Denotational Semantics

Ioannis Markakis

Lectures for Part II CST 2025/2026

The language Pcf

The language Pcf

Syntax

Types:

$$\tau ::= \texttt{nat} \mid \texttt{bool} \mid \tau \to \tau$$

Types: $\tau ::= \mathtt{nat} \mid \mathtt{bool} \mid \tau \to \tau$

Terms:

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

Types and terms of Pcf

Types:

$$\tau ::= \texttt{nat} \mid \texttt{bool} \mid \tau \to \tau$$

Terms:

$$\begin{aligned} t \quad ::= \quad & \texttt{0} \mid \texttt{succ}\,(t) \mid \texttt{pred}\,(t) \mid \\ & \texttt{true} \mid \texttt{false} \mid \texttt{zero?}\,(t) \mid \texttt{if}\ t\ \texttt{then}\ t\ \texttt{else}\ t \\ & x \mid \texttt{fun}\,x{:}\,\tau.\,t \mid t\ t \mid \texttt{fix}(t) \end{aligned}$$

- λ-calculus + base types/functions + `fix`
- tiny ML (without references, ADTs, polymorphism...)

Variables and terms: up to α-equivalence

Variables, substitutions and contexts

Variables and terms: up to α-equivalence

Substitution: $t[u/x]$

Variables and terms: up to α-equivalence

Substitution: $t[u/x]$

Typing contexts: $\Gamma ::= \cdot \mid \Gamma, x\colon \tau$

Variables, substitutions and contexts

Variables and terms: up to α-equivalence

Substitution: $t[u/x]$

Typing contexts: $\Gamma ::= \cdot \mid \Gamma, x\colon \tau$

- partial maps from variable to types
- finite lists $x_1\colon \tau_1, \ldots, x_n\colon \tau_n$

Typing for Pcf (I)

$\boxed{\Gamma \vdash t : \tau}$ The term t has type τ in context Γ

$$\textsc{Zero}\ \frac{}{\Gamma \vdash \mathtt{0} : \mathtt{nat}} \qquad \textsc{Succ}\ \frac{\Gamma \vdash t : \mathtt{nat}}{\Gamma \vdash \mathtt{succ}\,(t) : \mathtt{nat}} \qquad \textsc{Pred}\ \frac{\Gamma \vdash t : \mathtt{nat}}{\Gamma \vdash \mathtt{pred}\,(t) : \mathtt{nat}}$$

$\boxed{\Gamma \vdash t : \tau}$ The term t has type τ in context Γ

$$\textsc{Zero}\ \frac{}{\Gamma \vdash \mathtt{0} : \mathtt{nat}} \qquad \textsc{Succ}\ \frac{\Gamma \vdash t : \mathtt{nat}}{\Gamma \vdash \mathtt{succ}\,(t) : \mathtt{nat}} \qquad \textsc{Pred}\ \frac{\Gamma \vdash t : \mathtt{nat}}{\Gamma \vdash \mathtt{pred}\,(t) : \mathtt{nat}}$$

$$\textsc{True}\ \frac{}{\Gamma \vdash \mathtt{true} : \mathtt{bool}} \qquad \textsc{False}\ \frac{}{\Gamma \vdash \mathtt{false} : \mathtt{bool}} \qquad \textsc{IsZ}\ \frac{\Gamma \vdash t : \mathtt{nat}}{\Gamma \vdash \mathtt{zero?}\,(t) : \mathtt{bool}}$$

$$\textsc{If}\ \frac{\Gamma \vdash b : \mathtt{bool} \qquad \Gamma \vdash t : \tau \qquad \Gamma \vdash t' : \tau}{\Gamma \vdash \mathtt{if}\ b\ \mathtt{then}\ t\ \mathtt{else}\ t' : \tau}$$

Typing for Pcf (II)

$$\textsc{Var}\ \frac{\Gamma(x) = \tau}{\Gamma \vdash x : \tau} \qquad \textsc{Fun}\ \frac{\Gamma, x{:}\sigma \vdash t : \tau}{\Gamma \vdash \mathtt{fun}\, x{:}\sigma.\, t : \sigma \to \tau} \qquad \textsc{App}\ \frac{\Gamma \vdash f : \sigma \to \tau \qquad \Gamma \vdash u : \sigma}{\Gamma \vdash f\, u : \tau}$$

$$\textsc{Fix}\ \frac{\Gamma \vdash f : \tau \to \tau}{\Gamma \vdash \mathtt{fix}(f) : \tau}$$

Typing for Pcf (II)

$$\textsc{Var}\ \frac{\Gamma(x) = \tau}{\Gamma \vdash x : \tau} \qquad \textsc{Fun}\ \frac{\Gamma, x{:}\sigma \vdash t : \tau}{\Gamma \vdash \mathtt{fun}\, x{:}\sigma.\, t : \sigma \to \tau} \qquad \textsc{App}\ \frac{\Gamma \vdash f : \sigma \to \tau \qquad \Gamma \vdash u : \sigma}{\Gamma \vdash f\, u : \tau}$$

$$\textsc{Fix}\ \frac{\Gamma \vdash f : \tau \to \tau}{\Gamma \vdash \mathtt{fix}(f) : \tau}$$

$$\textsc{Pcf}_{\Gamma,\tau} \stackrel{\text{def}}{=} \{t \mid \Gamma \vdash t : \tau\} \qquad\qquad \textsc{Pcf}_{\tau} \stackrel{\text{def}}{=} \textsc{Pcf}_{\cdot,\tau}$$

Typing for Pcf (II)

$$\textsc{Var}\ \frac{\Gamma(x) = \tau}{\Gamma \vdash x : \tau} \qquad \textsc{Fun}\ \frac{\Gamma, x{:}\sigma \vdash t : \tau}{\Gamma \vdash \mathtt{fun}\, x{:}\sigma.\, t : \sigma \to \tau} \qquad \textsc{App}\ \frac{\Gamma \vdash f : \sigma \to \tau \qquad \Gamma \vdash u : \sigma}{\Gamma \vdash f\, u : \tau}$$

$$\textsc{Fix}\ \frac{\Gamma \vdash f : \tau \to \tau}{\Gamma \vdash \mathtt{fix}(f) : \tau}$$

$$\textsc{Pcf}_{\Gamma,\tau} \overset{\mathrm{def}}{=} \{t \mid \Gamma \vdash t : \tau\} \qquad\qquad \textsc{Pcf}_{\tau} \overset{\mathrm{def}}{=} \textsc{Pcf}_{\cdot,\tau}$$

The only programs we care about are the well-typed ones!

Lemma

If $\Gamma \vdash t : \tau$ and $\Gamma, x : \tau \vdash t' : \tau'$ both hold, then so does $\Gamma \vdash t'[t/x] : \tau'$.

The language Pcf

Operational semantics

Pcf values

Values: $v ::= \underbrace{\texttt{0} \mid \texttt{succ}\,(v)}_{\underline{n}} \mid \texttt{true} \mid \texttt{false} \mid \underbrace{\texttt{fun}\, x{:}\,\tau.\, t}_{\text{All functions } (<\, \texttt{fun}\, >)}$

Pcf values

Values: $v ::= \underbrace{\mathtt{0} \mid \mathtt{succ}\,(v)}_{\underline{n}} \mid \mathtt{true} \mid \mathtt{false} \mid \underbrace{\mathtt{fun}\, x{:}\,\tau.\, t}_{\text{All functions } (<\, \mathtt{fun}\, >)}$

We will only evaluate closed term to values.

Pcf evaluation

$$\textsc{Val}\ \frac{\vdash v : \tau}{v \Downarrow_\tau v}$$

PCF evaluation

$$\textsc{Val}\ \frac{\vdash v : \tau}{v \Downarrow_\tau v} \qquad \textsc{Succ}\ \frac{t \Downarrow_{\mathtt{nat}} v}{\mathtt{succ}\,(t) \Downarrow_{\mathtt{nat}} \mathtt{succ}\,(v)} \qquad \textsc{Pred}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{succ}\,(v)}{\mathtt{pred}\,(t) \Downarrow_{\mathtt{nat}} v}$$

PCF evaluation

$$\textsc{Val}\ \frac{\vdash v : \tau}{v \Downarrow_\tau v} \qquad \textsc{Succ}\ \frac{t \Downarrow_{\mathtt{nat}} v}{\mathtt{succ}\,(t) \Downarrow_{\mathtt{nat}} \mathtt{succ}\,(v)} \qquad \textsc{Pred}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{succ}\,(v)}{\mathtt{pred}\,(t) \Downarrow_{\mathtt{nat}} v}$$

$$\textsc{ZeroZ}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{0}}{\mathtt{zero?}\,(t) \Downarrow_{\mathtt{bool}} \mathtt{true}} \qquad \textsc{ZeroS}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{succ}\,(v)}{\mathtt{zero?}\,(t) \Downarrow_{\mathtt{bool}} \mathtt{false}}$$

PCF evaluation

$$\textsc{Val}\ \frac{\vdash v : \tau}{v \Downarrow_\tau v} \qquad \textsc{Succ}\ \frac{t \Downarrow_{\mathtt{nat}} v}{\mathtt{succ}\,(t) \Downarrow_{\mathtt{nat}} \mathtt{succ}\,(v)} \qquad \textsc{Pred}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{succ}\,(v)}{\mathtt{pred}\,(t) \Downarrow_{\mathtt{nat}} v}$$

$$\textsc{ZeroZ}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{0}}{\mathtt{zero?}\,(t) \Downarrow_{\mathtt{bool}} \mathtt{true}} \qquad \textsc{ZeroS}\ \frac{t \Downarrow_{\mathtt{nat}} \mathtt{succ}\,(v)}{\mathtt{zero?}\,(t) \Downarrow_{\mathtt{bool}} \mathtt{false}}$$

$$\textsc{IfT}\ \frac{b \Downarrow_{\mathtt{bool}} \mathtt{true} \qquad t_1 \Downarrow_\tau v}{\mathtt{if}\ b\ \mathtt{then}\ t_1\ \mathtt{else}\ t_2 \Downarrow_\tau v} \qquad \textsc{IfF}\ \frac{b \Downarrow_{\mathtt{bool}} \mathtt{false} \qquad t_2 \Downarrow_\tau v}{\mathtt{if}\ b\ \mathtt{then}\ t_1\ \mathtt{else}\ t_2 \Downarrow_\tau v}$$

$$\textsc{Val}\ \frac{\vdash v : \tau}{v \Downarrow_\tau v} \qquad \textsc{Succ}\ \frac{t \Downarrow_{\texttt{nat}} v}{\texttt{succ}\,(t) \Downarrow_{\texttt{nat}} \texttt{succ}\,(v)} \qquad \textsc{Pred}\ \frac{t \Downarrow_{\texttt{nat}} \texttt{succ}\,(v)}{\texttt{pred}\,(t) \Downarrow_{\texttt{nat}} v}$$

$$\textsc{ZeroZ}\ \frac{t \Downarrow_{\texttt{nat}} \texttt{0}}{\texttt{zero?}\,(t) \Downarrow_{\texttt{bool}} \texttt{true}} \qquad \textsc{ZeroS}\ \frac{t \Downarrow_{\texttt{nat}} \texttt{succ}\,(v)}{\texttt{zero?}\,(t) \Downarrow_{\texttt{bool}} \texttt{false}}$$

$$\textsc{IfT}\ \frac{b \Downarrow_{\texttt{bool}} \texttt{true} \qquad t_1 \Downarrow_\tau v}{\texttt{if}\ b\ \texttt{then}\ t_1\ \texttt{else}\ t_2 \Downarrow_\tau v} \qquad \textsc{IfF}\ \frac{b \Downarrow_{\texttt{bool}} \texttt{false} \qquad t_2 \Downarrow_\tau v}{\texttt{if}\ b\ \texttt{then}\ t_1\ \texttt{else}\ t_2 \Downarrow_\tau v}$$

$$\textsc{Fun}\ \frac{t \Downarrow_{\sigma\to\tau} \texttt{fun}\ x{:}\,\sigma.\, t' \qquad t'[u/x] \Downarrow_\tau v}{t\ u \Downarrow_\tau v} \qquad \textsc{Fix}\ \frac{t\,(\texttt{fix}(t)) \Downarrow_\tau v}{\texttt{fix}(t) \Downarrow_\tau v}$$

Example: Addition

Addition can be defined by recursion on the second argument.
It must satisfy the fixed point equation for $x, y : \mathtt{nat}$:

$$\mathtt{plus}\, x\, y = \mathtt{if\ zero?}(y)\ \mathtt{then}\, x\, \mathtt{else\ succ}(\mathtt{plus}\, x\ \mathtt{pred}(y))$$

Example: Addition

Addition can be defined by recursion on the second argument.
It must satisfy the fixed point equation for $x, y : \mathtt{nat}$:

$$\mathtt{plus}\; x\; y = \mathtt{if\ zero?}(y)\ \mathtt{then}\ x\ \mathtt{else\ succ}(\mathtt{plus}\; x\ \mathtt{pred}(y))$$

We may then define in context $x : \mathtt{nat}$:

$$\begin{aligned}\mathtt{plus}_x \overset{\text{def}}{=} {} & \mathtt{fix}(\mathtt{fun}(p{:}\,\mathtt{nat} \to \mathtt{nat})(y{:}\,\mathtt{nat}). \\ & \quad \mathtt{if\ zero?}(y)\ \mathtt{then}\ x\ \mathtt{else\ succ}(p\ \mathtt{pred}(y)))\end{aligned}$$

and then bind the variable x to get:

$$\mathtt{plus} = \mathtt{fun}\, x{:}\,\mathtt{nat}.\, \mathtt{plus}_x \tag{1}$$

$$\textsc{Fun}\ \frac{\texttt{plus} \Downarrow \texttt{fun x: nat. plus}_\texttt{x} \qquad \texttt{plus}_{\underline{3}}\ \underline{1} \Downarrow \underline{4}}{\texttt{plus}\ \underline{3}\ \underline{1} \Downarrow_{\texttt{nat}} \underline{4}}$$

$$\textsc{Fun}\ \dfrac{\texttt{plus} \Downarrow \texttt{fun}\ \texttt{x}{:}\ \texttt{nat}.\ \texttt{plus}_{\texttt{x}} \qquad \texttt{plus}_{\underline{3}}\ \underline{1} \Downarrow \underline{4}}{\texttt{plus}\ \underline{3}\ \underline{1} \Downarrow_{\texttt{nat}} \underline{4}}$$

$$\textsc{Fix}\ \dfrac{\textsc{Fun}\ \dfrac{\textsc{Val}\ \dfrac{}{(\texttt{fun}\ p{:}\ \texttt{nat} \to \texttt{nat}.\ \ldots) \Downarrow \ldots} \qquad \textsc{Val}\ \dfrac{}{(\texttt{fun}\ y{:}\ \texttt{nat}.\ \ldots)[\texttt{plus}_x / p] \Downarrow r_x}}{(\texttt{fun}(p{:}\ \texttt{nat} \to \texttt{nat})(y{:}\ \texttt{nat}).\ \ldots)\ \texttt{plus}_x \Downarrow r_x}}{\texttt{plus}_x \Downarrow \underbrace{\texttt{fun}\ y{:}\ \texttt{nat}.\ \texttt{if}\ \ \texttt{zero?}\,(y)\ \ \texttt{then}\ x\ \texttt{else}\ \ \texttt{succ}\,(\texttt{plus}_x\ \ \texttt{pred}\,(y))}_{r_x}}$$

$$
\textsc{Fun}\;\dfrac{\mathtt{plus}_{\underline{3}} \Downarrow \underline{r_3} \qquad \textsc{IfF}\;\dfrac{\textsc{ZeroS}\;\dfrac{\textsc{Val}\;\dfrac{}{\underline{1} \Downarrow \underline{1}}}{\mathtt{zero?}\left(\underline{1}\right) \Downarrow \mathtt{false}} \qquad \textsc{Succ}\;\dfrac{\dfrac{\dfrac{\textsc{ZeroZ}\;\dfrac{\textsc{Pred}\;\dfrac{\cdots}{\mathtt{pred}\left(\underline{1}\right) \Downarrow \mathtt{0}}}{\mathtt{zero?}\left(\mathtt{pred}\left(\underline{1}\right)\right) \Downarrow \mathtt{true}}}{\cdots}}{\mathtt{plus}_{\underline{3}}\ \mathtt{pred}\left(\underline{1}\right) \Downarrow \underline{3}}}{\mathtt{succ}\left(\mathtt{plus}_{\underline{3}}\ \mathtt{pred}\left(\underline{1}\right)\right) \Downarrow \underline{4}}}{\mathtt{if}\ \mathtt{zero?}\left(\underline{1}\right)\ \mathtt{then}\ \underline{3}\ \mathtt{else}\ \mathtt{succ}\left(\mathtt{plus}_{\underline{3}}\ \mathtt{pred}\left(\underline{1}\right)\right) \Downarrow \underline{4}}}{\mathtt{plus}_{\underline{3}}\ \underline{1} \Downarrow_{\mathtt{nat}} \underline{4}}
$$

Divergence ($t \Uparrow_\tau$):

$$t : \tau \quad \wedge \quad \nexists v.\, t \Downarrow_\tau v$$

Divergence ($t \Uparrow_\tau$):

$$t : \tau \quad \wedge \quad \nexists v.\, t \Downarrow_\tau v$$

$$\Omega_\tau \stackrel{\text{def}}{=} \mathtt{fix}(\mathtt{fun}\, x{:}\tau.\, x)$$

$$\Omega_\tau \Uparrow_\tau \qquad \text{(diverges)}$$

Divergence ($t \Uparrow_\tau$):

$$t : \tau \quad \wedge \quad \nexists v.\, t \Downarrow_\tau v$$

$$\Omega_\tau \stackrel{\text{def}}{=} \mathtt{fix}(\mathtt{fun}\, x{:}\, \tau.\, x)$$

$$\Omega_\tau \Uparrow_\tau \qquad \text{(diverges)}$$

$$\dfrac{\dfrac{\mathtt{fun}\, x{:}\, \tau.\, x \Downarrow \mathtt{fun}\, x{:}\, \tau.\, x \qquad \overset{\mathcal{P}}{\mathtt{fix}(\mathtt{fun}\, x{:}\, \tau.\, x) \Downarrow v}}{(\mathtt{fun}\, x{:}\, \tau.\, x)\,(\mathtt{fix}(\mathtt{fun}\, x{:}\, \tau.\, x)) \Downarrow v}}{\mathtt{fix}(\mathtt{fun}\, x{:}\, \tau.\, x) \Downarrow v}$$

Call-by-name and call-by-value

$$\textsc{Fun-CBN}\ \frac{t \Downarrow_{\sigma\to\tau} \mathtt{fun}\, x{:}\sigma.\, t' \qquad t'[u/x] \Downarrow_\tau v}{t\ u \Downarrow_\tau v}$$

$$\textsc{Fun-CBV}\ \frac{t \Downarrow_{\sigma\to\tau} \mathtt{fun}\, x{:}\sigma.\, t' \qquad u \Downarrow_\sigma v' \qquad t'[v'/x] \Downarrow_\tau v}{t\ u \Downarrow_\tau v}$$

$$\textsc{Fun-CBN}\ \frac{t \Downarrow_{\sigma\to\tau} \mathtt{fun}\, x{:}\, \sigma.\, t' \qquad t'[u/x] \Downarrow_\tau v}{t\ u \Downarrow_\tau v}$$

$$\textsc{Fun-CBV}\ \frac{t \Downarrow_{\sigma\to\tau} \mathtt{fun}\, x{:}\, \sigma.\, t' \qquad u \Downarrow_\sigma v' \qquad t'[v'/x] \Downarrow_\tau v}{t\ u \Downarrow_\tau v}$$

What does $(\mathtt{fun}\, x{:}\, \mathtt{nat}.\, \mathtt{0})\ \Omega_{\mathtt{nat}}$ denote?

Call-by-name and call-by-value

$$\textsc{Fun-CBN}\ \frac{t \Downarrow_{\sigma\to\tau} \mathtt{fun}\, x{:}\,\sigma.\, t' \qquad t'[u/x] \Downarrow_\tau v}{t\ u \Downarrow_\tau v}$$

$$\textsc{Fun-CBV}\ \frac{t \Downarrow_{\sigma\to\tau} \mathtt{fun}\, x{:}\,\sigma.\, t' \qquad u \Downarrow_\sigma v' \qquad t'[v'/x] \Downarrow_\tau v}{t\ u \Downarrow_\tau v}$$

What does $(\mathtt{fun}\, x{:}\,\mathtt{nat}.\, \mathtt{0})\ \Omega_{\mathtt{nat}}$ denote?

In call-by-value, all functions are **strict**... but the least-fixed points of a strict function is **always $\perp$**!

Small-step $t \rightsquigarrow_\tau u$:

$$\frac{}{(\mathsf{fun}\; x : \sigma .\, t)\; u \rightsquigarrow_\tau t[u/x]} \qquad \frac{t \rightsquigarrow_{\sigma \to \tau} t'}{t\; u \rightsquigarrow_\tau t'\; u} \qquad \dots$$

Small-step $t \rightsquigarrow_\tau u$:

$$\frac{}{(\mathsf{fun}\, x : \sigma .\, t)\; u \rightsquigarrow_\tau t[u/x]} \qquad \frac{t \rightsquigarrow_{\sigma \to \tau} t'}{t\; u \rightsquigarrow_\tau t'\; u} \qquad \dots$$

We have $t \Downarrow_\tau v$ iff $t \rightsquigarrow_\tau^\star u$.

PCF is Turing-complete: for every partial recursive function ϕ, there is a PCF term $\underline{\phi} \in \mathrm{PCF}_{\mathtt{nat} \to \mathtt{nat}}$ such that for all $n \in \mathbb{N}$, if $\phi(n)$ is defined then $\underline{\phi}\ \underline{n} \Downarrow_{\mathtt{nat}} \underline{\phi(n)}$.

Turing-completeness

PCF is Turing-complete: for every partial recursive function ϕ, there is a PCF term $\underline{\phi} \in \text{PCF}_{\mathtt{nat} \to \mathtt{nat}}$ such that for all $n \in \mathbb{N}$, if $\phi(n)$ is defined then $\underline{\phi}\ \underline{n} \Downarrow_{\mathtt{nat}} \underline{\phi(n)}$.

(Later on: $\phi = [\![\underline{\phi}]\!]$).

Evaluation in PCF is deterministic: if both $t \Downarrow_\tau v$ and $t \Downarrow_\tau v'$ hold, then $v = v'$.

Determinism

Evaluation in $\textsc{Pcf}$ is **deterministic**: if both $t \Downarrow_\tau v$ and $t \Downarrow_\tau v'$ hold, then $v = v'$.

By (rule) induction on evaluation $\Downarrow$:

$$P(t, \tau, v) \quad \stackrel{\text{def}}{=} \quad \forall v' \in \textsc{Pcf}_\tau \,.(t \Downarrow_\tau v' \Rightarrow v = v')$$

Intuition: there is always exactly one rule which applies.

The language Pcf

Contextual equivalence

Contextual equivalence – informal

Two phrases of a programming language are contextually equivalent if any occurrences of the first phrase in a complete program can be replaced by the second phrase without affecting the observable results of executing the program.

Contextual equivalence – informal

Two phrases of a programming language are **contextually equivalent** if any occurrences of the first phrase in a **complete program** can be replaced by the second phrase without affecting the **observable results** of executing the program.

The intuitive notion of **program equivalence** for programmers.

Two phrases of a programming language are **contextually equivalent** if any occurrences of the first phrase in a **complete program** can be replaced by the second phrase without affecting the **observable results** of executing the program.

The intuitive notion of **program equivalence** for programmers.

But what's a complete program? What's an observable result?

Evaluation contexts

"Term with a hole":

$$\begin{aligned}\mathcal{C} \quad ::= \quad & - \mid \texttt{succ}\,(\mathcal{C}) \mid \texttt{pred}\,(\mathcal{C}) \mid \texttt{zero?}\,(\mathcal{C}) \mid \\ & \texttt{if}\ \mathcal{C}\ \texttt{then}\ t\ \texttt{else}\ t \mid \texttt{if}\ t\ \texttt{then}\ \mathcal{C}\ \texttt{else}\ t \mid \texttt{if}\ t\ \texttt{then}\ t\ \texttt{else}\ \mathcal{C} \mid \\ & \texttt{fun}\, x{:}\,\tau.\, \mathcal{C} \mid \mathcal{C}\ t \mid t\ \mathcal{C} \mid \texttt{fix}(\mathcal{C})\end{aligned}$$

"Term with a hole":

$$
\begin{array}{rcl}
\mathcal{C} & ::= & - \mid \mathtt{succ}\,(\mathcal{C}) \mid \mathtt{pred}\,(\mathcal{C}) \mid \mathtt{zero?}\,(\mathcal{C}) \mid \\
& & \mathtt{if}\ \mathcal{C}\ \mathtt{then}\ t\ \mathtt{else}\ t \mid \mathtt{if}\ t\ \mathtt{then}\ \mathcal{C}\ \mathtt{else}\ t \mid \mathtt{if}\ t\ \mathtt{then}\ t\ \mathtt{else}\ \mathcal{C} \mid \\
& & \mathtt{fun}\ x{:}\tau.\ \mathcal{C} \mid \mathcal{C}\ t \mid t\ \mathcal{C} \mid \mathtt{fix}(\mathcal{C})
\end{array}
$$

Typing extended to evaluation contexts: $\Gamma \vdash_{\Delta,\sigma} \mathcal{C} : \tau$.

"Term with a hole":

$$\begin{array}{rcl}\mathcal{C} & ::= & - \mid \texttt{succ}\,(\mathcal{C}) \mid \texttt{pred}\,(\mathcal{C}) \mid \texttt{zero?}\,(\mathcal{C}) \mid \\ & & \texttt{if}\ \mathcal{C}\ \texttt{then}\ t\ \texttt{else}\ t \mid \texttt{if}\ t\ \texttt{then}\ \mathcal{C}\ \texttt{else}\ t \mid \texttt{if}\ t\ \texttt{then}\ t\ \texttt{else}\ \mathcal{C} \mid \\ & & \texttt{fun}\ x{:}\,\tau.\,\mathcal{C} \mid \mathcal{C}\ t \mid t\ \mathcal{C} \mid \texttt{fix}(\mathcal{C})\end{array}$$

Typing extended to evaluation contexts: $\Gamma \vdash_{\Delta,\sigma} \mathcal{C} : \tau$.

$$\frac{}{\Gamma \vdash_{\Gamma,\tau} - : \tau} \qquad \frac{\Gamma \vdash_{\Delta,\sigma} \mathcal{C} : \tau_1 \to \tau_2 \qquad \Gamma \vdash u : \tau_1}{\Gamma \vdash_{\Delta,\sigma} \mathcal{C}\ u : \tau_2} \qquad \dots$$

Given a type τ, a typing context Γ and terms $t, t' \in \textsc{Pcf}_{\Gamma,\tau}$, contextual equivalence, written $\Gamma \vdash t \cong_{\mathrm{ctx}} t' : \tau$ is defined to hold if for all evaluation contexts $\mathcal{C}$ such that $\cdot \vdash_{\Gamma,\tau} \mathcal{C} : \gamma$, where γ is `nat` or `bool`, and for all values $v \in \textsc{Pcf}_{\gamma}$,

$$\mathcal{C}[t] \Downarrow_{\gamma} v \Leftrightarrow \mathcal{C}[t'] \Downarrow_{\gamma} v.$$

When Γ is the empty context, we simply write $t \cong_{\mathrm{ctx}} t' : \tau$ for $\cdot \vdash t \cong_{\mathrm{ctx}} t' : \tau$.

Given a type τ, a typing context Γ and terms $t, t' \in \textsc{Pcf}_{\Gamma,\tau}$, contextual equivalence, written $\Gamma \vdash t \cong_{\mathrm{ctx}} t' : \tau$ is defined to hold if for all evaluation contexts $\mathcal{C}$ such that $\cdot \vdash_{\Gamma,\tau} \mathcal{C} : \gamma$, where γ is $\mathtt{nat}$ or $\mathtt{bool}$, and for all values $v \in \textsc{Pcf}_{\gamma}$,

$$\mathcal{C}[t] \Downarrow_{\gamma} v \Leftrightarrow \mathcal{C}[t'] \Downarrow_{\gamma} v.$$

When Γ is the empty context, we simply write $t \cong_{\mathrm{ctx}} t' : \tau$ for $\cdot \vdash t \cong_{\mathrm{ctx}} t' : \tau$.

Divergence is implicitly covered.

Denotational Semantics for Pcf

Denotational Semantics for Pcf

Introducing denotational semantics

The aims of denotational semantics

- a mapping of PCF types τ to domains $[\![\tau]\!]$;
- a mapping of PCF contexts Γ to domains $[\![\Gamma]\!]$;
- a mapping of closed, well-typed PCF terms $\cdot \vdash t : \tau$ to elements $[\![t]\!] \in [\![\tau]\!]$;
- denotation of open terms $\Gamma \vdash t : \tau$ will be continuous functions $[\![\Gamma]\!] \to [\![\tau]\!]$

- a mapping of PCF types τ to domains $[\![\tau]\!]$;
- a mapping of PCF contexts Γ to domains $[\![\Gamma]\!]$;
- a mapping of closed, well-typed PCF terms $\cdot \vdash t : \tau$ to elements $[\![t]\!] \in [\![\tau]\!]$;
- denotation of open terms $\Gamma \vdash t : \tau$ will be continuous functions $[\![\Gamma]\!] \to [\![\tau]\!]$

Compositionality: $[\![t]\!] = [\![t']\!] \Rightarrow [\![\mathcal{C}[t]]\!] = [\![\mathcal{C}[t']]\!]$.

Soundness: for any type τ, $t \Downarrow_\tau v \Rightarrow [\![t]\!] = [\![v]\!]$.

Adequacy: for $\gamma = \mathtt{bool}$ or $\mathtt{nat}$, if $t \in \mathrm{PCF}_\gamma$ and $[\![t]\!] = [\![v]\!]$ then $t \Downarrow_\gamma v$.

Adequacy for function types?

$$v \stackrel{\text{def}}{=} \mathtt{fun}\, x\colon \mathtt{nat}.\, (\mathtt{fun}\, y\colon \mathtt{nat}.\, y)\, \mathtt{0} \quad \text{and} \quad v' \stackrel{\text{def}}{=} \mathtt{fun}\, x\colon \mathtt{nat}.\, \mathtt{0}.$$

Proof principle: to show

$$t_1 \cong_{\mathrm{ctx}} t_2 : \tau$$

it suffices to establish

$$[\![t_1]\!] = [\![t_2]\!] \in [\![\tau]\!]$$

Proof principle: to show

$$t_1 \cong_{\mathrm{ctx}} t_2 : \tau$$

it suffices to establish

$$[\![t_1]\!] = [\![t_2]\!] \in [\![\tau]\!]$$

$$\begin{aligned}
\mathcal{C}[t_1] \Downarrow_{\mathtt{nat}} v &\Rightarrow [\![\mathcal{C}[t_1]]\!] = [\![v]\!] && \text{(soundness)} \\
&\Rightarrow [\![\mathcal{C}[t_2]]\!] = [\![v]\!] && \text{(compositionality on } [\![t_1]\!] = [\![t_2]\!]\text{)} \\
&\Rightarrow \mathcal{C}[t_2] \Downarrow_{\mathtt{nat}} v && \text{(adequacy)}
\end{aligned}$$

The power of denotational semantics

Proof principle: to show

$$t_1 \cong_{\text{ctx}} t_2 : \tau$$

it suffices to establish

$$[\![t_1]\!] = [\![t_2]\!] \in [\![\tau]\!]$$

$$\begin{aligned}
\mathcal{C}[t_1] \Downarrow_{\mathtt{nat}} v &\Rightarrow [\![\mathcal{C}[t_1]]\!] = [\![v]\!] && \text{(soundness)} \\
&\Rightarrow [\![\mathcal{C}[t_2]]\!] = [\![v]\!] && \text{(compositionality on } [\![t_1]\!] = [\![t_2]\!]\text{)} \\
&\Rightarrow \mathcal{C}[t_2] \Downarrow_{\mathtt{nat}} v && \text{(adequacy)}
\end{aligned}$$

and symmetrically for $\mathcal{C}[t_2] \Downarrow_{\mathtt{nat}} v \Rightarrow \mathcal{C}[t_1] \Downarrow_{\mathtt{nat}} v$, and similarly for `bool`.

Proof principle: to show

$$t_1 \cong_{\text{ctx}} t_2 : \tau$$

it suffices to establish

$$[\![t_1]\!] = [\![t_2]\!] \in [\![\tau]\!]$$

Denotational equality is sound, but is it complete?
Does equality in the model imply contextual equivalence?

Proof principle: to show

$$t_1 \cong_{\mathrm{ctx}} t_2 : \tau$$

it suffices to establish

$$[\![t_1]\!] = [\![t_2]\!] \in [\![\tau]\!]$$

Denotational equality is **sound**, but is it **complete**?
Does equality in the model imply contextual equivalence?

Full abstraction.

Denotational Semantics for Pcf

Definition

$$\llbracket \texttt{nat} \rrbracket \stackrel{\text{def}}{=} \mathbb{N}_\perp \qquad \text{(flat domain)}$$

$$\llbracket \texttt{bool} \rrbracket \stackrel{\text{def}}{=} \mathbb{B}_\perp \qquad \text{(flat domain)}$$

$$\llbracket \tau \to \tau' \rrbracket \stackrel{\text{def}}{=} \llbracket \tau \rrbracket \to \llbracket \tau' \rrbracket \qquad \text{(function domain)}$$

$$[\![\Gamma]\!] \stackrel{\text{def}}{=} \prod_{x\in\mathrm{dom}(\Gamma)} [\![\Gamma(x)]\!] \qquad \text{(environment)}$$

$$[\![\Gamma]\!] \stackrel{\text{def}}{=} \prod_{x \in \mathrm{dom}(\Gamma)} [\![\Gamma(x)]\!] \qquad \text{(environment)}$$

- $[\![\cdot]\!] = \mathbb{1}$ (one element set)
- $[\![x\colon \tau]\!] = (\{x\} \to [\![\tau]\!]) \cong [\![\tau]\!]$
- $[\![x_1\colon \tau_1, \dots, x_n\colon \tau_n]\!] \cong [\![\tau_1]\!] \times \cdots \times [\![\tau_n]\!]$

To every typing judgement

$$\Gamma \vdash t : \tau$$

we associate a continuous function

$$[\![\Gamma \vdash t : \tau]\!] : [\![\Gamma]\!] \to [\![\tau]\!]$$

between domains. In other words,

$$[\![-]\!] \colon \textsc{Pcf}_{\Gamma,\tau} \to [\![\Gamma]\!] \to [\![\tau]\!]$$

Denotation of operations on $\mathbb{B}$ and $\mathbb{N}$

$$\begin{aligned} \mathrm{succ} : \quad \mathbb{N} &\to \mathbb{N} \\ n &\mapsto n+1 \end{aligned}$$

$$\begin{aligned} \mathrm{pred} : \quad \mathbb{N} &\rightharpoonup \mathbb{N} \\ n+1 &\mapsto n \\ 0 & \quad \text{undefined} \end{aligned}$$

$$\begin{aligned} \mathrm{zero?} : \quad \mathbb{N} &\to \mathbb{B} \\ 0 &\mapsto \mathrm{true} \\ n+1 &\mapsto \mathrm{false} \end{aligned}$$

Denotation of operations on $\mathbb{B}$ and $\mathbb{N}$

$$\begin{aligned} \mathrm{succ}_\perp : \quad \mathbb{N}_\perp &\to \mathbb{N}_\perp \\ n &\mapsto n+1 \\ \perp &\mapsto \perp \end{aligned}$$

$$\begin{aligned} \mathrm{pred}_\perp : \quad \mathbb{N}_\perp &\to \mathbb{N}_\perp \\ n+1 &\mapsto n \\ 0 &\mapsto \perp \\ \perp &\mapsto \perp \end{aligned}$$

$$\begin{aligned} \mathrm{zero?}_\perp : \quad \mathbb{N}_\perp &\to \mathbb{B}_\perp \\ 0 &\mapsto \mathrm{true} \\ n+1 &\mapsto \mathrm{false} \\ \perp &\mapsto \perp \end{aligned}$$

Denotation of operations on $\mathbb{B}$ and $\mathbb{N}$

$$
\begin{aligned}
[\![\texttt{0}]\!](\rho) &\stackrel{\text{def}}{=} 0 && \in \mathbb{N}_\perp \\
[\![\texttt{true}]\!](\rho) &\stackrel{\text{def}}{=} \text{true} && \in \mathbb{B}_\perp \\
[\![\texttt{false}]\!](\rho) &\stackrel{\text{def}}{=} \text{false} && \in \mathbb{B}_\perp
\end{aligned}
$$

$$
\begin{aligned}
[\![\mathtt{0}]\!](\rho) &\stackrel{\text{def}}{=} 0 && \in \mathbb{N}_\perp \\
[\![\mathtt{true}]\!](\rho) &\stackrel{\text{def}}{=} \mathrm{true} && \in \mathbb{B}_\perp \\
[\![\mathtt{false}]\!](\rho) &\stackrel{\text{def}}{=} \mathrm{false} && \in \mathbb{B}_\perp
\end{aligned}
$$

$$
\begin{aligned}
[\![\mathtt{succ}\,(t)]\!](\rho) &\stackrel{\text{def}}{=} \mathrm{succ}_\perp([\![t]\!](\rho)) && \in \mathbb{N}_\perp \\
[\![\mathtt{pred}\,(t)]\!](\rho) &\stackrel{\text{def}}{=} \mathrm{pred}_\perp([\![t]\!](\rho)) && \in \mathbb{N}_\perp \\
[\![\mathtt{zero?}\,(t)]\!](\rho) &\stackrel{\text{def}}{=} \mathrm{zero?}_\perp([\![t]\!](\rho)) && \in \mathbb{B}_\perp
\end{aligned}
$$

$$
[\![\mathtt{succ}\,(t)]\!] = \mathrm{succ}_\perp \circ [\![t]\!]
$$

Denotation of operations on $\mathbb{B}$ and $\mathbb{N}$

$$
\begin{aligned}
[\![\texttt{0}]\!](\rho) &\stackrel{\text{def}}{=} 0 && \in \mathbb{N}_\perp \\
[\![\texttt{true}]\!](\rho) &\stackrel{\text{def}}{=} \text{true} && \in \mathbb{B}_\perp \\
[\![\texttt{false}]\!](\rho) &\stackrel{\text{def}}{=} \text{false} && \in \mathbb{B}_\perp
\end{aligned}
$$

$$
\begin{aligned}
[\![\texttt{succ}\,(t)]\!](\rho) &\stackrel{\text{def}}{=} \text{succ}_\perp([\![t]\!](\rho)) && \in \mathbb{N}_\perp \\
[\![\texttt{pred}\,(t)]\!](\rho) &\stackrel{\text{def}}{=} \text{pred}_\perp([\![t]\!](\rho)) && \in \mathbb{N}_\perp \\
[\![\texttt{zero?}\,(t)]\!](\rho) &\stackrel{\text{def}}{=} \text{zero?}_\perp([\![t]\!](\rho)) && \in \mathbb{B}_\perp
\end{aligned}
$$

$$
[\![\texttt{if}\ b\ \texttt{then}\ t\ \texttt{else}\ t']\!] \stackrel{\text{def}}{=} \text{if}\,([\![b]\!](\rho), [\![t]\!](\rho), [\![t']\!](\rho)) \in [\![\tau]\!]
$$

$$
[\![\texttt{if}\ b\ \texttt{then}\ t\ \texttt{else}\ t']\!] = \text{if} \circ \langle [\![b]\!], \langle [\![t]\!], [\![t']\!] \rangle \rangle
$$

$$[\![x]\!](\rho) \stackrel{\text{def}}{=} \rho(x) \qquad \in [\![\Gamma(x)]\!]$$

$$[\![x]\!](\rho) = \pi_x(\rho)$$

$$\begin{aligned} [\![x]\!](\rho) &\stackrel{\text{def}}{=} \rho(x) && \in [\![\Gamma(x)]\!] \\ [\![t_1\ t_2]\!](\rho) &\stackrel{\text{def}}{=} ([\![t_1]\!](\rho))\ ([\![t_2]\!](\rho)) \end{aligned}$$

$$[\![t_1\ t_2]\!] = \mathrm{eval} \circ \langle [\![t_1]\!], [\![t_2]\!] \rangle$$

Denotation of the λ-calculus operations

$$
\begin{aligned}
[\![x]\!](\rho) &\stackrel{\text{def}}{=} \rho(x) && \in [\![\Gamma(x)]\!] \\
[\![t_1\ t_2]\!](\rho) &\stackrel{\text{def}}{=} ([\![t_1]\!](\rho))\ ([\![t_2]\!](\rho)) && \\
[\![\mathsf{fun}\, x{:}\,\tau.\, t]\!](\rho) &\stackrel{\text{def}}{=} \lambda d \in [\![\tau]\!].\ [\![t]\!](\rho, d) &&
\end{aligned}
$$

$$
[\![\mathsf{fun}\, x{:}\,\tau.\, t]\!] = \mathrm{cur}([\![t]\!])
$$

Denotation of fixed points

$$[\![\texttt{fix}\, f]\!](\rho) \stackrel{\text{def}}{=} \mathrm{fix}([\![f]\!](\rho))$$

For any Pcf term t such that $\Gamma \vdash t : \tau$, the object $[\![t]\!]$
is well-defined and a continuous function $[\![t]\!] : [\![\Gamma]\!] \to [\![\tau]\!]$.

Denotation of Pcf terms

For any Pcf term t such that $\Gamma \vdash t : \tau$, the object $[\![t]\!]$
is well-defined and a continuous function $[\![t]\!] : [\![\Gamma]\!] \to [\![\tau]\!]$.

If $t \in \text{Pcf}_\tau$: $\quad [\![t]\!] \quad \in \quad [\![\cdot]\!] \to [\![\tau]\!] \quad = \quad \mathbb{1} \to [\![\tau]\!] \quad \cong \quad [\![\tau]\!]$

Denotational Semantics for Pcf

Compositionality

Suppose $t, u \in \textsc{Pcf}_{\Delta,\sigma}$, such that

$$[\![t]\!] = [\![u]\!] : [\![\Delta]\!] \to [\![\sigma]\!]$$

Suppose moreover that $\mathcal{C}[-]$ is a $\textsc{Pcf}$ context such that $\Gamma \vdash_{\Delta,\sigma} \mathcal{C} : \tau$. Then

$$[\![\mathcal{C}[t]]\!] = [\![\mathcal{C}[u]]\!] : [\![\Gamma]\!] \to [\![\tau]\!].$$

A denotation for evaluation contexts

If $\Gamma \vdash_{\Delta,\sigma} \mathcal{C} : \tau$, then define $[\![\mathcal{C}]\!]$ such that

$$[\![\mathcal{C}]\!] : ([\![\Delta]\!] \to [\![\sigma]\!]) \to [\![\Gamma]\!] \to [\![\tau]\!]$$

A denotation for evaluation contexts

If $\Gamma \vdash_{\Delta,\sigma} \mathcal{C} : \tau$, then define $[\![\mathcal{C}]\!]$ such that

$$[\![\mathcal{C}]\!] : ([\![\Delta]\!] \to [\![\sigma]\!]) \to [\![\Gamma]\!] \to [\![\tau]\!]$$

$$\begin{aligned} [\![-]\!](d) &= d \\ [\![\mathcal{C}\ t]\!](d)(\rho) &= ([\![\mathcal{C}]\!](d)(\rho))([\![t]\!](\rho)) \\ &\vdots \end{aligned}$$

A denotation for evaluation contexts

If $\Gamma \vdash_{\Delta,\sigma} \mathcal{C} : \tau$, then define $[\![\mathcal{C}]\!]$ such that

$$[\![\mathcal{C}]\!] : ([\![\Delta]\!] \to [\![\sigma]\!]) \to [\![\Gamma]\!] \to [\![\tau]\!]$$

$$\begin{aligned} [\![-]\!](d) &= d \\ [\![\mathcal{C}\; t]\!](d)(\rho) &= ([\![\mathcal{C}]\!](d)(\rho))([\![t]\!](\rho)) \\ &\vdots \end{aligned}$$

If $\Gamma \vdash_{\Delta,\sigma} \mathcal{C} : \tau$ and $\Delta \vdash t : \sigma$, then

$$[\![\mathcal{C}[t]]\!] = [\![\mathcal{C}]\!]([\![t]\!])$$

Assume

$$\Gamma \vdash u : \sigma$$
$$\Gamma, x{:}\sigma \vdash t : \tau$$

Then for all $\rho \in [\![\Gamma]\!]$

$$[\![t[u/x]]\!](\rho) = [\![t]\!](\rho[x \mapsto [\![u]\!](\rho)]).$$

In particular when $\Gamma = \cdot$, $[\![t]\!] : [\![\sigma]\!] \to [\![\tau]\!]$ and

$$[\![t[u/x]]\!] = [\![t]\!]([\![u]\!])$$

Denotational Semantics for Pcf

Soundness

Soundness

For all PCF types τ and all closed terms $t, v \in \text{PCF}_\tau$ with v a value, if $t \Downarrow_\tau v$ is derivable, then

$$[\![t]\!] = [\![v]\!] \in [\![\tau]\!]$$