Where we're at

We have a denotational semantics:

- a mapping of Pcf types τ to domains $[\![\tau]\!]$;
- a mapping of well-typed terms $\Gamma \vdash t : \tau$ into continuous functions $[\![t]\!] \in [\![\Gamma]\!] \to [\![\tau]\!]$;
- a mapping of well-typed, closed Pcf terms $\cdot \vdash t : \tau$ to elements $[\![t]\!] \in [\![\tau]\!]$.

Where we're at

We have a denotational semantics:

- a mapping of PCF types τ to domains $[\![\tau]\!]$;
- a mapping of well-typed terms $\Gamma \vdash t : \tau$ into continuous functions $[\![t]\!] \in [\![\Gamma]\!] \to [\![\tau]\!]$;
- a mapping of well-typed, closed PCF terms $\cdot \vdash t : \tau$ to elements $[\![t]\!] \in [\![\tau]\!]$.

such that:

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

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

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

Denotational Semantics for Pcf

Soundness

Typing and evaluation

$$\frac{t \Downarrow_{\sigma \to \tau} \mathrm{fun}\ x : \sigma .\ t' \qquad t'[u/x] \Downarrow_{\tau} v}{t\ u \Downarrow_{\tau} v}$$

If $t \Downarrow_\tau v$, then both $\vdash t : \tau$ and $\vdash v : \tau$.

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

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

By induction on $t \Downarrow_\tau v$

$$\overline{\sigma \Downarrow_c \sigma'} \rightsquigarrow [\![\sigma]\!] = [\![\sigma']\!]$$

$$\frac{t \Downarrow_{nat} v}{succ\ t \Downarrow_{nat} succ\ v} \rightsquigarrow$$

IH: $[\![t]\!] = [\![v]\!] \in \mathbb{N}_\perp$

$[\![succ\ t]\!] = succ_\perp [\![t]\!](e)$ $e : \emptyset \to \dots$

$= succ_\perp [\![u]\!]$

$= [\![succ\ u]\!]$

$$\frac{b \Downarrow_{\mathbb{B}} true \qquad t \Downarrow_\tau v}{if\ b\ then\ t\ else\ u \Downarrow_\tau v} \rightsquigarrow$$

IH: $[\![b]\!] = [\![true]\!] = true \in \mathbb{B}_\perp$

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

$[\![if\ b\ then\ t\ else\ u]\!] =$

$if < [\![b]\!], < [\![t]\!], [\![u]\!] >>$

$= if < true, < [\![t]\!], [\![u]\!] >> = [\![t]\!]$

$= [\![v]\!]$

$if \in \mathbb{B}_\perp \times D \times D \to D$

$$\frac{t \Downarrow_{\sigma\to\tau} \mathrm{fun}\, x{:}\sigma.t' \qquad t'[u/x] \Downarrow_\tau v}{t\, u \Downarrow_\tau v} \rightsquigarrow$$

IH: $[\![t]\!] = [\![\mathrm{fun}\, x{:}\sigma.t']\!]$
$= (\lambda d \in [\![\sigma]\!].[\![t']\!](x \mapsto d))$
$= (\lambda d.[\![\sigma]\!].[\![t']\!](d))$
$= [\![t']\!]$

$[\![t']\!] \in [\![x{:}\sigma]\!] \to [\![\tau]\!]$
$\quad\quad\quad\quad \| $
$\quad\quad\quad [\![\sigma]\!]$

$[\![t'[u/x]]\!] = [\![v]\!]$

<u>Lem:</u> $[\![t'[u/x]]\!] = [\![t']\!]\,[\![u]\!]$

But then $[\![t'[u/x]]\!] = [\![t]\!]\,[\![u]\!] = [\![t\,u]\!]$
$\quad\quad\quad\quad \|$
$\quad\quad\quad [\![v]\!]$

$$\frac{f\,(\mathrm{fix}\, f) \Downarrow_\tau v}{\mathrm{fix}\, f \Downarrow_\tau v} \rightsquigarrow$$

IH: $[\![f\,(\mathrm{fix}\, f)]\!] = [\![v]\!]$

$\vdash f : \tau \to \tau \qquad [\![f]\!] \in [\![\tau]\!] \to [\![\tau]\!]$

$$
\begin{aligned}
[\![\mathrm{fix}\, f]\!] &= \mathrm{fix}\, [\![f]\!] \\
&= [\![f]\!]\,(\mathrm{fix}\, [\![f]\!]) \\
&= [\![f]\!]\,([\![\mathrm{fix}\, f]\!]) \\
&= [\![f\,(\mathrm{fix}\, f)]\!] = [\![v]\!]
\end{aligned}
$$

If $t \in$ PCF$_{\mathtt{nat}}$ and $[\![t]\!] = \bot$, then $t \Uparrow_{\mathtt{nat}}$.

PCF$_{bool}$ - - - $t \Uparrow_{bool}$

$t \Uparrow_{nat} \overset{def}{\iff} \nexists v.\ t \Downarrow_{nat} v$

Assume $t \Downarrow_{nat} v$ then

$v = \underline{n}$ $\quad [\![v]\!] = [\![\underline{n}]\!] = n \in \mathbb{N}_\bot$

by soundness

$\bot = [\![t]\!] = [\![v]\!] = n$

$\bot = n \in \mathbb{N}_\bot$ $\quad ↯$

$\mathrm{fun}\ x : nat.\ \Omega_{nat} \Downarrow$ itself but $[\![\mathrm{fun}\ x : nat.\ \Omega_{nat}]\!]$

$= \lambda d \in \mathbb{N}_\bot.\ [\![\Omega_{nat}]\!] = \lambda d \in \mathbb{N}_\bot.\ \bot$

$= \bot_{\mathbb{N}_\bot \to \mathbb{N}_\bot}$

Adequacy

Reminder: Adequacy

For any closed PCF term t and value v of ground type $\gamma \in \{\texttt{nat}, \texttt{bool}\}$

true   false $\leftarrow [\![v]\!]$

$\perp$

$$[\![t]\!] = [\![v]\!] \in [\![\gamma]\!] \Rightarrow t \Downarrow_\gamma v$$

Reminder: Adequacy

For any **closed** PCF term t and value v of **ground** type $\gamma \in \{\mathtt{nat}, \mathtt{bool}\}$

$$[\![t]\!] = [\![v]\!] \in [\![\gamma]\!] \Rightarrow t \Downarrow_{\gamma} v$$

Adequacy does **not** hold at function types or for open terms

For any **closed** PCF term t and value v of **ground** type $\gamma \in \{\mathtt{nat}, \mathtt{bool}\}$

$$[\![t]\!] = [\![v]\!] \in [\![\gamma]\!] \Rightarrow t \Downarrow_\gamma v$$

Adequacy does **not** hold at function types or for open terms

$$[\![\mathtt{fun}\, x{:}\tau.\, (\mathtt{fun}\, y{:}\tau.\, y)\, x]\!] \quad = \quad [\![\mathtt{fun}\, x{:}\tau.\, x]\!] \quad : [\![\tau]\!] \to [\![\tau]\!]$$

but

$$\mathtt{fun}\, x{:}\tau.\, (\mathtt{fun}\, y{:}\tau.\, y)\, x \not\Downarrow_{\tau\to\tau} \mathtt{fun}\, x{:}\tau.\, x$$

For any **closed** PCF term t and value v of **ground** type $\gamma \in \{\mathtt{nat}, \mathtt{bool}\}$

$$[\![t]\!] = [\![v]\!] \in [\![\gamma]\!] \Rightarrow t \Downarrow_\gamma v$$

Adequacy does **not** hold at function types or for open terms

More serious: $\vdash f : nat \to bool$

$$[\![\mathtt{fun}\, x \colon \mathtt{nat}.\, (\mathtt{if}\ \ \mathtt{zero?}(f\ x)\ \mathtt{then}\ \mathtt{true}\ \mathtt{else}\ \mathtt{true})]\!]$$

$$\stackrel{?}{=} \quad [\![\mathtt{fun}\, x \colon \mathtt{nat}.\ \mathtt{true}]\!]$$

Adequacy

Formal approximation relation

How to prove adequacy

→ between $[\![\tau]\!]$ and PCF_τ

Proof idea: introduce a relation R such that

1. if $t \in \mathrm{PCF}_{\mathtt{nat}}$, $n \in \mathbb{N}$, and $R(n, t)$, then $t \Downarrow_\gamma \underline{n}$ (same for booleans);
2. for any well-typed term t, $R([\![t]\!], t)$.

Proof idea: introduce a relation R such that

1. if $t \in \mathrm{PCF}_{\mathtt{nat}}$, $n \in \mathbb{N}$, and $R(n, t)$, then $t \Downarrow_{\gamma} \underline{n}$ (same for booleans);
2. for any well-typed term t, $R(\llbracket t \rrbracket, t)$.

But at non-base types, adequacy does not hold.

Proof idea: introduce a relation R such that

1. if $t \in \textsc{Pcf}_{\mathtt{nat}}$, $n \in \mathbb{N}$, and $R(n, t)$, then $t \Downarrow_\gamma \underline{n}$ (same for booleans);
2. for any well-typed term t, $R(\llbracket t \rrbracket, t)$.

But at non-base types, adequacy does not hold.

We must define a **family** of relations, tailored for each type: **formal approximation**

$$\lhd_\tau \subseteq \llbracket \tau \rrbracket \times \textsc{Pcf}_\tau$$

Proof idea: introduce a relation R such that

1. if $t \in \textsc{Pcf}_{\texttt{nat}}$, $n \in \mathbb{N}$, and $R(n, t)$, then $t \Downarrow_{\gamma} \underline{n}$ (same for booleans);
2. for any well-typed term t, $R(\llbracket t \rrbracket, t)$.

But at non-base types, adequacy does not hold.

We must define a family of relations, tailored for each type: formal approximation

$$\lhd_\tau \subseteq \llbracket \tau \rrbracket \times \textsc{Pcf}_\tau$$

A logical relation.

How to prove adequacy

Proof idea: introduce a relation R such that

1. if $t \in \textsc{Pcf}_{\mathtt{nat}}$, $n \in \mathbb{N}$, and $R(n, t)$, then $t \Downarrow_{\gamma} \underline{n}$ (same for booleans);
2. for any well-typed term t, $R(\llbracket t \rrbracket, t)$.

But at non-base types, **adequacy does not hold**.

We must define a **family** of relations, tailored for each type: **formal approximation**

$$\lhd_{\tau} \subseteq \llbracket \tau \rrbracket \times \textsc{Pcf}_{\tau}$$

A **logical relation**.

Formal approximation at base types

$$d \lhd_{\mathtt{nat}} t \overset{\text{def}}{\Leftrightarrow} (d \in \mathbb{N} \Rightarrow t \Downarrow_{\mathtt{nat}} \underline{d})$$

$$\begin{aligned} d \lhd_{\mathtt{bool}} t \overset{\text{def}}{\Leftrightarrow} & (d = \text{true} \Rightarrow t \Downarrow_{\mathtt{bool}} \mathtt{true}) \\ & \wedge (d = \text{false} \Rightarrow t \Downarrow_{\mathtt{bool}} \mathtt{false}) \end{aligned}$$

Formal approximation at base types

$$d \lhd_{\texttt{nat}} t \overset{\text{def}}{\Leftrightarrow} (d \in \mathbb{N} \Rightarrow t \Downarrow_{\texttt{nat}} \underline{d})$$

$$\begin{aligned} d \lhd_{\texttt{bool}} t \overset{\text{def}}{\Leftrightarrow} & (d = \text{true} \Rightarrow t \Downarrow_{\texttt{bool}} \texttt{true}) \\ & \wedge (d = \text{false} \Rightarrow t \Downarrow_{\texttt{bool}} \texttt{false}) \end{aligned}$$

Exactly what we need to get 1.

$$d \lhd_{\mathtt{nat}} t \overset{\text{def}}{\Leftrightarrow} (d \in \mathbb{N} \Rightarrow t \Downarrow_{\mathtt{nat}} \underline{d})$$

$$\begin{aligned} d \lhd_{\mathtt{bool}} t \overset{\text{def}}{\Leftrightarrow} & (d = \text{true} \Rightarrow t \Downarrow_{\mathtt{bool}} \mathtt{true}) \\ & \wedge (d = \text{false} \Rightarrow t \Downarrow_{\mathtt{bool}} \mathtt{false}) \end{aligned}$$

Exactly what we need to get 1.

Note though that $\perp \lhd_{\mathtt{nat}} t$ for any $t \in \text{PCF}_{\mathtt{nat}}$.

Formal approximation at function types

1. if $t \in \text{PCF}_{\mathtt{nat}}$, $n \in \mathbb{N}$, and $R(n, t)$, then $t \Downarrow_\gamma \underline{n}$ (same for booleans); ✓
2. for any well-typed term t, $R(\llbracket t \rrbracket, t)$.

Formal approximation at function types

1. if $t \in \textsc{Pcf}_{\texttt{nat}}$, $n \in \mathbb{N}$, and $R(n, t)$, then $t \Downarrow_\gamma \underline{n}$ (same for booleans); ✓
2. for any well-typed term t, $R(\llbracket t \rrbracket, t)$.
    2.1 By induction on (the typing derivation of) t;
    2.2 we need to interpret each typing rule.

Formal approximation at function types

1. if $t \in \mathrm{PCF}_{\mathtt{nat}}$, $n \in \mathbb{N}$, and $R(n, t)$, then $t \Downarrow_\gamma \underline{n}$ (same for booleans); ✓
2. for any well-typed term t, $R(\llbracket t \rrbracket, t)$.
   2.1 By induction on (the typing derivation of) t;
   2.2 we need to interpret each typing rule.

$$\textsc{App}\ \frac{\vdash t : \tau \to \tau' \qquad \vdash u : \tau}{\vdash t\ u : \tau'}$$

$$\frac{\text{IH}: \llbracket t \rrbracket \lhd_{\sigma \to \tau} t \qquad \llbracket u \rrbracket \lhd_{\sigma} u}{\llbracket t \rrbracket \llbracket u \rrbracket \lhd_{\tau} t\,u}$$

1. if $t \in \mathrm{PCF}_{\mathtt{nat}}$, $n \in \mathbb{N}$, and $R(n, t)$, then $t \Downarrow_{\gamma} \underline{n}$ (same for booleans); ✓
2. for any well-typed term t, $R(\llbracket t \rrbracket, t)$.
   2.1 By induction on (the typing derivation of) t;
   2.2 we need to interpret each typing rule.

$$\textsc{App}\ \frac{\vdash t : \tau \to \tau' \qquad \vdash u : \tau}{\vdash t\ u : \tau'}$$

Assume $\llbracket u \rrbracket \lhd_{\tau} u$ and $\llbracket t \rrbracket \lhd_{\tau \to \tau'} t$, how do we get $\llbracket t\ u \rrbracket = \llbracket t \rrbracket (\llbracket u \rrbracket) \lhd_{\tau} t\ u$?

1. if $t \in \text{PCF}_{\texttt{nat}}$, $n \in \mathbb{N}$, and $R(n, t)$, then $t \Downarrow_\gamma \underline{n}$ (same for booleans); ✓
2. for any well-typed term t, $R(\llbracket t \rrbracket, t)$.
   2.1 By induction on (the typing derivation of) t;
   2.2 we need to interpret each typing rule.

$$\text{App}\ \frac{\vdash t : \tau \to \tau' \qquad \vdash u : \tau}{\vdash t\ u : \tau'}$$

Assume $\llbracket u \rrbracket \lhd_\tau u$ and $\llbracket t \rrbracket \lhd_{\tau \to \tau'} t$, how do we get $\llbracket t\ u \rrbracket = \llbracket t \rrbracket (\llbracket u \rrbracket) \lhd_\tau t\ u$?

Define

$$d \lhd_{\tau \to \tau'} t \overset{\text{def}}{\Leftrightarrow} \forall e \in \llbracket \tau \rrbracket, u \in \text{PCF}_\tau .(e \lhd_\tau u \Rightarrow d(e) \lhd_{\tau'} t\ u)$$

$$\textsc{Abs}\ \frac{\Gamma, x{:}\,\tau \vdash t : \tau'}{\Gamma \vdash \mathsf{fun}\, x{:}\,\tau.\, t : \tau \to \tau'}$$

To prove Item 2, we need to talk about **open** terms.

Formal approximation for open terms

$[\![\Gamma \vdash t]\!] : [\![\Gamma]\!] \times [\![\tau]\!] \to [\![\tau']\!]$

$$\textsc{Abs}\ \frac{\Gamma, x\!:\!\tau \vdash t : \tau'}{\Gamma \vdash \mathsf{fun}\, x\!:\!\tau.\, t : \tau \to \tau'}$$

To prove Item 2, we need to talk about **open** terms.

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

Semantic application ≈ syntactic substitution

$$\textsc{Abs}\ \frac{\Gamma, x{:}\tau \vdash t : \tau'}{\Gamma \vdash \mathsf{fun}\, x{:}\tau.\, t : \tau \to \tau'}$$

To prove Item 2, we need to talk about **open** terms.

$$[\![t]\!]\,([\![u]\!]) = [\![(t[u/x])]\!] \qquad \text{Semantic application} \approx \text{syntactic substitution}$$

$\rho \in [\![\Gamma]\!] = \prod_{x \in \mathrm{dom}(\Gamma)} [\![\Gamma(x)]\!]$

Parallel substitution: maps each $x \in \mathrm{dom}(\Gamma)$ to $\sigma(x) \in \textsc{Pcf}_{\Gamma(x)}$.

$$\rho \lhd_\Gamma \sigma \overset{\text{def}}{\Leftrightarrow} \forall x \in \mathrm{dom}(\Gamma), \rho(x) \lhd_{\Gamma(x)} \sigma(x)$$

For any

- context Γ and type τ
- term t such that $\Gamma \vdash t : \tau$
- environment ρ $\in [\![\Gamma]\!]$
- substitution σ for Γ
- such that $\rho \lhd_\Gamma \sigma$

we have

$$[\![\tau]\!] \qquad PCF_\tau$$

$$\cup\!\!\!\!\!\!\;| \qquad \cup\!\!\!\!\!\!\;|$$

$$[\![t]\!]\,(\rho) \lhd_\tau t[\sigma].$$

$$\left.\begin{array}{l}\Gamma, x : \tau' \vdash t : \tau \\ \Gamma \vdash u : \tau'\end{array}\right\} \Gamma \vdash t[{}^u/_x] : \tau$$

$$\left.\begin{array}{l}\Gamma \vdash t : \tau \\ \sigma \text{ is a subst. for } \Gamma\end{array}\right\} \cdot \vdash t[\sigma] : \tau$$

For any

- context Γ and type τ
- term t such that $\Gamma \vdash t : \tau$
- environment ρ
- substitution σ
- such that $\rho \lhd_\Gamma \sigma$

we have

$$[\![t]\!]\,(\rho) \lhd_\tau t[\sigma].$$

Corollary: if $\cdot \vdash t : \tau$,

$$[\![t]\!] \lhd_\tau t.$$

- $\vdash t : nat$ then $[\![t]\!] \lhd_{nat} t$

If $[\![t]\!] = [\![\underline{n}]\!] = n$ then $n \lhd_{nat} t$

$$\Updownarrow \text{def}$$

$$(\underbrace{n \in \mathbb{N}}_{\text{True}} \Rightarrow t \Downarrow_{nat} n)$$

$$\Downarrow$$

$$t \Downarrow_{nat} n$$