Topic 4

Scott Induction

Scott's Fixed Point Induction Principle

Let $f : D \to D$ be a continuous function on a domain D.

For any admissible subset $S \subseteq D$, to prove that the least fixed point of f is in S, *i.e.* that

$$\mathit{fix}(f) \in S \ ,$$

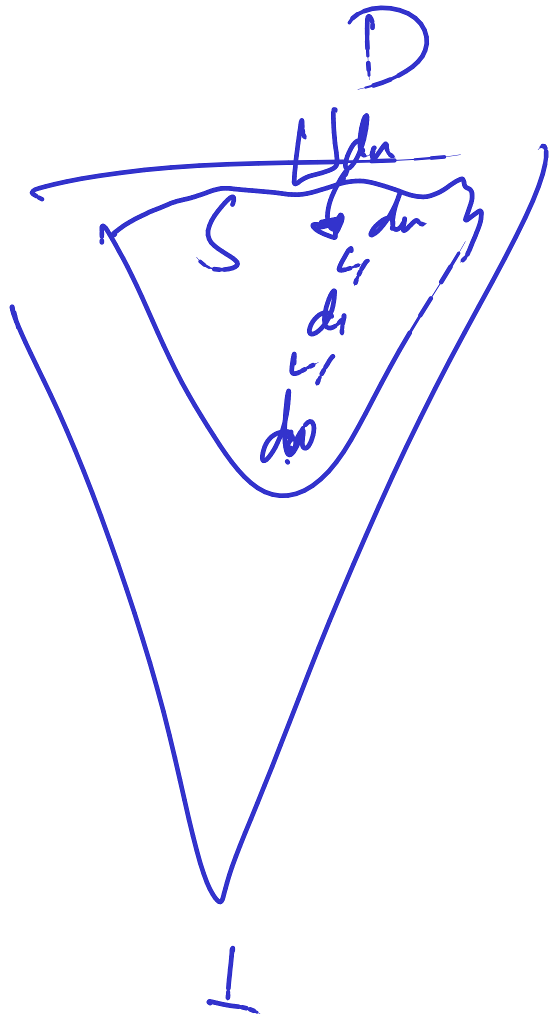
it suffices to prove

$$\forall d \in D \ (d \in S \ \Rightarrow \ f(d) \in S) \ .$$

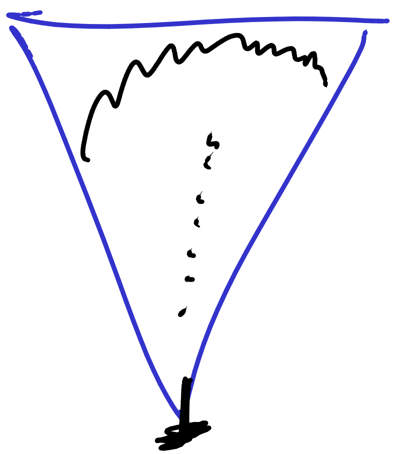
Chain-closed and admissible subsets

Let D be a cpo. A subset $S \subseteq D$ is called chain-closed iff for all chains $d_0 \sqsubseteq d_1 \sqsubseteq d_2 \sqsubseteq \ldots$ in D

$$(\forall n \geq 0 \,.\, d_n \in S) \;\Rightarrow\; \Big(\bigsqcup_{n \geq 0} d_n\Big) \in S$$



If D is a domain, $S \subseteq D$ is called admissible iff it is a chain-closed subset of D and $\perp \in S$.



admissible
S

Chain-closed and admissible subsets

Let D be a cpo. A subset $S \subseteq D$ is called chain-closed iff for all chains $d_0 \sqsubseteq d_1 \sqsubseteq d_2 \sqsubseteq \ldots$ in D

$$(\forall n \geq 0 \,.\, d_n \in S) \;\Rightarrow\; \Big(\bigsqcup_{n \geq 0} d_n\Big) \in S$$

If D is a domain, $S \subseteq D$ is called admissible iff it is a chain-closed subset of D and $\perp \in S$.

A property $\Phi(d)$ of elements $d \in D$ is called *chain-closed* (resp. *admissible*) iff $\{d \in D \mid \Phi(d)\}$ is a *chain-closed* (resp. *admissible*) subset of D.

$$\frac{\forall d.\ \Phi(d) \Rightarrow \Phi(fd)}{\Phi(fix\,f)}$$

Building chain-closed subsets (I)

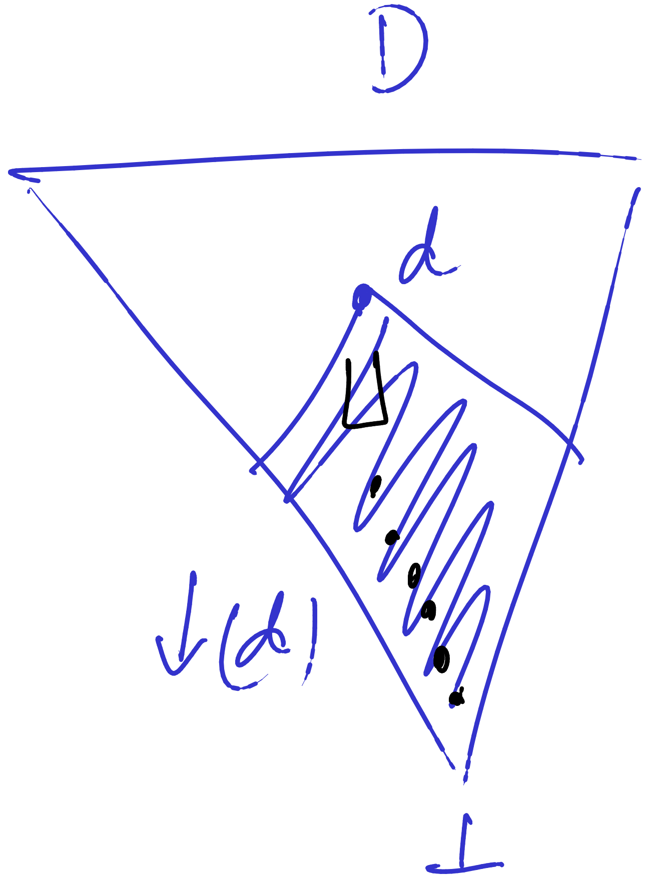
Let D, E be cpos.

Basic relations:

- For every $d \in D$, the subset

$$\downarrow(d) \stackrel{\text{def}}{=} \{\, x \in D \mid x \sqsubseteq d \,\}$$

of D is chain-closed.



$$\frac{x_i \sqsubseteq y_i}{\bigsqcup_i x_i \sqsubseteq \bigsqcup_i y_i} \quad \left(\begin{array}{l} x_i\ y_i \\ \text{chain} \end{array}\right)$$

Let D, E be cpos.

Basic relations:

- For every $d \in D$, the subset

$$\downarrow(d) \stackrel{\text{def}}{=} \{ x \in D \mid x \sqsubseteq d \}$$ in $D \times D$

  of D is chain-closed.

$(d_0, e_0) \sqsubseteq (d_1, e_1) \sqsubseteq \cdots \sqsubseteq (d_n, e_n) \sqsubseteq$

s.t. $d_i \sqsubseteq e_i$?

$\Longrightarrow \bigsqcup_n (d_n, e_n)$

$\quad\quad \| $

$\left(\bigsqcup_n d_n, \bigsqcup_n e_n\right)$

is s.t. $\bigsqcup_n d_n \sqsubseteq \bigsqcup_n e_n$

- The subsets

$$\{(x, y) \in D \times D \mid x \sqsubseteq y\}$$

  and

$$\{(x, y) \in D \times D \mid x = y\}$$

  of $D \times D$ are chain-closed.

Example (I): Least pre-fixed point property

Let D be a domain and let $f : D \to D$ be a continuous function.

$$\forall d \in D.\, f(d) \sqsubseteq d \implies \mathit{fix}(f) \sqsubseteq d$$

$x \in \downarrow(d) \Leftarrow)\quad x \sqsubseteq d \Rightarrow fx \sqsubseteq fd \sqsubseteq d \Longrightarrow fx \sqsubseteq d$

f mon $\qquad\qquad d$ a prefixed point

$$\Updownarrow$$

$fx \in \downarrow(d)$

$$\forall x \quad \frac{\dfrac{x \in \downarrow(d) \Rightarrow f(x) \in \downarrow(d)}{\mathit{fix}(f) \in \downarrow(d)}}{\mathit{fix}(f) \sqsubseteq d}$$

$S = \downarrow(d)$

Example (I): Least pre-fixed point property

Let D be a domain and let $f : D \to D$ be a continuous function.

$$\forall d \in D.\, f(d) \sqsubseteq d \implies \mathit{fix}(f) \sqsubseteq d$$

Proof by Scott induction.

Let $d \in D$ be a pre-fixed point of f. Then,

$$\begin{aligned} x \in {\downarrow}(d) \quad & \implies \quad x \sqsubseteq d \\ & \implies \quad f(x) \sqsubseteq f(d) \\ & \implies \quad f(x) \sqsubseteq d \\ & \implies \quad f(x) \in {\downarrow}(d) \end{aligned}$$

Hence,

$$\mathit{fix}(f) \in {\downarrow}(d) \ .$$

chain closed

$$\begin{tikzcd} f^{-1}(S) \arrow[d, phantom, "\cap"] & S \arrow[l, Rightarrow] \arrow[d, phantom, "\cap"] \\ D \arrow[r, "f"'] & E \end{tikzcd}$$

chain closed

Building chain-closed subsets (II)

Inverse image:

Let $f : D \to E$ be a continuous function.

If S is a chain-closed subset of E then the inverse image

$$f^{-1}S = \{x \in D \mid f(x) \in S\}$$

is an chain-closed subset of D.

Example (II)

$\forall d.\ f(gd) \sqsubseteq g(fd)$

Let D be a domain and let $f, g : D \to D$ be continuous functions such that $f \circ g \sqsubseteq g \circ f$. Then,

$$f(\perp) \sqsubseteq g(\perp) \implies \mathit{fix}(f) \sqsubseteq \mathit{fix}(g) .$$

$ff\perp \sqsubseteq fg\perp$

$\sqsubseteq gf\perp$

$\sqsubseteq gg\perp$ ✓

?

$ff\perp \sqsubseteq gg\perp$

$gf\perp \sqsubseteq gg\perp$

✓

$\perp \sqsubseteq \perp,\ f(\perp) \sqsubseteq g(\perp),\ \ldots,\ f^k(\perp) \sqsubseteq g^k(\perp),\ \ldots$

$\forall n \quad f^n(\perp) \sqsubseteq g^n(\perp)$

$\bigsqcup_n f^n(\perp) \sqsubseteq \bigsqcup_n g^n(\perp)$

$\mathit{fix}(f) \sqsubseteq \mathit{fix}(g)$

Example (II)

(*) Hence $fix(g)$ is a prefixed point of f and so $fix(f) \sqsubseteq fix(g)$

Let D be a domain and let $f, g : D \to D$ be continuous functions such that $f \circ g \sqsubseteq g \circ f$. Then,

$$f(\perp) \sqsubseteq g(\perp) \implies fix(f) \sqsubseteq fix(g) \,.$$

Proof by Scott induction.

Consider the admissible property $\Phi(x) \equiv (f(x) \sqsubseteq g(x))$ of D.

the inverse image of an admissible set

Since

$$f(x) \sqsubseteq g(x) \Rightarrow g(f(x)) \sqsubseteq g(g(x)) \Rightarrow f(g(x)) \sqsubseteq g(g(x))$$

$\Phi(x)$ $\qquad$ $\Phi(g\,x)$

we have that

$$f(fix(g)) \sqsubseteq g(fix(g)) \,.$$

$= fix(g)$ (*)

$\Longrightarrow \Phi(fix\, g)$

Building chain-closed subsets (III)

Logical operations:

- If $S, T \subseteq D$ are chain-closed subsets of D then

$$S \cup T \qquad \text{and} \qquad S \cap T$$

  are chain-closed subsets of D.

- If $\{ S_i \}_{i \in I}$ is a family of chain-closed subsets of D indexed by a set I, then $\bigcap_{i \in I} S_i$ is a chain-closed subset of D.

- If a property $P(x, y)$ determines a chain-closed subset of $D \times E$, then the property $\forall x \in D.\ P(x, y)$ determines a chain-closed subset of E.

Example (III): Partial correctness

Let $\mathcal{F} : State \rightharpoonup State$ be the denotation of

$$\textbf{while } X > 0 \textbf{ do } (Y := X * Y; X := X - 1) \ .$$

partial correctness

For all $x, y \geq 0$,

$$\mathcal{F}[X \mapsto x, Y \mapsto y] \downarrow$$

$$\implies \mathcal{F}[X \mapsto x, Y \mapsto y] = [X \mapsto 0, Y \mapsto !x \cdot y].$$

x factorial

Recall that

$$\mathcal{F} = \mathit{fix}(f)$$

where $f : (State \rightharpoonup State) \to (State \rightharpoonup State)$ is given by

$$f(w) = \lambda(x, y) \in State. \begin{cases} (x, y) & \text{if } x \leq 0 \\ w(x - 1, x \cdot y) & \text{if } x > 0 \end{cases}$$

Proof by Scott induction.

We consider the admissible subset of $(State \rightharpoonup State)$ given by

$$S = \left\{ w \;\middle|\; \begin{array}{l} \forall x, y \geq 0. \\ \quad w[X \mapsto x, Y \mapsto y]\downarrow \\ \qquad \Rightarrow w[X \mapsto x, Y \mapsto y] = [X \mapsto 0, Y \mapsto !x \cdot y] \end{array} \right\}$$

and show that

$$w \in S \implies f(w) \in S \; .$$

Topic 5

PCF

PCF syntax

Types

$$\tau ::= nat \mid bool \mid \tau \to \tau$$

PCF syntax

Types

$$\tau ::= nat \mid bool \mid \tau \to \tau$$

Expressions

$$M \quad ::= \quad \mathbf{0} \mid \mathbf{succ}(M) \mid \mathbf{pred}(M)$$

PCF syntax

Types

$$\tau ::= nat \mid bool \mid \tau \to \tau$$

Expressions

$$\begin{aligned} M \quad ::= \quad & \mathbf{0} \mid \mathbf{succ}(M) \mid \mathbf{pred}(M) \\ \mid \quad & \mathbf{true} \mid \mathbf{false} \mid \mathbf{zero}(M) \end{aligned}$$

PCF syntax

Types

$$\tau ::= nat \mid bool \mid \tau \to \tau$$

Expressions

$$\begin{array}{rcl} M & ::= & \mathbf{0} \mid \mathbf{succ}(M) \mid \mathbf{pred}(M) \\ & \mid & \mathbf{true} \mid \mathbf{false} \mid \mathbf{zero}(M) \\ & \mid & x \mid \mathbf{if}\ M\ \mathbf{then}\ M\ \mathbf{else}\ M \end{array}$$

PCF syntax

Types

$$\tau ::= nat \mid bool \mid \tau \to \tau$$

Expressions

$$\begin{array}{rcl} M & ::= & \mathbf{0} \mid \mathbf{succ}(M) \mid \mathbf{pred}(M) \\ & \mid & \mathbf{true} \mid \mathbf{false} \mid \mathbf{zero}(M) \\ & \mid & x \mid \mathbf{if}\ M\ \mathbf{then}\ M\ \mathbf{else}\ M \\ & \mid & \mathbf{fn}\, x : \tau \,.\, M \mid M\ M \mid \mathbf{fix}(M) \end{array}$$

where $x \in \mathbb{V}$, an infinite set of variables.

PCF syntax

Types

$$\tau ::= nat \mid bool \mid \tau \to \tau$$

Expressions

$$\begin{array}{rcl} M & ::= & \mathbf{0} \mid \mathbf{succ}(M) \mid \mathbf{pred}(M) \\ & \mid & \mathbf{true} \mid \mathbf{false} \mid \mathbf{zero}(M) \\ & \mid & x \mid \mathbf{if}\ M\ \mathbf{then}\ M\ \mathbf{else}\ M \\ & \mid & \mathbf{fn}\, x : \tau \,.\, M \mid M\ M \mid \mathbf{fix}(M) \end{array}$$

where $x \in \mathbb{V}$, an infinite set of variables.

Technicality: We identify expressions up to α-conversion of bound variables (created by the $\mathbf{fn}$ expression-former): by definition a PCF term is an α-equivalence class of expressions.

PCF typing relation, $\Gamma \vdash M : \tau$

- Γ is a type environment, *i.e.* a finite partial function mapping variables to types (whose domain of definition is denoted $dom(\Gamma)$)
- M is a term
- τ is a type.

PCF typing relation, $\Gamma \vdash M : \tau$

- Γ is a type environment, *i.e.* a finite partial function mapping variables to types (whose domain of definition is denoted $dom(\Gamma)$)
- M is a term
- τ is a type.

Notation:

$M : \tau$ means M is closed and $\emptyset \vdash M : \tau$ holds.

$\mathrm{PCF}_\tau \stackrel{\text{def}}{=} \{M \mid M : \tau\}$.

PCF typing relation (sample rules)

$$(:_{\text{fn}}) \quad \frac{\Gamma[x \mapsto \tau] \vdash M : \tau'}{\Gamma \vdash \mathbf{fn}\, x : \tau \,.\, M : \tau \to \tau'} \quad \text{if } x \notin dom(\Gamma)$$

PCF typing relation (sample rules)

$$(:_{\mathrm{fn}})\quad \frac{\Gamma[x \mapsto \tau] \vdash M : \tau'}{\Gamma \vdash \mathbf{fn}\, x : \tau \,.\, M : \tau \to \tau'} \quad \text{if } x \notin dom(\Gamma)$$

$$(:_{\mathrm{app}})\quad \frac{\Gamma \vdash M_1 : \tau \to \tau' \quad \Gamma \vdash M_2 : \tau}{\Gamma \vdash M_1\, M_2 : \tau'}$$

PCF typing relation (sample rules)

$$(:_{\mathrm{fn}}) \quad \frac{\Gamma[x \mapsto \tau] \vdash M : \tau'}{\Gamma \vdash \mathbf{fn}\, x : \tau \,.\, M : \tau \to \tau'} \quad \text{if } x \notin dom(\Gamma)$$

$$(:_{\mathrm{app}}) \quad \frac{\Gamma \vdash M_1 : \tau \to \tau' \quad \Gamma \vdash M_2 : \tau}{\Gamma \vdash M_1\, M_2 : \tau'}$$

$$(:_{\mathrm{fix}}) \quad \frac{\Gamma \vdash M : \tau \to \tau}{\Gamma \vdash \mathbf{fix}(M) : \tau}$$

$h = fix(\lambda k. \lambda x. \lambda z.$ if (zero z) then $f\ x$
else $g\ x$ (pred z) ($k\ x$ (pred z)))

Partial recursive functions in PCF

- Primitive recursion.

$$\begin{cases} h(x,0) = f(x) \\ h(x,y+1) = g(x,y,h(x,y)) \end{cases}$$

$$\begin{cases} h\ x\ 0 = f x \\ h\ x\ (y+1) = g\ x\ y\ (h\ x\ y) \end{cases}$$

zero(z)

pred(z)

$h\ x\ z =$ if $z=0$ then $f(x)$
else $g\ x\ (z-1)\ (h\ x\ (z-1))$

Partial recursive functions in PCF

- Primitive recursion.

$$\begin{cases} h(x, 0) = f(x) \\ h(x, y+1) = g(x, y, h(x, y)) \end{cases}$$

- Minimisation.

$$m(x) \;=\; \text{the least } y \geq 0 \text{ such that } k(x, y) = 0$$