Predicates on Least Fixed Points

Apiros3

First Version : Jun 25, 2025
Last Update : – –, 2025

1 Introduction

The motivation for this notes comes from wanting to clarify why we get to use the phrase 'by chain completeness' to make certain proofs regarding programs in the Functional Programming course at Oxford.

2 Setup

2.1 Basic Definitions

Definition 2.1.1. *Let D be a set equipeed with a partial order $\sqsubseteq$. The* ***least upper bound*** *of some set S is an element $y \in D$ such that*

- *for all $x \in S$, $x \sqsubseteq y$.*
- *y is the least element that satisfies the above condition.*

If a least upper bound exists, we write $\bigsqcup S$ to refer to it.

Definition 2.1.2. *A* ***cpo (complete partial order)*** *is a set D with a partial order $\sqsubseteq$ such that for any chain, the least upper bound exists in the set. Specifically, given an increasing sequence (chain)*

$$x_0 \sqsubseteq x_1 \sqsubseteq x_2 \sqsubseteq \cdots$$

The least upper bound, $\bigsqcup_n x_n$ exists and is in D.

Remark 2.1.3. Note that the term cpo and complete partial order may be used to refer to different levels of completeness, where in our case we have chosen closure under chains, but some texts may mean closure under directed sets.

Some objects are not cpos, and a simple example is $\mathbb{N}$. However, we can make this into a cpo by adding a point at infinity, and then $\mathbb{N} \cup \infty$ is a cpo.

Definition 2.1.4. *A cpo D is said to have a bottom, if there is an element in D that is less than or equal to any other element in D. Such an element is called the bottom element and is written $\perp$.*

Proposition 2.1.5. *Let D be a cpo with bottom. Then the bottom element is unique.*

Proof. Let $\perp, \perp' \in D$ be elements which are minimal in D. Then,

$$\perp \sqsubseteq \perp' \qquad \perp' \sqsubseteq \perp$$

and so $\perp = \perp'$. □

2.2 Continuity and Predicates

Definition 2.2.1. *Let $f : D \to D'$ be a function between cpos D and D'. f is* ***monotonic*** *(increasing) if for any $x, y \in D$,*

$$x \sqsubseteq y \implies f(x) \sqsubseteq f(y)$$

That is, f preserves ordering.

Definition 2.2.2. *Let $f : D \to D'$ be a monotonic (increasing) function between cpos D and D'. f is* ***continuous*** *if given any increasing chain*

$$x_0 \sqsubseteq x_1 \sqsubseteq x_2 \sqsubseteq \cdots$$

in D, we have

$$f(\bigsqcup_n x_n) = \bigsqcup_n f(x_n)$$

That is, f preserves chain suprema.

Definition 2.2.3. *Let $f : D \to D$ be a function over a cpo D. An element $y \in D$ is a* ***fixed point*** *if $f(y) = y$. Let S be the set of all such fixed points. If S has some minimal element, then this is the* ***least fixed point****. We may write $\mathrm{lfp}(f)$ to refer to this element.*

Theorem 2.2.4 (Knaster-Tarski-Kleene Least Fixed Point Theorem)**.** *Let $f : D \to D$ be a continuous function over a cpo D with $\perp$. Then,*

$$\bigsqcup_{n \in \mathbb{N}} f^n(\perp)$$

is the least fixed point of f, where $f^0(\perp) = \perp$ and $f^{n+1}(\perp) = f(f^n(\perp))$.

Proof. First,

$$f\left(\bigsqcup_{n \in \mathbb{N}} f^n(\perp)\right) = \bigsqcup_{n \in \mathbb{N}} f^{n+1}(\perp) = \bigsqcup_{n \in \mathbb{N}} f^n(\perp)$$

where the last equality comes from the fact that $\perp \sqsubseteq f(\perp)$, so this is indeed a fixed point.

Suppose that y is another such fixed point. Then, noting we have $\perp \sqsubseteq y$ and $f(\perp) \sqsubseteq f(y) = y$ from monotonicity, we inductively have that $f^n(\perp) \sqsubseteq y$. In particular, y is an upper bound for the set $\{f^n(\perp) \mid n \in \mathbb{N}\}$. Thus by definition,

$$\bigsqcup_{n \in \mathbb{N}} f^n(\perp) \sqsubseteq y$$

□

Definition 2.2.5. *A predicate P is called* ***admissible*** *or* ***ω-continuous*** *if for every ascending ω-chain (x_n), we have*

$$(\forall n \in \mathbb{N},\ P(x_n)) \implies P\left(\bigsqcup_n x_n\right)$$

Remark 2.2.6. We focus only on ω-continuous predicates, as otherwise our proof strategy does not scale. When working with explicit datatypes, it is often the case that predicates are ω-continuous as we will see in a later section.

It is therefore important that our predicates are indeed ω-continuous. For example,

- $D = \mathcal{P}(\mathbb{N})$ with $P(S) = S$ is finite is not an ω-admissible statement by considering the chain $\{0\} \subseteq \{0,1\} \subseteq \{0,1,2\} \cdots$
- Taking $D = [0,2]$ with the order on $\mathbb{R}$, any increasing chain in $[0,2]$ has a supremum in $[0,2]$. Then the predicate $P(x) := x < 1$ is not ω-admissible by considering the sequence by $x_n := 1 - \frac{1}{n}$.

3 Proof Tactic

To prove a predicate holds for a least upper bound over a chain, if the predicate is ω-continuous, it suffices to show that it is satisfied for every element in the chain (by definition). So, given an infinite object, we will decompose them into a countable set of finite objects that have some inductive notion attached to it. Then, we show that predicates definied using this inductive structure are ω-continuous.

3.1 Lists

In this section, we will work with the following definition of lists:

```
data List A  where
    Nil  :: () → List A
  | Cons :: A → List A → List A
```

Remark 3.1.1. Note that while we usually write `Nil :: List A`, this is semantically just the constant map `1` $\to$ `List A`, noting the canonical isomorphism

$$(1 \to \texttt{List A}) \to \texttt{List A} \qquad f \mapsto f()$$

so we do not need to worry about this difference.

Proposition 3.1.2. *The constructors on `List A` are continuous.*

Proof. ◻

We consider finite approximants of the list via the functor

$$F(X) = 1 + A \times X$$

This naturally generates an ω-chain $(F^n(\bot))$ by

1. $F^0(\bot) = \bot$ acts like 'no lists'
2. $F^1(\bot) = 1$ is the empty list (nil)
3. $F^2(\bot) = 1 + A$ are lists of length < 2
4. $F^n(\bot)$ are lists of length at most n.

Let $\ell = [a_0, a_1, a_2, \dots]$ be an infinite list.