

$$I - pom^*(\langle \sigma, x \rangle) \in D \quad \text{and} \quad J = pom^*(\langle \sigma_n, x \rangle) \in D.$$ 

By definition of $I_n$ and $J_n$ we have

$$I = \bigsqcup_{n \geq 0} I_n, \quad J = \bigsqcup_{n \geq 0} J_n.$$ 

By Lemma 8.4(c) and (d) we have

$$I = \bigsqcup_{n=0}^{\infty} I_n \subseteq \bigsqcup_{n=0}^{\infty} J_n = J,$$

$$J = \bigsqcup_{n=0}^{\infty} J_n \subseteq \bigsqcup_{n=0}^{\infty} I_{n+1} = I. \quad \Box$$

Hence our assumption of the existence of a partial order satisfying the conditions above leads to the contradiction $I = J$.

Appendix A. The definitions of the operators on prime event structures

A.I. The prefixing operator [3, 14]

$x. e$ describes a process which first performs $x$ and then behaves like $e$. We get $x. e$ by creating a new event $e_0$ labelled by $x$ which has no predecessors. All events of $e$ are successors of $e_0$.

Definition A.1. Let $e = (E, \preceq, \#, l) \in PrimeEv$ and $x \in Act$. Then

$$x. e = (E \cup \{ e_0 \}, \preceq \cup \{ (e_0, e): e \in E \cup \{ e_0 \} \}, \#, l \cup \{ (e_0, x) \}),$$

where $e_0 \notin E$. 


A.2. The nondeterministic choice operator $+$ [14]

$e_1 + e_2$ describes a process which behaves either like $e_1$ or like $e_2$. The decision for which alternative is chosen is given by performing the first action. We get $e_1 + e_2$ by taking the union where each pair $(e_1, e_2)$ of events $e_i$ in $e_i$ is in conflict.

**Definition A.2.** Let $e_i = (E_i, \leq_i, \#_i, l_i) \in \text{PrimeEv}, i = 1, 2, E_1 \cap E_2 = \emptyset$. Then

$$e_1 + e_2 = (E_1 \cup E_2, \leq_1 \cup \leq_2, \#_1 \cup \#_2, l_1 \cup l_2),$$

where $\# = \#_1 \cup \#_2 \cup E_1 \times E_2 \cup E_1 \times E_1$.

**Example A.3.**

$$
\begin{align*}
\begin{array}{c}
\vdash \\
\vdash
\end{array}
\end{align*}
\quad + \quad 
\begin{align*}
\begin{array}{c}
\vdash \\
\vdash
\end{array}
\end{align*}
\quad = 
\begin{align*}
\begin{array}{c}
\vdash \\
\vdash
\end{array}
\end{align*}
$$

A.3. The parallel operator $\parallel$ without synchronisation [3]

$e_1 \parallel e_2$ stands for the process which performs $e_1$ and $e_2$ in parallel (without synchronisation). We get $e_1 \parallel e_2$ by taking the “independent union”. This means

$$\neg(e_1 \# e_2) \land \neg(e_1 \leq e_2) \land \neg(e_2 \# e_1)$$

for each pair $(e_1, e_2)$ of events $e_i$ in $e_i$.

**Definition A.4.** Let $e_i = (E_i, \leq_i, \#_i, l_i) \in \text{PrimeEv}, i = 1, 2, E_1 \cap E_2 = \emptyset$. Then

$$e_1 \parallel e_2 = (E_1 \cup E_2, \leq_1 \cup \leq_2, \#_1 \cup \#_2, l_1 \cup l_2).$$

**Example A.5.**

$$
\begin{align*}
\begin{array}{c}
\vdash \\
\vdash
\end{array}
\quad \parallel \quad 
\begin{array}{c}
\vdash \\
\vdash
\end{array}
\end{align*}
\quad = 
\begin{align*}
\begin{array}{c}
\vdash \\
\vdash
\end{array}
\end{align*}
$$

A.4. The parallel operator $\mid$ with synchronisation [13]

**Definition A.6.** Let $e_i = (E_i, \leq_i, \#_i, l_i), i = 1, 2$, be prime event structures. We assume w.l.o.g. that $E_1 \cap E_2 = \emptyset$.

$\mathcal{C}_{\text{comm}}(e_1, e_2)$ denotes the set of possible communications:

$$\mathcal{C}_{\text{comm}} = \{(e_1, e_2) \in E_1 \times E_2 : l_1(e_1) = l_2(e_2) \neq \tau\},$$

and let $\mathcal{C}$ denote the set of all possible events $\mathcal{C} = E_1 \cup E_2 \cup \mathcal{C}_{\text{comm}}$. 
Let $*$ be a symbol which neither belongs to $E_1$ nor to $E_2$. We identify each event $e$ in $e_1$ resp. $e_2$ with $(e, *)$ resp. $(*, e)$. We extend $\leq_i$ and the conflict relation on $E_i \cup \{*, t\}$ in the following way:

1. $(e \leq_i *) \lor (*) \leq_i e, e \in E_i \cup \{*, t\} \Rightarrow e = * \forall e \in E_i \cup \{*, t\}$.
2. $\neg(e \#_i *) \land \neg(* \#_i e) \forall e \in E_i \cup \{*, t\}$.

Let $R_\#$ be the transitive, reflexive closure of $\rightarrow$ where the binary relation $\rightarrow$ on $\mathcal{C}$ is given by

$$(e_1, e_2) \rightarrow (e_1', e_2') \iff [(e_1 \leq_1 e_1') \land \neg(e_2 >_2 e_2')] \lor [(e_2 \leq_2 e_2') \land \neg(e_1 >_1 e_1')]$$

The conflict relation $\#_\#$ on $\mathcal{C}$ is given by

$$(e_1, e_2) \#_\# (e_1', e_2') \iff (e_1 \#_1 e_1') \lor (e_2 \#_2 e_2')$$

$$\lor [(e_1 = e_1') \land (e_2 \neq e_2')] \lor [(e_2 = e_2') \land (e_1 \neq e_1')]$$

A nonempty subset $C$ of $\mathcal{C}$ is called

- **left-closed** if for each pair $(e_1, e_2) \in C$ we have:
  1. If $e_1'$ is an event in $e_1$ such that $e_1' \leq_1 e_1$ then there exists an event $e_2'$ in $e_2$ such that $(e_1', e_2') \in C$ and $(e_1, e_2') \rightarrow (e_1, e_2)$.
  2. If $e_2'$ is an event in $e_2$ such that $e_2' \leq_2 e_2$ then there exists an event $e_1'$ in $e_1$ such that $(e_1', e_2') \in C$ and $(e_1', e_2) \rightarrow (e_1, e_2)$.

- **conflict-free** if $\neg(\xi \#_\# \xi')$ for all $\xi, \xi' \in C$.

- **linear** if $\leq_\# = R_\# \cap C \times C$ is a partial order on $C$ (i.e. $\leq_\#$ is antisymmetric) and if there exists a unique maximal element (with respect to $\leq_\#$) in $C$. (This is denoted by $\text{max}(C)$.)

- **depth-finite** if

$$\max_{e \in \pi_i(C)} \text{depth}_i(e) < \infty, \quad i = 1, 2,$$

where

$$\pi_i(C) = \{e \in E_i : (e, e') \in C \text{ for some } e' \in E_2 \cup \{*, t\} \},$$

$$\pi_2(C) = \{e \in E_2 : (e', e) \in C \text{ for some } e' \in E_1 \cup \{*, t\} \}.$$

Let $E$ denote the set of all left-closed, conflict-free, linear and depth-finite subsets of $\mathcal{C}$. We define a conflict relation $\#$ on $E$ as follows:

$$C_1 \# C_2 \iff \exists \xi_1 \in C_1 \exists \xi_2 \in C_2 \xi_1 \notin \xi_2.$$

Then

$$e_1|e_2 = (E, \leq, \#)$$
is an event structure where the labelling function $l : E \to \text{Act}$ is given by

$$l(C) = \begin{cases} \ell_{i} \left( \text{max}(C) \right) & \text{if } \text{max}(C) \in \mathcal{E}_i, \; i = 1, 2, \\ \tau & \text{if } \text{max}(C) \notin \mathcal{E}_{\text{comm}}. \end{cases}$$

**Example A.7.** Let $s_1, s_2, s_3$ resp. $s_4$ be given by

$$\begin{array}{cccc}
\tau & \rightarrow & \gamma \\
\tau & \rightarrow & \gamma \\
\beta & \rightarrow & \gamma \\
\tau & \rightarrow & \gamma \\
\end{array}$$

Since $s_2$ and $s_3$ do not contain complementary actions no communication in $s_2|s_3$ is possible. We get

$$s_2|s_3 = s_2 \parallel s_3 = \begin{array}{cccc}
\tau & \rightarrow & \gamma \\
\beta & \rightarrow & \gamma \\
\end{array}$$

The events labelled by $\bar{c}$ resp. $c$ in $s_3$ resp. $s_4$ are able to communicate. We get

$$s_4|s_3 = \begin{array}{cccc}
\tau & \rightarrow & \gamma \\
\beta & \rightarrow & \gamma \\
\end{array}$$

Next we look for $s_1|s_3$. The $\bar{c}$-event in $s_3$ has two possibilities to communicate. We get

$$s_1|s_3 = \begin{array}{cccc}
\tau & \rightarrow & \gamma \\
\beta & \rightarrow & \gamma \\
\end{array}$$

A.5. The restriction operator

If $L \subseteq \text{Act} \setminus \{\tau\}$ then $e \setminus L$ describes a process which behaves like $e$ when all actions in $L \cup \bar{L}$ are forbidden.

**Definition A.8.** Let $e = (E, \preceq, \#, l) \in \text{PrimeEv}$ and $L \subseteq \text{Act} \setminus \{\tau\}$. Then $e \setminus L = e \upharpoonright E'$ where

$$E' = \{e \in E_L : e \downarrow \subseteq E_L\}, \quad E_L = \{e \in E : l(e) \notin L \cup \bar{L}\}.$$

A.6. The relabelling operator

If $\lambda : \text{Act} \to \text{Act}$ is a relabelling function then $\epsilon \{\lambda\}$ behaves like $\epsilon$ where each action $a$ is substituted by $\lambda(a)$. 

Denotational semantics 219
Definition A.9. Let \( E= (E, \leq, \# , I) \in \text{PrimeEv} \) and \( \lambda : \text{Act} \rightarrow \text{Act} \) a relabelling function. Then

\[
\epsilon[\lambda] = (E, \leq, \# , \lambda \cdot I).
\]

It is easy to see that \( + \) and \( \parallel \) are commutative and associative. \( : \) is an associative operator with \( (e_1 + e_2) \parallel e = e_1 \parallel e + e_2 \parallel e \). \( \emptyset \) is neutral with respect to \( +, \parallel \) and \( : \). Prefixing is a special case of the sequence operator:

\[
\alpha \cdot e = \overline{\alpha} ; e.
\]

References