1 Trouble with while

If we try to define $\mathcal{C}[\![\textbf{while}\, b\, \textbf{do}\, c]\!]$ in the obvious manner, we get

$$\mathcal{C}[\![\textbf{while}\, b\, \textbf{do}\, c]\!]\sigma = \mathit{if}\ \neg\mathcal{B}[\![b]\!] \quad \mathit{then} \quad \sigma \quad \mathit{else} \quad \mathcal{C}[\![\textbf{while}\, b\, \textbf{do}\, c]\!](\mathcal{C}[\![c]\!]\sigma).$$

However, $\mathcal{C}[\![\textbf{while}\, b\, \textbf{do}\, c]\!]$ appears on both sides—this is really an equation, not a definition[1]. Looking at this more generally, $\mathcal{C}[\![\textbf{while}\, b\, \textbf{do}\, c]\!]$ is a solution to the equation

$$x = \Gamma(x)$$

where

$$\Gamma = \lambda f \in \Sigma_\perp \to \Sigma_\perp .\, \lambda\sigma \in \Sigma_\perp .\, \mathit{if}\ \neg\mathcal{B}[\![b]\!]\ \mathit{then}\ \sigma\ \mathit{else}\ f(\mathcal{C}[\![c]\!]\sigma).$$

What we would like to do is define

$$\begin{aligned} \mathcal{C}[\![\textbf{while}\, b\, \textbf{do}\, c]\!] &= \mathit{fix}(\Gamma) \\ &= \mathit{fix}(\lambda f \in \Sigma_\perp \to \Sigma_\perp .\, \lambda\sigma \in \Sigma_\perp .\, \mathit{if}\ \neg\mathcal{B}[\![b]\!]\sigma\ \mathit{then}\ \sigma\ \mathit{else}\ f(\mathcal{C}[\![c]\!]\sigma)) \end{aligned}$$

But which fixed point of Γ do we want? We would like to take the "least" fixed point, in the sense that we want $\mathcal{C}[\![\textbf{while}\, b\, \textbf{do}\, c]\!]$ to give a non-$\perp$ result only when required by the intended semantics. (For example, we want $\mathcal{C}[\![\textbf{while}\ \textbf{true}\ \textbf{do}\ \textbf{skip}]\!]\sigma = \perp$ for all σ.) The rest of this lecture will expand on this notion of least fixed point, with a look at the underlying theory of *partial orders*.

Iterating Γ allows us to create a sequence of approximations for $\mathcal{C}[\![\textbf{while}\, b\, \textbf{do}\, c]\!]$:

$$\begin{aligned} f_0 &= \perp \text{ (more precisely, } \perp_{\Sigma_\perp \to \Sigma_\perp}) \\ f_1 &= \Gamma(\perp) \\ &= \lambda\sigma.\, \mathit{if}\ \neg\mathcal{B}[\![b]\!]\ \mathit{then}\ \sigma\ \mathit{else}\ \perp \\ f_2 &= \Gamma(\Gamma(\perp)) \\ &= \lambda\sigma.\, \mathit{if}\ \neg\mathcal{B}[\![b]\!]\sigma\ \mathit{then}\ \sigma\ \mathit{else} \\ &\qquad \mathit{if}\ \neg\mathcal{B}[\![b]\!]\mathcal{C}[\![c]\!]\sigma\ \mathit{then}\ \mathcal{C}[\![c]\!]\sigma\ \mathit{else}\ \perp \\ f_3 &= \Gamma(\Gamma(\Gamma(\perp))) \\ &= \lambda\sigma.\, \mathit{if}\ \neg\mathcal{B}[\![b]\!]\sigma\ \mathit{then}\ \sigma\ \mathit{else} \\ &\qquad \mathit{if}\ \neg\mathcal{B}[\![b]\!]\mathcal{C}[\![c]\!]\sigma\ \mathit{then}\ \mathcal{C}[\![c]\!]\sigma\ \mathit{else} \\ &\qquad\ \ \mathit{if}\ \neg\mathcal{B}[\![b]\!]\mathcal{C}[\![c]\!]\mathcal{C}[\![c]\!]\sigma\ \mathit{then}\ \mathcal{C}[\![c]\!]\mathcal{C}[\![c]\!]\sigma\ \mathit{else}\ \perp \\ &\ \ \vdots \\ f_n &= \Gamma^n(\perp) \\ &\ \ \vdots \end{aligned}$$

The "limit" of this sequence will be the denotation of $\textbf{while}\, b\, \textbf{do}\, c$. To take this "limit", we will consider the approximations as an increasing sequence $f_0 \leq f_1 \leq f_2 \leq \cdots$, and then take the least upper bound. We must first study partial orders to get the needed machinery.

[1] It's important to point out here that our denotations will be defined by structural induction, so that it is okay in this case to assume that $\mathcal{B}[\![b]\!]$ and $\mathcal{C}[\![c]\!]$ are defined.

2 Partial Orders

A *partial order* (also known as a *partially ordered set* or *poset*) is a pair $(S, \sqsubseteq)$, where

- S is a set of elements.
- $\sqsubseteq$ is a relation on S which is:
  - *i.* reflexive: $x \sqsubseteq x$
  - *ii.* transitive: $(x \sqsubseteq y \wedge y \sqsubseteq z) \Rightarrow x \sqsubseteq z$
  - *iii.* anti-symmetric: $(x \sqsubseteq y \wedge y \sqsubseteq x) \Rightarrow x = y$

Examples:

- $(\mathbf{Z}, \leq)$, where $\mathbf{Z}$ is the integers and $\leq$ is the usual ordering.
- $(\mathbf{Z}, =)$ (Note that unequal elements are incomparable in this order. Partial orders ordered by the identity relation, $=$, are called *discrete*.)
- $(2^S, \subseteq)$ (Here, 2^S denotes the powerset of S, the set of all subsets of S, often written $\mathcal{P}(S)$, and in Winskel, $\mathcal{P}ow(S)$.)
- $(2^S, \supseteq)$
- $(S, \sqsupseteq)$, if we are given that $(S, \sqsubseteq)$ is a partial order.
- $(\omega, |)$, where $\omega = \{0, 1, 2, \ldots\}$ and $a|b \Leftrightarrow (a \text{ divides } b) \Leftrightarrow (b = ka \text{ for some } k \in \omega)$. Note that for any $n \in \omega$, we have $n|0$; we call 0 an upper bound for ω (but only in this ordering, of course!).

Non-examples:

- $(\mathbf{Z}, <)$ is not a partial order, because $<$ is not reflexive.
- $(\mathbf{Z}, \sqsubseteq)$, where $m \sqsubseteq n \Leftrightarrow |m| \leq |n|$, is not a partial order because $\sqsubseteq$ is not anti-symmetric: $-1 \sqsubseteq 1$ and $1 \sqsubseteq -1$, but $-1 \neq 1$.

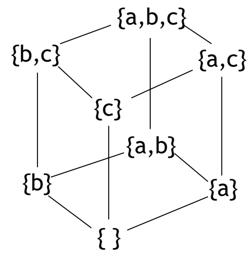
The "partial" in partial order comes from the fact that our definition does not require these orders to be total; *e.g.*, in the partial order $(2^{\{a,b\}}, \subseteq)$, the elements $\{a\}$ and $\{b\}$ are incomparable: neither $\{a\} \subseteq \{b\}$ nor $\{b\} \subseteq \{a\}$ hold.

Hasse diagrams Partial orders can be described pictorially using *Hasse diagrams*[2]. In a Hasse diagram, each element of the partial order is displayed as a (possibly labeled) point, and lines are drawn between these points, according to these rules:

1. If x and y are elements of the partial order, and $x \sqsubseteq y$, then the point corresponding to x is drawn lower in the diagram than the point corresponding to y.
2. A line is drawn between the points representing two elements x and y *iff* $x \sqsubseteq y$ and $\neg \exists z$ in the partial order, distinct from x and y, such that $x \sqsubseteq z$ and $z \sqsubseteq y$ (*i.e.*, the ordering relation between x and y is not due to transitivity).

An example of a Hasse diagram for the partial order on the set $2^{\{a,b,c\}}$ using $\subseteq$ as the binary relation is:

[2] Named after Helmut Hasse, 1898-1979. Hasse published fundamental results in algebraic number theory, including the Hasse (or "local-global") principle. He succeeded Hilbert and Weyl as the chair of the Mathematical Institute at Göttingen.



Least upper bounds Given a partial order $(S, \sqsubseteq)$, and a subset $B \subseteq S$, y is an *upper bound* of B *iff* $\forall x \in B . x \sqsubseteq y$. In addition, y is a *least upper bound iff* y is an upper bound and $y \sqsubseteq z$ for all upper bounds z of B. We may abbreviate "least upper bound" as LUB or lub. We shall notate the LUB of a subset B as $\bigsqcup B$. We may also make this an infix operator, as in $\bigsqcup\{x_1, \ldots, x_m\} = x_1 \sqcup \ldots \sqcup x_m$.

Chains A *chain* is a pairwise comparable sequence of elements from a partial order (*i.e.*, elements $x_0, x_1, x_2 \ldots$ such that $x_0 \sqsubseteq x_1 \sqsubseteq x_2 \sqsubseteq \ldots$). For any finite chain, its LUB is its last element (*e.g.*, $\bigsqcup\{x_0, x_1, \ldots, x_n\} = x_n$). Infinite chains (Winskell: ω-chains) may also have LUBs.

Complete partial orders A *complete partial order* (cpo or CPO) is a partial order in which every chain has a LUB. Note that the requirement for *every* chain is trivial for finite chains (and thus finite partial orders) – it is the infinite chains that can cause trouble.

Some examples of cpos:

- $(2^S, \subseteq)$ Here S itself is the LUB for the chain of all elements.
- $(\omega \cup \{\infty\}, \leq)$ Here ∞ is the LUB for any infinite chain: $\forall w \in \omega . w \leq \infty$.
- $([0,1], \leq)$ where $[0,1]$ is the closed continuum, and 1 is a LUB for infinite chains. Note that making the continuum open at the top – $[0,1)$ – would cause this to no longer be a cpo, since there would be no LUB for infinite chains such as $\frac{1}{2}, \frac{2}{3}, \frac{3}{4}, \ldots$
- $(S, =)$ This is a discrete cpo, just as it is a discrete partial order. The only infinite chains are of the sort $x_i \sqsubseteq x_i \sqsubseteq x_i \ldots$, of which x_i is itself a LUB.

Even if $(S, \sqsubseteq)$ is a cpo, $(S, \sqsupseteq)$ is not necessarily a cpo. Consider $((0,1], \leq)$, which is a cpo. Reversing its binary relation yields $((0,1], \geq)$ which is not a cpo, just as $([0,1), \leq)$ above was not.

CPOs can also have a least element, written $\bot$, such that $\forall x . \bot \sqsubseteq x$. We call a cpo with such an element a *pointed cpo*. Winskel instead uses *cpo with bottom*.

3 Least fixed points of functions

Recall that at the end of the last lecture we were attempting to define the least fixed point operator *fix* over the domain $(\Sigma_\bot \to \Sigma_\bot)$ so that we could determine calculate fixed points of $\Gamma : (\Sigma_\bot \to \Sigma_\bot) \to (\Sigma_\bot \to \Sigma_\bot)$. It was unclear, however, what the "least" fixed point of this domain would be – how is one function from states to states "less" than another? We've now developed the theory to answer that question.

We define the ordering of states by *information content*: $\sigma \sqsubseteq \sigma'$ *iff* σ gives less (or at most as much) information than σ'. Non-termination is defined to provide less information than any other state: $\forall \sigma \in \Sigma_\bot . \bot \sqsubseteq \sigma$. In addition, we have that $\sigma \sqsubseteq \sigma$. No other pairs of states are defined to be comparable. The lifted set of possible states $\Sigma_\bot$ can now be characterized as a flat pointed cpo (also, in other sources: flat cpo, discrete cpo with bottom):

- Its elements are elements of $\Sigma \cup \{\bot\}$.
- The ordering relation $\sqsubseteq$ satisfies the reflexive, transitive, and anti-symmetric properties.

- There are three types of infinite chains, each with a LUB:

  1. $\bot \sqsubseteq \bot \sqsubseteq \ldots$, LUB $= \bot$
  2. $\sigma \sqsubseteq \sigma \sqsubseteq \ldots$, LUB $= \sigma$
  3. $\bot \sqsubseteq \bot \sqsubseteq \ldots \sqsubseteq \sigma \sqsubseteq \sigma \sqsubseteq \ldots$, LUB $= \sigma$

We are at least ready to define an ordering relation on functions. Functions will be ordered using a *pointwise ordering* on their results. Given a cpo E, a domain D, $f \in D \to E$, and $g \in D \to E$:

$$f \sqsubseteq_{D \to E} g \triangleq \forall x \in D. f(x) \subseteq_E g(x)$$

Note that we are defining a new cpo over $D \to E$, and that this cpo is pointed if E is pointed, since $\bot_{D \to E} = \lambda x \in D. \bot_E$.

As an example, consider two functions $\mathbf{Z} \to \mathbf{Z}_\bot$:

$$\begin{aligned} f &= \lambda x \in \mathbf{Z}. \mathbf{if}\ x = 0\ \mathbf{then}\ \bot\ \mathbf{else}\ x \\ g &= \lambda x \in \mathbf{Z}. x \end{aligned}$$

We conclude $f \sqsubseteq g$ because $f(x) \sqsubseteq g(x)$ for all x; in particular, $f(0) = \bot \sqsubseteq 1 = g(0)$.

4 Back to while

It's now time to unify our dual understanding of the denotation of **while** as both a limit and a fixed point.

We previously defined the denotation of **while** as both:

$$\begin{aligned} \mathcal{C}[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!] &= \mathit{fix}(\Gamma) \\ &= \text{limit of } \Gamma^n(\bot) \end{aligned}$$

However, we did not know how to define the *fix* operator over the range of Γ, nor did we have a definition for the least fixed point of Γ to take as its limit. CPOs have given us the machinery to handle these definitions now.

We assert that:

$$\mathcal{C}[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!] = \bigsqcup_{n \in \omega} \Gamma^n(\bot)$$

As an example to give us confidence that this is the correct definition, we see that:

$$\begin{aligned} \mathcal{C}[\![\mathbf{while}\ \mathbf{true}\ \mathbf{do}\ \mathbf{skip}]\!] &= \bigsqcup_{n \in \omega} \Gamma^n(\bot) \\ &= \bot_{\Sigma_\bot \to \Sigma_\bot} \\ &= \lambda \sigma \in \Sigma_\bot . \bot \end{aligned}$$

As we begin to construct a proof that this denotation is correct, we want to show that this limit, or LUB, is a least fixed point of Γ. That is, we want to show that

$$\bigsqcup_{n \in \omega} \Gamma^n(\bot)$$

is the least solution to

$$x = \Gamma(x)$$

This will not be true for arbitrary Γ! We need Γ to be both monotonic and continuous.

Consider a non-monotonic Γ:

$$\begin{aligned}\Gamma(x) \quad = \quad & \textbf{if } x = \bot \textbf{ then } 1 \\ & \textbf{else if } x = 1 \textbf{ then } \bot \\ & \textbf{else if } x = 0 \textbf{ then } 0\end{aligned}$$

Although 0 is clearly a fixed point of this Γ, $\Gamma^n(\bot)$ is not a chain (the elements cycle between $\bot$ and 1), and so we cannot take the LUB of it. Thus we need monotonicity.

Even monotonicity is not enough. Consider a monotonic but non-continuous Γ defined over the complete partial order $(\mathbf{R} \cup \{-\infty, \infty\}, \leq)$:

$$\Gamma(x) = \textbf{if}\, x < 0 \,\textbf{then}\, \tan^{-1}(x) \,\textbf{else}\, 1$$

The least fixed point of this Γ is 1. However,

$$\begin{aligned}\Gamma^1(\bot) \quad &= \quad \tan^{-1}(-\infty) = -\frac{\pi}{2} \\ \Gamma^2(\bot) \quad &= \quad \tan^{-1}(-\frac{\pi}{2}) = \ldots\end{aligned}$$

and $\Gamma^n(\bot)$ approaches 0, so its LUB is 0. But $\Gamma(0) = 1$, so the LUB is not a fixed point! The least fixed point of this monotonic function is actually $1 = \Gamma(1)$. We need some form of continuity in Γ for *fix* to yield a fixed point.

We continue toward our goal of proving the denotation of **while** correct in the next lecture.