Applications of Compactness in the Smyth Powerdomain of Streams

(Extended Abstract)

J.-J.Ch. Meyer & E.P. de Vink

Department of Mathematics and Computer Science

Free University Amsterdam

Abstract

We show in a uniform setting the crucial role of compactness in the theory of the Smyth powerdomain of streams. The topological notion of compactness is characterized in an order-theoretical manner, involving a notion of bounded sets. We obtain general results on the continuity of operators, and consider applications as diverse as interleaving, hiding and stream programming operators.

Section 1 Introduction

This paper originated from writing a tutorial for students on (denotational) stream semantics of concurrency, such as used by e.g. Broy ([Br1], [Br2]), Back ([Ba]) and De Bakker et al. ([BKMOZ]). In the setting of this tutorial programs are built from atomic actions, sequential composition, alternative composition, parallel composition (interleaving or merge) and recursive constructs. In [Me1], [Me2] it was studied how to develop a uniform semantics for such a language based upon a powerdomain ordered by the Smyth ordering (cf. [Sm1]), which appeared to be equivalent to an observation-based semantics in the style of Hoare et al., (cf. [OH], [BMO]). (A recent application of the Smyth powerdomain to the semantics of more refined concurrency languages can be found in [Ma].) In the framework of [Me1], [Me2] atomic actions are left uninterpreted, so denotations of program statements are sets of streams (traces) of atomic actions. Moreover, it is assumed that the set of atomic actions is finite. Some of the proofs, however, are difficult and not suited for lecture notes for students.

In the meantime we found that using the Egli-Milner order instead of the Smyth ordering provided a simplification of the proofs. So when we wrote the tutorial we chose for a stream powerdomain ordered by the Egli-Milner ordering, and as such the material could be viewed as an extension of Back's work [Ba] for concurrent operators such as interleaving.

During the elaboration of the material for the lecture notes we discovered that regarding the continuity proofs of the operators, viz. sequential, alternative and parallel composition, the condition of the finiteness of the alphabet of atomic actions could be relaxed to a certain condition of boundedness of the stream-sets that occur as denotations. Informally, a stream-set is bounded if all sets of truncations up to a finite length are finite. The continuity proofs now became very elegant and uniform.

After the completion of the lecture notes we felt that it was interesting enough to write an article on the subject. At some moment we realized that the concept of boundedness could also be employed profitably in the case of the Smyth powerdomain. Moreover, we learned that in this case it was even *more* crucial. Whereas in the study of the Egli-Milner domain the boundedness concept proved to be a *useful* tool to ease the continuity proofs, in the Smyth powerdomain it is an *essential* requirement for the continuity of the

operators under consideration.

After work done in [Me1] we appreciated the fact that the finiteness of the alphabet is not needed when dealing with the Egli-Milner ordering. This was suggested already by the fact that in [Ba] Back uses the Egli-Milner ordering in order to obtain a continuous semantics of unbounded nondeterminism in sequential programming. Thus in the Egli-Milner domain also the restriction to bounded sets may not be strictly needed for continuity. (For the tutorial, the condition of boundedness simplified proofs and it held anyway in the case considered.) For continuity in the Smyth powerdomain of stream-sets, however, the restriction to bounded sets proves to be pivotal: for unbounded sets continuity simply fails to hold. Since for a research paper it is more interesting to show the extreme cases instead of just the easy cases, we have chosen to treat the topic of boundedness in the Smyth framework.

We shall give the notion of boundedness, which - in conjunction with closedness - is an alternative characterization of the well-known topological notion of compactness, (cf. [Du], [En]). We shall prove a fundamental theorem concerning converging sequences in bounded sets, upon which we base the rest of our proofs. It is in fact this theorem that enables us to give a neat and unified treatment of the issue of continuity. We consider functions on sets of streams that are lifted from continuous functions on streams and establish a result concerning the continuity of these lifted functions.

This theorem on lifted functions provides us with a uniform treatment of several diverse applications. We consider, for example, operators of uniform concurrency (sequential, alternative and parallel composition and hiding, cf. [BKMOZ], [Me1]) and, by interpreting the elements of the alphabet as data elements rather than actions, operators of stream programming such as *tail* (cf. [dBK]). Furthermore we indicate an application of our theory in the realm of logic programming, (cf. [dV]). Here we use our theory to define a continuous operator that dovetails stream-sets of substitutions in order to obtain a denotational semantics.

In [Sm2] a general domain theoretic characterization of the Smyth powerdomain is given using advanced topological techniques. In our paper we deal with a concrete domain, viz. that of sets of streams, which fact enables us to prove our results using elementary means only. Therefore no knowledge is required beyond some elementary metric topology and cpo theory.

Finally a word on the organization of this extended abstract. In order to keep our submission within the prescribed length we have omitted many proofs of the more elementary propositions and theorems. These proofs can be found in the full version of our paper [MV].

Section 2 The Smyth Powerdomain of Streams

In this section we shall establish some useful properties of the Smyth powerdomain in the special case of streams.

This domain is based upon streams of actions: For an (finite or infinite) alphabet A of actions, we define the set A^{st} of streams over A by $A^{st} = A^* \cup A^* \cdot \perp \cup A^\omega$. Let $x \in A^{st}$. x is called finished iff $x \in A^*$, x is called unfinished iff $x \in A^* \cdot \perp$ and x is called infinite iff $x \in A^\omega$.

We define the function $strip : A^{st} \rightarrow A^* \cup A^\omega$ by $strip(x) = x'$ if $x = x' \cdot \perp$ and $strip(x) = x$ otherwise. The prefix ordering $\leq_{pr}$ on $A^* \cup A^\omega$ is defined as usually. For concatenation we use the convention $x \cdot y =_{def} x$ for $x \in A^\omega$. The length of a stream x is denoted by $|x|$. If $x \in A^\omega$ then $|x| = \infty$. The norm $\|x\|$ of a stream x is defined by $\|x\| = |strip(x)|$. For $x \in A^{st}$ we define $x[n]$ by $x[n] = x$ if $|x| < n$ and $x[n] = x'$ if $|x| \geq n$ and where $x' \leq_{pr} x$ s.t. $|x'| = n$. For all $x \in A^{st}$ and for all n : $|x[n]| \leq n$ implies

$x = x[n]$. For $n \leq |x|$, we denote the n -th action of x by $x(n)$.

We put the following ordering on streams:

(2.1) DEFINITION $x \leq_{st} y$ iff $\begin{cases} \text{if } x \text{ is finished or infinite then } x = y, \\ \text{if } x \text{ is unfinished then } strip(x) \leq_{pr} strip(y). \end{cases}$

The terminology of unfinished vs. finished is motivated by the view that an element ending with $\perp$ may approximate another - more defined - element, whereas a finite element without $\perp$ cannot approximate any element but itself.

(2.2) PROPOSITION $(A^{st}, \leq_{st}, \perp)$ is a cpo. $\square$

The ordering $\leq_{st}$ on A^{st} has the following special property: if two streams x_1 and x_2 have a common upperbound, i.e. both are majorized by the same stream x_3 , then x_1 and x_2 are comparable with respect to $\leq_{st}$. This property will be frequently used in the sequel.

In the languages we are interested in, there are non-deterministic operators. So in our domain we must consider sets of streams. The next step is to define a suitable ordering on these sets of streams. We shall use the Smyth ordering ([Sm1]).

(2.3) DEFINITION

- (a) A subset $X \subseteq A^{st}$ is called *flat* iff $\forall x_1, x_2 \in X: x_1 \leq_{st} x_2 \Rightarrow x_1 = x_2$.
- (a) Let $\mathcal{P}_f(A^{st})$ be the collection of all flat subsets of $\mathcal{P}(A^{st})$.
- (b) For $X \in \mathcal{P}_f(A^{st})$, $n \geq 0$, $X[n] = \{x[n] \mid x \in X\}$ and $X(n) = \{x(n) \mid x \in X\}$.
- (c) The binary relation $\leq_s$ on $\mathcal{P}_f(A^{st})$ is for all $X_1, X_2 \in \mathcal{P}_f(A^{st})$ defined by:
 $X_1 \leq_s X_2$ iff $\forall x_2 \in X_2 \exists x_1 \in X_1: x_1 \leq_{st} x_2$.
- (d) For $X_1, X_2 \in \mathcal{P}_f(A^{st})$: $X_1 =_s X_2$ iff $X_1 \leq_s X_2$ and $X_2 \leq_s X_1$.

(2.4) LEMMA $\leq_s$ is a partial ordering on $\mathcal{P}_f(A^{st})$. $\square$

We next prove that $\leq_s$ is a complete partial ordering on $\mathcal{P}_f(A^{st})$. To this end we need a characterization of the least upperbound $\text{lub}_i X_i$ of a chain $\langle X_i \rangle_i$ in $\mathcal{P}_f(A^{st})$.

(2.5) DEFINITION Let $\langle X_i \rangle_i$ be a chain in $\mathcal{P}_f(A^{st})$. Then $\text{LUB}_i X_i =_{\text{def}} \{ \text{lub}_i x_i \mid \forall i: x_i \in X_i; \langle x_i \rangle_i \text{ is a chain} \}$.

(2.6) PROPOSITION For $X_i \in \mathcal{P}_f(A^{st})$ such that $\langle X_i \rangle_i$ is a chain, $\text{LUB}_i X_i \in \mathcal{P}_f(A^{st})$. and $\text{lub}_i X_i = \text{LUB}_i X_i$. $\square$

(2.7) PROPOSITION (Interpolation) Let $X_1 \leq_s X_2 \leq_s X_3$, X_1 flat and $x_1 \in X_1$, $x_3 \in X_3$ such that $x_1 \leq_{st} x_3$. Then there exists $x_2 \in X_2$ such that $x_1 \leq_{st} x_2 \leq_{st} x_3$. $\square$

(2.8) THEOREM $(\mathcal{P}_f(A^{st}), \leq_s, \{\perp\})$ is a cpo. $\square$

Finally in this section we consider the operator *min*, which is used to "flatten" sets within the same Smyth equivalence class:

(2.9) DEFINITION Let $X \subseteq A^{st}$. Then $\text{min}(X)$ is the set of minimal elements in X , formally given by $\text{min}(X) = \{x \in X \mid \neg \exists x' \in X: x' \leq_{st} x \wedge x' \neq x\}$.

If $X \in \mathcal{P}(A^{st})$ then $\min(X)$ is flat and $\min(X) \equiv_S X$. If $X_1, X_2 \in \mathcal{P}(A^{st})$ then $X_1 \leq_S X_2$ iff $\min(X_1) \leq_S \min(X_2)$, and $X_1 \equiv_S X_2$ iff $\min(X_1) = \min(X_2)$.

For $X \in \mathcal{P}(A^{st})$ and for all $n \geq 0$: $\min(X[n]) = (\min(X))[n]$. and for any chain $\langle X_i \rangle_i$ in $\mathcal{P}(A^{st})$ it holds that $\text{lub}_i \min(X_i) = \min(\text{lub}_i X_i)$.

Section 3 Compactness in the Smyth Powerdomain of Streams

As was already mentioned in the introduction, the notion of compactness plays an important role in the theory of powerdomains of streams. In this section we shall introduce this concept and characterize it in terms of properties of streams. After this we shall prove a fundamental theorem which enables us to establish several continuity results in section 4.

§3.1 Introduction to Compactness

Compactness is a well-known and well-studied notion in topology. (Cf. [Du], [En]). A general topological space M is called compact iff every open cover of M contains a finite subcover. In a metric space, however, we have some equivalent conditions of compactness:

(3.1) THEOREM In a metric space M the following three conditions are equivalent:

- (1) M is compact;
- (2) M is sequentially compact, i.e. every sequence in M has a converging subsequence;
- (3) M has the Bolzano-Weierstrass property, i.e. every infinite subset of M has a limit point. $\square$

In computer science compactness is used indirectly in papers on bounded nondeterminism (e.g. [dB], [Ku], [Pl]) and more directly in process theory (e.g. [BZ]) and the theory of infinite trees (e.g. [Ni]). Also, in some way, compactness was used in [Me1] and [Me2] in the form of an assumption of having a *finite* alphabet of atomic actions to ensure continuity of certain semantical operators. In our present paper, however, we shall use a more local kind of compactness to ensure continuity of semantical operators. We do not require the semantical domain as a whole to be compact (as is the case when the alphabet of atomic actions is assumed to be finite), but only the stream-sets that are to act as denotations, will have to be compact.

First of all we want to characterize the notion of a compact set in terms of streams and their orderings. We shall see that we obtain a kind of "Heine-Borel" characterization: a stream-set is compact iff it is closed and bounded.

§3.2 Closedness

We now define the notion of a closed stream-set.

(3.2) DEFINITION Let $X \subseteq A^{st}$. X is closed if $\forall x \in A^\omega \forall n \exists x_n \in X: x[n] \leq_{st} x_n \Rightarrow x \in X$.

REMARK This notion of closedness is essentially the same as that of [Ba] and can be shown equivalent to the notion of closedness induced by the standard metric on streams.

(3.3) DEFINITION The class of closed subsets of A^{st} is denoted by $\mathcal{P}_c(A^{st})$.

We note that closedness is preserved by *lub* (in the context of flat sets).

(3.4) THEOREM Let $\langle X_i \rangle_i$, $X_i \in \mathcal{P}_{cf}(A^{st})$, be a chain. Then $\text{lub}_i X_i \in \mathcal{P}_{cf}(A^{st})$. $\square$

Next, we obtain the following useful properties of closed sets.

(3.5) PROPOSITION

- (a) For any $X \in \mathcal{P}_{cf}(A^{st})$, $X = \text{lub}_n X[n]$.
- (b) For any $X \in \mathcal{P}_c(A^{st})$, $\text{min}(X) \in \mathcal{P}_{cf}(A^{st})$. $\square$

Finally, in this subsection on closedness we obtain a basic property of chains of flat, closed sets of streams, which we shall use in the subsection 3.4 for the proof of our main theorem.

(3.6) LEMMA Let $\langle X_i \rangle_i$ be a chain in $\mathcal{P}_{cf}(A^{st})$ and $x \in A^{st}$ such that $\forall i: x[i] \in X_i[i]$. Then $x \in \text{lub}_i X_i$. $\square$

§3.3 Boundedness

We proceed with the introduction of the central notion of boundedness. Intuitively a stream-set X is bounded if X can be represented by a finitely branching tree. (The streams in the set X correspond to the paths in the tree.) Formally we give a definition based on truncations.

(3.7) DEFINITION A set $X \subseteq A^{st}$ is called bounded iff $\forall n: X[n]$ is finite.

REMARK Note that if A is finite, any $X \subseteq A^{st}$ is trivially bounded (but obviously not vice versa). Thus this paper can be viewed as a proper extension of the treatment of the Smyth domain in [Me1], [Me2].

(3.8) DEFINITION The class of flat and bounded stream-sets is denoted by $\mathcal{P}_{bf}(A^{st})$.

Next we prove a few properties of a chain of flat and bounded stream-sets that we shall need below. The first one states that the infinite union of the sets in a $\leq_s$ -chain in $\mathcal{P}_{bf}(A^{st})$ is still bounded. (Note that this is clearly not generally the case for an infinite union of *arbitrary* sets in $\mathcal{P}_{bf}(A^{st})$.)

(3.9) PROPOSITION Let $\langle X_i \rangle_i$ be a chain in $\mathcal{P}_{bf}(A^{st})$. Then $\bigcup_i X_i$ is bounded. $\square$

The following proposition will be of help when we are dealing with operators in section 4. (Compare with proposition 3.5b.)

(3.10) PROPOSITION If $X \subseteq A^{st}$ is bounded, then $\text{min}(X)$ is bounded as well. $\square$

The next property we shall prove is rather technical, but plays an essential role in the next subsection where we explore properties of compact sets of streams. But before we shall prove this technical lemma we repeat a version of König's lemma (see [Ni]), which will be of considerable help in our proofs.

(3.11) PROPOSITION (König's lemma) Let X be a set, $\langle X_n \rangle_n$ a sequence in $\mathcal{P}(X)$ and $R \subseteq X \times X$ a relation on X such that

- (i) X_n is non-empty, for all n .
- (ii) X_n is finite, for all n .
- (iii) $\forall n \forall x_{n+1} \in X_{n+1} \exists x_n \in X_n: R(x_n, x_{n+1})$.
- (iv) $\bigcup_n X_n$ is an infinite set

Then there exists a sequence $\langle x_n \rangle_n$ in X such that $\forall n: x_n \in X_n$ and $\forall n: R(x_n, x_{n+1})$. $\square$

We now arrive at our main, technical lemma on chains of flat and bounded sets of streams:

(3.12) LEMMA Let $\langle X_i \rangle_i$ be a chain in $\mathcal{P}_{br}(A^{st})$, and $\langle x_i \rangle_i$ an arbitrary sequence such that $\forall i: x_i \in X_i$. Then there exists a subsequence $\langle x_{i_n} \rangle_n$ of $\langle x_i \rangle_i$ such that $\langle x_{i_n}[n] \rangle_n$ is a chain.

PROOF Put $X = \{ x_i \mid i \geq 0 \}$. If X is finite, then there is some $x \in X$ such that $x = x_i$ for infinitely many i . Trivially, a subsequence $\langle x_{i_j} \rangle_j$ with $x_{i_j} = x$, (for all j), satisfies the requirement. Now consider the case that X is infinite. Define $Y_i =_{\text{def}} X \setminus \{ x_0, \dots, x_{i-1} \}$ for $i \geq 0$. Note that $Y_i[i]$ is non-empty, for all i . Furthermore, $Y_i[i]$ is finite for all i , since $Y_i[i] \subseteq \left(\bigcup_i X_i \right)[i]$ and $\bigcup_i X_i$ is bounded by proposition 3.9. Moreover, from $Y_i \supseteq Y_{i+1}$ we have that $Y_i \leq_s Y_{i+1}$ and so also $Y_i[i] \leq_s Y_{i+1}[i] \leq_s Y_{i+1}[i+1]$. Consequently: $\forall i \forall y_{i+1} \in Y_{i+1}[i+1] \exists y_i \in Y_i[i]: y_i \leq_{st} y_{i+1}$.

Next we check that $\bigcup_i \left\{ Y_i[i] \right\}$ is an infinite set. It is sufficient to prove that

$\forall r \forall i_0, \dots, i_r \forall y_0 \in Y_{i_0}[i_0], \dots, y_r \in Y_{i_r}[i_r] \exists i \exists y \in Y_i[i]: y \in \{ y_0, \dots, y_r \}$. Suppose this is not true. Then we may choose $r, i_0, \dots, i_r, y_0 \in Y_{i_0}[i_0], \dots, y_r \in Y_{i_r}[i_r]$ such that $\forall i \forall y \in Y_i[i]: y \notin \{ y_0, \dots, y_r \}$.

Consequently, $x_i[i] \in \{ y_0, \dots, y_r \}$ for all i . Put $m = \max\{ i_0, \dots, i_r \}$ and let $i > m$. Then for every $0 \leq j \leq r$, $|y_j| \leq m < i$, and thus $|x_i[i]| < i$. Consequently, $x_i = x_i[i] \in \{ y_0, \dots, y_r \}$. Hence $Y_{m+1} = \{ x_{m+1}, x_{m+2}, \dots \} \subseteq \{ y_0, \dots, y_r \}$. Thus Y_{m+1} is finite, which contradicts the fact that $X = Y_{m+1} \cup \{ x_0, \dots, x_m \}$ is infinite. Therefore we may conclude that $\bigcup_i \left\{ Y_i[i] \right\}$ is infinite.

Now choose by means of König's lemma a sequence $\langle y_i \rangle_i$ such that $\forall i: y_i \in Y_i[i]$ and $\forall i: y_i \leq_{st} y_{i+1}$. We now construct a subsequence $\langle x_{i_n} \rangle_n$ of $\langle x_i \rangle_i$ such that $\langle x_{i_n}[n] \rangle_n$ is a chain. Put $i_{-1} = 1$. Choose i_{n+1} inductively such that $x_{i_{n+1}} \in Y_{i_n+1}$ and $x_{i_{n+1}}[i_n+1] = y_{i_n+1}$. Note that $i_{n+1} > i_n$, because $x_k \in Y_{i_n+1} = X \setminus \{ x_0, \dots, x_{i_n} \}$ implies $k > i_n$, for every k (and consequently for $k = i_{n+1}$).

Now we may check whether $\langle x_{i_n}[n] \rangle_n$ is indeed a chain: from $i_{n-1} < i_n$ it follows that $x_{i_n}[i_{n-1}+1] = y_{i_{n-1}+1} \leq_{st} y_{i_n+1} = x_{i_{n+1}}[i_n+1]$, so that (by the fact that $\forall n: i_n \geq n$) $x_{i_n}[n] \leq_{st} x_{i_{n+1}}[n] \leq_{st} x_{i_{n+1}}[n+1]$. $\square$

§3.4 Properties of Compact Stream-sets

In this subsection we shall explore some basic properties of flat and compact sets of streams, combining the facts we have seen already for closed and bounded sets. In particular we shall prove a fundamental - though technical - property of chains of those compact stream-sets, on which we shall base further proofs. We show that the Smyth domain of flat and compact stream-sets is a complete partial order. But first we shall prove a "Heine-Borel" characterization of compactness, which enables us to use the result of subsections 3.2 and 3.3.

(3.13) THEOREM A set $X \subseteq A^{st}$ is compact with respect to the standard metric on streams iff it is closed and bounded. $\square$

(3.14) DEFINITION The class of flat, closed and bounded stream-sets is denoted by $\mathcal{P}^*(A^{st})$.

REMARK By theorem 3.13 we are justified to refer to elements of $\mathcal{P}^*(A^{st})$ as flat and compact.

Next we prove our fundamental theorem on chains of compact, (i.e. closed and bounded) and flat stream-sets. To this end we use the following easy proposition on flat sets.

(3.15) PROPOSITION Let X be flat, and $\langle x_n \rangle_n$ a sequence such that $\langle x_{i_k}[k] \rangle_k$ is a chain. Moreover, let

$x \in X$ be such that $x = \text{lub}_k x_{n_k}[k]$. Then it holds that $x[k] = x_{n_k}[k]$ for all k .

PROOF Since $x_{n_k}[k] \leq_{\text{st}} \text{lub}_k \left(x_{n_k}[k] \right) = x$, we have that $x_{n_k}[k] = x_{n_k}[k][k] \leq_{\text{st}} x[k]$. Now if $x_{n_k}[k] <_{\text{st}} x[k]$, then $|x_{n_k}[k]| < k$, so $x_{n_k}[k] = x_{n_k}$. Consequently, $x_{n_k} = x_{n_k}[k] <_{\text{st}} x[k] \leq_{\text{st}} x$, contradicting the flatness of X . Hence, $x_{n_k}[k] = x[k]$. $\square$

The lemmas 3.12 and 3.6 and proposition 3.15 can be combined into the following theorem which is an order-theoretical analogue of sequential compactness of compact sets.

(3.16) THEOREM

- (a) Let $\langle X_i \rangle_i$ be a chain in $\mathcal{O}^*(A^{\text{st}})$ and $\langle x_i \rangle_i$ a sequence in A^{st} such that $\forall i: x_i \in X_i$. Then there is a subsequence $\langle x_{i_k} \rangle_k$ of $\langle x_i \rangle_i$ such that $\langle x_{i_k}[k] \rangle_k$ is a chain with $\text{lub}_k x_{i_k}[k] \in \text{lub}_i X_i$.
- (b) Let $X \in \mathcal{O}^*(A^{\text{st}})$ and $\langle x_i \rangle_i$ a sequence in X . Then there exists a subsequence $\langle x_{i_k} \rangle_k$ of $\langle x_i \rangle_i$ such that $\langle x_{i_k}[k] \rangle_k$ is a chain with $\text{lub}_k x_{i_k}[k] \in X$.

PROOF (a) By lemma 3.6 there is a subsequence $\langle x_{i_k} \rangle_k$ such that $\langle x_{i_k}[k] \rangle_k$ is a chain. Now consider the (sub-)chain $\langle X_{i_k} \rangle_k$. By proposition 3.15, $\left(\text{lub}_k x_{i_k}[k] \right)[k] = x_{i_k}[k] \in X_{i_k}[k]$. By lemma 3.12 and by proposition 2.7, $\text{lub}_k x_{i_k}[k] \in \text{lub}_k X_{i_k} = \text{lub}_i X_i$.

(b) Directly from (a), since $\langle X \rangle_i$ is a (constant) chain. $\square$

By means of this theorem we are able to establish some important facts in the sequel. But first we observe the following:

(3.17) LEMMA Let $\langle X_i \rangle_i$ be a chain in $\mathcal{O}^*(A^{\text{st}})$. Then $\text{lub}_i X_i$ is bounded. $\square$

(3.18) THEOREM $(\mathcal{O}^*(A^{\text{st}}), \leq_s, \{\perp\})$ is a complete partial order.

PROOF By proposition 2.6, theorems 2.8, 3.4 and 3.16 and lemma 3.17. $\square$

Chains $\langle X_m \rangle_m$ of bounded sets satisfy a curious property: if we take n -truncations $X_m[n]$ of X_m , then we know already that $\langle X_m[n] \rangle_m$ is also a chain: but in the case of bounded sets this chain stabilises precisely at the n -truncation of the least upperbound $\text{lub}_m X_m$. In other words, $\left(\text{lub}_m X_m \right)[n]$ can be approximated in a finite number of steps via the truncations $X_m[n]$. More formally:

(*) Let $\langle X_m \rangle_m$ be a $\leq_s$ -chain in $\mathcal{O}^*(A^{\text{st}})$. Then $\forall n \exists m: X_m[n] = \left(\text{lub}_k X_k \right)[n]$.

For the proof we need the following lemma, which states the continuity of truncation.

(3.19) LEMMA For any n , $\cdot[n]: \mathcal{O}^*(A^{\text{st}}) \rightarrow \mathcal{O}^*(A^{\text{st}})$ is continuous. $\square$

We proceed with a few additional properties of chains of bounded sets, on which we shall presently base the proof of (*).

(3.20) LEMMA Let $\langle X_m \rangle_m$ be a chain of bounded sets and let $n < \omega$ be fixed. Then $\exists m \forall m' \geq m: X_m[n] \supseteq X_{m'}[n]$. $\square$

(3.21) LEMMA Let $\langle X_m \rangle_m$ be a $\supseteq$ -chain of non-empty sets. Let X_0 be finite. Then there exists an m_0 such that $\bigcap_m X_m = X_{m_0} \neq \emptyset$. $\square$

(3.22) COROLLARY Let $\langle X_m \rangle_m$ be a chain of non-empty, bounded sets of streams and let $n < \omega$ be fixed. Then $\exists m_o: \text{lub}_m X_m[n] = X_{m_o}[n]$. $\square$

(3.23) COROLLARY Let $\langle X_m \rangle_m$ be a chain in $\mathcal{P}^*(A^{st})$ such that $X_m \neq \emptyset$, for all m , and let $n < \omega$ be fixed. Then there exists an m_o such that $\text{lub}_m X_m[n] = X_{m_o}[n] \neq \emptyset$. $\square$

REMARK This, by the way, answers a question raised in [BBKM] (Remark 2.10.1), whether for a $\supseteq$ -chain or (more generally, $\leq_S$ -chain) $\langle X_n \rangle_n$ such that $X_n \neq \emptyset$ for all n , it holds that $\bigcap_n X_n \neq \emptyset$ (or $\text{lub}_n X_n \neq \emptyset$, respectively) under a weaker condition than finiteness of A : the answer is that it also holds under the condition that the X_n are bounded.

By means of these results we get:

(3.24) THEOREM Let $\langle X_m \rangle_m$ be a $\leq_S$ -chain in $\mathcal{P}^*(A^{st})$. Then $\forall n \exists m: X_m[n] = \left(\text{lub}_k X_k \right)[n]$. $\square$

From theorem 3.24 we infer a quick corollary which we shall use in section 4.

(3.25) COROLLARY Let $\langle X_m \rangle_m$ be a chain in $\mathcal{P}^*(A^{st})$. Then $\forall n \exists m \forall l \geq m: X_l[n] = (\text{lub}_k X_k)[n]$. $\square$

Finally in this section we show that the properties of flat, compact stream-sets, such as stated in theorems 3.16 and 3.24 do not generally hold for (flat) non-compact sets.

EXAMPLES (1) Take $A = \{a\} \cup \{a_i \mid i \geq 0\}$, and $X_i = \{aa_k \mid k \geq i\}$. Note X_i is not bounded for any i . Since $\langle X_i \rangle_i$ is a $\supseteq$ -chain, it is also a $\leq_S$ -chain, with $\text{lub}_i X_i = \bigcap_i X_i = \emptyset$. Note that $X_i[1] = \{a\perp\}$ for all i . Hence, for all m , $\{a\perp\} = X_m[1] = \text{lub}_i \left(X_i[1] \right) \neq \left(\text{lub}_i X_i \right)[1] = \emptyset[1] = \emptyset$. So $\nexists m: X_m[1] = \left(\text{lub}_i X_i \right)[1]$, thus theorem 3.24 does not hold for unbounded sets. Moreover, take the sequence $\langle x_i \rangle_i = \langle aa_i \rangle_i$. Now $aa_i \in X_i$, all i , but there is clearly no subsequence $\langle x_{i_k} \rangle_k$ of $\langle x_i \rangle_i$ such that $\langle x_{i_k}[k] \rangle_k$ is a chain and $\text{lub}_k x_{i_k} \in \text{lub}_i X_i = \emptyset$.

(2) Take $X_i = \{a^k \mid k > i\}$ ($i \geq 0$). Note that $a^\omega \notin X_i$, so X_i is not closed, (all i). Again $\text{lub}_i X_i = \bigcap_i X_i = \emptyset$, so $\left(\text{lub}_i X_i \right)[1] = \emptyset$, but $\text{lub}_i X_i[1] = \text{lub}_i \{a\perp\} = \{a\perp\}$. Furthermore, take $a^{i+1} \in X_i$, for each $i \geq 0$. Now although $\langle a^{i+1}[i] \rangle_i = \langle a^i \perp \rangle_i$ is a chain, its least upper bound $a^\omega \notin \text{lub}_i X_i = \emptyset$.

Section 4 Lifting Stream-Based Functions

After having investigated some basic properties of compact (and flat) sets of streams we now turn to *functions* with values in the domain of compact and flat stream-sets. In particular we will concentrate in this section on functions that are defined originally on streams and that are then lifted to functions defined on compact sets of streams. We shall see that in this case well-definedness of the functions thus lifted is guaranteed and that, moreover, lifting preserves continuity. This notion of lifted functions can be applied in several cases, as will be shown in section 5. Without loss of generality we shall restrict ourselves to unary functions only.

Let f be a stream-based function, i.e. $f: A^{st} \rightarrow \mathcal{P}^*(A^{st})$. We are now interested in a version $\hat{f}$ of f lifted to (compact and flat) sets as defined in:

(4.1) DEFINITION For $f: A^{st} \rightarrow \mathcal{P}^*(A^{st})$ we define $f[\cdot]$, $\hat{f}: \mathcal{P}^*(A^{st}) \rightarrow \mathcal{P}^*(A^{st})$ by $f[X] = \bigcup_{x \in X} f(x)$

and $\hat{f}(X) = \min(f[X])$.

REMARKS

- (1) So $\hat{f}$ is the collection of minimal elements of the collection of (element-wise) applications of f on the elements of its argument. The operator $\min$ is added in order to guarantee flatness of the result $\hat{f}(X)$.
- (2) Note that also simple functions $f: A^{st} \rightarrow A^{st}$ can be considered in our treatment by viewing f as the function $\lambda x. \{f(x)\}$ of functionality $A^{st} \rightarrow \mathcal{P}^*(A^{st})$.

We are now interested in properties of well-definedness and continuity of the function $\hat{f}$ lifted from a stream-based function f , where f may be assumed to be continuous already. First we note that the monotonicity of $f[\cdot]$ and $\hat{f}$ is easy to prove:

(4.2) PROPOSITION Let $f: A^{st} \rightarrow \mathcal{P}^*(A^{st})$ be monotonic. Then $f[\cdot]$ and $\hat{f}$ are monotonic as well. $\square$

In order to derive the well-definedness of $\hat{f}$ (i.e. $\hat{f}$ preserves closedness and boundedness) and its continuity we employ the following lemma.

(4.3) LEMMA Let $f: A^{st} \rightarrow \mathcal{P}^*(A^{st})$ be continuous and let $X \in \mathcal{P}^*(A^{st})$. Then $\langle f[X[i]] \rangle_i$ is a chain in $\mathcal{P}(A^{st})$ and $f[X] = \text{lub}_i f[X[i]]$. $\square$

By means of this lemma we obtain:

(4.4) THEOREM Let $f: A^{st} \rightarrow \mathcal{P}^*(A^{st})$ be continuous. Then, for all $X \in \mathcal{P}^*(A^{st})$, $\hat{f}(X) \in \mathcal{P}^*(A^{st})$ and $\hat{f}(X) = \text{lub}_n \hat{f}(X[n])$. $\square$

To get the continuity of $\hat{f}$ we use:

(4.5) LEMMA Let $F: \mathcal{P}^*(A^{st}) \rightarrow \mathcal{P}^*(A^{st})$ be monotonic. Then F is continuous iff $\forall X \in \mathcal{P}^*(A^{st}): F(X) = \text{lub}_n F(X[n])$.

PROOF " $\Leftarrow$ " Let $\langle X_m \rangle_m$ be a chain in $\mathcal{P}^*(A^{st})$ and let $X = \text{lub}_m X_m \in \mathcal{P}^*(A^{st})$. By the monotonicity of F , both $\langle F(X_m) \rangle_m$ and $\langle F(X[n]) \rangle_n$ are chains, and, moreover, $\forall m \forall n: F(X_m[n]) \leq_s F(X_m)$, since $X_m[n] \leq_s X_m$ (all m, n). We now choose inductively on n a subsequence $\langle X_{m_n} \rangle_n$ of $\langle X_m \rangle_m$ such that $\forall n: X_{m_n}[n] = X[n]$: By theorem 3.24, $\exists m_0: X_{m_0}[0] = X$; Given $m_1, \dots, m_n$, by corollary 3.25 $\exists m \forall k > m: X_k[n+1] = X[n+1]$; take $m_{n+1} = \max\{m, m_1+1, \dots, m_n+1\}$; then $m_{n+1} > m_n$ and $X_{m_{n+1}}[n+1] = X[n+1]$. So now $F(X) = \text{lub}_n F(X[n]) = \text{lub}_n F(X_{m_n}[n]) \leq_s \text{lub}_n F(X_{m_n}) = \text{lub}_m F(X_m)$ using proposition 2.7. $\square$

REMARK Note that by combining lemma 4.5 with the well-definedness proof of theorem 4.4 we obtain the fact that a continuous function preserves compactness, which is well-known in a purely topological setting. (Cf. [Du], [En].)

By means of lemma 4.5 we obtain immediately

(4.6) THEOREM Let $f: A^{st} \rightarrow \mathcal{P}^*(A^{st})$ be continuous. Then $\hat{f}$ is continuous. $\square$

REMARK Note that the truncation function $\cdot[n]: \mathcal{P}^*(A^{st}) \rightarrow \mathcal{P}^*(A^{st})$ is a special case of a continuous function $f: A^{st} \rightarrow \mathcal{P}^*(A^{st})$ which is lifted: for $X \in \mathcal{P}^*(A^{st})$, $X[n] = \bigcup_{x \in X} \{x[n]\} = \min \left(\bigcup_{x \in X} \{x[n]\} \right)$. (Note that $X[n]$ is flat already, so $X[n] = (\min X[n])$.) The reason why we have

chosen for a separate proof of the continuity of $\cdot[n]$ on $\mathcal{P}^*(A^{st})$ in section 3 is that we needed this result for the lemmas leading up to theorem 4.6. By taking an alternative route (directly via theorem 3.16) it becomes possible to derive lemma 3.19 from theorem 4.6 without circularity of reasoning.

Finally a few words about the role of flatness in our theory. Although the properties of closedness and boundedness have been given the principal part in our work, flatness plays a very important, and perhaps underestimated, subsidiary part. As will be clear when checking the proofs, flatness is omnipresent in these proofs, from the most trivial propositions up to the main lemmas and theorems. Whereas the notions of closedness and of boundedness have a topological analogue, flatness is typical for our order-theoretic approach. It is intrinsic in our domain and without it most of our results would not hold any longer. For instance, the "unflattened" lifted version $f[\cdot]$ of a continuous function $f: A^{st} \rightarrow \mathcal{P}(A^{st})$ does not preserve compactness, as is shown by the following example.

EXAMPLE Let $f: A^{st} \rightarrow \mathcal{P}(A^{st})$ be defined by

$$f(x) = \begin{cases} \{x\} & \text{if } x \in A^* \\ \{\perp\} & \text{if } x \in A^* \cdot \perp \cup A^\omega \end{cases}$$

Clearly, f is monotonic and continuous. However, $f': \mathcal{P}(A^{st}) \rightarrow \mathcal{P}(A^{st})$ defined by $f'(X) = f[X]$, does not preserve closedness: $f'(a^* \cup a^\omega) = a^* \cup \{\perp\}$, which is not closed. On the other hand, of course $\hat{f} = \min(f[\cdot])$ does preserve closedness (and boundedness), (theorem 4.4).

Section 5 Applications

The theory of sections 3 and 4 can be applied fruitfully in a number of areas in denotational semantics of programming languages. We shall briefly touch upon a few examples, viz. uniform imperative concurrency ([BKMOZ], [Br1], [Mi], [BK]), stream programming ([dBK], [Bu], [Tu]) and logic programming ([JM], [Li], [dV]).

§5.1 Uniform Imperative Concurrency

In [BKMOZ] an outline is given of several approaches to semantics of imperative languages with concurrency and recursion. In [MO] also hiding is considered. The syntax of the language L_{UIC} we shall concern ourselves with in this example is as follows.

(5.1) DEFINITION For $s \in L_{UIC}$: $s ::= a \mid s_1; s_2 \mid s_1 \cup s_2 \mid s_1 \parallel s_2 \mid s \backslash a \mid s \langle a_2/a_1 \rangle \mid x \mid \mu x[s]$.

REMARKS $a \in A$ is an atomic action in a (possibly infinite) alphabet A ; it is left uninterpreted, hence the name "uniform concurrency"; $s_1; s_2$ stands for sequential composition; $s_1 \cup s_2$ for nondeterministic choice; $s_1 \parallel s_2$ for interleaving; $s \backslash a$ for hiding a from s ; $s \langle a_2/a_1 \rangle$ for renaming a_1 into a_2 and $\mu x[s]$ is a recursive call of s , (possibly) containing the variable $x \in Stmv$, the class of statement variables. (Cf. [BKMOZ], [MO]).

The semantical operators associated with $;$, $\cup$, $\parallel$, $\backslash$ and $\langle / \rangle$ are the following:

(5.2) DEFINITION Let $A_f = A^* \cup A^* \cdot \perp$.

(a) concatenation $\circ: A_f \times A_f \rightarrow \mathcal{P}(A_f)$ is given by:

- $\epsilon \circ x = \{x\}$, $\perp \circ x = \{\perp\}$ and $ax \circ y = \{a(x \circ y)\}$.
- (b) union $+$: $A_f \times A_f \rightarrow \mathcal{P}(A_f)$ by $x + y = \min \{x, y\}$.
- (c) merge $\parallel$: $A_f \times A_f \rightarrow \mathcal{P}(A_f)$ and leftmerge $\ll$: $A_f \times A_f \rightarrow \mathcal{P}(A_f)$ by
 $\epsilon \parallel x = \{x\}$, $\perp \parallel y = \{\perp\}$, $(ax) \parallel y = a(x \parallel y)$ and $x \parallel y = (x \ll y) + (y \ll x)$.
- (d) hiding $\backslash a$: $A_f \rightarrow \mathcal{P}(A_f)$ by
 $\epsilon \backslash a = \{\epsilon\}$, $\perp \backslash a = \{\perp\}$, $(ax) \backslash a = x \backslash a$ and $(a'x) \backslash a = a'(x \backslash a)$ if $a \neq a'$.
- (e) renaming $\langle a_2/a_1 \rangle$: $A_f \rightarrow \mathcal{P}(A_f)$ by
 $\epsilon \langle a_2/a_1 \rangle = \{\epsilon\}$, $\perp \langle a_2/a_1 \rangle = \{\perp\}$, $(a_1x) \langle a_2/a_1 \rangle = a_2(x \langle a_2/a_1 \rangle)$ and
 $(ax) \langle a_2/a_1 \rangle = a(x \langle a_2/a_1 \rangle)$ if $a \neq a_1$.

We shall now illustrate our theory by means of these functions. In fact, it is possible to obtain continuous lifted versions of these functions via two ways.

We next describe the first one.

(5.3) DEFINITION For $\otimes \in \{\circ, +, \parallel\}$, $x, y \in A^{st}$ we define

$$x \otimes y = \text{lub}_n \ x[n] \otimes y[n], \ x \backslash a = \text{lub}_n \ x[n] \backslash a \text{ and } x \langle a_2/a_1 \rangle = \text{lub}_n \ x[n] \langle a_2/a_1 \rangle.$$

These functions can be shown to be continuous:

(5.4) PROPOSITION $\circ, +, \parallel$: $A^{st} \times A^{st} \rightarrow \mathcal{P}^*(A^{st})$ and $\backslash a, \langle a_2/a_1 \rangle$: $A^{st} \rightarrow \mathcal{P}^*(A^{st})$ are continuous. $\square$

Furthermore, define lifted versions of $\circ, +, \parallel$ and $\backslash a$ as usual: for $\otimes \in \{\circ, +, \parallel\}$ and $X, Y \in \mathcal{P}^*(A^{st})$,
 $X \otimes \hat{\ } Y = \min (\otimes [X, Y])$, $X(\backslash a) \hat{\ } = \min ((\backslash a)[X])$
and $X \langle a_2/a_1 \rangle \hat{\ } = \min (\langle a_2/a_1 \rangle [X])$.

These functions are obviously monotonic. Now, by theorems 4.4 and 4.6 we obtain immediately.

(5.5) THEOREM $\circ \hat{\ }, + \hat{\ }, \parallel \hat{\ }, \backslash a \hat{\ }$ and $\langle a_2/a_1 \rangle \hat{\ }$: $\mathcal{P}^*(A^{st}) \rightarrow \mathcal{P}^*(A^{st})$ are well-defined and continuous. $\square$

The continuity of the operators allows us to give a denotational semantics for L_{UIC} as usual, (cf. e.g. [dB]).

REMARKS

(1) The synchronization merge of [BKMOZ] can also be treated in this way. Our theory thus provides an alternative and more uniform proof method of the continuity of the semantical operators occurring in [BKMOZ] and [Me1], [MO].

(2) Note again the crucial role of compactness. For instance, consider $X_i = \{a^k \mid i < k < \omega\}$, for $i \geq 0$. Then $\langle X_i \rangle_i$ is a chain of non-closed, but bounded stream-sets and $\text{lub}_i X_i = \bigcap_i X_i = \emptyset$. Furthermore, take $Y_i = \{a^\omega\}$, for all $i \geq 0$. So $\text{lub}_i X_i \circ Y_i = \text{lub}_i \{a^\omega\} = \{a^\omega\}$, whereas $(\text{lub}_i X_i) \circ (\text{lub}_i Y_i) = \emptyset \circ \{a^\omega\} = \emptyset$. By taking $A = \{a_0, a_1, \dots\}$, $X'_i = \{a_0^\omega\}$ and $Y'_i = A \setminus \{a_0, \dots, a_i\}$ we obtain a similar example of discontinuity in the case of non-bounded, but closed sets: $\text{lub}_i X'_i \circ Y'_i = \text{lub}_i \{a_0^\omega\} = \{a_0^\omega\}$, but $(\text{lub}_i X'_i) \circ (\text{lub}_i Y'_i) = \{a_0^\omega\} \circ \emptyset = \emptyset$.

Quite some work is still involved with the proof of the continuity of the function $\parallel$: $A^{st} \times A^{st} \rightarrow \mathcal{P}^*(A^{st})$. Therefore it might be more convenient to take another route for "difficult" functions like $\parallel$: Define a version of $\parallel$ on the collection $\mathcal{P}_{ff}(A_f)$ of finite and flat subsets of A_f . This is done as follows: For $X, Y \in \mathcal{P}_{ff}(A_f)$, $X \parallel \tilde{\ } Y = \min (\cup \{x \parallel y \mid x \in X, y \in Y\})$. Extend this $\parallel \tilde{\ }$ to a function $\parallel^\circ$

on $\mathcal{P}^*(A^{st})$: for $X, Y \in \mathcal{P}^*(A^{st})$, $X \parallel^\circ Y = \text{lub}_n X[n] \parallel Y[n]$. Then we have: for every $X, Y \in \mathcal{P}^*(A^{st})$: $X \parallel^\circ Y = \text{lub}_n X[n] \parallel^\circ Y[n]$. So $\parallel^\circ$ is continuous by lemma 4.5.

§5.2 Stream Programming

By viewing the alphabet A of section 5.1 as a set of data elements rather than uninterpreted actions and adding functions $A^{st} \rightarrow A^{st}$, we obtain a stream programming language in the style of [Br2], [Bu], [Tu] and [dBK].

(5.6) DEFINITION $S \in L_{SP}$ is given by:

$$s ::= a \mid s_1 \cdot s_2 \mid \underline{f}(s) \mid x \mid \mu X[s], \quad S ::= s \mid S_1 \cdot S_2 \mid S_1 \cup S_2 \mid S_1 \parallel S_2 \mid X \mid \mu X[S].$$

REMARKS

- (1) $a \in A$ is a data element from the (possibly infinite) set A of data elements.
- (2) Denotations of s are streams in A^{st} . Those of S are stream-sets in $\mathcal{P}^*(A^{st})$.
- (3) Since in this subsection we do not consider our language to be imperative, we write the concatenation operator as " $\cdot$ " rather than as " $;$ ".
- (4) The semantical counterparts $\cdot$, $\cup$ and $\parallel$ are as in section 5.2; so we may assume continuous, lifted versions $\cdot^\wedge$, $\cup^\wedge$ and $\parallel^\wedge$ with the properties desired.
- (5) Operators $\underline{f}$ are meant to denote $\leq_{st}$ -continuous stream-based functions $f: A^{st} \rightarrow \mathcal{P}^*(A^{st})$. So by theorems 4.4 and 4.6 they have well-defined and continuous, lifted versions $\hat{f}: \mathcal{P}^*(A^{st}) \rightarrow \mathcal{P}^*(A^{st})$.

So in a similar way, as in section 5.1 we obtain immediately a denotational semantics of L_{SP} based on fixed point theory.

EXAMPLE Take $A = \mathbb{N}$, $\underline{f}_1(s) = s+1$; $\underline{f}_2 = \text{tail}$ with meanings f_1 and f_2 , given by:

$$\begin{aligned} f_1(\epsilon) &= f_2(\epsilon) = \{\epsilon\}, \\ f_1(\perp) &= f_2(\perp) = \{\perp\}, \\ f_1(ax) &= (a+1) \cdot f_1(x), \text{ for } x \in A_f \\ f_1(x) &= \text{lub}_n f_1(x[n]), \text{ for } x \in A^\omega \\ f_2(ax) &= \{x\}, \text{ for } x \in A^{st}. \end{aligned}$$

Clearly, f_1 and f_2 are continuous functions $A^{st} \rightarrow \mathcal{P}^*(A^{st})$. So by theorems 4.4. and 4.6 their lifted versions $\hat{f}_1, \hat{f}_2: \mathcal{P}^*(A^{st}) \rightarrow \mathcal{P}^*(A^{st})$ are well-defined and continuous.

In this language one can program streams like Fibonacci's: $\mu X[1(X+0 \cdot X)]$.

REMARK Note that, since we do not employ contractive functions as in [dBK], we do not have to bother about the notion of guardedness as in [dBK]. Thus, in our framework we can treat a proper super-set of the functions dealt with in [dBK]. However, we still have problems with functions such as *perm* defined by

$$\text{perm}(x) = \begin{cases} \text{the set of all permutations of } x, & \text{if } x \in A^* \cup A^\omega \\ \text{perm}(x') \cdot \perp, & \text{if } x = x' \cdot \perp \end{cases}$$

for two reasons.

- (1) *perm*(x) is not a continuous function $A^{st} \rightarrow \mathcal{P}(A^{st})$: $\text{perm}(1\perp) = \{1\perp\} \not\leq_s \{12\perp, 21\perp\} = \text{perm}(12\perp)$.

(2) $\text{perm}(x)$ is not bounded for $x \in A^{\text{st}}$: $\text{perm}(a_1 a_2 a_3 \cdot \cdot \cdot) = a_1 X_1 \cup a_2 X_2 \cup a_3 X_3 \cup \cdot \cdot \cdot$ for some X_i ($i \geq 1$) and so $(\text{perm}(a_1 a_2 a_3 \cdot \cdot \cdot))[1] = \{a_1, a_2, a_3, \cdot \cdot \cdot\}$ is infinite!

§5.3 Logic Programming

The notion of bounded stream-sets was originally investigated by one of the authors (EdeV) in the context of infinite logic programming (see [NA], [dV]). In order to give a denotational semantics of infinite logic programs one has to handle infiniteness and non-determinacy. Moreover, the semantics should be compositional.

Infiniteness of computations can be modeled by (finite or infinite) streams of substitutions. Thus, the n -th substitution in a stream reflects the resolution of all subgoals of depth n . Non-determinism can be embedded in the semantics by the use of sets of streams instead of just streams. Compositionality can be handled by dovetailing the denotations of the conjuncts of a conjunction in order to obtain the denotation of the conjunction itself.

To be more specific, let *Prog* be the collection of logic programs, i.e. the collections of finite sets of program-clauses and let *Goal* be the collections of goal-clauses. (We follow the terminology of [L1].) Let *Subst* be the collection of substitutions and let *Env* be the collections of environments, i.e. $\text{Env} = \text{Goal} \rightarrow \mathcal{P}^*(\text{Subst}^{\text{st}})$. (The environments are used to treat recursion.) We want a function $D: \text{Prog} \rightarrow \text{Env} \rightarrow \text{Goal} \rightarrow \mathcal{P}^*(\text{Subst}^{\text{st}})$ to give meaning to a logic program and a goal in a certain environment.

First we define the dovetailing of streams $\&_o: \text{Subst}^* \cup \text{Subst}^* \cdot \perp \rightarrow \text{Subst}^*$ is defined by $x \&_o \epsilon = x$ and $\epsilon \&_o y = y$, $x \&_o \perp = \perp \&_o y = \perp$ and $\sigma x \&_o \theta y = \text{mgu}(\sigma, \theta) \cdot (x \&_o y)$. (where *mgu* stands for a most general unifier) and $\&: \text{Subst}^{\text{st}} \times \text{Subst}^{\text{st}} \rightarrow \mathcal{P}^*(\text{Subst}^{\text{st}})$ by $x \& y = \{ \text{lub}_n x[n] \&_o y[n] \}$. It is easily verified that $\&$ is well-defined and continuous. So we can lift $\&$ to a map $\hat{\&}: \mathcal{P}^*(\text{Subst}^{\text{st}}) \times \mathcal{P}^*(\text{Subst}^{\text{st}}) \rightarrow \mathcal{P}^*(\text{Subst}^{\text{st}})$.

If we want to take advantage of the theory of section 4 we must be sure that all stream-sets involved are bounded. Informally this is checked by the observation that the resolution of an atom yields only finitely many subgoals. Then we can derive from theorems 4.4 and 4.6 that $\hat{\&}$ is continuous. We define $D(P)(\gamma)(c_1 \wedge c_2) = D(P)(\gamma)(c_1) \hat{\&} D(P)(\gamma)(c_2)$ for $P \in \text{Prog}$, $\gamma \in \text{Env}$ and $c_1, c_2 \in \text{Goal}$. $D(P)(\gamma)(c_1 \wedge c_2)$ is well-defined, provided that $D(P)(\gamma)(c_1)$ and $D(P)(\gamma)(c_2)$ are bounded.

The continuity of $\hat{\&}$ is important. It guarantees that the function D will be a continuous mapping of functionality $\text{Prog} \rightarrow \text{Env} \rightarrow \text{Goal} \rightarrow \mathcal{P}^*(\text{Subst}^{\text{st}})$ or equivalently $\text{Prog} \rightarrow \text{Env} \rightarrow \text{Env}$. Hence for a fixed program P , $D(P)$ has a least fixed point which will be the main ingredient for the denotation of the logic program P . Further details are given in [dV].

Section 6 Concluding Remarks

We have developed a powerdomain theory for streams around a certain notion of compactness, characterized by notions of closedness and boundedness. These notions appeared to be very useful in order to establish continuity results for functions that are lifted from streams to sets of streams. Moreover, we discovered that compactness is essential for these continuity results in the Smyth powerdomain of streams. Allowing non-compact stream-sets leads to discontinuity.

After an investigation of the basic properties of compact stream-sets, we turned to our main problem,

viz. the well-definedness and continuity of functions that are lifted versions of continuous stream-based ones. We proved a theorem which stated that these functions are indeed well-defined and continuous. This general theorem was applied in several diverse areas such as uniform concurrency and stream programming. Our uniform method is amenable in all cases where compactness is available.

As was said before, non-compactness leads to discontinuity of the functions concerned in our Smyth framework. Whether in the case of unbounded but closed stream-sets the problem of discontinuity can be evaded by adopting the Egli-Milner ordering instead, remains an unsolved question for the present. Although it is clear to us that certain specific operators such as sequential composition and interleaving do allow such an escape route, we have serious doubts, whether this can be done in general for all lifted functions. The situation of non-closed (but possibly bounded) sets of streams seems to be even less clear. This is the area of issues of fairness (cf. [Me2]) and it is also an interesting object of future study. Perhaps lemmas 3.6 and 3.12 are still of some use in these situations.

Section 7 References

- [Ba]. R.J.R. Back, "A Continuous Semantics for Unbounded Nondeterminism," *Theoretical Computer Science* **23**, pp. 187-210 (1983).
- [dB]. J.W. de Bakker, *Mathematical Theory of Program Correctness*, Prentice-Hall (1980).
- [BBKM]. J.W. de Bakker, J.A. Bergstra, J.W. Klop, and J.-J.Ch. Meyer, "Linear Time and Branching Time Semantics for Recursion with Merge," *Theoretical Computer Science* **34**, pp. 135-156 (1984).
- [dBK]. J.W. de Bakker and J.N. Kok, "Towards a Uniform Topological Treatment of Streams and Functions on Streams," *Proceedings of the 12th ICALP*, pp. 140-148, Springer, LNCS 194 (1985).
- [BKMOZ]. J.W. de Bakker, J.N. Kok, J.-J.Ch. Meyer, E.-R. Olderog and J.I. Zucker, "Contrasting Themes in the Semantics of Imperative Concurrency," in *Current Trends in Concurrency: Overviews and Tutorials*, pp. 51-121, Springer, LNCS 224 (1986).
- [BMO]. J.W. de Bakker, J.-J.Ch. Meyer, and E.-R. Olderog, "Infinite Streams and Finite Observations in the Semantics of Uniform Concurrency," *Proceedings of the 12th ICALP*, pp. 149-157, Springer, LNCS 194 (1985).
- [BZ]. J.W. de Bakker and J.I. Zucker, "Compactnes in Semantics for Merge and Fair Merge," *Proceedings Workshop Logics of Programs*, Pittsburgh, pp. 18-33, Springer, LNCS 154 (1983).
- [BK]. J.A. Bergstra and J.W. Klop, "Process Algebra for Synchronous Communication," *Information and Control* **60**, pp. 109-137 (1984).
- [Br1]. M. Broy, "Fixed Point Theory for Communication and Concurrency," *Proceedings of the IFIP TC2 Working Conference '82*, Garmisch-Partenkirchen, North-Holland (1983).
- [Br2]. M. Broy, "Extensional Behaviour of Concurrent, Nondeterministic, Communicating Systems," pp. 229-276 in *Control Flow and Data Flow: Concepts of Distributed Programming*, ed. M. Broy, Springer (1985).
- [Bu]. W.H. Burge, "Stream Processing Functions," *IBM Journal of Research and Development* **19**, pp. 12-25 (1975).
- [Du]. J. Dugundji, *Topology*, Allyn and Bacon (1966).

- [En]. R. Engelking, *General Topology*, Polish Scientific Publishers (1977).
- [JM]. N.D. Jones and A. Mycroft, "Stepwise Development of Operational and Denotational Semantics for Prolog," *1984 International Symposium on Logic Programming*, Atlantic City, pp. 281-288 (1984).
- [Ku]. R. Kuiper, "An Operational Semantics for Bounded Nondeterminism Equivalent to a Denotational One," *IFIP TC2 - MC Symposium on Algorithmic Languages*, Amsterdam, pp. 373-398, North-Holland (1981).
- [Li]. J.W. Lloyd, *Foundations of Logic Programming*. Springer (1984).
- [Ma]. M. Maïne, "Demons, Chaos and Communicating Processes: A Model Theory for Milner's CCS in Terms of Smyth Powerdomains," *Proceedings of the Workshop on Mathematical Foundations of Programming Semantics* (1986).
- [Me1]. J.-J.Ch. Meyer, *Programming Calculi Based on Fixed Point Transformations: Semantics and Applications*, Free University, Dissertation, Amsterdam (1985).
- [Me2]. J.-J.Ch. Meyer, "Merging Regular Processes by Means of Fixed Point Theory," *Theoretical Computer Science*, to appear.
- [MO]. J.-J.Ch. Meyer and E.-R. Olderog, "Hiding in Stream Semantics of Uniform Concurrency," Technical Report, Free University, Amsterdam. In preparation.
- [MV]. J.-J.Ch. Meyer and E.P. de Vink, "Applications of Compactness in the Smyth Powerdomain of Streams," Technical Report IR-110, Free University, Amsterdam (1986).
- [Mi]. R. Milner, *A Calculus for Communicating Processes*, Springer, LNCS 92 (1980).
- [NA]. M.A. Nait Abdallah, "On the Interpretation of Infinite Computations in Logic Programming," *Proceedings of the 11th International Colloquium on Automata Languages and Programming*, Antwerp, pp. 358-370, Springer, LNCS 172 (1984).
- [Ni]. M. Nivat, "Infinite Words, Infinite Trees, Infinite Computations," pp. 1-52 in *Foundations of Computer Science III*, ed. J.W. de Bakker & J. van Leeuwen, Mathematical Centre, Amsterdam (1979).
- [OH]. E.-R. Olderog and C.A.R. Hoare, "Specification-Oriented Semantics for Communicating Processes," *Acta Informatica* **23**, pp. 9-66 (1986).
- [Pi]. G.D. Plotkin, "A Power Domain Construction," *SIAM Journal on Computing* **5**, pp. 452-487 (1976).
- [Sm1]. M.B. Smyth, "Power Domains," *Journal of Computer and System Sciences* **16**, pp. 23-36 (1978).
- [Sm2]. M.B. Smyth, "Power Domains and Predicate Transformers: A Topological View," *Proceeding of the 10th ICALP*, pp. 662-675, Springer, LNCS 154 (1983).
- [Tu]. D.A. Turner, "Recursion Equations as a Programming Language," pp. 1-28 in *Functional Programming and Its Applications*, ed. J. Darlington, P. Henderson & D.A. Turner, Cambridge University Press (1982).
- [dV]. E.P. de Vink, "Stream Semantics for Infinite Logic Programming," Technical Report, Free University, Amsterdam. In preparation.