Dedekind completion as a method for constructing new Scott domains

Klaus Grue

DIKU (Dept.Comp.Sci.)
University of Copenhagen
Copenhagen, Denmark

Abstract

Many operations exist for constructing Scott-domains. This paper presents Dedekind completion as a new operation for constructing such domains and outlines an application of the operation. Dedekind complete Scott domains are of particular interest when modeling versions of λ-calculus that allow quantification over sets of arbitrary cardinality. Hence, it is of interest when constructing models of powerful specification languages and when using λ-calculus as a foundation for mathematics.

1 Introduction

As is well known, Scott domains and inverse limit constructions [12,1] are convenient for solving domain equations and for constructing models of λ-calculus, and thus they are convenient for modeling computable functions.

As was noted already by Scott, the notion of a Scott domain can be generalized to the notion of a κ-Scott domain [2,3,4] for any set κ whose cardinality is regular (the notion merely depends on the cardinality of κ). $\mathbb{N}$-Scott domains (i.e. ordinary Scott domains) are suited for modeling computable functions. $\mathbb{R}$-Scott domains are suited for modeling parallelism or for modeling λ-calculi that contain a quantifier that quantifies over some countable set. For sufficiently large κ, κ-Scott domains can model theories that are strong enough to prove the consistency of ZFC [2,4].

Since the inverse limit construction, other methods for constructing Scott domains have been developed. Those methods often lead to models that are easier to work with. In particular, as pointed out in [5], the inverse limit construction is heavy to use in connection with construction of κ-Scott domains for uncountable κ. One such alternative method is to use webbed domains as presented in [11]. Two approaches to webbed domains are Girards coherent

[1] Email:grue@diku.dk

This is a preliminary version. The final version will be published in
Electronic Notes in Theoretical Computer Science
URL: www.elsevier.nl/locate/entcs

spaces [6] and Krivines initial segments [9,10]. The approach of this paper to webbed domains follows [2].

The Dedekind operation defined later is useful in connection with webbed domains. When applicable, it allows to find new solutions D to domain equations that have been solved before, but such that the new solutions have the following closure property: Every set S of compact elements of D that has a compact upper bound also has a "compact supremum". A "compact supremum" is the least upper bound of S among all compact elements of D (a set S may have both a compact supremum and an ordinary supremum, in which case the compact supremum is greater than or equal the ordinary one). The closure property also ensures existence of compact infima. Existence of such suprema and infima can be handy when introducing quantifiers into lambda calculi.

The paper starts with some chores such as defining what a pcs is. Section 3 describes how to construct Scott domains from pcs's. Section 4 describes operations on pcs's that are well known. Section 5 gives an example of how to solve a domain equation with the given machinery.

Section 6 defines the notion of a Dedekind cut in the context of pcs's and Scott domains. That section also defines the Dedekind operation presented in this paper. Section 7 states that, under reasonable conditions, the Dedekind operation commutes with most of the standard operations. Commutation is what allows the Dedekind operation to find new solutions to given domain equations, i.e. to perturb solutions without changing their equational properties. Section 8 gives an example of finding a new solution to the domain equation from Section 5.

Section 9 hints at the application for which the Dedekind operation was originally devised: the Dedekind operation is needed in a crucial step deep inside a rather large consistency proof for the theory presented in [8] (the consistency proof itself is not yet published). The consistency proof builds on the domain from Section 8.

The present paper, however, focuses on the Dedekind operation as a general, new tool for constructing Scott domains and investigates the general properties of the operation.

Non-trivial proofs are in the appendix. Trivial proofs are omitted.

2 Preordered coherent spaces (pcs's)

We shall say that a triple $P = \langle A, B, C \rangle$ is a *triplet* if A is a set and B and C are relations over A (i.e. $B, C \subseteq A \times A$). We shall say that a triplet $P = \langle A, B, C \rangle$ is a *preordered, coherent space* (*pcs*) if

- B is reflexive and transitive on A,
- C is reflexive and symmetric on A, and
- B and C are *compatible* in the sense defined in a moment.

For all triplets $P = \langle A, B, C\rangle$ and all $x, y \in A$ we shall say that x is *below* y (w.r.t. P), written $x \preceq_P y$, if $\langle x, y\rangle \in B$. If $x \preceq_P y$ then we shall also say that y is *above* x, that x is *an inferior* of y, and that y is *a superior* of x.

We shall say that x is *coherent with* y (w.r.t. P), written $x \sim_P y$, if $\langle x, y\rangle \in C$. If $x \sim_P y$ then we shall also say that x and y *are coherent*.

We shall refer to A as the *domain* of P. We shall use D_P to denote the domain of P.

The definition of the notion of a pcs referred to the notion of "compatibility". For a triplet $P = \langle A, B, C\rangle$, B and C are "compatible" if the following holds: If x and y are compatible, then all inferiors of x are compatible with all inferiors of y.

In summary, a triplet P is a pcs if the following hold:

$$\forall x{\in}D_P\colon x \preceq_P x$$

$$x \preceq_P y \wedge y \preceq_P z \Rightarrow x \preceq_P z$$

$$\forall x{\in}D_P\colon x \sim_P x$$

$$x \sim_P y \Rightarrow y \sim_P x$$

$$\overline{x} \sim_P \overline{y} \wedge x \preceq_P \overline{x} \wedge y \preceq_P \overline{y} \Rightarrow x \sim_P y$$

We shall say that x and y are *consistent* if they have a common superior. A pcs S is a *κ-Scott pcs* if $\langle D_S, \preceq_S\rangle$ is a κ-Scott domain and if the coherence relation of S coincides with the consistency relation (i.e. $x \sim_S y \Leftrightarrow \exists z{\in}D_S\colon x \preceq_S z \wedge y \preceq_S z$).

Hence, a κ-Scott pcs is just a κ-Scott domain with an extra relation $x \sim_S y$. In a κ-Scott pcs, $x \sim_S y$ merely contains redundant information.

There clearly is a one-to-one correspondence between κ-Scott domains and κ-Scott pcs's. We merely introduce κ-Scott pcs's here for notational convenience.

3 Construction of κ-Scott pcs's

The motivation for looking at pcs's is that one can construct κ-Scott domains from them. The power pcs operator defined below shows how to do that.

Let P be a pcs and let $A, B \subseteq D_P$. A is an *initial segment* (w.r.t. P) if all superiors of all elements of A are themselves elements of A. A is *self-coherent* (w.r.t. P) if all elements of A are coherent to each other (i.e. $\forall x, y{\in}A\colon x \sim_P y$). A and B are *set-coherent* (w.r.t. P), written $A \sim_P^* B$, if all elements of A are coherent to all elements of B. A is *set-below* B, written $A \preceq_P^* B$, if all elements of A have a superior in B. Let the *power pcs* $\mathcal{P}(P)$ denote the unique triplet R for which:

(1) D_R is the set of self-coherent, initial segments of P

(2) $x \preceq_R y \Leftrightarrow x \preceq_P^* y$

(3) $x \sim_R y \Leftrightarrow x \sim^*_P y$

Note that one can always define a triplet R uniquely by specifying D_R, $x \preceq_R y$, and $x \sim_R y$. In (2) and (3) it is understood that x and y range over D_R and that $x \preceq_R y$ and $x \sim_R y$ are false when $x \notin D_R \vee y \notin D_R$. This generally is to be understood in definitions of triplets.

The power pcs operator is useful for constructing κ-Scott pcs's because of the following simple fact:

Lemma 3.1 *If P is a pcs then $\mathcal{P}(P)$ is a κ-Scott pcs.*

In other words, one may construct κ-Scott domains thus: Construct a pcs P. As long as P is a pcs,

$$\langle D_{\mathcal{P}(P)}, \preceq_{\mathcal{P}(P)} \rangle$$

is a κ-Scott domain. Hence, the more operations one has available on pcs's, the more κ-Scott domains will one be able to construct.

A more verbose formulation of the lemma above reads: If P is a pcs then $\mathcal{P}(P)$ is a κ-Scott pcs for all sufficiently large κ.

4 Standard operations on pcs's

Now that we have seen that operations on pcs's are useful, it is worth looking at the "standard" operations on pcs's rather than jumping to the Dedekind operation, which is the new operation proposed in this paper. One reason is that the Dedekind operation has the very convenient property that it commutes with the "standard" operations, so the "standard" operations have to be defined somewhere in the paper anyway. This section defines the following operations:

- E: the empty pcs.
- $\bigcup S$: the union of a set of pcs's.
- $\mathrm{U}(a)$: the unit pcs (that merely contains the object a).
- P_a: the pcs P with a added at the bottom.
- $P \oplus Q$: the direct sum of the pcs's P and Q.
- $\updownarrow P$: the pcs P reversed bottom-up.
- $P \otimes Q$: an operation that is useful for defining $P \rightarrow Q$.
- $P \rightarrow Q$: a model for functions from the pcs P to the pcs Q.
- $\mathcal{P}_\kappa(P)$: a bounded power pcs operator. A set A is said to be B-*small* if the cardinality of A is less than the cardinality of B. As examples, $\mathbb{N}$ and $\mathbb{Z}$ are $\mathbb{R}$-small whereas $\mathbb{N}$ is not $\mathbb{Z}$-small. $\mathcal{P}_\kappa(P)$ is a pcs of κ-small subsets of D_P.

Table 1 defines all the concepts above except $P \rightarrow Q$ whose definition reads

$$P \rightarrow Q = \updownarrow P \otimes Q$$

	D_R	$x \preceq_R y$	$x \sim_R y$
$R{=}\mathsf{E}$	$\emptyset$	$0{=}1$	$0{=}1$
$R{=}\bigcup S$	$\bigcup_{P\in S} D_P$	$\exists P{\in}S\colon x{\preceq_P}y$	$\exists P{\in}S\colon x{\sim_P}y$
$R{=}\mathrm{U}(a)$	$\{a\}$	$0{=}0$	$0{=}0$
$R{=}P_a$	$D_P{\cup}\{a\}$	$x{=}a \vee x{\preceq_P}y$	$x{=}a \vee y{=}a \vee x{\sim_P}y$
$R{=}P{\oplus}Q$	$D_P{\cup}D_Q$	$x{\preceq_P}y \vee x{\preceq_Q}y$	$x{\sim_P}y \vee x{\sim_Q}y$
$R{=}{\updownarrow}P$	D_P	$y{\preceq_P}x$	$x{\not\sim_P}y$
$R{=}P{\otimes}Q$	$D_P{\times}D_Q$	$x_1{\preceq_P}y_1 \wedge x_2{\preceq_Q}y_2$	$x_1{\sim_P}y_1 \vee x_2{\sim_Q}y_2$
$R{=}\mathcal{P}_\kappa(P)$	(*)	$x{\preceq^*_P}y$	$x{\sim^*_P}\, y$

(*) the set of κ-small, self-coherent subsets of P.

Table 1
Definitions of various triplets R

As an example of reading Table 1, the fifth line says that $P \oplus Q$ is the unique triplet R for which $D_R = D_P{\cup}D_Q$, $x{\preceq_R}y \Leftrightarrow x{\preceq_P}y \vee x{\preceq_Q}y$, and $x{\sim_R}y \Leftrightarrow x{\sim_P}y \vee x{\sim_Q}y$. In the table, 0=0 and 0=1 are used as statements that are always true and false, respectively. In the penultimate line of Table 1, x_n and y_n denote the n'th element of the tuples x and y, respectively.

Under reasonable conditions, the above triplets are pcs's:

Lemma 4.1 E *is a pcs.*

For triplets $P = \langle A, B, C\rangle$ and $Q = \langle A', B', C'\rangle$ define

$$P{\sqsubseteq}Q \Leftrightarrow A{\subseteq}A' \wedge B{\subseteq}B' \wedge C{\subseteq}C'$$

A set S of pcs's is a *chain* if $\forall P, Q{\in}S\colon P \sqsubseteq Q \vee Q \sqsubseteq P$.

Lemma 4.2 $\bigcup S$ *is a pcs for all chains S of pcs's.*

Lemma 4.3 $\mathrm{U}(a)$ *is a pcs for all objects a (of ZFC).*

Lemma 4.4 *If P is a pcs and $a \notin D_P$ then P_a is a pcs*

Lemma 4.5 *If P and Q are pcs's and D_P and D_Q are disjoint, then $P \oplus Q$ is a pcs.*

A triplet P is an *anti-pcs* if it is a pcs except that $x \sim_P y$ is anti-reflexive: $\forall x{\in}D_P\colon x \not\sim_P x$.

Lemma 4.6 ${\updownarrow}P$ *is an anti-pcs for all pcs's P.*

Lemma 4.7 *$P \otimes Q$ is a pcs for all anti-pcs's P and all pcs's Q.*

Lemma 4.8 *$P \rightarrow Q$ is a pcs for all pcs's P and Q.*

Lemma 4.9 *$\mathcal{P}_\kappa(P)$ is a pcs for all pcs's P and all sets κ.*

5 Example of solving a domain equation

A κ-Scott domain that satisfies the domain equation

(4) $M = (M \rightarrow M)_\perp + \{T\}$

may be constructed thus:

(5) $t = 0$
(6) $f = 1$
(7) $H_\kappa(P) = (\mathcal{P}_\kappa(P) \rightarrow P)_f \oplus \mathrm{U}(t)$
(8) $M_\alpha = \bigcup\{H_\kappa(M_\beta) | \beta \in \alpha\}$
(9) γ is large enough to satisfy $M_\gamma = M_{\gamma+1}$
(10) $M = \mathcal{P}(M_\gamma)$

The operations in (4) are the normal Scott operations rather than those defined in the previous section. Definition (8) is by transfinite recursion in α. The constructed κ-Scott domain is $\langle D_M, \preceq_M\rangle$. For further information on the construction above consult [2].

6 Dedekind cuts

¿From now on, if P is a triplet then $x \in P$ and $S \subseteq P$ are shorthand for $x \in D_P$ and $S \subseteq D_P$, respectively.

Let P be a pcs and let $S \subseteq P$. x is an *upper bound* of S if x is above all elements of S; x is a *lower bound* of S if it is below all elements of S. The *upper cone*, denoted $\vee_P(S)$, is the set of all upper bounds of S. The symbol $\vee$ is used to denote upper cones because it looks like a cone that stands on its sharp end. In $\vee_P(S)$, the index P prevents confusion with "logical or". The *lower cone*, denoted $\wedge_P(S)$, is the set of all lower bounds of S.

Informally, a *Dedekind cut* is a pair $\langle L, U\rangle$ of non-empty subsets of P such that U is the upper cone of L and L is the lower cone of U. It is inconvenient, however, to represent Dedekind cuts by pairs of sets when they can be represented equally well by either their lower or upper cone. In the following, the arbitrary convention is made to represent Dedekind cuts by their upper cones:

Definition 6.1 For all triplets P, U is a Dedekind cut in P if

$U \subseteq D_P$,
$U \neq \emptyset$,
$\wedge_P(U) \neq \emptyset$, and
$U = \vee_P(\wedge_P(U))$.

The pcs $\mathcal{D}(P)$ of Dedekind cuts in P is defined thus:

Definition 6.2 For all triplets P, let $\mathcal{D}(P)$ denote the triplet R for which

D_R is the set of Dedekind cuts in P,
$x \preceq_R y \Leftrightarrow y \subseteq x$
$x \sim_R y \Leftrightarrow \wedge_P(x) \sim^*_P \wedge_P(y)$

Lemma 6.3 *For all (anti-)pcs's P, $\mathcal{D}(P)$ is an (anti-)pcs.*

The following lemma says that if one identifies elements x of D_P with $\vee_P(\{x\})$, then P becomes a subpcs of $\mathcal{D}(P)$:

Lemma 6.4 *If P is an (anti-)pcs and $x, y \in P$ then*
$\vee_P(\{x\}) \in D_{\mathcal{D}(P)}$,
$x \preceq_P y \Leftrightarrow \vee_P(\{x\}) \preceq_{\mathcal{D}(P)} \vee_P(\{y\})$, and
$x \sim_P y \Leftrightarrow \vee_P(\{x\}) \sim_{\mathcal{D}(P)} \vee_P(\{y\})$.

If P and Q are triplets, then ϕ is an *isomorphism* from P to Q if ϕ is a bijective function from D_P onto D_Q which satisfies $x \preceq_P y \Leftrightarrow \phi(x) \preceq_Q \phi(y)$ and $x \sim_P y \Leftrightarrow \phi(x) \sim_Q \phi(y)$. P and Q are *isomorphic*, written $P \cong Q$, if there exists an isomorphism from P to Q. The following lemma expresses that the Dedekind operation is a closure operation:

Lemma 6.5 *$\mathcal{D}(\mathcal{D}(P)) \cong \mathcal{D}(P)$ for all pcs's P.*

7 Commutation rules

Before outlining applications of the Dedekind operation, it will be stated that, under suitable conditions, the Dedekind operation commutes with the standard operations. Non-trivial proofs are in the appendix. Trivial ones are omitted.

Lemma 7.1 $\mathcal{D}(\mathsf{E}) \cong \mathsf{E}$

Lemma 7.2 *$\mathcal{D}(\mathrm{U}(a)) \cong \mathrm{U}(a)$ for all objects a (of ZFC).*

Lemma 7.3 *If P is a pcs and $a \notin P, \mathcal{D}(P)$ then $\mathcal{D}(P_a) \cong \mathcal{D}(P)_a$*

Lemma 7.4 *If P and Q are pcs's and P and Q are disjoint then $\mathcal{D}(P)$ and $\mathcal{D}(Q)$ are disjoint, and $\mathcal{D}(P \oplus Q) \cong \mathcal{D}(P) \oplus \mathcal{D}(Q)$.*

A pcs P is said to be *consistently coherent* if bounded, set-coherent subsets of P have coherent upper bounds. In other words, P is consistently coherent if

$$\vee_P(A){\neq}\emptyset \wedge \vee_P(B){\neq}\emptyset \wedge A{\sim}^*_P B \Rightarrow \exists a{\in}\vee_P(A)\exists b{\in}\vee_P(B){:}\, a{\sim}_P b$$

Lemma 7.5 *For all consistently coherent pcs's P, $\mathcal{D}(\updownarrow P) \cong \updownarrow\mathcal{D}(P)$*

Lemma 7.6 *$\mathcal{D}(P \otimes Q) \cong \mathcal{D}(P) \otimes \mathcal{D}(Q)$ for all anti-pcs's P and pcs's Q.*

Lemma 7.7 *$\mathcal{D}(P \to Q) \cong \mathcal{D}(P) \to \mathcal{D}(Q)$ for all pcs's Q and all consistently coherent pcs's P.*

A pcs P is said to be *κ-consistently coherent* if it is consistently coherent and, furthermore, any κ-small subset A of P that has a lower bound also has a greatest lower bound. The greatest lower bound will be referred to as the *infimum* of A.

Lemma 7.8 *For all κ-consistently coherent pcs's P, $\mathcal{P}_\kappa(P)$ is consistently coherent.*

Lemma 7.9 $\mathcal{D}(\mathcal{P}_\kappa(P)) \cong \mathcal{P}_\kappa(\mathcal{D}(P))$ *for all sets κ and all κ-consistently coherent pcs's P.*

8 Examples of using Dedekind cuts

In continuation of Section 5, consider the following lemmas:

Lemma 8.1 $\mathcal{D}(H_\kappa(P)) \cong H_\kappa(\mathcal{D}(P))$ *if P is a κ-consistently coherent pcs.*

Lemma 8.2 M_γ *is a κ-consistently coherent pcs.*

Lemma 8.3 $H_\kappa(\mathcal{D}(M_\gamma)) \cong \mathcal{D}(M_\gamma)$

Section 5 stated that

$$\mathcal{P}(M_\gamma)$$

was a solution to the domain equation

$$M = (M \to M)_\perp + \{T\}$$

Lemma 8.3 above has the consequence that

$$\mathcal{P}(\mathcal{D}(M_\gamma))$$

is another solution to that equation; it is closed under compact suprema and infima as mentioned in Section 1.

9 Intended application

While the Dedekind operation has properties that look interesting in themselves, it was developed with a particular application in mind, namely to simplify the use of λ-calculus as a foundation of mathematics.

In the following, let $\perp = (\lambda x.xx)(\lambda x.xx)$. To use λ-calculus as a foundation of classical mathematics,

(i) one must be able to represent truth and falsehood,

(ii) one must have a selection function $\mathrm{if}(x, y, z)$ whose value is y or z or $\perp$ depending on x, and

(iii) one must have a quantifier.

The second requirement rules out pure λ-calculus since the range of any function in pure λ-calculus has either one or infinitely many elements.

An obvious choice then is to include two Ur-elements T and F in λ-calculus to stand for truth and falsehood. A slight simplification can be obtained, however, by using the trick to let e.g. $\lambda x.x$ stand for falsehood so that one merely has to include T. A model of such a λ-calculus must satisfy the domain equation (4) considered in Section 5 and 8. In the following, suppose M satisfies the given domain equation.

To satisfy the third requirement, one has to find a suitable subset Φ of M which the quantifier should quantify over. To model classical mathematics, Φ must have complexity like the universe of ZFC. The choice of Φ made in [2] and [7] is the following: Let σ be an inaccessible ordinal. Let $\kappa = \mathcal{P}(\sigma)$. Let M be a κ-Scott domain that satisfies (4). Let xy denote x applied to y, and let it be defined on M as in [2]. For all subsets G of M define

$$G^\circ = \{x{\in}M | \forall y_1, y_2, \ldots {\in} G \exists n {\in} \mathbb{N}\colon x y_1 y_2 \cdots y_n = T\}$$

Then let Φ be the least subset of M for which $G^{\circ\circ} \subseteq \Phi$ for all κ-small subsets G of Φ.

The resulting theory has consistency power like ZFC and is suited as a foundation of mathematics, but the axiomatization [7] is rather complicated and non-intuitive.

This is where the Dedekind operation comes in. If one uses $M = \mathcal{P}(\mathcal{D}(M_\gamma))$ as the domain for the theory, then one can extend Φ in such a way that every chain in Φ has an infimum in Φ; it is the Dedekind operation that guarantees the existence of such infima. This gives rise to a theory with much nicer properties and a much simpler axiomatization. The simplified axiomatization is presented in [8]. The model construction has about the same size as [2] so, unfortunately, a detailed example of use of the Dedekind operation does not fit the present paper.

10 Conclusion

The Dedekind operation has been introduced and a number of lemmas have been stated. The general nature of the operation has been investigated and one application has been sketched.

A Proofs

This appendix contains proofs; it also contains some auxiliary definitions and lemmas. Trivial proofs are omitted.

Let S be a subset of a triplet P. S is said to be *upward bounded* (w.r.t P) if S has an upper bound. In other words, S is upward bounded if the upper cone of S is non-empty. Likewise, S is *downward bounded* if the lower cone of S is non-empty.

Lemma A.1 *If P is a triplet and $S, T \subseteq P$ then*

- *If $S \subseteq T$ then $\vee_P(T) \subseteq \vee_P(S)$.*
- *If $S \subseteq T$ then $\wedge_P(T) \subseteq \wedge_P(S)$.*

- *If S is non-empty and upward bounded and if $T = \vee_P(S)$ then T is a Dedekind cut. Furthermore, T is the least element of $\mathcal{D}(P)$ (least w.r.t. $\preceq_{\mathcal{D}(P)}$, largest w.r.t. $\subseteq$) whose lower cone contains S.*
- *If S is non-empty and downward bounded and if $T = \vee_P(\wedge_P(S))$, then T is a Dedekind cut. Furthermore, T is the greatest element of $\mathcal{D}(P)$ which contains S.*

Lemma A.2 *If P is consistently coherent then for all $x, y \in \mathcal{D}(P)$ we have $x \sim_{\mathcal{D}(P)} y \Leftrightarrow \exists \overline{x}{\in}x \exists \overline{y}{\in}y{:}\, \overline{x} \sim_P \overline{y}$.*

Proof of 7.5 We shall construct an isomorphism ϕ from $\updownarrow\mathcal{D}(P)$ to $\mathcal{D}(\updownarrow P)$. Elements of $\updownarrow\mathcal{D}(P)$ are upper cones of Dedekind cuts in P. Elements of $\mathcal{D}(\updownarrow P)$ are lower cones of Dedekind cuts in P. For all $x \in \updownarrow\mathcal{D}(P)$ define

$$\phi(x) = \wedge_P(x)$$

To see that ϕ is an isomorphism, proceed thus: Clearly, ϕ is injective and the range of ϕ is $\mathcal{D}(\updownarrow P)$. If $x, y \in \updownarrow\mathcal{D}(P)$ then

$$\begin{aligned}
\phi(x) \preceq_{\mathcal{D}(\updownarrow P)} \phi(y) &\Leftrightarrow \wedge_P(x) \preceq_{\mathcal{D}(\updownarrow P)} \wedge_P(y)\\
&\Leftrightarrow \wedge_P(y) \subseteq \wedge_P(x)\\
&\Leftrightarrow x \subseteq y\\
&\Leftrightarrow y \preceq_{\mathcal{D}(P)} x\\
&\Leftrightarrow x \preceq_{\updownarrow\mathcal{D}(P)} y\\
\phi(x) \sim_{\mathcal{D}(\updownarrow P)} \phi(y) &\Leftrightarrow \wedge_P(x) \sim_{\mathcal{D}(\updownarrow P)} \wedge_P(y)\\
&\Leftrightarrow \forall \overline{x}{\in}\wedge_{\updownarrow P}(\wedge_P(x)) \forall \overline{y}{\in}\wedge_{\updownarrow P}(\wedge_P(y)){:}\, \overline{x} \sim_{\updownarrow P} \overline{y}\\
&\Leftrightarrow \forall \overline{x}{\in}\vee_P(\wedge_P(x)) \forall \overline{y}{\in}\vee_P(\wedge_P(y)){:}\, \overline{x} \sim_{\updownarrow P} \overline{y}\\
&\Leftrightarrow \forall \overline{x}{\in}x \forall \overline{y}{\in}y{:}\, \neg(\overline{x} \sim_P \overline{y})\\
&\Leftrightarrow \neg\exists \overline{x}{\in}x \exists \overline{y}{\in}y{:}\, \overline{x} \sim_P \overline{y}\\
&\Leftrightarrow \neg(x \sim_{\mathcal{D}(P)} y)\\
&\Leftrightarrow x \sim_{\updownarrow\mathcal{D}(P)} y
\end{aligned}$$

□

Definition A.3 For all sets R define

$$\begin{aligned}
\mathrm{dom}(R) &= \{x | \exists y{:}\, \langle x, y\rangle \in R\}\\
\mathrm{rng}(R) &= \{y | \exists x{:}\, \langle x, y\rangle \in R\}
\end{aligned}$$

Lemma A.4 *If P and Q are triplets and if $R \subseteq P \otimes Q$ then*

$$\begin{aligned}
\wedge_{P\otimes Q}(R) &= \wedge_P(\mathrm{dom}(R) \times \wedge_Q(\mathrm{rng}(R))\\
\vee_{P\otimes Q}(R) &= \vee_P(\mathrm{dom}(R) \times \vee_Q(\mathrm{rng}(R))
\end{aligned}$$

Lemma A.5 *If P and Q are triplets and if $R \in \mathcal{D}(P \otimes Q)$ then*

$$\begin{aligned}
&R = \mathrm{dom}(R) \times \mathrm{rng}(R)\\
&\mathrm{dom}(R) \in \mathcal{D}(P)\\
&\mathrm{rng}(R) \in \mathcal{D}(Q)
\end{aligned}$$

Proof of 7.6 For all $R \in \mathcal{D}(P \otimes Q)$ define

$$\phi(R) = \langle \mathrm{dom}(R), \mathrm{rng}(R) \rangle.$$

For all $R \in \mathcal{D}(P \otimes Q)$ we have $\phi(R) \in \mathcal{D}(P) \otimes \mathcal{D}(Q)$, and ϕ is clearly both injective and surjective. It is trivial to prove

$$\phi(R_1) \preceq_{\mathcal{D}(P)\otimes\mathcal{D}(Q)} \phi(R_2) \Leftrightarrow R_1 \preceq_{\mathcal{D}(P\otimes Q)} R_2.$$

Finally, let $R_1 = P_1 \times Q_1 \in \mathcal{D}(P \otimes Q)$ and $R_2 = P_2 \times Q_2 \in \mathcal{D}(P \otimes Q)$. We have

$$\begin{aligned}
& R_1 \sim_{\mathcal{D}(P\otimes Q)} R_2 \\
\Leftrightarrow\; & P_1 \times Q_1 \sim_{\mathcal{D}(P\otimes Q)} P_2 \times Q_2 \\
\Leftrightarrow\; & \forall \langle p_1, q_1 \rangle {\in} \wedge_{P\otimes Q}(P_1 \times Q_1) \forall \langle p_2, q_2 \rangle {\in} \wedge_{P\otimes Q}(P_2 \times Q_2)\colon \\
& \langle p_1, q_1 \rangle \sim_{P\otimes Q} \langle p_2, q_2 \rangle \\
\Leftrightarrow\; & \forall p_1 {\in} \wedge_P(P_1) \forall q_1 {\in} \wedge_Q(Q_1) \forall p_2 {\in} \wedge_P(P_2) \forall q_2 {\in} \wedge_Q(Q_2)\colon \\
& p_1 \sim_P p_2 \wedge q_1 \sim_Q q_2 \\
\Leftrightarrow\; & (\forall p_1 {\in} \wedge_P(P_1) \forall p_2 {\in} \wedge_P(P_2)\colon p_1 \sim_P p_2) \vee \\
& (\forall q_1 {\in} \wedge_Q(Q_1) \forall q_2 {\in} \wedge_Q(Q_2)\colon q_1 \sim_Q q_2) \\
\Leftrightarrow\; & P_1 \sim_{\mathcal{D}(P)} P_2 \vee Q_1 \sim_{\mathcal{D}(Q)} Q_2 \\
\Leftrightarrow\; & \langle P_1, Q_1 \rangle \sim_{\mathcal{D}(P)\otimes\mathcal{D}(Q)} \langle P_2, Q_2 \rangle \\
\Leftrightarrow\; & \phi(R_1) \sim_{\mathcal{D}(P)\otimes\mathcal{D}(Q)} \phi(R_2)
\end{aligned}$$

□

Lemma A.6 *If P is a κ-consistently coherent pcs and if A is a κ-small self-coherent subset of $\mathcal{D}(P)$, then there exists a κ-small self-coherent subset $\overline{A}$ of P that intersects all elements of A.*

Proof of A.6 Let $f\colon A \times A \to P$ satisfy

$$\forall \overline{x} {\in} \wedge_P(x)\colon \overline{x} \preceq_P f(x,y) \text{ and}$$
$$f(x,y) \sim_P f(y,x)$$

for all $x, y \in A$. f exists because P is consistently coherent. We have $f(x,y) \in x$. For all $x \in A$ define

$$S(x) = \{f(x,y) | y \in A\}.$$

$S(x)$ is a subset of x and x is a Dedekind cut, so any element of the lower cone of x is a lower bound of $S(x)$. Furthermore, $S(x)$ is κ-small. Now let $h(x)$ be the infimum of $S(x)$. $h(x)$ is greater than any element in the lower cone of x so $h(x) \in x$. Furthermore, if $x, y \in A$ then $h(x) \preceq_P f(x,y)$ and $h(y) \preceq_P f(y,x)$ which gives $h(x) \sim_P h(y)$. Hence, $\overline{A} = \{h(x) | x \in A\}$ satisfies the lemma. □

Proof of 7.8 In this proof, $\bigcup$ denotes the union set operator of ZFC.

Assume A and B are bounded, set-coherent subsets of $\mathcal{P}_\kappa(P)$. Let a and b be upper bounds of A and B, respectively. For all $x \in a$ define $\overline{a}(x) = \{y \in \bigcup A | y \preceq_P x\}$. Define $\overline{b}(x)$ similarly. Let $\tilde{A} = \{\overline{a}(x) | x \in a\} \setminus \{\emptyset\}$ and $\tilde{B} = \{\overline{b}(x) | x \in b\} \setminus \{\emptyset\}$. We have

$$\bigcup A = \bigcup \tilde{A}$$
$$\bigcup B = \bigcup \tilde{B}$$

The sets A and B are bounded, set-coherent sets of κ-small, bounded, set-coherent subsets of P whereas $\tilde{A}$ and $\tilde{B}$ are κ-small, bounded, set-coherent sets of bounded, set-coherent, non-empty subsets of P.

Let $f(x, y)$ satisfy

$$\forall \overline{x} \in x \colon \overline{x} \preceq_P f(x, y) \text{ and}$$
$$f(x, y) \sim_P f(y, x)$$

for all $x, y \in \tilde{A} \cup \tilde{B}$. f exists because P is consistently coherent. For all $x \in \tilde{A} \cup \tilde{B}$ define

$$S(x) = \{f(x, \overline{y}) | \overline{y} \in \tilde{A} \cup \tilde{B}\}$$

$S(x)$ is κ-small. All elements of $S(x)$ are greater than all elements of the non-empty set x , so $S(x)$ has an infimum $h(x)$ which is greater than all elements of x. Furthermore, $S(x) \sim_P S(y)$ for all $x, y \in \tilde{A} \cup \tilde{B}$. Now define

$$\overline{A} = \{h(x) | x \in \tilde{A}\}$$
$$\overline{B} = \{h(x) | x \in \tilde{B}\}$$

$\overline{A}$ and $\overline{B}$ are κ-small, self-coherent subsets of P so $\overline{A}, \overline{B} \in \mathcal{P}_\kappa(P)$. Furthermore, $\overline{A} \sim_{\mathcal{P}_\kappa(P)} \overline{B}$. Finally,

$$\forall \tilde{a} \in \bigcup \tilde{A} \exists \overline{a} \in \overline{A} \colon \tilde{a} \preceq_P \overline{a}$$
$$\forall \tilde{b} \in \bigcup \tilde{B} \exists \overline{b} \in \overline{B} \colon \tilde{b} \preceq_P \overline{b}$$

so $\overline{A}$ and $\overline{B}$ are bounds of A and B, respectively. The existence of $\overline{A}$ and $\overline{B}$ proves that $\mathcal{P}_\kappa(P)$ is consistently coherent.

□

Proof of 7.9 In this proof, $\bigcup$ denotes the union set operator of ZFC.

Let choice(x) have the property

$$\text{choice}(x) \in x$$

for all non-empty sets x. For all $x \in \mathcal{D}(\mathcal{P}_\kappa(P))$ and all $p \in P$ define

$$c(x, p) = \{y \in \bigcup \wedge_{\mathcal{P}_\kappa(P)}(x) | y \preceq_P p\}$$

Furthermore define

$$L(x) = \{c(x, p) | p \in \text{choice}(x)\} \setminus \{\emptyset\}$$

For all $x \in \mathcal{D}(\mathcal{P}_\kappa(P))$ we have

$$\bigcup \wedge_{\mathcal{P}_\kappa(P)}(x) = \bigcup L(x)$$

$\wedge_{\mathcal{P}_\kappa(P)}(x)$ is a bounded set of self-coherent, κ-small subsets of P whereas $L(x)$ is a self-coherent, κ-small set of bounded, non-empty subsets of P. Now define

$$\phi(x) = \{\vee_P(y) | y \in L(x)\}.$$

We have $\phi(x) \in \mathcal{P}_\kappa(\mathcal{D}(P))$.

We now prove that ϕ is injective. If $\phi(x) = \phi(\overline{x})$ then $L(x) = L(\overline{x})$ so

$$\bigcup \wedge_{\mathcal{P}_\kappa(P)}(x) = \bigcup \wedge_{\mathcal{P}_\kappa(P)}(x)$$

which entails

$$x = \vee_{\mathcal{P}_\kappa(P)}\{\bigcup \wedge_{\mathcal{P}_\kappa(P)}(x)\} = \vee_{\mathcal{P}_\kappa(P)}\{\bigcup \wedge_{\mathcal{P}_\kappa(P)}(\overline{x})\} = \overline{x}$$

Hence, ϕ is injective.

To prove that ϕ is surjective, assume $y \in \mathcal{P}_\kappa(\mathcal{D}(P))$, let

$$\overline{L} = \{\wedge_P(\overline{y}) | \overline{y} \in y\}$$
$$x = \vee_{\mathcal{P}_\kappa(P)}\{\bigcup \overline{L}\}$$

and let $\overline{A}$ be a κ-small set of pairwise coherent elements of P that intersects all elements of y (such an $\overline{A}$ exists according to Lemma A.6). We have that x is non-empty since it contains $\overline{A}$ and that the lower cone of $\overline{A}$ is non-empty since it contains the empty set, so x is a Dedekind cut. Furthermore, $y = \phi(x)$ so ϕ is surjective.

Now let $x, \overline{x} \in \mathcal{D}(\mathcal{P}_\kappa(P))$. We have

$$\begin{aligned}
x \preceq_{\mathcal{D}(\mathcal{P}_\kappa(P))} \overline{x} &\Leftrightarrow \forall p \in P \colon c(x,p) \subseteq c(\overline{x},p) \\
&\Leftrightarrow \forall z \in L(x) \exists \overline{z} \in L(\overline{x}) \colon z \subseteq \overline{z} \\
&\Leftrightarrow \forall z \in \phi(x) \exists \overline{z} \in \phi(\overline{x}) \colon z \preceq_{\mathcal{D}(P)} \overline{z} \\
&\Leftrightarrow \phi(x) \preceq_{\mathcal{P}_\kappa(\mathcal{D}(P))} \phi(\overline{x}) \\
x \sim_{\mathcal{D}(\mathcal{P}_\kappa(P))} \overline{x} &\Leftrightarrow \forall y \in \wedge_{\mathcal{P}_\kappa(P)}(x) \forall \overline{y} \in \wedge_{\mathcal{P}_\kappa(P)}(\overline{x}) \colon y \sim_{\mathcal{P}_\kappa(P)} \overline{y} \\
&\Leftrightarrow \forall y \in \bigcup \wedge_{\mathcal{P}_\kappa(P)}(x) \forall \overline{y} \in \bigcup \wedge_{\mathcal{P}_\kappa(P)}(\overline{x}) \colon y \sim_P \overline{y} \\
&\Leftrightarrow \forall y \in \bigcup L(x) \forall \overline{y} \in \bigcup L(\overline{x}) \colon y \sim_P \overline{y} \\
&\Leftrightarrow \forall y \in L(x) \forall z \in y \forall \overline{y} \in L(\overline{x}) \forall \overline{z} \in \overline{y} \colon z \sim_P \overline{z} \\
&\Leftrightarrow \{\vee_P(y) | y \in L(x)\} \sim_{\mathcal{P}_\kappa(P)} \{\vee_P(y) | y \in L(\overline{x})\} \\
&\Leftrightarrow \phi(x) \sim_{\mathcal{P}_\kappa(\mathcal{D}(P))} \phi(\overline{x})
\end{aligned}$$

□

References

[1] Barendregt, H., "The Lambda Calculus, Its Syntax and Semantics," Studies in Logic and The Foundation of Mathematics **103**, North-Holland, 1984.

[2] Berline, C. and K. Grue, *A κ-denotational semantics for Map Theory in ZFC+SI*, Theoretical Computer Science **179** (1997), pp. 137–202.

[3] Flagg, R., *K-continuous lattices and comprehension principles for Frege structures*, Annals of Pure and Applied Logic **36** (1987), pp. 1–16.

[4] Flagg, R. and J. Myhill, *A type-free system extending ZFC*, Annals of Pure and Applied Logic **43** (1989), pp. 79–97.

[5] Gianantonio, P. D., F. Honsell and G. Plotkin, *Uncountable limits and the lambda-calculus*, Nordic Journal of Computing **2** (1995), pp. 126–145.

[6] Girard, J., *The system F of variable types, fifteen years later*, Theoretical Computer Science **45** (1986), pp. 159–192.

[7] Grue, K., *Map theory*, Theoretical Computer Science **102** (1992), pp. 1–133.

[8] Grue, K., *λ-calculus as a foundation for mathematics*, in: C. A. Anderson and M. Zelëny, editors, *Logic, Meaning and Computation*, Synthese Library **305**, Kluwer Academic Publishers, 2001 .

[9] Krivine, J., "Lambda-calcul: types et Modèles," Masson, Paris, 1990.

[10] Krivine, J., "Lambda-calculus, types and models," Ellis & Horwood, 1993.

[11] Larsen, K. and G. Winskel, *Using information systems to solve recursive domain equations*, in: *Lecture Notes in Computer Science*, Lecture Notes in Computer Science **173 (Semantics of data types)**, Springer-Verlag, 1984 pp. 109–130.

[12] Scott, D., *Continuous lattices*, in: F. Lawvere, editor, *Lecture Notes in Mathematics*, Lecture Notes in Mathematics **274, Toposes, algebraic geometry and logic**, Proceedings of Dalhousie Conference (1972), pp. 97–136.