Introductory course on domain theory

Paul-André Melliès

Modèles des langages de programmation
Master Parisien de Recherche en Informatique

1 Ordered set

Definition 1.1 (ordered set) *An order relation $\leq$ on a set A is a binary relation which is reflexive:*

$$\forall a \in A, \qquad a \leq a$$

transitive:

$$\forall a, b, c \in A, \qquad (a \leq b \text{ et } b \leq c) \Rightarrow a \leq c$$

and antisymmetry:

$$\forall a, b \in A, \qquad (a \leq b \text{ et } b \leq a) \Rightarrow a = b.$$

A set A equipped with an order relation $\leq$ is called an ordered set.

Definition 1.2 (monotone function) *A monotone function*

$$f \quad : \quad (A, \leq_A) \quad \longrightarrow \quad (B, \leq_B)$$

is a function

$$f \quad : \quad A \quad \longrightarrow \quad B$$

between the underlying sets, such that

$$\forall a_1, a_2 \in A, \qquad a_1 \leq_A a_2 \quad \Rightarrow \quad f(a_1) \leq_B f(a_2).$$

Exercise. Show that the ordered set and that monotone functions define a category.

Exercise*. Show that this category is cartesian closed.

2 Least upper bound

We suppose given an ordered set $(A, \leq)$ and a subset $\mathcal{F}$ of the underlying set A.

Definition 2.1 (upper bound) *An upper bound of $\mathcal{F}$ is an element $m \in A$ such that*

$$\forall a \in \mathcal{F}, \qquad a \leq m.$$

Notation. One writes $\mathcal{F} \leq m$ when m is an upper bound of $\mathcal{F}$.

Definition 2.2 (least upper bound) *One calls the least upper bound (or lub) of $\mathcal{F}$ any element $m \in A$ which satisfies the two properties below:*

- *m is an upper bound of $\mathcal{F}$*
- *every upper bound of $\mathcal{F}$ is larger than m.*

Exercise. Show that the set $\mathcal{F}$ has one least upper bound at most.

Notation. When it exists, the least upper bound is noted $\bigvee F$.

So, when it exists, the least upper bound of a set $\mathcal{F}$ is the unique element $\bigvee \mathcal{F}$ which satisfies the two properties below:

$$\mathcal{F} \leq \bigvee \mathcal{F}$$

$$\forall a \in A, \qquad \mathcal{F} \leq a \quad \Rightarrow \quad \bigvee \mathcal{F} \leq a.$$

Property. Suppose that $\mathcal{F}$ has a least upper bound $\bigvee \mathcal{F}$ in the ordered set $(A, \leq_A)$ and that

$$f \quad : \quad (A, \leq_A) \quad \longrightarrow \quad (B, \leq_B)$$

is a monotone function. In that case,

$$f(\mathcal{F}) \quad \leq_B \quad f(\bigvee \mathcal{F}).$$

In particular, if the set $f(\mathcal{F})$ has a least upper bound $\bigvee f(\mathcal{F})$ in the ordered set $(B, \leq_B)$, then

$$\bigvee f(\mathcal{F}) \quad \leq_B \quad f(\bigvee \mathcal{F}).$$

Proof. Suppose that $x \in f(\mathcal{F})$. By definition of $f(\mathcal{F})$, there exists $a \in \mathcal{F}$ such that $x = f(a)$. It follows from $a \in \mathcal{F}$ that $a \leq_A \bigvee \mathcal{F}$. One deduces that $f(a) \leq_B f(\bigvee \mathcal{F})$ from the hypothesis that the function f is monotone. From this, it follows that $f(\mathcal{F}) \ \leq_B \ f(\bigvee \mathcal{F})$ since the property has been chosen for an arbitrary element $x = f(a)$ in the set $f(\mathcal{F})$. In order to obtain the second property, it is sufficient to notice that $f(\bigvee \mathcal{F})$ is an upper bound of $f(\mathcal{F})$ by the property just established, and that the least upper bound $\bigvee f(\mathcal{F})$ is thus smaller.

3 Streams

Definition 3.1 *We suppose given an ordered set $(A, \leq_A)$. One denotes*

$$\mathbf{Stream}(A) \quad = \quad \mathbb{N} \Rightarrow A$$

the set of total functions from natural numbers to A. This set of ordered by pointwise ordering $\leq$, defined as the relation:

$$\forall f, g \in \mathbf{Stream}(A), \qquad f \leq g \quad \Longleftrightarrow \quad \forall n \in \mathbb{N}, \quad f(n) \leq_A g(n).$$

Definition 3.2 (flat order) *For every set X, one defines the set*

$$X_\perp \quad = \quad X \cup \{\perp\}$$

equipped with the order relation $\leq$ defined as follows:

$$\forall x \in X, \qquad \perp \leq x$$

$$\forall x, y \in X, \qquad x \leq y \Rightarrow x = y.$$

In other words, the element $\perp$ is the smallest element of $X_\perp$ and all the elements of X are incomparable. This ordered set is called the flat order associated to the set X.

Exercise. Explain in what sense an element of $\mathbf{Stream}(\mathbb{N}_\perp)$ can be seen as a partial function from natural numbers to natural numbers. Explicate the order relation between two such partial functions. Give the example of a subset $\mathcal{F}$ of the set $\mathbf{Stream}(\mathbb{N}_\perp)$ such that $\mathcal{F}$ has a least upper bound $\bigvee \mathcal{F}$ which is not an element of the set $\mathcal{F}$.

4 Filters

At this point, we are ready to introduce the key notion of *filter* on an ordered set $(A, \leq_A)$.

Definition 4.1 *A filter of* $(A, \leq_A)$ *is a non-empty subset* $\mathcal{F}$ *of* A *such that:*

$$\forall a, b \in \mathcal{F}, \quad \exists c \in \mathcal{F}, \qquad a \leq c \textit{ et } b \leq c.$$

Property. If

$$f \quad : \quad (A, \leq_A) \quad \longrightarrow \quad (B, \leq_B)$$

is a monotone function and $\mathcal{F}$ is a filter of $(A, \leq_A)$, then $f(\mathcal{F})$ is a filter of $(B, \leq_B)$.

Proof. Let x, y be two elements of $f(\mathcal{F})$. By definition of $f(\mathcal{F})$, there exists a pair of elements a, b of $\mathcal{F}$ such that $x = f(a)$ and $y = f(b)$. By hypothesis that $\mathcal{F}$ is a filter, there exists an element $c \in \mathcal{F}$ such that $a \leq_A c$ and $b \leq_A c$. Since f is monotone, it follows that $f(a) \leq_B f(c)$ and $f(b) \leq_B f(c)$. One concludes from the fact that $f(c)$ is an element of $f(\mathcal{F})$ and that the two elements x and y were chosen as arbitrary elements of $f(\mathcal{F})$. dans $f(\mathcal{F})$.

5 Domains

Definition 5.1 *A domain* $(D, \leq)$ *is an ordered set such that*

- *there exists a smallest element noted* $\perp$*,*
- *every filter* $\mathcal{F}$ *of* $(D, \leq)$ *has a least upper bound.*

Property. Suppose that $(D, \leq_D)$ is a domain. Then, the ordered set $\mathbf{Stream}(D)$ is itself a domain.

Proof. The smallest element of $\mathbf{Stream}(D)$ is given by the constant function which associates the element $\perp_D$ to every natural number $n \in \mathbb{N}$. There remains to establish that every filter of $(\mathbf{Stream}(D), \leq)$ has a least upper bound. A simple way to establish the property is to use the projection function

$$\pi_n \quad : \quad \mathbf{Stream}(D) \quad \longrightarrow \quad D$$

which transports every sequence $(x_n)_{n\in\mathbb{N}}$ to its n-th element x_n. This function π_n is monotone and thus transports every filter $\mathcal{F}$ of **Stream**(D) to a filter $\pi_n(\mathcal{F})$ in the domain D. For convenience, we will write $\mathcal{F}_n$ for this filter in the domain D. By definition:

$$\mathcal{F}_n \quad = \quad \{ \quad x_n \quad | \quad (x_n)_{n\in\mathbb{N}} \in \mathcal{F} \quad \}$$

By hypothesis, $(D, \leq_D)$ is a domain. It follows that the filter $\mathcal{F}_n$ has a least upper bound $\bigvee \mathcal{F}_n$. So, one may define the stream

$$\varphi \quad = \quad (\bigvee \mathcal{F}_0, \cdots, \bigvee \mathcal{F}_n, \cdots).$$

obtained by putting together all the least upper bounds obtained for each $n \in \mathbb{N}$. By construction, this sequence φ is an upper bound of $\mathcal{F}$. There remains to establish that it is the least upper bound. So, let us suppose that ψ is an upper bound of $\mathcal{F}$. In that case,

$$\bigvee \mathcal{F}_n \quad \leq \quad \pi_n(\psi)$$

follows from the fact that π_n is monotone. Since the property is true for every natural number n, one deduces that

$$\phi \quad \leq \quad \psi$$

This establishes that ϕ provides the least upper bound $\sup \mathcal{F}$ of the filter $\mathcal{F}$, and concludes the proof.

Exercise. Use the same proof in order to establish that the ordered set $A \Rightarrow B$ is a domain when B is a domain. Here, the ordered set $A \Rightarrow B$ is defined for any two ordered sets $(A, \leq_A)$ and $(B, \leq_B)$ as the set of ordered functions, ordered by the pointwise ordering:

$$f \leq g \quad \iff \quad \forall a \in A, \quad f(a) \ \leq_B \ g(a).$$

Exercise*. Deduce that the category of domains and monotone functions is cartesian closed.

6 Continuous functions

In this section, we introduce the imposition notion of *continuous function* between domains.

Definition 6.1 (continuous functions) *A monotone function*

$$f \quad : \quad (D, \leq_D) \quad \longrightarrow \quad (E, \leq_E)$$

between domains is called continuous when

$$\bigvee f(\mathcal{F}) \quad = \quad f(\bigvee \mathcal{F})$$

for all filters $\mathcal{F}$ of the domain $(D, \leq)$.

Property. The ordered set $D \Rightarrow E$ of continuous functions ordered by pointwise ordering

$$f \leq g \quad \iff \quad \forall a \in D, \quad f(a) \ \leq_E \ g(a)$$

defines a domain.

Proof. As previously, the smallest element is defined as the constant function which associates the element $\bot_E$ to every element of D. This function is obviously continuous. Now, suppose given a filter $\mathcal{F}$ of $D \Rightarrow E$. It is essentially immediate to construct a function

$$\varphi \quad : \quad D \quad \longrightarrow \quad E$$

defined as

$$\varphi \quad : \quad a \quad \mapsto \quad \bigvee \mathcal{F}_a$$

where $\mathcal{F}_a$ is the filter obtained by projecting the filter $\mathcal{F}$ on the component a. The construction is done in the same way as in the case of streams, using the projection function

$$\pi_a \quad : \quad D \Rightarrow E \quad \longrightarrow \quad E$$

for every element $a \in D$. Now, let us show that the function

$$\varphi \quad : \quad D \quad \longrightarrow \quad E$$

just constructed is monotone. Quite obviously,

$$\forall a_1, a_2 \in D, \qquad a_1 \leq_D a_2 \quad \Rightarrow \quad \mathcal{F}_{a_1} \leq \mathcal{F}_{a_2}$$

where $\mathcal{F}_{a_1} \leq \mathcal{F}_{a_2}$ means that

$$\forall x \in \mathcal{F}_{a_1} \exists y \in \mathcal{F}_{a_2}, \quad x \leq_E y.$$

From this, one deduces that

$$a_1 \leq_D a_2 \quad \Rightarrow \quad \bigvee \mathcal{F}_{a_1} \leq_E \bigvee \mathcal{F}_{a_2}$$

this establishing that the function φ is monotone. Now, let us show that the function φ is continuous. Let $\mathcal{G}$ be a filter in D. Since the function φ is monotone, one knows that it satisfies the following inequality:

$$\bigvee \varphi(\mathcal{G}) \quad \leq \quad \varphi(\bigvee \mathcal{G}).$$

There remains to show that

$$\varphi(\bigvee \mathcal{G}) \quad \leq \quad \bigvee \varphi(\mathcal{G})$$

or formulated in another way that

$$\begin{array}{rcll}
\varphi(\bigvee \mathcal{G}) & = & \bigvee \mathcal{F}_{\bigvee \mathcal{G}} & \text{by definition of } \varphi \\
& = & \bigvee \{ f(\bigvee \mathcal{G}) | f \in \mathcal{F} \} & \text{by definition of } \mathcal{F}_a \\
& = & \bigvee \{ \bigvee f(\mathcal{G}) | f \in \mathcal{F} \} & \text{by continuity of } f \in \mathcal{F} \\
& = & \bigvee \left\{ \bigvee \{ f(x) | x \in \mathcal{G} \} \;\middle|\; f \in \mathcal{F} \right\} & \text{by definition} \\
& = & \bigvee \left\{ \bigvee \{ f(x) \,|\, f \in \mathcal{F} \} \;\middle|\; x \in \mathcal{G} \right\} & \text{by Fubini property} \\
& = & \bigvee \{ \phi(x) | x \in \mathcal{G} \} & \text{by definition of } \varphi \\
& = & \bigvee \phi(\mathcal{G}) & \text{by definition.}
\end{array}$$

Exercise*. Deduce that the category of domains and continuous function is cartesian closed.

7 Computational intuitions and motivations

The guiding intuition is that among all the functions

$$f \quad : \quad \mathbf{Stream}(\mathbb{N}) \quad \longrightarrow \quad \mathbf{Stream}(\mathbb{N})$$

only the functions which can be extend to a continuous function

$$\varphi \quad : \quad \mathbf{Stream}(\mathbb{N}_\perp) \quad \longrightarrow \quad \mathbf{Stream}(\mathbb{N}_\perp)$$

can be implemented by an algorithm. Typically, the function

$$f \quad : \quad (x_n)_{n\in\mathbb{N}} \quad \mapsto \quad (x_n + x_{n+1})_{n\in\mathbb{N}}$$

may be extended to the continuous function

$$\varphi \quad : \quad (x_n)_{n\in\mathbb{N}} \quad \mapsto \quad (y_n)_{n\in\mathbb{N}}$$

where

$$y_n \quad = \quad \begin{cases} x_n + x_{n+1} & \text{if } x_n \in \mathbb{N} \text{ and } x_{n+1} \in \mathbb{N}. \\ \bot & \text{if } x_n = \bot \\ \bot & \text{if } x_{n+1} = \bot \end{cases}$$

This function f is computable (in the lazy sense) by an algorithm which for each natural number n computes $y_n = x_n + x_{n+1}$ and loops if one of the two values x_n or x_{n+1} is not available in the input stream. On the other hand, it is not possible to implement the function

$$f \quad : \quad (x_n)_{n\in\mathbb{N}} \quad \mapsto \quad \begin{cases} (x_n)_{n\in\mathbb{N}} & \text{if } (x_n)_{n\in\mathbb{N}} \text{ is bounded} \\ (0)_{n\in\mathbb{N}} & \text{sinon} \end{cases}$$

because it cannot be extended to a continuous function

$$\varphi \quad : \quad \mathbf{Stream}(\mathbb{N}_\bot) \quad \longrightarrow \quad \mathbf{Stream}(\mathbb{N}_\bot)$$

Remark. In order to establish the property in a formal and rigourous way, one needs to interpret every program of a given programming language as a continuous function. As we will see, this is achieved by building a cartesian closed category of domains and continuous functions. From this, one deduces that a non-continuous function cannot be implemented in the language.