Notes on dI domains
Semantics and Domain Theory course

Herman Geuvers

November 25, 2019

1 Stable models

These notes are derived from [5, 3, 4]. For further reading on domain theory in general, see [2].

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* $y \sqsubseteq z$*.*

$(D, \sqsubseteq)$ *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,* $\bot$*.*

Remark 1.2 *Just like in the case for lubs of chains, the lub of a set is characterized by two properties:*

(lub1) $\forall d \in X (d \sqsubseteq \bigsqcup X)$,
$\bigsqcup X$ *is an upperbound to all the elements in* X*.*

(lub2) If $\forall d \in X (d \sqsubseteq e)$*, then* $\bigsqcup X \sqsubseteq e$,
$\bigsqcup X$ *is the least of all upperbound of* X*.*

Remark 1.3 *The cpos of [1] 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 1.4 *An element* $d \in D$ *is* compact *(also called* finite *or* isolated*) 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 compact elements of* D *is* $\mathbf{B}_D$*, so*

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

If d is compact and $d = \bigsqcup_{i\in\mathbb{N}} d_i$, then $d = d_i$ for some $i \in \mathbb{N}$. So, in an infinitary approximation of a compact element we have to use the element itself.

Definition 1.5 *A dcpo* $(D, \sqsubseteq)$ *is* algebraic *if each element is the lub of its compact 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 [1] does. The idea is that the compact (or finite, or isolated) elements form a "basis" for the dcpo: every element is the lub of the compact (finite) elements below it. As examples, the flat domain $\mathbb{N}_\perp$ and the dcpo $\mathbb{N}_\perp \to \mathbb{N}_\perp$ of Scott-continuous functions from $\mathbb{N}_\perp$ to $\mathbb{N}_\perp$ (ordered pointwise) are algebraic dcpos.

Definition 1.6 *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 1.7 *Many of the examples of dcpos we have seen are also bounded complete, but there are examples of dcpos that are not bounded complete.*

Lemma 1.8 *A bounded complete dcpo is pointed (i.e. contains* $\perp$*).*

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 $\bigsqcup \emptyset = \perp$

Definition 1.9 *Given elements* $x, y \in D$, *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* $z \sqsubseteq x \sqcap y$.

Similarly, the least upper bound *(lub) of* x *and* y, *also called the* join of x and y, *denoted by* $x \sqcup y$, *is the element satisfying*

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

In a partial order, not every pair of elements needs to have a glb (or a lub), but in case the glb (or lub) exists, it is unique, as can easily be shown.

Notation 1.10 *If the set* $\{x, y\}$ *is bounded we write* $x \uparrow y$*. Note that, in a bounded dcpo, if* $x \uparrow y$*, then* $\bigsqcup\{x, y\}$ $(x \sqcup y)$ *exists.*

Lemma 1.11 *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 1.12 *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 $\mathbb{N}_\perp \to \mathbb{N}_\perp$ 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

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

A function $f : \mathbb{N}_\perp \to \mathbb{N}_\perp$ is very finite if and only if its *domain* is finite, that is: $\{n \mid f(n) \neq \perp\}$ is finite. So in $\mathbb{N}_\perp \to \mathbb{N}_\perp$, elements are compact if and only if they are very finite.

Definition 1.13 *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 compact element 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 now a different function space can be defined (i.e. different from the Scott-continuous functions ordered point-wise), that excludes inherently parallel functions like parallel or. These dI-domains were introduced by Berry [4] to analyze the notion of sequentiality and to give a semantics for PCF that only includes sequential functions.

Definition 1.14 *Let* D *and* E *be dI-domains. The function* $f : D \to E$ *is* stable *if it is continuous and it satisfies*

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

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

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

The real idea behind stability may not be clear from its definition. The intuition is that if $f(x) = z$, there is a fixed *least amount of information from x needed to compute z*. So there is a least $x' \sqsubseteq x$, 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 $f(x' \sqcap x'') = z$. (If both $f(x') = z$ and $f(x'') = z$ then the information shared by x' and x'' is enough to compute z.)

To make the intuition more precise we give (without proof) a proposition. In fact this gives the original definition of stability from [4]. Definition 1.14 was proved by Berry as an equivalent characterization, but nowadays it is used as the definition and we follow that habit.

Proposition 1.15 *Let D and E be dI-domains. The continuous function $f : D \to E$ is stable if and only if*

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

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 [1] 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 1.14) 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 1.16 *The parallel or function* $\mathsf{por} : \mathbb{B}_\perp \times \mathbb{B}_\perp \to \mathbb{B}_\perp$ *is not stable.*

Proof Remember that $\mathsf{por}(\perp, \mathsf{tt}) = \mathsf{por}(\mathsf{tt}, \perp) = \mathsf{tt}$ and $\mathsf{por}(\perp, \perp) = \perp$. $\mathsf{por}(\mathsf{ff}, \mathsf{ff}) = \mathsf{ff}$. If por would be stable, we would have
$\mathsf{tt} = \mathsf{por}(\perp, \mathsf{tt}) \sqcap \mathsf{por}(\mathsf{tt}, \perp) = \mathsf{por}((\perp, \mathsf{tt}) \sqcap (\mathsf{tt}, \perp)) = \mathsf{por}(\perp, \perp) = \perp$,
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 1.16.

Proposition 1.17 *The parallel or function is not definable in PCF.*

2 Exercises

1. Suppose that a monotonic function $p : (\mathbb{B}_\perp \times \mathbb{B}_\perp) \to \mathbb{B}_\perp$ satisfies

   - $p(\mathsf{tt}, \perp) = \mathsf{tt}$,
   - $p(\perp, \mathsf{tt}) = \mathsf{tt}$,
   - $p(\mathsf{ff}, \mathsf{ff}) = \mathsf{ff}$.

   Show that p coincides with the parallel-or function (on Slide 72 of the notes of Pitts) in the sense that $p(d_1, d_2) = \mathrm{por}(d_1)(d_2)$, for all $d_1, d_2 \in \mathbb{B}_\perp$.

2. Show that the evaluation relation for PCF+por (Slide 77, where rules for $\mathbf{por}(M_1, M_2) \Downarrow V$ have been added to PCF) is still deterministic: If $M \Downarrow V$, then for all V', if $M \Downarrow V'$, then $V = V'$. This is again proved by induction on the derivation of $M \Downarrow V$; do the case for the new **por**-rules.

3. (a) Describe the compact elements of $\mathbb{N}_\perp \to \mathbb{N}_\perp$.
   (b) Show that $\mathbb{N}_\perp \to \mathbb{N}_\perp$ is an algebraic dcpo.

4. Let X be a set and $\wp(X)$ the power set of X ordered by inclusion $\subseteq$.

   (a) Describe the compact elements of $\wp(X)$.
   (b) Show that $\wp(X)$ is an algebraic dcpo.

5. Suppose we are in a dcpo where each pair of elements has a glb. Show that $\forall x, y(x \sqsubseteq y \Leftrightarrow x = x \sqcap y)$.

6. (a) Prove that $\mathbb{N}_\perp$ satisfies (axiom d):

   $$\forall x, y, z \in \mathbb{N}_\perp(y \uparrow z \Rightarrow x \sqcap (y \sqcup z) = (x \sqcap y) \sqcup (x \sqcap z).$$

   (b) Prove that $\mathbb{N}_\perp \to \mathbb{N}_\perp$ (the set of Scott continuous functions) satisfies (axiom d).

7. We are in a bounded complete p.o. and we consider the property $(*)$ (used in stability)

   $$\forall x, y \in D(x \uparrow y \to f(x \sqcap y) = f(x) \sqcap f(y)) \qquad (*)$$

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

8. (a) Define all possible different "AND" functions as monotone functions (in $\mathbb{B}_\perp \times \mathbb{B}_\perp \to \mathbb{B}_\perp$).
   (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

9. Prove that the identity function $I : \mathbb{N}_\perp \to \mathbb{N}_\perp$ is not very finite.

10. Prove that, in a bounded complete dcpo, every non-empty set X has a greatest lower bound, $\bigsqcap X$.

References

[1] M. Fiore A. M. Pitts, G. Winskel. Lecture notes on denotational semantics, part ii of the computer science tripos 2018/19, https://www.cl.cam.ac.uk/teaching/1819/denotsem/denotsem2018.pdf, 2018.

[2] Samson Abramsky and Achim Jung. Domain theory. In *Handbook of Logic in Computer Science*, pages 1–168. Clarendon Press, 1994.

[3] G. Berry and Pierre-Louis Curien. Sequential algorithms on concrete data structures. *Theor. Comput. Sci.*, 20:265–321, 1982.

[4] 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.

[5] Thierry Coquand, Carl A. Gunter, and Glynn Winskel. Domain theoretic models of polymorphism. *Inf. Comput.*, 81(2):123–167, 1989.