DRAFT chapter from
E. A. Lee, Concurrent Models of
Computation, January, 2014

1 Partially Ordered Sets

Contents

1.1 Sets **2**
1.2 Relations and Functions **3**
1.2.1 Restriction and Projection 6
1.3 Sequences **7**
1.4 Partial Orders **7**
Insight: Exponential Notation for Sets of Functions 8
1.4.1 Orders on Tuples 10
1.4.2 Upper and Lower Bounds 11
1.4.3 Complete Partial Orders 12
1.4.4 Flat Partial Orders 16
1.4.5 Lattices 17
1.5 Functions on Posets **18**
1.6 Fixed Points **22**
1.7 Summary **23**
Sidebar: Fixed Point Theorems 24
Exercises **25**

This chapter provides mathematical preliminaries used in subsequent chapters to develop the theory of concurrent systems. It reviews basic ideas and notation in logic, with particular emphasis on sets, functions, partial orders, and fixed-point theorems. The applications

in subsequent chapters are essential to develop a full understanding of the role that these mathematical models play in concurrent systems.

1.1 Sets

In this section, we review the notation for sets. A **set** is a collection of objects. When object a is in set A, we write $a \in A$. We define the following sets:

- $\mathbb{B} = \{0, 1\}$, the set of **binary digits**.
- $\mathbb{T} = \{false, true\}$, the set of **truth values**.
- $\mathbb{N} = \{0, 1, 2, \cdots\}$, the set of **natural numbers**.
- $\mathbb{Z} = \{\cdots, -1, 0, 1, 2, \cdots\}$, the set of **integers**.
- $\mathbb{R}$, the set of **real numbers**.
- $\mathbb{R}_+$, the set of **non-negative real numbers**.

When set A is entirely contained by set B, we say that A is a **subset** of B and write $A \subseteq B$. For example, $\mathbb{B} \subseteq \mathbb{N} \subseteq \mathbb{Z} \subseteq \mathbb{R}$. The sets may be equal, so the statement $\mathbb{N} \subseteq \mathbb{N}$ is true, for example. The **powerset** of a set A is defined to be the set of all subsets. It is written $\wp(A)$ or 2^A (for a justification of the latter notation, see the sidebar on page 8). The **empty set**, written $\emptyset$, is always a member of the powerset, $\emptyset \in \wp(A)$.

We define **set subtraction** as follows,

$$A \backslash B = \{a \in A \ : \ a \notin B\}$$

for all sets A and B. This notation is read "the set of elements a from A such that a is not in B."

A **cartesian product** of sets A and B is a set written $A \times B$ and defined as follows,

$$A \times B = \{(a, b) \ : \ a \in A, b \in B\}.$$

A member of this set (a, b) is called a **tuple**. This notation is read "the set of tuples (a, b) such that a is in A and b is in B." A cartesian product can be formed with three or more

sets, in which case the tuples have three or more elements. For example, we might write $(a,b,c) \in A \times B \times C$. A cartesian product of a set A with itself is written $A^2 = A \times A$. A cartesian product of a set A with itself n times, where $n \in \mathbb{N}$ is written A^n. A member of the set A^n is called an ***n*-tuple**. By convention, A^0 is a **singleton set**, or a set with exactly one element, regardless of the size of A. Specifically, we define $A^0 = \{\emptyset\}$. Note that A^0 is not itself the empty set. It is a singleton set containing the empty set (for insight into the rationale for this definition, see the box on page 8).

1.2 Relations and Functions

A **relation** from set A to set B is a subset of $A \times B$. A **partial function** f from set A to set B is a relation where $(a,b) \in f$ and $(a,b') \in f$ imply that $b = b'$. Such a partial function is written $f : A \rightharpoonup B$. A **total function** or just **function** f from A to B is a partial function where for all $a \in A$, there is a $b \in B$ such that $(a,b) \in f$. Such a function is written $f : A \to B$, and the set A is called its **domain** and the set B its **codomain**. Rather than writing $(a,b) \in f$, we can equivalently write $f(a) = b$.

Example 1.1: An example of a partial function is $f : \mathbb{R} \rightharpoonup \mathbb{R}$ defined by $f(x) = \sqrt{x}$ for all $x \in \mathbb{R}_+$. It is undefined for any $x < 0$ in its domain $\mathbb{R}$.

A partial function $f : A \rightharpoonup B$ may be defined by an **assignment rule**, as done in the above example, where an assignment rule simply explains how to obtain the value of $f(a)$ given $a \in A$. Alternatively, the function may be defined by its **graph**, which is a subset of $A \times B$.

Example 1.2: The same partial function from the previous example has the graph $f \subseteq \mathbb{R}^2$ given by

$$f = \{(x,y) \in \mathbb{R}^2 \ : \ x \geq 0 \text{ and } y = \sqrt{x}\} \ .$$

Note that we use the same notation f for the function and its graph when it is clear from context which we are talking about.

The set of all functions $f: A \to B$ is written $(A \to B)$ or B^A. The former notation is used when the exponential notation proves awkward. For a justification of the notation B^A, see the box on page 8.

The **function composition** of $f: A \to B$ and $g: B \to C$ is written $(g \circ f): A \to C$ and defined by

$$(g \circ f)(a) = g(f(a))$$

for any $a \in A$. Note that in the notation $(g \circ f)$, the function f is applied first. For a function $f: A \to A$, the composition with itself can be written $(f \circ f) = f^2$, or more generally

$$\underbrace{(f \circ f \circ \cdots \circ f)}_{n \text{ times}} = f^n$$

for any $n \in \mathbb{N}$. In case $n = 1$, $f^1 = f$. For the special case $n = 0$, the function f^0 is by convention the **identity function**, so $f^0(a) = a$ for all $a \in A$. When the domain and codomain of a function are the same, i.e. $f \in A^A$, then $f^n \in A^A$ for all $n \in \mathbb{N}$.

For every function $f: A \to B$, there is an associated function $\hat{f}: \wp(A) \to \wp(B)$ defined on the powerset of A as follows,

$$\forall\, A' \subseteq A, \quad \hat{f}(A') = \{b \in B \,:\, \exists\, a \in A',\ f(a) = b\}.$$

We call $\hat{f}$ the **lifted** version of f. When there is no ambiguity, we may write the lifted version of f simply as f rather than $\hat{f}$ (see problem 3(c) for an example of a situation where there is ambiguity).

For any $A' \subseteq A$, $\hat{f}(A')$ is called the **image** of A' for the function f. The image $\hat{f}(A)$ of the domain is called the **range** of the function f.

Example 1.3: The image $\hat{f}(\mathbb{R})$ of the function $f: \mathbb{R} \to \mathbb{R}$ defined by $f(x) = x^2$ is $\mathbb{R}_+$.

A function $f: A \to B$ is **onto** (or **surjective**) if $\hat{f}(A) = B$. A function $f: A \to B$ is **one-to-one** (or **injective**) if for all $a, a' \in A$,

$$a \neq a' \Rightarrow f(a) \neq f(a'). \tag{1.1}$$

That is, no two distinct values in the domain yield the same values in the codomain. A function that is both one-to-one and onto is called a **bijection**.

Example 1.4: The function $f: \mathbb{R} \to \mathbb{R}$ defined by $f(x) = 2x$ is a bijection. The function $f: \mathbb{Z} \to \mathbb{Z}$ defined by $f(x) = 2x$ is one-to-one, but not onto. The function $f: \mathbb{R}^2 \to \mathbb{R}$ defined by $f(x,y) = xy$ is onto but not one-to-one.

The previous example underscores the fact that an essential part of the definition of a function is its domain and codomain.

Proposition 1.1. *If $f: A \to B$ is onto, then there is a one-to-one function $h: B \to A$.*

Proof. Let h be defined by $h(b) = a$ where a is any element in A such that $f(a) = b$. There must always be at least one such element because f is onto. We can now show that h is one-to-one. To do this, consider any two elements $b, b' \in B$ where $b \neq b'$. We need to show that $h(b) \neq h(b')$. Assume to the contrary that $h(b) = h(b') = a$ for some $a \in A$. But then by the definition of h, $f(a) = b$ and $f(a) = b'$, which implies $b = b'$, a contradiction. □

The converse of this proposition is also easy to prove.

Proposition 1.2. *If $h: B \to A$ is one-to-one, then there is an onto function $f: A \to B$.*

Any bijection $f: A \to B$ has an **inverse** $f^{-1}: B \to A$ defined as follows,

$$f^{-1}(b) = a \in A \text{ such that } f(a) = b\,, \tag{1.2}$$

for all $b \in B$. This function is defined for all $b \in B$ because f is onto. And for each $b \in B$ there is a single unique $a \in A$ satisfying (1.2) because f is one-to-one. For any bijection f, its inverse is also a bijection.

1.2.1 Restriction and Projection

Given a function $f: A \to B$ and a subset $C \subseteq A$, we can define a new function $f|_C$ that is the **restriction** of f to C. It is defined so that for all $x \in C$, $f|_C(x) = f(x)$.

Example 1.5: The function $f: \mathbb{R} \to \mathbb{R}$ defined by $f(x) = x^2$ is not one-to-one. But the function $f|_{\mathbb{R}_+}$ is.

Consider an n-tuple $a = (a_0, a_1, \cdots, a_{n-1}) \in A_0 \times A_1 \times \cdots \times A_{n-1}$. A **projection** of this n-tuple extracts elements of the tuple to create a new tuple. Specifically, let

$$I = (i_0, i_1, \cdots, i_m) \in \{0, 1, \cdots, n-1\}^m$$

for some $m \in \mathbb{N} \backslash \{0\}$. That is, I is an m-tuple of indexes. Then we define the projection of a onto I by

$$\pi_I(a) = (a_{i_0}, a_{i_1}, \cdots, a_{i_m}) \in A_{i_0} \times A_{i_1} \times \cdots \times A_{i_m} .$$

The projection may be used to permute elements of a tuple, to discard elements, or to repeat elements.

Projection of a tuple and restriction of a function are related. An n-tuple $a \in A^n$ where $a = (a_0, a_1, \cdots, a_{n-1})$ may be considered a function of the form $a: \{0, 1, \cdots, n-1\} \to A$, in which case $a(0) = a_0$, $a(1) = a_1$, etc. Projection is similar to restriction of this function, differing in that restriction, by itself, does not provide the ability to permute, repeat, or renumber elements. But conceptually, the operations are similar, as illustrated by the following example.

Example 1.6: Consider a 3-tuple $a = (a_0, a_1, a_2) \in A^3$. This is represented by the function $a: \{0, 1, 2\} \to A$. Let $I = \{1, 2\}$. The projection $b = \pi_I(a) = (a_1, a_2)$, which itself can be represented by a function $b: \{0, 1\} \to A$, where $b(0) = a_1$ and $b(1) = a_2$.

The restriction $a|_I$ is not exactly the same function as b, however. The domain of the first function is $\{1, 2\}$, whereas the domain of the second is $\{0, 1\}$. In particular, $a|_I(1) = b(0) = a_1$ and $a|_I(2) = b(1) = a_2$.

A projection may lifted just like ordinary functions. Given a set of n-tuples $B \subseteq A_0 \times A_1 \times \cdots \times A_{n-1}$ and an m-tuple of indexes $I \in \{0,1,\cdots,n-1\}^m$, the **lifted projection** is

$$\hat{\pi}_I(B) = \{\pi_I(b) \ : \ b \in B\} \ .$$

1.3 Sequences

A tuple $(a_0, a_1) \in A^2$ can be interpreted as a sequence of length 2. The order of elements in the sequence matters, and is in fact captured by the natural ordering of the natural numbers. The number 0 comes before the number 1. We can generalize this and recognize that a **sequence** of elements from set A of length n is an n-tuple in the set A^n. A^0 represents the set of empty sequences, a singleton set (there is only one empty sequence). We denote the empty sequence λ.

The set of all **finite sequences** of elements from the set A is written A^*, where we interpret $*$ as a wildcard that can take on any value in $\mathbb{N}$. Since $0 \in \mathbb{N}$, $\lambda \in A^*$. A member of this set with length $n > 0$ is an n-tuple. The $*$ operator here is known as the **Kleene star** (or **Kleene operator** or **Kleene closure**), after American mathematician Stephen Cole Kleene (1909-1994).

The set of **infinite sequences** of elements from A is written $A^{\mathbb{N}}$ or A^{ω}. The set of **finite and infinite sequences** is written

$$A^{**} = A^* \cup A^{\mathbb{N}} \ .$$

Finite and infinite sequences play an important role in the semantics of concurrent programs. They can be used, for example, to represent streams of messages sent from one part of the program to another. Or they can represent successive assignments of values to a variable. For programs that terminate, finite sequences will be sufficient. For programs that do not terminate, we need infinite sequences.

1.4 Partial Orders

The notion of **order** is central to much of the structure in concurrent systems. Examples of ordering relationships include the usual intuitive ordering of numbers (0 is less than 1, etc.), but also many other concepts. For example, event a causes event b, 3.14159 is a better approximation of π than 3.14, and John is Sarah's father are all elements of ordering

Insight: Exponential Notation for Sets of Functions

The exponential notation B^A for the set of functions of form $f: A \to B$ is worth explaining. Recall that A^2 is the cartesian product of set A with itself, and that $\wp(A)$ is the powerset of A. These two notations are naturally thought of as sets of functions. A construction attributed to John von Neumann defines the natural numbers as follows,

$$\begin{aligned} \mathbf{0} &= \emptyset \\ \mathbf{1} &= \{\mathbf{0}\} = \{\emptyset\} \\ \mathbf{2} &= \{\mathbf{0}, \mathbf{1}\} = \{\emptyset, \{\emptyset\}\} \\ \mathbf{3} &= \{\mathbf{0}, \mathbf{1}, \mathbf{2}\} = \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\} \\ \ldots & \end{aligned}$$

With this definition, the powerset $\mathbf{2}^A$ is the set of functions mapping the set A into the set $\mathbf{2}$. Consider one such function, $f \in \mathbf{2}^A$. For each $a \in A$, either $f(a) = \mathbf{0}$ or $f(a) = \mathbf{1}$. If we interpret "$\mathbf{0}$" to mean "nonmember" and "$\mathbf{1}$" to mean "member," then indeed the set of functions $\mathbf{2}^A$ represents the set of all subsets of A. Each such function defines a subset.

Similarly, the cartesian product $A^\mathbf{2}$ can be interpreted as the set of functions of form $f: \mathbf{2} \to A$, or using von Neumann's numbers, $f: \{\mathbf{0}, \mathbf{1}\} \to A$. Consider a tuple $a = (a_0, a_1) \in A^\mathbf{2}$. It is natural to associate with this tuple a function $a: \{\mathbf{0}, \mathbf{1}\} \to A$ where $a(\mathbf{0}) = a_0$ and $a(\mathbf{1}) = a_1$. The argument to the function is the index into the tuple. We can now interpret the set of functions B^A of form $f: A \to B$ as a set of tuples indexed by the set A instead of by the natural numbers.

Let $\omega = \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \cdots\}$ represent the set of **von Neumann numbers**. This set is closely related to the set $\mathbb{N}$ (see problem 3). Given a set A, it is now natural to interpret A^ω as the set of all infinite sequences of elements from A, the same as $A^\mathbb{N}$.

The singleton set $A^\mathbf{0}$ can now be interpreted as the set of all functions whose domain is the empty set and codomain is A. There is exactly one such function (no two such functions are distinguishable), and that function has an empty graph. Before, we defined $A^\mathbf{0} = \{\emptyset\}$. Using von Neumann numbers, $A^\mathbf{0} = \mathbf{1}$, corresponding nicely with the definition of a zero exponent on ordinary numbers. Moreover, you can think of $A^\mathbf{0} = \{\emptyset\}$ as the set of all functions with an empty graph.

It is customary in the literature to omit the bold face font for $A^\mathbf{0}$, $\mathbf{2}^A$, and $A^\mathbf{2}$, writing instead simply A^0, 2^A, and A^2.

relations. For many such ordering relations, not all elements of the sets of interest are ordered. For example, John and Jane may not be related at all, unlike John and Sarah. Such partial orders play a central role in mathematical models of concurrent systems.

Definition 1.1. *A **partial order** on a set A is a relation from A to A satisfying the following properties. For all $a, b, c \in A$, the relation is*

1. ***reflexive**: $a \leq a$*
2. ***antisymmetric**: $a \leq b$ and $b \leq a$ implies that $a = b$.*
3. ***transitive**: $a \leq b$ and $b \leq c$ implies that $a \leq c$.*

In this definition, we have written the relation using the symbol $\leq$. Specifically, the relation is $\leq$, where $\leq \subseteq A \times A$, and $(a_0, a_1) \in \leq$ is equivalently written $a_0 \leq a_1$. The latter notation is referred to as **infix notation**.

A **partially ordered set** or **poset** is a set A and a partial order relation $\leq$ on that set. We can write a poset as a tuple $(A, \leq)$, or $(A, \leq_A)$ if we need to make a distinction between a partial order $\leq_A$ on set A and another partial order $\leq_B$ on another set B.

Example 1.7: A set A of sets and the subset relation $\subseteq$ form a poset $(A, \subseteq)$. The relation $\subseteq$ is referred to as the **subset order**.

For sets of sequences, there is a natural partial order relation based on the notion of a prefix.

Definition 1.2. *For a set A and the set of finite and infinite sequences A^{**} of elements of A, the **prefix order** is a relation $\sqsubseteq$ from A^{**} to A^{**} such that for any $s, s' \in A^{**}$, $s \sqsubseteq s'$ if either s is the empty sequence, or for all $n \in \mathbb{N}$ where $s(n)$ is defined, $s'(n)$ is defined and is equal to $s(n)$.*

It is easy to show that $(A, \sqsubseteq)$ is a poset by showing that the prefix order conforms with Definition 1.1.

Given a poset $(A, \leq)$ and two elements $a, a' \in A$, if either $a \leq a'$ or $a' \leq a$, then a and a' are said to be **comparable**. Otherwise, they are **incomparable**.

Definition 1.3. *A **chain** $C \subseteq A$ is a subset of a poset $(A, \leq)$ where any two members of the subset are comparable.*

A **total order** is a poset $(A, \leq)$ where A itself is a chain.

Example 1.8: The posets $(\mathbb{N}, \leq)$, $(\mathbb{Z}, \leq)$, and $(\mathbb{R}, \leq)$, where $\leq$ is the ordinary **numeric order** are total orders.

Example 1.9: The powerset $2^{\mathbb{N}}$ is the set of all sets of natural numbers. $(2^{\mathbb{N}}, \subseteq)$, where $\subseteq$ is the subset order, is a poset but not a total order.

Given a poset $(A, \leq)$, we can induce another relation $<$ called the **strict partial order** relation defined as follows.

$$\forall\, a, a' \in A, \quad a < a' \Leftrightarrow a \leq a' \text{ and } a \neq a' \ .$$

$(A, <)$ is called a **strict poset**.

1.4.1 Orders on Tuples

Given a poset $(A, \leq_A)$, the set A^2 has a **pointwise order** defined by

$$(a_0, a_1) \leq (a'_0, a'_1) \Leftrightarrow a_0 \leq_A a'_0 \text{ and } a_1 \leq_A a'_1,$$

for all $(a_0, a_1), (a'_0, a'_1) \in A^2$. With this order relation, $(A^2, \leq)$ is clearly a poset. An alternative order for A^2 is the **lexicographic order**, defined by

$$(a_0, a_1) \leq (a'_0, a'_1) \Leftrightarrow a_0 <_A a'_0 \text{ or } (a_0 = a'_0 \text{ and } a_1 \leq_A a'_1),$$

for all $(a_0, a_1), (a'_0, a'_1) \in A^2$.

Both pointwise order and lexicographic order generalize trivially to A^m for $m > 2$. They also generalize to A^*, $A^{\mathbb{N}}$, and A^{**}.

Example 1.10: Let $A = \{a, b, c, \cdots, z\}$ be the letters of the alphabet and $\leq$ be the usual **alphabetical order** for those letters. Then A^* is the set of all finite sequences of letters, which includes the set of all words. The lexicographic order for A^* is called the **dictionary order** because it gives the order in which words appear in a dictionary. The dictionary order is a total order, whereas the pointwise order for this poset would not be. In particular, under the pointwise order, the sequence (a, b) would be incomparable to the sequence (b, a), whereas under the dictionary order, the first is less than the second.

1.4.2 Upper and Lower Bounds

Given a poset $(A, \leq)$ and a subset $B \subseteq A$, an **upper bound** of B, if it exists, is an element $a \in A$ such that for all $b \in B$, $b \leq a$. A **least upper bound** or **LUB**, if it exists, is an upper bound a such that for all other upper bounds a' we have $a \leq a'$. A set may have an upper bound and no LUB.

Example 1.11: Consider the poset $(\mathbb{Q}, \leq)$ of **rational numbers**, where $\leq$ is the natural numeric order. Let $B \subset \mathbb{Q}$ be the subset of rational numbers whose value is less than the irrational number π (this relation "less than" is not the $\leq$ relation in this poset, since π is not a member of the poset, but we understand the definition of this subset anyway). In this poset, B has many upper bounds, but no least upper bound. Its LUB would have to be the greatest rational less than π, and there is no such greatest rational.

If a set $B \subseteq A$ has a least upper bound in the poset $(A, \leq)$, then it is said to be **joinable** in $(A, \leq)$, and the LUB is called the **join** of B and written $\bigvee B$.

Example 1.12: Consider the poset $(2^{\mathbb{N}}, \subseteq)$ of sets of natural numbers under the subset order. Every subset of $B \subseteq 2^{\mathbb{N}}$ is joinable and the join is the union of the sets

in B,

$$\bigvee B = \bigcup_{b \in B} b \, .$$

It is easy to see that this is a LUB. Any other bound must contain at least this union. It is not accidental that the notation $\bigvee$ is similar to $\cup$. This similarity can be very helpful in getting used to the notation.

Correspondingly, a subset $B \in A$ may have a **lower bound** in the poset $(A, \leq)$. This will be an element $a \in A$ such that for all $b \in B$, $a \leq b$. The set B may also have a **greatest lower bound** or **GLB**, defined similarly to the LUB. The GLB, if it exists, is called the **meet** of B in $(A, \leq)$, and is written $\bigwedge B$.

Example 1.13: Any subset B as in the previous example has a meet given by

$$\bigwedge B = \bigcap_{b \in B} b \, .$$

Again, it is not accidental that the notation $\bigwedge$ reminds us of $\cap$.

1.4.3 Complete Partial Orders

A **pointed** poset $(A, \leq)$ is one that has a **bottom element**, often written using a special upside-down 'T' character, as in

$$\bot_A = \bigwedge A \in A \, .$$

The bottom element is less than or equal to every element in A. When the set is understood, the bottom element may be written simply $\bot$.

Definition 1.4. *A nonempty subset $D \subseteq A$ of poset $(A, \leq)$ is a **directed set** if every pair of elements in D has an upper bound in D. Equivalently, D is directed if every non-empty finite subset of D is joinable in D.*

Every chain is a directed set. In fact, directed sets can be viewed as generalizations of the idea of chains.

Definition 1.5. *A **complete partial order** or **CPO** $(A, \leq)$ is a pointed poset where every directed subset is joinable in A.*

CPOs have extremely useful properties and are widely used in computer science to study semantics. To understand why they are so useful, note first that any *finite* directed subset is trivially joinable, by the definition of a directed set. To be a CPO, this property has to extend to *infinite* directed subsets. This may seem like a small step, but it is not. Directed subsets in a CPO may be thought of as being "directed" towards some goal. That goal is the least upper bound, whose existence is guaranteed by the fact that the set is a CPO. This LUB is in a sense the "**limit**" of the directed set. This notion of a limit turns out to be very powerful and surprisingly widely applicable.

The intuition behind the LUB of a directed set is easiest to understand for a chain $C \subseteq A$, which is a particularly simple kind of directed set. In this case, interpreting the LUB as a limit of the chain is completely natural. Being able to rely on the existence of this limit is valuable. Any ordered sequence in a CPO, therefore, has a limit. This is not a trivial property.

Example 1.14: None of the sets $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{R}$ with numeric order is a CPO. Neither is the set of von Neumann numbers ω under the subset order. Each of these sets is itself a chain with no least upper bound, so clearly it cannot be that every chain has a least upper bound. If we augment $\mathbb{N}$ with an infinite element ∞, bigger than all elements in $\mathbb{N}$, then the resulting set $\mathbb{N} \cup \{\infty\}$ with the (extended) numeric order is a CPO. Similarly, the set $\omega \cup \{\omega\}$ is a CPO under the subset order, because every element of ω is a subset of ω.

The sets considered in the previous example are all totally ordered. For such sets, every directed subset is a chain. In fact, every subset is a chain. So to determine whether the set is a CPO, we only need to consider whether every chain has a least upper bound. Some sets that are not totally ordered also have the property that every directed set is chain.

Example 1.15: Given any set A, the set of finite sequences A^* of elements from A with the prefix order $\sqsubseteq$ is not a CPO. It is easy to construct a chain of growing finite sequences that has no upper bound that is itself a finite sequence. However,

the set of finite and infinite sequences A^{**} is a CPO under the prefix order. To show this, first note that every directed subset of A^{**} is a chain (given two sequences that are both a prefix of a third, one of the two must be a prefix of the other). Thus, we simply note that given any chain, if the chain is finite, its least upper bound is its maximal element; if the chain is infinite, its least upper bound is the infinite sequence defined by the union of the elements of the chain. The notion of limits of such chains plays a major role in the semantics of concurrent systems.

It turns out that to decide whether any poset is a CPO, it is enough to consider only whether every chain has a least upper bound. It is not necessary to consider directed sets that are not chains. This follows from the following alternative definition of a CPO.

Definition 1.6. *A pointed poset $(A, \leq)$ is a **complete partial order (CPO)** if every chain in A is joinable in A.*

This definition captures the intuition every chain has a limit, in the sense of a least upper bound. To proof that this definition is equivalent to Definition 1.5 appears to be quite difficult. It seems to require the machinery of ordinals and the axiom of choice and is beyond the scope of this text. For a discussion, see Davey and Priestly (2002). In this text, we will use both definitions, though most of the time the second one will prove easier to use. In the case of the prefix order, every directed set is a chain, so the equivalence of the two definitions is trivial.

The following proposition enables us to construct more complicated CPOs from simpler ones.

Proposition 1.3. *Given a CPO $(A, \leq_A)$, $(A^n, \leq)$ is a CPO for any $n \in \mathbb{N}$, where $\leq$ is the pointwise order.*

Proof. First note that if $n = 0$, the proposition is trivial since the set has only one element, and every finite pointed poset is a CPO. It is also trivial for $n = 1$. For $n > 1$, we use definition 1.6 and consider only chains in A^n. Denote such a chain by

$$C = \{(a_{1,1}, a_{1,2}, \cdots, a_{1,n}), (a_{2,1}, a_{2,2}, \cdots, a_{2,n}), \cdots (a_{i,1}, a_{i,2}, \cdots, a_{i,n}), \cdots\}$$

where $i \leq j$ implies that $a_{i,m} \leq_A a_{j,m}$ for all $m \in \{1, \cdots, n\}$. Let $A_j = \{a_{1,j}, a_{2,j}, \cdots, a_{i,j}, \cdots\}$ for $j = 1, 2, \cdots$. It is easy to see then that

$$\bigvee C = (\bigvee A_1, \bigvee A_2, \cdots, \bigvee A_n) .$$

Hence, the chain has a LUB. Since every such chain has a LUB, the poset is a CPO. □

The following proposition follows trivially using the same proof technique, and proves quite useful.

Proposition 1.4. *Given a CPO $(A, \leq_A)$ and any set B, $(A^B, \leq)$ is a CPO, where $\leq$ is the pointwise order.*

The following example illustrates how we can pull together several of these ideas to begin addressing questions of program semantics.

Example 1.16: Let B be a set of variable names in a computer program and let A be the set of values that those variables can take on. During the (possibly nonterminating) execution of the program, a variable $b \in B$ takes on a (possibly infinite) sequence of values. The sequence of values is a member of the set A^{**}. We can define the **semantics** of the program to be a function $f : B \to A^{**}$ that for every $b \in B$ yields a sequence $f(b) \in A^{**}$. Since A^{**} is a CPO under the prefix order, by proposition 1.4 $(A^{**})^B$ is also a CPO under the **pointwise prefix order**.

Let a **partial execution** of the program be a function $f_i : B \to A^*$ for $i \in \mathbb{N}$. A partial execution always yields a finite sequence. If the execution of the program can be described as a chain of such partial executions (in the pointwise prefix order), then the limit of this chain in A^{**} can be taken to be the semantics f.

Describing an execution of program in this way is natural for many programs. Consider a determinate sequential program that updates values for variables as it executes. Each such update appends a new value to the end of the sequence consisting of the previous values of that variable. The sequence of growing sequences of values of variables is clearly a chain in the prefix order.

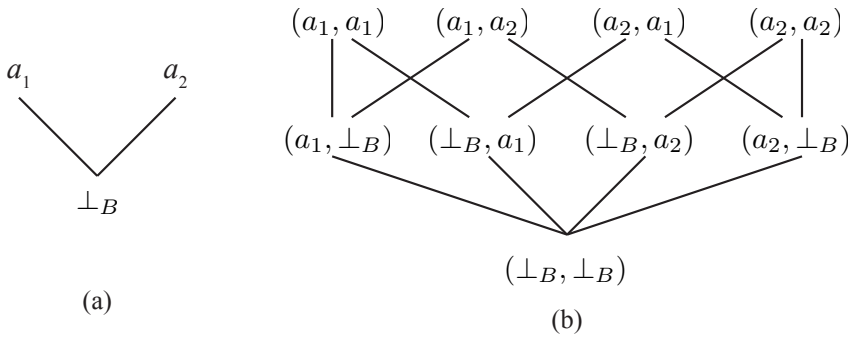


Figure 1.1: (a) Flat partial order for $B = \{\perp_B, a_1, a_2\}$. (b) Pointwise order on B^2.

1.4.4 Flat Partial Orders

Any set can be turned into a CPO by choosing a **flat order relation**. Given an arbitrary set A, augment the set with one additional element that serves as the bottom element. Specifically, let B be the augmented set and the one additional element be $\perp_B$, so $B = A \cup \{\perp_B\}$. Define a poset $(B, \leq)$, where the order relation is such that $\perp_B \leq a$ for all $a \in A$ and $a \leq a' \Rightarrow a = a'$ for all $a, a' \in A$. This is called a **flat partial order**. Any two distinct elements a and a' in A are incomparable. Moreover, every directed subset of B is a chain with either one or two elements. Trivially, each such chain has an upper bound. Hence, $(B, \leq)$ is a CPO. In this CPO, all chains are finite, so the notion of a limit of a chain is rather trivial.

Example 1.17: An example of a flat partial order is illustrated in figure 1.1(a), where $A = \{a_1, a_2\}$ and $B = \{\perp_B, a_1, a_2\}$. Diagrams of that sort are known as **Hasse diagrams**. In a Hasse diagram, elements of the poset are placed above one another with line segments indicating the order relation. If two elements a and b are joined by a line segment, and a is below b in the diagram, then $a < b$.

Example 1.18: Suppose that $(B, \leq_B)$ is the flat partial order of the previous example. Then $(B^2, \leq)$, where $\leq$ is the pointwise order, is illustrated by the Hasse diagram in figure 1.1(b).

1.4.5 Lattices

Some partial orders have more structure than we have so far assumed.

Definition 1.7. *A **lattice** is a poset $(A, \leq)$ where any two elements $a, a' \in A$ have a unique greatest lower bound $a \bigwedge a' \in A$ and a unique least upper bound $a \bigvee a' \in A$.*

Example 1.19: Given a set of sets A, the poset $(A, \subseteq)$ formed with the subset order is a lattice. Given two sets $a, a' \in A$, their GLB is their intersection $a \cap a'$, and their LUB is the union $a \cup a'$. If A is a powerset, then this lattice is known as the **powerset lattice**.

Example 1.20: The posets in figure 1.1 are not lattices. For example, in figure 1.1(a), the subset a_1, a_2 has no upper bound, much less a least upper bound. Adding an artificial top element (commonly denoted $\top$), would change these posets so that they both become lattices.

Some posets have some of the structure of lattices, but not all of it. A **lower semilattice** or **meet semilattice** is a poset $(A, \leq)$ where any two elements $a, a' \in A$ have a unique greatest lower bound $a \bigwedge a' \in A$.

Example 1.21: The posets in figure 1.1 are lower semilattices. You can exhaustively verify that for any pair of elements, there is a single unique GLB.

Example 1.22: The poset $(A^{**}, \sqsubseteq)$ of finite and infinite sequences from the set A is a lower semilattice under the prefix order. Any two sequences in A^{**} have a unique common prefix (which ma y be the empty sequence).

An **upper semilattice** or **join semilattice** is dually defined as a poset $(A, \leq)$ where any two elements $a, a' \in A$ have a unique least upper bound $a \bigvee a' \in A$.

A **complete lattice** is a poset $(A, \leq)$ where every subset (not just every pair elements) has a LUB and GLB (and similarly for **complete lower semilattice** or **complete upper semilattice**). A complete lattice is also a CPO.

Example 1.23: The powerset $2^{\mathbb{R}}$ of the reals with the subset order, $(2^{\mathbb{R}}, \subseteq)$ is a complete lattice. Given any set of sets of reals, the GLB is the intersection of the sets and the LUB is the union.

1.5 Functions on Posets

A function from a poset to a poset may preserve order.

Definition 1.8. *A function $f : A \to B$ from poset $(A, \leq_A)$ to poset $(B, \leq_B)$ is **monotonic** or **order preserving** if $a \leq_A a' \implies f(a) \leq_B f(a')$.*

Example 1.24: Consider a function $f : A^{**} \to A^{**}$ that is monotonic in the prefix order. This means that if a sequence s is a prefix of another s', i.e. that $s \sqsubseteq s'$, then $f(s) \sqsubseteq f(s')$. Functions with this property have many special qualities. The property implies that it is "safe" to evaluate the function with partial information about its input. Given only a prefix of what the input will eventually be, we can nonetheless evaluate the function using only the prefix, and we will get partial information (a prefix) about what the output will eventually be when the input is complete.

A function $f: A \to B$ is an **order embedding** if

$$a \leq_A a' \iff f(a) \leq_B f(a') \,. \tag{1.3}$$

Note that an order embedding is monotonic, but a monotonic function is not necessarily an order embedding. An order embedding is also necessarily one-to-one (see exercise 2). An order embedding $f: A \to B$ that is onto is called an **order isomorphism** from A to B. Since by exercise 2 it is also one-to-one, an order isomorphism is necessarily a bijection. Two posets are **order isomorphic** if there is an order isomorphism from one to the other. Order isomorphism is a rather strong relationship between posets. The posets are essentially the same, in that one can be obtained from the other by just renaming the elements.

Example 1.25: Let $A = \{0, 1, 2, \cdots, 255\} \subset \mathbb{N}$ and $B = \mathbb{B}^8$. Here, B is the set of all sequences of 8 binary digits. Let $f: B \to A$ be a function that for each $b \in B$ yields $f(b) \in A$ that is the number represented by b when b is interpreted as a binary unsigned number with the high-order bit first. Assume $\mathbb{B}$ is ordered so that $0 < 1$ and that B is endowed with a lexicographic order. Assume A is endowed with the usual numeric order. Then f is an order isomorphism.

If we are considering functions on CPOs rather than arbitrary posets, then an even more useful property than monotonicity is continuity.

Definition 1.9. *A function $f: A \to B$ from CPO $(A, \leq_A)$ to CPO $(B, \leq_B)$ is **continuous** if for all chains $C \subseteq A$,*

$$f(\bigvee C) = \bigvee \hat{f}(C) \,,$$

where $\hat{f}$ is the lifted version of f.

The **set of all continuous functions** from A to B is denoted $[A \to B]$, and is obviously a subset of the set B^A of all functions from A to B.

Proposition 1.5. *Every continuous function $f: A \to B$, where $(A, \leq_A)$ and $(B, \leq_B)$ are CPOs, is monotonic.*

Proof. Consider any $a, a' \in A$ where $a \leq a'$. Let $C = \{a, a'\}$, a chain. Note that $\bigvee C = a'$. Since f is continuous, we know that

$$\bigvee \hat{f}(C) = \bigvee \{f(a), f(a')\} = f(\bigvee C) = f(a') \; .$$

Hence, $f(a) \leq f(a')$, so the function is monotonic. □

Not every monotonic function is continuous however.

Example 1.26: Consider the CPO $(A, \leq)$, where $A = \mathbb{N} \cup \{\infty\}$ and $\leq$ is the numeric order. Let $f : A \to A$ be given by

$$f(a) = \begin{cases} 1 & \text{if } a \neq \infty \\ 2 & \text{otherwise} \end{cases}$$

This function is obviously monotonic. But it is not continuous. To see that, let $C = \mathbb{N}$ and note that $\bigvee C = \infty$. Hence, $f(\bigvee C) = 2$. However, $\hat{f}(C) = \{1\}$, because every element of C is finite. Hence, $\bigvee \hat{f}(C) = 1 \neq 2$. So the function is not continuous.

The following proposition specializes proposition 1.4 to continuous functions.

Proposition 1.6. *Given two CPOs $(A, \leq_A)$ and $(B, \leq_B)$, let $[B \to A] \subset A^B$ be the set of all continuous functions from B to A. Then $([B \to A], \leq)$ is a CPO under the pointwise order $\leq$.*

Proof. First we need to show that $[B \to A]$ has a bottom element. This is easy. The bottom element is a function $g \in [B \to A]$ where for all $b \in B$, $g(b) = \bot$. This function is obviously continuous and total, and hence is in $[B \to A]$.

Second, we need to show that any chain of functions in $[B \to A]$ has a LUB and that the LUB is continuous. Consider a chain of functions

$$C = \{f_1, f_2, ...\} \subset [B \to A].$$

Since each of these functions is continuous (and hence monotonic), then for any $b \in B$, the following set is also a chain,

$$C'_b = \{f_1(b), f_2(b), ...\} \subset A.$$

Since A is a CPO, this set has a LUB. Define the function $g: B \to A$ such that for all $b \in B$,

$$g(b) = \bigvee C'_b.$$

Then in the pointwise order, it must be that

$$g = \bigvee C = \bigvee \{f_1, f_2, ...\}.$$

It remains to show that g is in $[B \to A]$. To show this, we must show that it is continuous. We must show that for all chains $D \subset B$,

$$g(\bigvee D) = \bigvee \hat{g}(D).$$

Writing the elements of $D = \{d_1, d_2, ...\}$, observe that

$$\begin{aligned}
\bigvee \hat{g}(D) &= \bigvee \{g(d_1), g(d_2), ...\} \\
&= \bigvee \{\bigvee \{f_1(d_1), f_2(d_1), ...\}, \bigvee \{f_1(d_2), f_2(d_2), ...\} ...\} \\
&= \bigvee \{\bigvee \{f_1(d_1), f_1(d_2), ...\}, \bigvee \{f_2(d_1), f_2(d_2), ...\} ...\} \\
&= \bigvee \{\bigvee \hat{f}_1(D), \bigvee \hat{f}_2(D), ...\} \\
&= \bigvee \{f_1(\bigvee D), f_2(\bigvee D), ...\} \\
&= g(\bigvee D).
\end{aligned}$$

Note that the above implicitly uses the axiom of choice, which states that given a set of sets, one can construct a new set by collecting one element from each of the sets in the set of sets.

□

1.6 Fixed Points

Given a function $f: A \to A$, if there is value $a \in A$ such that $f(a) = a$, that value is called a **fixed point**. Existence and uniqueness of fixed point points play a central role in the semantics of programs. Existence of a fixed point will be interpreted as "the program has a meaning." Uniqueness will be interpreted as "the program has no more than one meaning." For functions that have multiple fixed points, we are often interested in the **least fixed point**, which is a fixed point $a \in A$ such that for any other fixed point $a' \in A$, $a \leq a'$. The following proposition assures the existence and uniqueness of such a least fixed point for continuous functions, and moreover gives a constructive procedure to determine that least fixed point.

Proposition 1.7. ***Kleene fixed-point theorem.*** *For any monotonic function $f: A \to A$ on CPO $(A, \leq)$, let*

$$C = \{f^n(\bot) \ : \ n \in \mathbb{N}\} \ .$$

Then if $\bigvee C = f(\bigvee C)$, $\bigvee C$ is the least fixed point of f. Moreover, if f is also continuous, then $\bigvee C = f(\bigvee C)$.

Proof. The first part of this theorem does not require that f be continuous, but only that it be monotonic. Suppose $\bigvee C = f(\bigvee C)$. This is a fixed point. Let a be any other fixed point, i.e. $f(a) = a$. We can show that $\bigvee C \leq a$, and hence $\bigvee C$ is the least fixed point. First, observe that $\bot \leq a$. Since f is monotonic, this implies that $f(\bot) \leq f(a) = a$. Again, since f is monotonic, this implies that $f(f(\bot)) \leq f(f(a)) = f(a) = a$. Continuing in this fashion, for any $n \in \mathbb{N}$,

$$f^n(\bot) \leq f^n(a) = a \ .$$

Hence, a is an upper bound of C. Since $\bigvee C$ is the least upper bound of C, it follows that $\bigvee C \leq a$, and hence $\bigvee C$ is the least fixed point of f.

For the second part of this theorem, we require that f be continuous, and not just monotonic. First, we observe that C is a chain in the CPO $(A, \leq)$. To see that, note that $\bot \leq f(\bot)$. Since f is monotonic, this implies that $f(\bot) \leq f(f(\bot))$. Continuing, we see that for all $n \in \mathbb{N}$, $f^n(\bot) \leq f^{n+1}(\bot)$, so C is a chain. Since C is a chain, it has a LUB $\bigvee C$.

Next note that $\hat{f}(C) \cup \{\bot\} = C$. Moreover, $\bigvee(\hat{f}(C) \cup \{\bot\}) = \bigvee \hat{f}(C)$ (an additional $\bot$ in a chain will not change its least upper bound). Combining these two facts, we conclude

that $\bigvee \hat{f}(C) = \bigvee C$. But since f continuous, we also know that $\bigvee \hat{f}(C) = f(\bigvee C)$. Hence, $f(\bigvee C) = \bigvee C$, and hence $\bigvee C$ is a fixed point. □

This theorem has profound consequences. It states that for a continuous function f, the least fixed point of this function can be found by first evaluating $f(\bot)$, then $f(f(\bot))$, then $f^3(\bot)$, etc. The result of this sequence of evaluations is a chain in A, and since A is a CPO, this chain has a limit. That limit is the least fixed point of the function. Hence, this theorem offers a **constructive procedure** for finding the least fixed point of a continuous function. The ascending chain $C = \{f^n(\bot) \ : \ n \in \mathbb{N}\}$ is known as the **Kleene chain** of the function f. Applications of this theorem are scattered throughout the text.

1.7 Summary

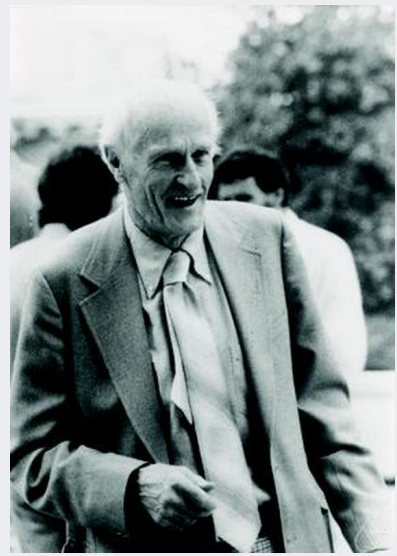
This chapter has introduced some of the mathematical tools that we will need to study concurrent models of computation in subsequent chapters. The most essential of these is the Kleene fixed-point theorem, which gives a constructive procedure (the Kleene chain) for finding the least fixed point of a continuous function. This result proves to have an astonishing variety of applications, many of which are explored in subsequent chapters. In particular, we will show that it gives a deterministic semantics to certain concurrent models of computation. It also suggests an execution policy for such models of computation.

Sidebar: Fixed Point Theorems

Proposition 1.7 is a variant of the **Kleene fixed-point theorem**, named after American mathematician Stephen Cole Kleene (1909–1994).

The Kleene fixed-point theorem is often attributed to Alfred Tarski, (1901–1983), a Polish-American logician and mathematician and a professor of mathematics at the University of California, Berkeley. **Tarski's fixed-point theorem** is similar to the Kleene fixed-point theorem, but in its original statement, it is about monotone functions on complete lattices. Because of this attribution, techniques used in computer science that are based on fixed-point theorems are often called **Tarskian**.

Another well-known theorem in this family is the **Knaster-Tarski fixed-point theorem**, developed earlier by Tarski and Bronislaw Knaster. This theorem is a special case of Tarski's fixed-point theorem that applies to the powerset lattice.



Stephen Cole Kleene (1909-1994). Photo by Konrad Jacobs, Erlangen, copyright (1978) MFO, Mathematisches Forschungsinstitut Oberwolfach, licensed under the Creative Commons Attribution-Share Alike 2.0 Germany license.

Exercises

1. This problem explores properties of onto and one-to-one functions.

   (a) Show that if $f: A \to B$ and $g: B \to C$ are onto, then $(g \circ f): A \to C$ is onto.

   (b) Show that if $f: A \to B$ is one-to-one and $g: B \to C$ is one-to-one, then $(g \circ f): A \to C$ is one-to-one.

2. For two posets A and B:

   (a) Show that if $f: A \to B$ is an order embedding, then f is one-to-one.

   (b) Show that if $f: A \to B$ is an order isomorphism, then there is an order isomorphism $g: B \to A$.

3. Let $\omega = \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \cdots\}$ be the von Neumann numbers as defined on page 8. This problem explores the relationship between this set and $\mathbb{N}$.

   (a) Let $f: \omega \to \mathbb{N}$ be defined by

   $$f(x) = |x|, \quad \forall\, x \in \omega\,.$$

   That is, $f(x)$ is the size of the set x. Show that f is a bijection.

   (b) Let $(\omega, \subseteq)$ and $(\mathbb{N}, \leq)$ be posets, where $\subseteq$ is the usual subset order and $\leq$ is the usual numeric order. Show that these posets are order isomorphic.

   (c) The lifted version of the function f in part (a) is written $\hat{f}$. What is the value of $\hat{f}(\{\emptyset, \{\emptyset\}\})$? What is the value of $f(\{\emptyset, \{\emptyset\}\})$? Note that on page 4 it is noted that when there is no ambiguity, $\hat{f}$ may be written simply f. For this function, is there such ambiguity?

4. Classify each of the following as a partial order relation, a total order relation, or neither. State whether it is a CPO.

   (a) The set of words A^* over a non-empty alphabet A with a relation $\sqsubseteq$ defined by

   $$x \sqsubseteq y \quad \Longleftrightarrow \quad \exists\, z \in A^* : x.z = y,$$

   where $x.z$ denotes concatenation.

   (b) The set $A = \{\text{seconds, grams, meters, pounds, yards}\}$ of units and a relation R defined by $(a, a') \in A$ if a can be converted to a' by scaling by a unitless constant.

(c) The set D of all countable (including finite) subsets of $\mathbb{R}$, ordered by the subset order. **Hint:** It may be easier to use Definition 1.5 for a CPO, rather than Definition 1.6.

(d) The set $\{1/n \mid n \in \mathbb{N}\} \cup \{0\}$, ordered by the natural numeric order (where $1/0$ is interpreted as ∞).

(e) The set $\{1/n \mid n \in \mathbb{N}\}$, ordered by the reverse numeric order (where again $1/0$ is interpreted as ∞).

5. State whether each of the following posets is a CPO, and also where it is a lattice, lower semilattice, or upper semilattice. If you do not have enough information to determine whether this is a CPO or a lattice, then state this.

(a) The set $A = [0,1) \subseteq \mathbb{R}$ under the numeric order, where $[0,1) = \{x \in \mathbb{R} : 0 \leq x < 1\}$.

(b) The set $A = [0,1] \subseteq \mathbb{R}$ under the numeric order, where $[0,1] = \{x \in \mathbb{R} : 0 \leq x \leq 1\}$.

(c) The set $A = \{x \in \mathbb{Q} : 0 \leq x \leq 1\}$ under the numeric order.

(d) A pointed poset $(A, \leq)$ where every directed subset is finite.

(e) A set A where every directed subset is a chain.

6. Assume two CPOs $(A, \leq)$ and $(B, \leq)$. Consider a poset $(A \times B, \leq)$ where the order is the lexicographic order.

(a) Show that $(A \times B, \leq)$ is a CPO.

(b) Suppose that $A = T_A^{**}$, $B = T_B^{**}$, and both CPOs use the prefix order, for arbitrary sets T_A and T_B. Suppose that $a \in T_A$ and $b_1, b_2 \in T_B$. Consider the set of sequences

$$C = \{((a),(b_1)),((a,a),(b_2)),((a,a,a),(b_1)),...\}$$

Under the lexicographic order, this is a chain. Find its LUB.