Semantics of Programming Languages

Denotational Semantics

Prof. Peter Müller

Software Component Technology

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

3. Denotational Semantics

Partial Ordering

- A **partial order** is a relation that is
  - reflexive: $d \sqsubseteq d$,
  - transitive: $d_1 \sqsubseteq d_2 \wedge d_2 \sqsubseteq d_3 \Rightarrow d_1 \sqsubseteq d_3$, and
  - anti-symmetric: $d_1 \sqsubseteq d_2 \wedge d_2 \sqsubseteq d_1 \Rightarrow d_1 = d_2$
- We formalize the requirements for the desired fixed point by introducing a partial order $\sqsubseteq$ on partial functions $\text{State} \hookrightarrow \text{State}$
- We set $g_1 \sqsubseteq g_2$ when the partial function $g_1 : \text{State} \hookrightarrow \text{State}$ **shares its results** with the partial function $g_2 : \text{State} \hookrightarrow \text{State}$ in the sense that $g_1(\sigma) = \sigma' \Rightarrow g_2(\sigma) = \sigma'$ for all σ, σ'

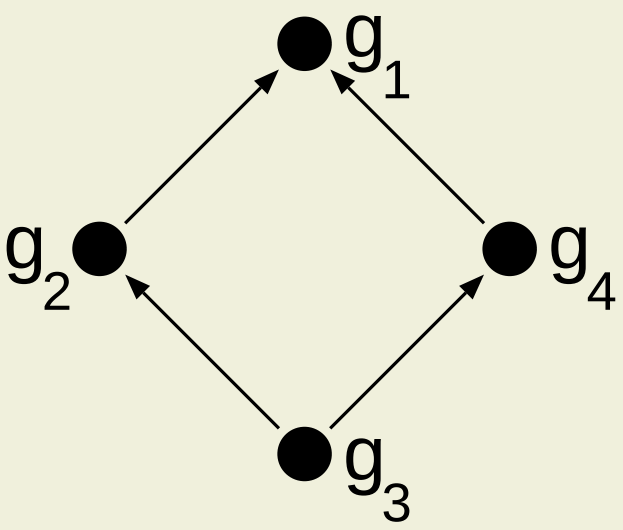
Partial Ordering: Example

$$g_1(\sigma) = \sigma$$

$$g_2(\sigma) = \begin{cases} \sigma & \text{if } \sigma(x) \geq 0 \\ \text{undefined} & \text{otherwise} \end{cases}$$

$$g_3(\sigma) = \begin{cases} \sigma & \text{if } \sigma(x) = 0 \\ \text{undefined} & \text{otherwise} \end{cases}$$

$$g_4(\sigma) = \begin{cases} \sigma & \text{if } \sigma(x) \leq 0 \\ \text{undefined} & \text{otherwise} \end{cases}$$



Partially Ordered Sets

- Definition

> A *partially ordered set* is a pair $(D, \sqsubseteq_D)$ where D is a set and $\sqsubseteq_D$ is a partial order on D.

- We say that d_1 **shares information** with d_2 if $d_1 \sqsubseteq_D d_2$
- We omit the subscript from $\sqsubseteq_D$ if it is clear from the context

Least Elements

- Definition

> An element d of D satisfying
> $$\forall d' \in D : d \sqsubseteq_D d'$$
> is a *least element* of the partially ordered set $(D, \sqsubseteq_D)$

- We say that a least element contains **no information**

Unique Least Elements

- Lemma 3.1:

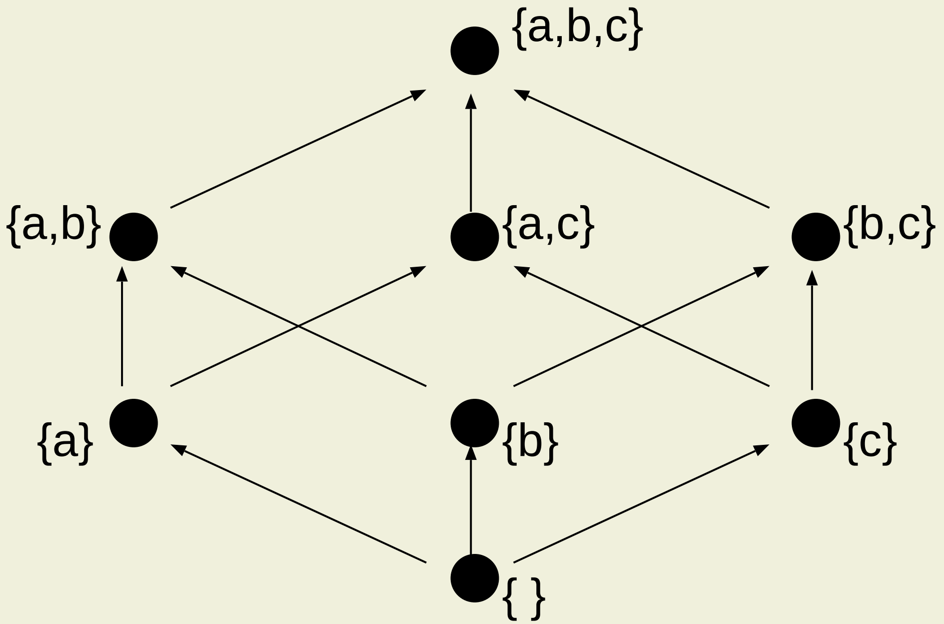
  If a partially ordered set $(D, \sqsubseteq_D)$ has a least element d, then d is unique

- Proof
  - Assume that d_1 and d_2 are two least elements of $(D, \sqsubseteq_D)$
  - By the definition of least elements, we get $d_1 \sqsubseteq d_2$ and $d_2 \sqsubseteq d_1$
  - Anti-symmetry implies $d_1 = d_2$

- The least element of D, if one exists, is denoted by $\perp_D$ (or simply $\perp$)

Partially Ordered Sets: Example

- Let S be a non-empty set and $\mathcal{P}(S) = \{K | K \subseteq S\}$ the power set of S
- $(\mathcal{P}(S), \subseteq)$ is a partially ordered set
  - $\subseteq$ is reflexive: $K \subseteq K$
  - $\subseteq$ is transitive: $K_1 \subseteq K_2 \wedge K_2 \subseteq K_3 \Rightarrow K_1 \subseteq K_3$
  - $\subseteq$ is anti-symmetric: $K_1 \subseteq K_2 \wedge K_2 \subseteq K_1 \Rightarrow K_1 = K_2$



Ordering for $S = \{a, b, c\}$

Back to Semantics

- Lemma 3.2:

> $(\text{State} \hookrightarrow \text{State}, \sqsubseteq)$ is a partially ordered set. The partial function $\bot$: $\text{State} \hookrightarrow \text{State}$ defined by
>
> $$\bot(\sigma) = \text{undefined} \quad \text{for all } \sigma$$
>
> is the least element of $\text{State} \hookrightarrow \text{State}$

- Proof
  - Part 1: $\sqsubseteq$ is a partial order
  - Part 2: $\bot$ is the least element of $\text{State} \hookrightarrow \text{State}$

Proof: Part 1

- Recall: $g_1 \sqsubseteq g_2$ means that $g_1(\sigma) = \sigma' \Rightarrow g_2(\sigma) = \sigma'$ for all σ, σ'

- Reflexivity: $g \sqsubseteq g$ since $g(\sigma) = \sigma' \Rightarrow g(\sigma) = \sigma'$

- Transitivity: $g_1 \sqsubseteq g_2 \wedge g_2 \sqsubseteq g_3 \Rightarrow g_1 \sqsubseteq g_3$ follows from the transitivity of implication "$\Rightarrow$"

- Anti-symmetry: $g_1 \sqsubseteq g_2 \wedge g_2 \sqsubseteq g_1 \Rightarrow g_1 = g_2$
  - If $g_1(\sigma) = \sigma'$ then $g_2(\sigma) = \sigma'$
  - If $g_1(\sigma) =$ undefined then $g_2(\sigma) =$ undefined (otherwise, we would get a contradiction)

Proof: Part 2

- We show that $\perp$ is the least element of $\text{State} \hookrightarrow \text{State}$

- $\perp$ is an element of $\text{State} \hookrightarrow \text{State}$
- $\perp \sqsubseteq g$ holds for all g since $\perp(\sigma) = \sigma'$ vacuously implies $g(\sigma) = \sigma'$

The Desired Fixed Point

Requirements on $FIX\ F$:

- $FIX\ F$ is a fixed point of F, that is

$$F(FIX\ F) = FIX\ F$$

- $FIX\ F$ is the least fixed point of F, that is
  if $F(g) = g$ then $FIX\ F \sqsubseteq g$

3. Denotational Semantics

Least Upper Bounds

- Definition of **upper bound**

Let $(D, \sqsubseteq)$ be a partially ordered set and Y a subset of D. An element d of D is an *upper bound* of Y if

$$\forall d' \in Y : d' \sqsubseteq d$$

- Definition of **least upper bound**

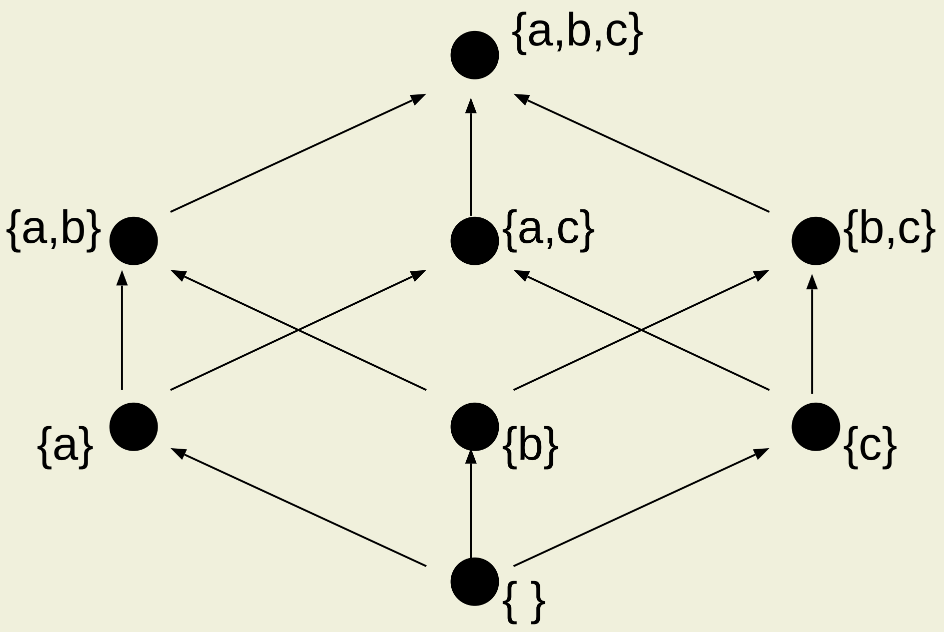
An upper bound d of Y is a *least upper bound* if and only if

d'' is an upper bound implies that $d \sqsubseteq d''$

- If Y has a least upper bound, then it is unique (see exercise session 7), and denoted by $\sqcup Y$

Least Upper Bounds: Example

- Let S be a non-empty set and $\mathcal{P}(S) = \{K | K \subseteq S\}$ the power set of S
- Every subset Y of $(\mathcal{P}(S), \subseteq)$ has the least upper bound $\bigcup_{d \in Y} d$



Ordering for $S = \{a, b, c\}$

Least Upper Bounds: Example

- $\bigcup_{d \in Y} d$ is an upper bound
  - $\bigcup_{d \in Y} d$ is in $\mathcal{P}(S)$ since it is a subset of S
  - $\forall d' \in Y : d' \subseteq \bigcup_{d \in Y} d$
- $\bigcup_{d \in Y} d$ is the least upper bound
  - We have to show that if u is an upper bound, then $\bigcup_{d \in Y} d \subseteq u$
  - For all $x \in S$, we get: $x \in \bigcup_{d \in Y} d \Rightarrow \exists d' \in Y : x \in d' \Rightarrow x \in u$ because d' has to be a subset of the upper bound u

Chains

- A subset Y is called a **chain** if it is consistent in the sense that if we take any two elements of Y then one will share its information with the other

- Definition

A subset Y is called a *chain* if

$$\forall d_1, d_2 \in Y : d_1 \sqsubseteq d_2 \vee d_2 \sqsubseteq d_1$$

Example: Power Sets

- Consider the partially ordered set $(\mathcal{P}(\{a, b, c\}), \subseteq)$
- $Y_1 = \{ \quad \{\}, \{a\}, \{a, c\} \quad \}$ is a chain
  - $\{a, c\}$ and $\{a, b, c\}$ are upper bounds of Y_1
  - $\{a, c\}$ is the least upper bound of Y_1
  - $\{a, b\}$ is **not** an upper bound of Y_1 because $\{a, c\} \not\subseteq \{a, b\}$
- $Y_2 = \{ \quad \{\}, \{a\}, \{c\}, \{a, c\} \quad \}$ is **not** a chain because $\{a\}$ and $\{c\}$ are unrelated by the ordering
  - $\{a, c\}$ is the least upper bound of Y_2
- $Y_3 = \{\}$ is a chain
  - Any element of $\mathcal{P}(\{a, b, c\})$ is an upper bound of $\{\}$
  - $\{\}$ is the least upper bound of $\{\}$

Example: Chains and Upper Bounds

- Let S be a non-empty set and
  $\mathcal{P}_{fin}(S) = \{K | K \text{ is finite and } K \subseteq S\}$
- For some choices of S, there are chains of $(\mathcal{P}_{fin}(S), \subseteq)$ that do not have an upper bound
- For $\mathcal{P}_{fin}(\mathbb{N})$, the infinite chain of finite subsets of $\mathbb{N}$

$$Y = \bigcup_{n \in \mathbb{N}} \{i | i \leq n\} = \{ \quad \{0\}, \{0,1\}, \{0,1,2\}, \ldots \quad \}$$

  has no upper bound, because $\mathbb{N}$ is the only superset of all sets in Y
  - $\mathbb{N}$ is an infinite set: $\mathbb{N} \notin \mathcal{P}_{fin}(\mathbb{N})$

Example: Partial Functions

- Let g_n : State $\hookrightarrow$ State be defined by

$$g_n(\sigma) = \begin{cases} \text{undefined} & \text{if } \sigma(x) > n \\ \sigma[x \mapsto -1] & \text{if } 0 \leq \sigma(x) \leq n \\ \sigma & \text{if } \sigma(x) < 0 \end{cases}$$

- $n \leq m \Rightarrow g_n \sqsubseteq g_m$ because g_n will be undefined for more states than g_m
- $Y_0 = \{g_n | n \geq 0\}$ is a chain.
- The partial function g is the least upper bound of Y_0

$$g(\sigma) = \begin{cases} \sigma[x \mapsto -1] & \text{if } 0 \leq \sigma(x) \\ \sigma & \text{if } \sigma(x) < 0 \end{cases}$$

CCPOs and Complete Lattices

- Definition of **Chain complete partially ordered set**

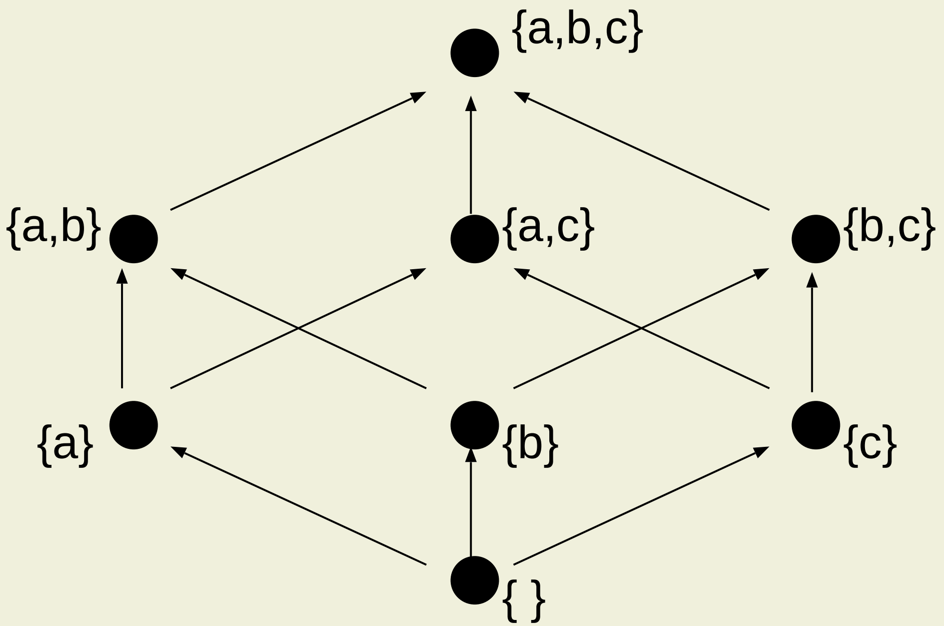
> A partially ordered set $(D, \sqsubseteq)$ is a *chain complete partially ordered set* (ccpo) whenever $\sqcup Y$ exists for all chains Y

- Definition of **Complete lattice**

> A partially ordered set $(D, \sqsubseteq)$ is a *complete lattice* whenever $\sqcup Y$ exists for all subsets Y of D

Examples

- $(\mathcal{P}(S), \subseteq)$ is a complete lattice (and, thus, a ccpo)
  - We have shown that each subset has a least upper bound
- $(\mathcal{P}_{fin}(\mathbb{N}), \subseteq)$ is neither a complete lattice nor a ccpo



Ordering for $S = \{a, b, c\}$

Least Elements

Lemma 3.3:

> If $(D, \sqsubseteq)$ is a ccpo then it has a least element $\bot = \bigsqcup \emptyset$

Proof:

- $\emptyset$ is a chain
- Since $(D, \sqsubseteq)$ is a ccpo, $\bigsqcup \emptyset$ exists
- All elements d of D are upper bounds of $\emptyset$:
  $\forall d' \in \emptyset : d' \sqsubseteq d$
- Since $\bigsqcup \emptyset$ is the least upper bound, we get $\bigsqcup \emptyset \sqsubseteq d$

CCPO of Semantic Functions

- Lemma 3.4:

  $(\text{State} \hookrightarrow \text{State}, \sqsubseteq)$ is a ccpo. The least upper bound $\sqcup Y$ of a chain Y is given by

  $$\sqcup Y(\sigma) = \begin{cases} \sigma' & \text{if } \exists g \in Y : g(\sigma) = \sigma' \\ \text{undefined} & \text{otherwise} \end{cases}$$

- For the proof, we have to show that
  1. $\sqcup Y$ is indeed a partial function in State $\hookrightarrow$ State
  2. $\sqcup Y$ is an upper bound of Y
  3. $\sqcup Y$ is the least upper bound of Y

Proof: Part 1—Partial Function

- Let g_1 and g_2 be two functions in Y with
  - $g_1(\sigma) = \sigma'$
  - $g_2(\sigma) = \sigma''$
- We prove that $\sigma' = \sigma''$
- Since Y is a chain, we have $g_1 \sqsubseteq g_2$ or $g_2 \sqsubseteq g_1$
- By the definition of $\sqsubseteq$, we get
  $g_1(\sigma) = \sigma' \Rightarrow g_2(\sigma) = \sigma'$ or $g_2(\sigma) = \sigma' \Rightarrow g_1(\sigma) = \sigma'$
- Therefore, we have $\sigma' = \sigma''$

Proof: Part 2—Upper Bound

- For any function $g \in Y$, we have to show $g \sqsubseteq \sqcup Y$, that is $g(\sigma) = \sigma' \Rightarrow \sqcup Y(\sigma) = \sigma'$
- This is a trivial consequence of the definition of $\sqcup Y$

$$\sqcup Y(\sigma) = \begin{cases} \sigma' & \text{if } \exists g \in Y : g(\sigma) = \sigma' \\ \text{undefined} & \text{otherwise} \end{cases}$$

Proof: Part 3—Least Upper Bound

- We show that $\sqcup Y$ is less than any other upper bound of Y

- Let g' be an upper bound of Y, that is, $\forall g \in Y : g \sqsubseteq g'$
- This means that if there is a function $g \in Y$ with $g(\sigma) = \sigma'$, then $g'(\sigma) = \sigma'$
- By the definition of $\sqcup Y$, we get $\sqcup Y(\sigma) = \sigma' \Rightarrow (\exists g \in Y : g(\sigma) = \sigma') \Rightarrow g'(\sigma) = \sigma'$
- Therefore, we have $\sqcup Y \sqsubseteq g'$, which completes the proof

3. Denotational Semantics

Monotone Functions

- Recall that we want to ensure that $\mathcal{S}_{DS}$ and, in particular, $F(g) = cond(\mathcal{B}[\![b]\!], g \circ \mathcal{S}_{DS}[\![s]\!], id)$ always has a least fixed point
- Since F maps functions in State $\hookrightarrow$ State to functions in State $\hookrightarrow$ State, we consider functions on ccpo's
- Definition of **Monotone Functions**

> Let $(D, \sqsubseteq)$ and $(D', \sqsubseteq')$ be ccpo's, and $f : D \rightarrow D'$ a (total) function. f is *monotone* if
> $$\forall d_1, d_2 \in D : d_1 \sqsubseteq d_2 \Rightarrow f(d_1) \sqsubseteq' f(d_2)$$

Monotone Functions: Examples

- Consider the ccpo's $(\mathcal{P}(\{a,b\}), \subseteq)$ and $(\mathcal{P}(\{d,e\}), \subseteq)$
- The function f_1 is monotone

X	$\{a,b\}$	$\{a\}$	$\{b\}$	$\emptyset$
$f_1(X)$	$\{d,e\}$	$\{d\}$	$\{e\}$	$\emptyset$

f_1 changes a's to d's and b's to e's

- The function f_2 is not monotone

X	$\{a,b\}$	$\{a\}$	$\{b\}$	$\emptyset$
$f_2(X)$	$\{d\}$	$\{d\}$	$\{e\}$	$\{e\}$

f_2 maps sets that contain an a to $\{d\}$ and sets that do not contain an a to $\{e\}$

- $\{b\} \subseteq \{a,b\}$ but $f_2(\{b\}) \not\subseteq f_2(\{a,b\})$

Composition of Monotone Functions

Lemma 3.5:

> Let $(D, \sqsubseteq)$, $(D', \sqsubseteq')$, and $(D'', \sqsubseteq'')$ be ccpo's and let $f : D \to D'$ and $f' : D' \to D''$ be monotone functions. Then $f' \circ f : D \to D''$ is a monotone function

Proof:

- Assume that $d_1 \sqsubseteq d_2$
- Monotonicity of f gives $f(d_1) \sqsubseteq' f(d_2)$
- Monotonicity of f' gives $f'(f(d_1)) \sqsubseteq'' f'(f(d_2))$

Monotonicity and Chains

- Lemma 3.6:

> Let $(D, \sqsubseteq)$ and $(D', \sqsubseteq')$ be ccpo's and let $f : D \to D'$ be a monotone function. If Y is a chain in D then $\{f(d) | d \in Y\}$ is a chain in D'. Furthermore, $\bigsqcup' \{f(d) | d \in Y\} \sqsubseteq' f(\bigsqcup Y)$

- Proof of Case 1: $Y = \emptyset$
  - $\emptyset$ is a chain in both D and D'
  - By the monotonicity of f, we get $\perp' \sqsubseteq' f(\perp)$ and, thus, $\bigsqcup' \emptyset \sqsubseteq' f(\bigsqcup \emptyset)$

Proof of Case 2—Chain

- We show that $\{f(d)|d \in Y\}$ is a chain in D'

- Let d'_1 and d'_2 be two elements of $\{f(d)|d \in Y\}$
- There are elements d_1 and d_2 with $d'_1 = f(d_1)$ and $d'_2 = f(d_2)$
- Since Y is a chain, we have $d_1 \sqsubseteq d_2$ or $d_2 \sqsubseteq d_1$
- By the monotonicity of f, we get $f(d_1) \sqsubseteq' f(d_2)$ or $f(d_2) \sqsubseteq' f(d_1)$
- Consequently, we have $d'_1 \sqsubseteq' d'_2$ or $d'_2 \sqsubseteq' d'_1$

Proof of Case 2—Least Upper Bound

- We show that $\sqcup'\{f(d)|d \in Y\} \sqsubseteq' f(\sqcup Y)$

- For an arbitrary $d \in Y$, we have $d \sqsubseteq \sqcup Y$
- By the monotonicity of f, we get $f(d) \sqsubseteq' f(\sqcup Y)$
- Since this property holds for all $d \in Y$, we get that $f(\sqcup Y)$ is an upper bound on $\{f(d)|d \in Y\}$

Monotonicity and Least Upper Bounds

- Monotone functions preserve chains, but not necessarily least upper bounds (see Example 4.31 in the book)
- Monotone functions that do preserve least upper bounds are called **continuous functions**
- Such functions satisfy $\sqcup'\{f(d) | d \in Y\} = f(\sqcup Y)$
- Intuitively, we obtain the same information independently of whether we determine the least upper bound before or after applying the monotone function f

Continuous Functions

- Definition of **Continuous Functions**

A function $f : D \to D'$ defined on ccpo's $(D, \sqsubseteq)$ and $(D', \sqsubseteq')$ is *continuous* if it is monotone and $\sqcup'\{f(d) | d \in Y\} = f(\sqcup Y)$ holds for all *non-empty* chains Y

- Definition of **Strict Functions**

A function is *strict* if $\sqcup'\{f(d) | d \in Y\} = f(\sqcup Y)$ holds for the empty chain, that is $\perp' = f(\perp)$

Continuous Functions: Example

- Consider the ccpo's $(\mathcal{P}(\{a,b\}), \subseteq)$ and $(\mathcal{P}(\{d,e\}), \subseteq)$
- The function f_1 is continuous

X	$\{a,b\}$	$\{a\}$	$\{b\}$	$\emptyset$
$f_1(X)$	$\{d,e\}$	$\{d\}$	$\{e\}$	$\emptyset$

f_1 changes a's to d's and b's to e's

- Proof
  - Let X_0 be the least upper bound of a chain Y of $\mathcal{P}(\{a,b\})$
  - Since $X_0 \in Y$, we get $f_1(\sqcup Y) = f_1(X_0) \subseteq \sqcup\{f_1(X) | X \in Y\}$
  - By Lemma 3.6, we get $\sqcup\{f_1(X) | X \in Y\} \subseteq f_1(\sqcup Y)$

- f_1 is strict because $f_1(\emptyset) = \emptyset$

Composition of Continuous Functions

Lemma 3.7:

> Let $(D, \sqsubseteq)$, $(D', \sqsubseteq')$, and $(D'', \sqsubseteq'')$ be ccpo's and let $f : D \to D'$ and $f' : D' \to D''$ be continuous functions. Then $f' \circ f : D \to D''$ is a continuous function

Proof:

- From Lemma 3.5, we know that $f' \circ f$ is monotone
- It remains to show that least upper bounds are preserved

Proof

- Let y be a non-empty chain in D
- The continuity of f gives $\sqcup'\{f(d)|d \in Y\} = f(\sqcup Y)$
- Since $\{f(d)|d \in Y\}$ is a non-empty chain in D', we get by the continuity of f':

$$\sqcup''\{f'(d')|d' \in \{f(d)|d \in Y\}\} = f'(\sqcup'\{f(d)|d \in Y\})$$

- This is equivalent to $\sqcup''\{f'(f(d))|d \in Y\} = f'(f(\sqcup Y))$

Knaster-Tarski Fixed Point Theorem

- Theorem 3.8

> Let $f : D \to D$ be a continuous function on the ccpo $(D, \sqsubseteq)$ with least element $\perp$. Then
>
> $$FIX f = \sqcup\{f^n(\perp) | n \geq 0\}$$
>
> defines an element of D that is the least fixed point of f

where $f^0 = id$ and $f^{n+1} = f \circ f^n$ for $n \geq 0$

- We have to prove that

1. $FIX f$ is well-defined
2. $FIX f$ is a fixed point of f
3. $FIX f$ is the least fixed point of f

Proof: Part 1—Well-Definedness

- Since D is a ccpo, $FIX f$ exists if $\{f^n(\bot) | n \geq 0\}$ is a non-empty chain of D

- $\{f^n(\bot) | n \geq 0\}$ is non-empty since it contains $\bot$

- By a trivial induction, one can show that $f^n(\bot) \sqsubseteq f^n(d)$ holds for all $d \in D$
- We use this result to prove $f^n(\bot) \sqsubseteq f^m(\bot)$ for $n \leq m$
  - $f^n(\bot) \sqsubseteq f^n(f^{m-n}(\bot)) = f^m(\bot)$

Proof: Part 2—Fixed Point

- We have to show that $f(FIX\,f) = FIX\,f$

$f(FIX\,f) =$	[Definition of $FIX\,f$]
$f(\sqcup\{f^n(\perp) \mid n \geq 0\}) =$	[Continuity of f]
$\sqcup\{f(f^n(\perp)) \mid n \geq 0\} =$	
$\sqcup\{f^n(\perp) \mid n \geq 1\} =$	[$\sqcup\,(Y \cup \{\perp\}) = \sqcup Y$]
$\sqcup(\{f^n(\perp) \mid n \geq 1\} \cup \{\perp\}) =$	[$f^0(\perp) = \perp$]
$\sqcup\{f^n(\perp) \mid n \geq 0\} =$	[Definition of $FIX\,f$]
$FIX\,f$	

Proof: Part 3—Least Fixed Point

- We show that $FIX\, f$ is less than any other fixed point of f

- Let d be a fixed point of f
- We have $f^n(\bot) \sqsubseteq f^n(d) = d$ for $n \geq 0$
- Thus, d is an upper bound on $\{f^n(\bot) | n \geq 0\}$
- Since $FIX\, f = \sqcup\{f^n(\bot) | n \geq 0\}$ is the **least** upper bound, we get $FIX\, f \sqsubseteq d$

Fixed Point Iteration: Example

- We determine the least fixed point of the functional F'

$$F'(g)\sigma = \begin{cases} g(\sigma) & \text{if } \sigma(x) \neq 0 \\ \sigma & \text{otherwise} \end{cases}$$

Fixed Point Theory: Summary

To guarantee the existence of a least fixed point, the following steps have been taken

1. We restrict ourselves to **chain complete partially ordered sets**—ccpo's
2. We restrict ourselves to **continuous functions** on ccpo's
3. We show that continuous functions on ccpo's always have **least fixed points**