Domain Theory

Part 2: Recursively Defined Programs

Dr. Liam O'Connor

based on material from
Graham Hutton, Dana Scott, Joseph E. Stoy, Carl Gunter, Glynn Winskel

March 6, 2024

1 Introduction

In the last lecture, we proposed modelling the semantics of programs by monotonic functions on pointed posets, and suggested that this would allow us to give a semantics to recursive programs.

In this lecture, we will refine our understanding of recursive programs to be more precise, which will show our existing formulation insufficient: our posets must not just be pointed but also complete (i.e. a cpo), and our functions must not just be monotonic but instead continuous.

2 Fixed Points

Consider our recursive while loop semantics attempt again:

$$[\![\mathsf{while}\ b\ \mathsf{do}\ c\ \mathsf{od}]\!]_{\mathcal{C}}\ \sigma \ = \begin{cases} [\![\mathsf{while}\ b\ \mathsf{do}\ c\ \mathsf{od}]\!]_{\mathcal{C}}\ ([\![c]\!]_{\mathcal{C}}\ \sigma) & \text{if } [\![b]\!]_{\mathcal{B}}\ \sigma = \mathsf{T} \\ \sigma & \text{otherwise} \end{cases}$$

How do we ensure that solutions exist for such recursive equations? And, if multiple solutions exist, how do we decide which one to pick? To address these questions, let's factor out everything in our definition except the recursion into a separate (higher-order) function:

$$[\![\mathsf{while}\ b\ \mathsf{do}\ c\ \mathsf{od}]\!]_{\mathcal{C}} = f([\![\mathsf{while}\ b\ \mathsf{do}\ c\ \mathsf{od}]\!]_{\mathcal{C}})$$

$$\text{where } f(X)\ \sigma \ \triangleq \begin{cases} X\ ([\![c]\!]_{\mathcal{C}}\ \sigma) & \text{if } [\![b]\!]_{\mathcal{B}}\ \sigma = \mathsf{T} \\ \sigma & \text{otherwise} \end{cases}$$

Looking at it this way, we can see that any solution to this equation must be an element of our semantic domain $X \in \mathbf{C}$ such that $f(X) = X$. In other words, the problem of finding solutions to our recursive equations can be cast as the problem of finding fixed points for *non-recursive* higher-order functions.

Problems

1. *Not all* monotonic functions on posets have fixed points!
2. Some monotonic functions on posets have *multiple* fixed points! Which should we choose?

So, monotonicity and posets are not enough. Intuitively, recursive programs are executed by "unfolding" as much as necessary to get a result. We would like to characterise our domains to ensure that solutions always exist, and to allow us to pick the solutions that are "minimal" in the sense that they rely on a minimal amount of unfolding.

3 Chains and Directed Sets

Let us imagine the ascending Kleene chain of our function f[1]:

$$\bot \sqsubseteq f(\bot) \sqsubseteq f(f(\bot)) \sqsubseteq f(f(f(\bot))) \sqsubseteq \cdots$$

We know this chain exists as $\bot \sqsubseteq f(\bot)$ (as $\bot$ is the bottom element), and applying monotonicity repeatedly gives us all the other links in the chain. Intuitively, the fixed point we are looking for ought to be the *limit* of this chain, i.e. $f(f(f(f(f(f(\dots))))))$. First, let us define chains and directed sets. Then we will discuss the properties our posets and functions must satisfy to ensure such limits exist, and that they correspond to fixed points.

Chains

A chain in a poset X is a totally ordered subset of X. That is, a subset $Y \subseteq X$ is a chain iff

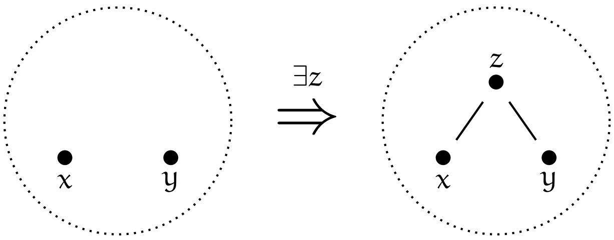
$$\forall x, y \in Y.\ x \sqsubseteq y \lor y \sqsubseteq x$$

In domain theory, we usually only care about chains with a countable number of elements, also known as ω-chains. Such ω-chains can be expressed as an infinite sequence of elements $x_1 \sqsubseteq x_2 \sqsubseteq x_3 \sqsubseteq \cdots$. The ascending Kleene chain above is an example of an ω-chain.

3.1 Directed Sets

While chains are technically sufficient for our purposes here, later on it will be convenient to talk instead about directed sets, which can be intuitively described as sets that are "going somewhere" — given two elements we can always find a "greater" one in the set. Formally, A non-empty subset $Y \subseteq X$ of a poset X is directed iff:

$$\forall x, y \in Y.\ \exists z \in Y.\ x \sqsubseteq z \land y \sqsubseteq z$$

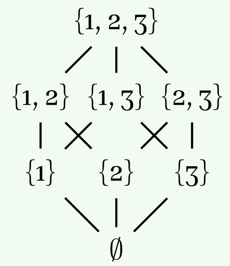


This z is an upper bound of x and y. Hence, a non-empty set is directed iff every pair of values has an upper bound *in the set*.

[1] where $\bot : C$ here refers to the constant function that just returns $\bot : \Sigma_\bot$.

Example

- The power set $\mathcal{P}(X)$ of any X is directed under $\subseteq$.
  Right: $\mathcal{P}(\{1,2,3\})$
- Any non-empty chain is directed.

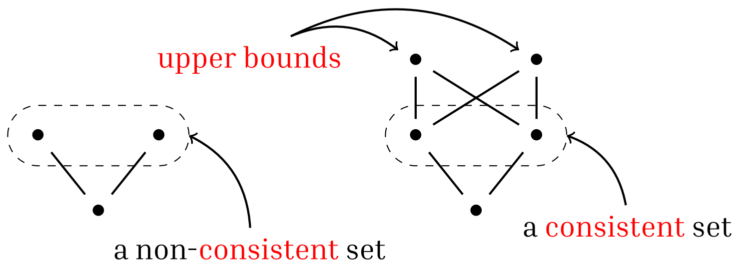


3.1.1 An Equivalent Definition

A subset $Y \subseteq X$ of a poset X is consistent iff

$$\exists x \in X. \forall y \in Y. y \sqsubseteq x$$

Such an x is called an upper bound of Y.



Alternative Definition of Directedness

A subset $Y \subseteq X$ of a poset X is directed iff every *finite* subset of $Y' \subseteq Y$ has an upper bound in Y. Briefly sketching the equivalence to our previous definition:

- **New** $\implies$ **Old**: Every pair of elements x and y has an upper bound as we can just take an upper bound of the set x, y.
- **Old** $\implies$ **New**: Given a finite set $X = \{x_1, x_2, \ldots, x_n\}$, we can show it has an upper bound by an inductive process, first taking an upper bound of x_1 and x_2, and then an upper bound of that and x_3, and so on until we have an upper bound for the whole set. This induction works because the set X is finite.

A corollary of this is that a finite set S is directed iff it has a top element $\top$, i.e. $\forall x \in S. x \sqsubseteq \top$.

4 Least Upper Bounds

Let $X \subseteq Y$ be a subset of a poset Y. An element $y \in Y$ is a *least upper bound* (lub) for X iff:

1. **it is an upper bound**: $\forall x \in X. x \sqsubseteq y$
2. **that is less than any other upper bound**: $\forall y' \in Y. (\forall x \in X. x \sqsubseteq y') \Rightarrow y \sqsubseteq y'$

It follows from this definition that the lub is unique if it exists.

Notation

We write the lub of a set X as $\bigsqcup X$, and usually write $x \sqcup y$ as a shorthand for $\bigsqcup\{x, y\}$.

If all binary lubs exist, then this binary $\sqcup$ operator is idempotent, symmetric, and associative, and $x \sqsubseteq y$ iff $x \sqcup y = y$.

When the poset Y in question is ordered by our information ordering $\sqsubseteq$, the intuition of the lub $\bigsqcup X$ is that it combines all of the information content of all elements of X, but it does *not* add any additional information (hence *least*).

Example (in $\mathbb{B}_\perp \times \mathbb{B}_\perp$)

- $(\perp, F) \sqcup (T, \perp) = (T, F)$
- $(\perp, F) \sqcup (\perp, \perp) = (\perp, F)$
- $(\perp, F) \sqcup (T, T)$ does not exist

Our poset Y is *pointed* with a bottom element iff $\bigsqcup \emptyset$ exists, as the lub of an empty set is just the least element in the poset, i.e. $\bigsqcup \emptyset = \perp$.

5 Complete Partial Orders

A complete partial order (i.e. cpo) (more specifically a dcpo or directed complete partial order) is a poset where lubs exist for the empty set and for all directed subsets. That is, a cpo is poset A such that:

1. A has a bottom element, i.e. $\perp \in A$, and
2. $\bigsqcup X$ exists for all directed $X \subseteq A$.

If our intuition for directed sets was that they were "going somewhere", then in a cpo the sets "get there" in the sense that the final destination, $\bigsqcup X$, exists in the poset (although it need not be in X itself).

Other Kinds of Completeness

Instead of directed completeness, we may consider chain completeness instead, where our second requirement instead states that $\bigsqcup X$ exists for all chains $X \subseteq A$. These formulations are equivalent, however the proof is non-trivial. If I can find a nice one, I will provide a link to a proof somewhere on the course webpage.

If we further weaken this requirement to ω-chain completeness, which only requires that lubs exist for countable ω-chains, we get what is called an ω-cpo. Directed completeness implies ω-chain completeness (so, all dcpos are ω-cpos) but not vice versa.

We could work only with chain or even ω-chain completeness, and it would be technically sufficient, however working with directed completeness simplifies some of the properties we will discuss later in the course, so we will stick with dcpos for now.

Examples and Counterexamples

- Any pointed finite poset is a cpo.
- $(\mathcal{P}(S), \subseteq)$ is a cpo: the lub is just the union.
- $(\mathbb{N}, \leqslant)$ is *not* a cpo, the ω-chain $1 \leqslant 2 \leqslant 3 \leqslant \cdots$ has no lub.

- $(\mathbb{N} \cup \{\infty\}, \leqslant)$ is a cpo, as ∞ is the lub of any non-repeating chain.
- $([0,1] \subseteq \mathbb{R}, \leqslant)$ is a cpo, but $([0,1) \subseteq \mathbb{R}, \leqslant)$ is not.
- $(\mathbb{Q}, \leqslant)$ is *not* a cpo, not just because it lacks a lub for $\mathbb{Q}$ itself, but also it doesn't contain $\sqrt{2}$, which can be expressed as the lub of an infinite sequence of rational approximations.
- A flat domain $S_\perp$ is a cpo, as the largest chains have two elements, and we always pick the non-$\perp$ one as the lub.

If we require that our semantic domains are cpos, we know that the ascending Kleene chain we saw earlier has a limit:

$$\perp \sqsubseteq f(\perp) \sqsubseteq f(f(\perp)) \sqsubseteq f(f(f(\perp))) \sqsubseteq \cdots$$

The limit of this chain is[2] $\bigsqcup\{f^n(\perp) \mid n \in \mathbb{N}\}$, where we are operating in the cpo of functions $\Sigma_\perp \to \Sigma_\perp$.

Thesis

Semantic domains are cpos, ensuring the presence of these limits.

6 Continuity

By choosing a cpo for our semantic domain, we can ensure that the ascending Kleene chain has a limit. However, it is not guaranteed that the limit we find will be a fixed point to our monotonic function f.

Discontinuity

Consider this monotonic function g defined over a cpo $(\mathbb{R} \cup \{\infty, -\infty\}, \leqslant)$:

$$g(x) = \begin{cases} \tan^{-1} x & \text{if } x < 0 \\ 1 & \text{otherwise} \end{cases}$$

Note this function is not continuous at 0. Starting from our $\perp$ element $-\infty$, we can see that the limit of the ascending Kleene chain is 0:

$$\begin{aligned} g(-\infty) &= -\tfrac{\pi}{2} \\ g(-\tfrac{\pi}{2}) &= -1 \\ g(-1) &\approx -0.78 \end{aligned}$$

But $g(0) = 1$! – the lub of the ascending Kleene chain is *not* a fixed point!

To address this problem, we shall strengthen our requirement on functions from mere monotonicity to *continuity*. Formally, a function $f : A \to B$ on cpos A and B is continuous iff for all directed $X \subseteq A$,

1. $\bigsqcup\{f(x) \mid x \in X\}$ exists, and
2. $f(\bigsqcup X) = \bigsqcup\{f(x) \mid x \in X\}$, i.e., f preserves lubs.

The intuition behind continuity is that "nothing is suddenly invented at infinity": our function will behave analogously at the limit as it does for each element in our chain.

[2]The notation f^n here refers to the n-fold self-composition of f.

Continuity implies Monotonicity

All continuous functions are monotonic. To see why, consider $x \sqsubseteq y$. Then $\{x, y\}$ is directed with a lub of y. By the second condition above, we get $f(y) = f(x) \sqcup f(y)$ which is equivalent to $f(x) \sqsubseteq f(y)$.

Example (Monotonic but not Continuous)

As an example of another function that is monotonic but is not continuous, consider this function from $\mathbb{N} \cup \{\infty\} \to \{\top, \bot\}$ defined by:

$$f(x) = \begin{cases} \bot & \text{if } x \in \mathbb{N} \\ \top & \text{otherwise} \end{cases}$$

Taking $X = \mathbb{N}$ (which is a directed subset of the cpo $\mathbb{N} \cup \{\infty\}$), then $f(\bigsqcup \mathbb{N}) = f(\infty) = \top$, but $\bigsqcup\{f(n) \mid n \in \mathbb{N}\} = \bigsqcup\{\bot\} = \bot$. Thus, this function is not continuous.

While monotonicity does not imply continuity (as we can see from the example above), it does imply the first condition of continuity[3]. Thus, we can revise our definition of continuity to the more commonly used definition below:

Alternative Definition of Continuity

A function $f : A \to B$ on cpos A and B is continuous iff:

1. f is monotonic, and
2. $f(\bigsqcup X) = \bigsqcup\{f(x) \mid x \in X\}$, for all directed $X \subseteq A$

Thesis

Computable functions are continuous functions on cpos.

7 Recursive Programs

Armed with all of our new mathematics, we can return to the problem of assigning semantics to recursively defined programs.

The Kleene Fixed Point Theorem

Let $(S, \sqsubseteq)$ be a cpo and $f : S \to S$ be a continuous function. Then the lub of the ascending Kleene chain $\bigsqcup_{n \in \mathbb{N}} f^n(\bot)$ is the least fixed point of f.

Proof: We apply continuity to show that it is a fixed point:

$$\begin{aligned} f(\textstyle\bigsqcup_{n \in \mathbb{N}} f^n(\bot)) &= \textstyle\bigsqcup_{n \in \mathbb{N}} f(f^n(\bot)) && \text{(continuity)} \\ &= \textstyle\bigsqcup_{n \in \mathbb{N}} f^{n+1}(\bot) \\ &= \textstyle\bigsqcup_{n = 1,2\ldots} f^n(\bot) && \text{(reindexing)} \\ &= \bot \sqcup \textstyle\bigsqcup_{n = 1,2\ldots} f^n(\bot) \\ &= \textstyle\bigsqcup_{n \in \mathbb{N}} f^n(\bot) \end{aligned}$$

[3]The proof of this is an exercise.

To see that it is the *least* fixed point: Let y be a fixed point of f. We know that $\bot \sqsubseteq y$ by definition of $\bot$. Taking f of both sides, we get $f(\bot) \sqsubseteq y$. We can continue this inductively and thus we know that, for all $n \in \mathbb{N}$, $f^n(\bot) \sqsubseteq y$. Because y is, therefore, an upper bound of the ascending Kleene chain, it must also be at least as large as the lub of that chain.

Armed with this theorem, we can return to our semantics of while loops, and at last define their semantics without relying on dubious recursive definitions:

$$\llbracket \mathsf{while}\ b\ \mathsf{do}\ c\ \mathsf{od} \rrbracket_{\mathcal{C}} = \mathbf{fix}(f)$$
$$\text{where } f(X)\ \sigma \triangleq \begin{cases} X\ (\llbracket c \rrbracket_{\mathcal{C}}\ \sigma) & \text{if } \llbracket b \rrbracket_{\mathcal{B}}\ \sigma = \mathsf{T} \\ \sigma & \text{otherwise} \end{cases}$$

Here $\mathbf{fix}(f)$ is the least fixed point that we get from Kleene's fixed point theorem, i.e. $\bigsqcup_{n \in \mathbb{N}} f^n(\bot)$. Proof that f is continuous is technically required but is omitted here.

Example

- What is $\mathbf{fix}(\mathrm{id})$ where id is the identity function on $\mathbb{B}_\bot$?

$$\bigsqcup \{\bot, \mathrm{id}(\bot), \mathrm{id}(\mathrm{id}(\bot)), \ldots\} = \bot$$

- What is $\mathbf{fix}(\kappa_{\mathsf{F}})$ where κ_{F} is the constant function that always returns F?

$$\bigsqcup \{\bot, \kappa_{\mathsf{F}}(\bot), \kappa_{\mathsf{F}}(\kappa_{\mathsf{F}}(\bot)), \ldots\} = \mathsf{F}$$

- What is $\mathbf{fix}(f)$ where

$$f : [\mathbb{N}_\bot] \to [\mathbb{N}_\bot]$$
$$f(x) = 1 :: x$$

(assuming these are Haskell-style lists)?

$$\bigsqcup \{\bot, 1 :: \bot, 1 :: 1 :: \bot, \ldots\} = 1 :: 1 :: 1 :: \cdots$$

Here $\mathbf{fix}(f)$ would be the semantics of the recursive definition $\mathsf{ones} = 1 :: \mathsf{ones}$.

We can at last use $\mathbf{fix}(f)$ to give semantics to recursive programs. Consider the function $\Phi \in (\mathbb{N}_\bot \to \mathbb{N}_\bot) \to (\mathbb{N}_\bot \to \mathbb{N}_\bot)$, given by:

$$\Phi(f) = \lambda n.\ \text{if } n = 0 \text{ then } 1 \text{ else } n \cdot f(n-1)$$

Looking at the first few elements of our ascending Kleene chain, we get:

$$\begin{array}{lcll} \Phi^0(\bot) & = & \lambda n.\ \bot & (\text{i.e. } \bot \text{ in } \mathbb{N}_\bot \to \mathbb{N}_\bot) \\ \Phi^1(\bot) & = & \lambda n.\ \text{if } n = 0 \text{ then } 1 \text{ else } \bot & \\ \Phi^2(\bot) & = & \lambda n.\ \text{if } n = 0 \text{ then } 1 \text{ else } n \cdot (\text{if } n - 1 = 0 \text{ then } 1 \text{ else } \bot) & \\ \ldots & & & \end{array}$$

Continuing on, we see that the mth approximation $\Phi^m(\bot)$ to $\mathbf{fix}(\Phi)$ is the function that gives $x!$ for all x up to m, and diverges for all other arguments. Hence the limit $\mathbf{fix}(\Phi)$ is the factorial function on $\mathbb{N}_\bot$!

Exercises

1. Give an example of a poset A and a monotonic function $f : A \to A$ such that f *doesn't* have a fixed point.

2. What is the lub operator on subsets $X \subseteq \mathbb{N}$ of the poset $(\mathbb{N}, \leqslant)$?

3. Show that if a function $f : A \to B$ on cpos A and B is monotonic and A is finite, then f is continuous.
   Hint: Finite directed sets contain their lub

4. Show that if a function $f : A \to B$ on cpos A and B is monotonic, then $\bigsqcup\{f(x) \mid x \in X\}$ exists.
   Hint: It suffices to show that $\{f(x) \mid x \in X\}$ is directed.

5. a) Show that if A and B are posets and $X \subseteq A \times B$ is directed, then the subsets $\pi_0(X) \subseteq A$ and $\pi_1(X) \subseteq B$ (defined below) are also directed.

$$\begin{aligned} \pi_0(X) &= \{a \in A \mid \exists b \in B.\ (a, b) \in X\} \\ \pi_1(X) &= \{b \in B \mid \exists a \in A.\ (a, b) \in X\} \end{aligned}$$

   b) Give an example of a set $X \subseteq \{\top, \bot\} \times \{\top, \bot\}$ such that $\pi_0(X)$ and $\pi_1(X)$ are directed, but X is not.

   c) Show that if A and B are cpos and $X \subseteq A \times B$ is directed, then

$$\bigsqcup X = \left(\bigsqcup \pi_0(X), \bigsqcup \pi_1(X)\right)$$

   Note: Together with $\bot_{A \times B} = (\bot_A, \bot_B)$ this shows that the Cartesian product of two cpos is itself a cpo.

6. Write down the first few approximations $\Phi^m(\bot)$ to $\textbf{fix}(\Phi)$, where the higher order function $\Phi : (\mathbb{Z}_\bot \to [\mathbb{Z}_\bot]) \to (\mathbb{Z}_\bot \to [\mathbb{Z}_\bot])$ is given by:

$$\Phi(f) = \lambda n.\ n :: f(n + 1)$$

   What is $\Phi^m(\bot)$? What is $\textbf{fix}(\Phi)$?

7. Repeat the same process as the previous question, but this time for a new higher order function $\Phi : (\mathbb{Z}_\bot \to \mathbb{Z}_\bot) \to (\mathbb{Z}_\bot \to \mathbb{Z}_\bot)$ given by:

$$\Phi(f) = \lambda n.\ \text{if } n = 0 \text{ then } 0 \text{ else } f(n - 2)$$

Glossary

ω-chain A chain with countable elements. 2, 4, 8, 9

ω-cpo A cpo which is complete for ω-chains, as opposed to the stronger dcpos which are complete for directed subsets. 4, 9

ascending Kleene chain The ω-chain that results from the iterated application of a monotonic function f to $\bot$, i.e. $\bot \sqsubseteq f(\bot) \sqsubseteq f(f(\bot)) \sqsubseteq f(f(f(\bot))) \sqsubseteq \cdots$. 2, 5–7

associative An operator $\sqcup$ is *associative* if $x \sqcup (y \sqcup z) = (x \sqcup y) \sqcup z$. 4

chain A *chain* in a poset X is a totally ordered subset of X. 2–5, 8

complete A poset is *complete* if lubs exist for the empty set (i.e. a bottom element) and for all directed subsets. 1, 4, 8, 9

consistent A set is *consistent* if it has an upper bound. 3

continuous A function $f : A \to B$ on cpos A and B is continuous iff f is monotonic and $f(\bigsqcup X) = \bigsqcup\{f(x) \mid x \in X\}$ for all directed $X \subseteq A$. 1, 5–7, 9

cpo A cpo or a complete partial order is a poset which is complete. 1, 4–6, 8, 9

dcpo A cpo which is complete for directed subsets, as opposed to ω-cpos which are only complete for ω-chains. 4, 8

directed A non-empty set is *directed* if every pair of values has an upper bound *in the set*. 2–6, 8, 9

fixed point A value x is a *fixed point* of a function f if $f(x) = x$. 1, 2, 5–7

idempotent An operator $\sqcup$ is *idempotent* if $x \sqcup x = x$. 4

lub The *least* upper bound. 3–8

symmetric An operator $\sqcup$ is *symmetric* if $x \sqcup y = y \sqcup x$. 4

top A *top* element of a set X, written $\top$, is an element that is an upper bound to all elements in the set, i.e. $\forall x \in X . x \sqsubseteq \top$. 3

totally ordered A poset is *totally ordered* if all elements are comparable. 2, 8

upper bound The *upper bound* of a set Y is some x such that $\forall y \in Y . y \sqsubseteq x$. 2, 3, 7–9