

Notes on dI domains

Semantics and Domain Theory course

Herman Geuvers

May 20, 2014

1 Stable models

These notes are derived from [4, 2, 3]. For further reading on domain theory in general, see [1].

Definition 1.1 *Let (D, \sqsubseteq) be a partial order. A set $X \subseteq D$ is directed if it is non-empty and for every $x, y \in X$, there is a $z \in X$ such that $x \sqsubseteq z$ and $y \sqsubseteq z$. The p.o. is called directed complete, or a dcpo, if every directed subset X has a lub, $\bigsqcup X$. The p.o. is called pointed if it has a least element, \perp .*

Remark 1.2 *The definition of cpo of Winskell [5] uses so called chain completeness instead of directed completeness. The cpos of [5] are also called chain-complete partial orders, ccpos. Directed completeness implies chain completeness: every chain is also a directed set. The other way around: chain completeness also implies directed completeness, but the proof is hard and requires the axiom of choice. So dcpos and ccpos are the same, but in this note we use the definition of dcpo.*

From now on, (D, \sqsubseteq) will denote a dcpo (Definition 1.1).

Definition 1.3 *An element $d \in D$ is isolated (also called finite or compact) if for every directed $X \subseteq D$, if $d \sqsubseteq \bigsqcup X$, then there is an $x \in X$ such that $d \sqsubseteq x$. The collection of isolated elements of D is \mathbf{B}_D , so*

$$\mathbf{B}_D := \{d \in D \mid d \text{ is isolated}\}$$

If d is isolated and $d = \bigsqcup_{i \in \mathbb{N}} d_i$, then $d = d_i$ for some $i \in \mathbb{N}$. So, for isolated elements we can not give an infinitary approximation without using the element itself.

Definition 1.4 *A dcpo (D, \sqsubseteq) is algebraic if each element is the lub of its isolated (finite) approximations, more precisely if for each $d \in D$,*

- $M_d := \{x \in \mathbf{B}_D \mid x \sqsubseteq d\}$ is directed and

- $d = \bigsqcup M_d$.

In various places in the literature, a domain is identified with an algebraic dcpo, and not with a cpo with bottom element, as [5] does. The idea is that the isolated (or finite, or compact) elements form a “basis” for the dcpo: every element is the lub of the isolated (finite) elements below it.

As examples, the flat domain \mathbb{N}_\perp and the dcpo $\mathbb{N}_\perp \rightarrow \mathbb{N}_\perp$ of Scott-continuous functions from \mathbb{N}_\perp to \mathbb{N}_\perp (ordered pointwise) are algebraic dcpos.

Definition 1.5 *A subset X of D is bounded if there is a $d \in D$ such that $\forall x \in X (x \sqsubseteq d)$. (Note the difference with directedness; the d need not be an element of X)*

The dcpo (D, \sqsubseteq) is bounded complete if each bounded subset X has a lub $\bigsqcup X$.

The idea of a set X being bounded is that it gives *consistent* information: If X is bounded, then there is a way to “unify all information in X ” into a whole, namely the d that is an upperbound of X .

Remark 1.6 *Many (most? all?) of the examples of dcpos we have seen are also bounded complete, but there are examples of dcpos that are not bounded complete.*

Lemma 1.7 *A bounded complete dcpo is pointed.*

Proof The empty set is bounded, so $\bigsqcup \emptyset$ exists. From the definition of \bigsqcup it follows immediately that $\forall x \in D, \bigsqcup \emptyset \sqsubseteq x$, so $\bigsqcup \emptyset = \perp$.

Notation 1.8 *If the set $\{x, y\}$ is bounded we write $x \uparrow y$ and we denote $\bigsqcup \{x, y\}$ by $x \sqcup y$. This is also called the join of x and y .*

Definition 1.9 *Given elements $x, y, z \in D$, the greatest lower bound (glb) of x and y , also called the meet of x and y , denoted by $x \sqcap y$, is the element satisfying*

- $x \sqcap y \sqsubseteq x$ and $x \sqcap y \sqsubseteq y$,
- for all z , if $z \sqsubseteq x$ and $z \sqsubseteq y$, then $z \sqsubseteq x \sqcap y$.

In a dcpo, not every pair of elements needs to have a glb, but in case the glb exists, it is unique, as can easily be shown.

Lemma 1.10 *In a bounded complete dcpo, every pair of elements x, y has a greatest lower bound, $x \sqcap y$.*

Proof Define $z := \bigsqcup \{d \in D \mid d \sqsubseteq x \wedge d \sqsubseteq y\}$ and prove that this lub exists and that z is the glb of x and y .

Definition 1.11 *An element $d \in D$ is very finite if the set $\{x \in D \mid x \sqsubseteq d\}$ is finite.*

In flat domains like \mathbb{N}_\perp and \mathbb{B}_\perp all elements are very finite. In $\mathbb{N}_\perp \rightarrow \mathbb{N}_\perp$ there are very finite elements, but also ones that are not. For example the identity function is not very finite, but the following function f is

$$f(x) := \begin{cases} \perp & \text{if } x = \perp \\ \perp & \text{if } x \in \mathbb{N}, x > 10 \\ x & \text{if } x \in \mathbb{N}, x \leq 10 \end{cases}$$

In general, a function $f : \mathbb{N}_\perp \rightarrow \mathbb{N}_\perp$ is very finite if and only if its *domain* is finite, that is: $\{n \mid f(n) \neq \perp\}$ is finite.

Definition 1.12 *A dI-domain is an algebraic bounded-complete dcpo which satisfies*

axiom d For every $x, y, z \in D$, if $y \uparrow z$, then $x \sqcap (y \sqcup z) = (x \sqcap y) \sqcup (x \sqcap z)$.

axiom I Every isolated point is very finite

Note that the right hand side of (axiom d) in the definition makes sense: the join of $x \sqcap y$ and $x \sqcap z$ exists because x is a bound of $\{x \sqcap y, x \sqcap z\}$.

The importance of dI-domains lies in the fact that a different function space can be defined on them (that is: different from the Scott-continuous functions ordered point-wise), that excludes all kinds of inherently parallel functions like parallel or (por). These dI-domains were introduced by Berry [3] to analyse the notion of sequentiality and to give a semantics of PCF that only includes sequential functions.

Definition 1.13 *Let D and E be dI-domains. The function $f : D \rightarrow E$ is stable if it is continuous and it satisfies*

$$\forall x, y \in D (x \uparrow y \rightarrow f(x \sqcap y) = f(x) \sqcap f(y)).$$

The function space $[D \rightarrow_s E]$ consists of the stable functions from D to E ordered by the stable function ordering: $f \sqsubseteq_s g$ if

- $\forall x \in D (f(x) \sqsubseteq g(x))$,
- $\forall x, y \in D (x \sqsubseteq y \rightarrow f(x) = f(y) \sqcap g(x))$

The real idea behind stability may not be clear from its definition. The intuition is that in case $f(x) = z$, there is a fixed *least amount of information from x needed*, some $x' \sqsubseteq x$, such that $f(x') = z$. In the definition of stability we can see a way that this is being expressed, because if $f(x) = z$ and $x', x'' \sqsubseteq x$ with $f(x') = f(x'') = z$, then also $f(x' \sqcap x'') = z$.

To make the intuition more precise we give (without proof) a proposition. In fact this gives the original definition of stability from [3]. Definition 1.13 was proved by Berry as an equivalent characterization, but nowadays it is used as the definition and we follow that habit.

Proposition 1.14 *Let D and E be dI-domains. The continuous function $f : D \rightarrow E$ is stable if and only if*

$$\begin{aligned} \forall x \in D, \forall y \sqsubseteq f(x), \exists M \sqsubseteq x (& y \sqsubseteq f(M) \\ \wedge \quad & \forall z \sqsubseteq x (y \sqsubseteq f(z) \rightarrow M \sqsubseteq z)) \end{aligned}$$

The M in the proposition is often given parameters: $M(f, x, y)$ to express its dependency on f, x, y . It can be shown that this M is unique. If we take $y := f(x)$, then M represents the least amount of information needed from x to compute y .

Constructions on domains that we know from [5] can also be carried out on dI-domains. For example the product of two dI-domains is a dI-domain and the functions space (of stable functions, ordered by the stable ordering of Definition 1.13) between dI-domains is a dI-domain. Especially, “eval” and “curry” are stable functions. In categorical terminology: the dI-domains form a cartesian closed category.

Lemma 1.15 *The parallel or function $\text{por} : \mathbb{B}_\perp \times \mathbb{B}_\perp \rightarrow \mathbb{B}_\perp$ is not stable.*

Proof Remember that $\text{por}(\perp, \text{tt}) = \text{por}(\text{tt}, \perp) = \text{tt}$ and $\text{por}(\perp, \perp) = \perp$. $\text{por}(\text{ff}, \text{ff}) = \text{ff}$. If por would be stable, we would have $\text{tt} = \text{por}(\perp, \text{tt}) \sqcap \text{por}(\text{tt}, \perp) = \text{por}((\perp, \text{tt}) \sqcap (\text{tt}, \perp)) = \text{por}(\perp, \perp) = \perp$, so por is not stable.

The semantics of PCF can also be given in terms of dI-domains. All terms of PCF get interpreted as stable functions, as can be checked by induction on the derivation of the term. As a consequence we have the following, which immediately follows from Lemma 1.15.

Proposition 1.16 *The parallel or function is not definable*

2 Exercises

1. Suppose we are in a dcpo where each pair of elements has a glb. Show that $\forall x, y (x \sqsubseteq y \Leftrightarrow x = x \sqcap y)$.
2. We are in a bounded complete p.o. and we consider the property $(*)$ (used in stability)

$$\forall x, y \in D (x \uparrow y \rightarrow f(x \sqcap y) = f(x) \sqcap f(y)) \quad (*)$$

Show that, if f satisfies $(*)$, then it is monotone.

3. (a) Define all possible different “AND” functions as monotone functions (in $\mathbb{B}_\perp \times \mathbb{B}_\perp \rightarrow \mathbb{B}_\perp$).
- (b) Show that 3 of your functions can be defined in PCF.
- (c) Now show that one of your functions cannot be defined in PCF

- i. by semantic means (using dI-domains).
 - ii. by using the non-definability of `por`
4. Prove that the identity function $I : \mathbb{N}_\perp \rightarrow \mathbb{N}_\perp$ is not very finite.
 5. Prove that, in a bounded complete dcpo, every non-empty set X has a greatest lower bound, $\bigsqcap X$.

References

- [1] Samson Abramsky and Achim Jung. Domain theory. In *Handbook of Logic in Computer Science*, pages 1–168. Clarendon Press, 1994.
- [2] G. Berry and Pierre-Louis Curien. Sequential algorithms on concrete data structures. *Theor. Comput. Sci.*, 20:265–321, 1982.
- [3] Gérard Berry. Stable models of typed lambda-calculi. In Giorgio Ausiello and Corrado Böhm, editors, *ICALP*, volume 62 of *Lecture Notes in Computer Science*, pages 72–89. Springer, 1978.
- [4] Thierry Coquand, Carl A. Gunter, and Glynn Winskel. Domain theoretic models of polymorphism. *Inf. Comput.*, 81(2):123–167, 1989.
- [5] Glynn Winskel and Andy Pitts. Lecture notes on denotational semantics, <http://www.cl.cam.ac.uk/~gw104/dens.pdf>, 2005.