



PSC 2021/22 (375AA, 9CFU)

Principles for Software Composition

Roberto Bruni

<http://www.di.unipi.it/~bruni/>

<http://didawiki.di.unipi.it/doku.php/magistraleinformatica/psc/start>

12b - HOFL Operational Semantics

Disclaim

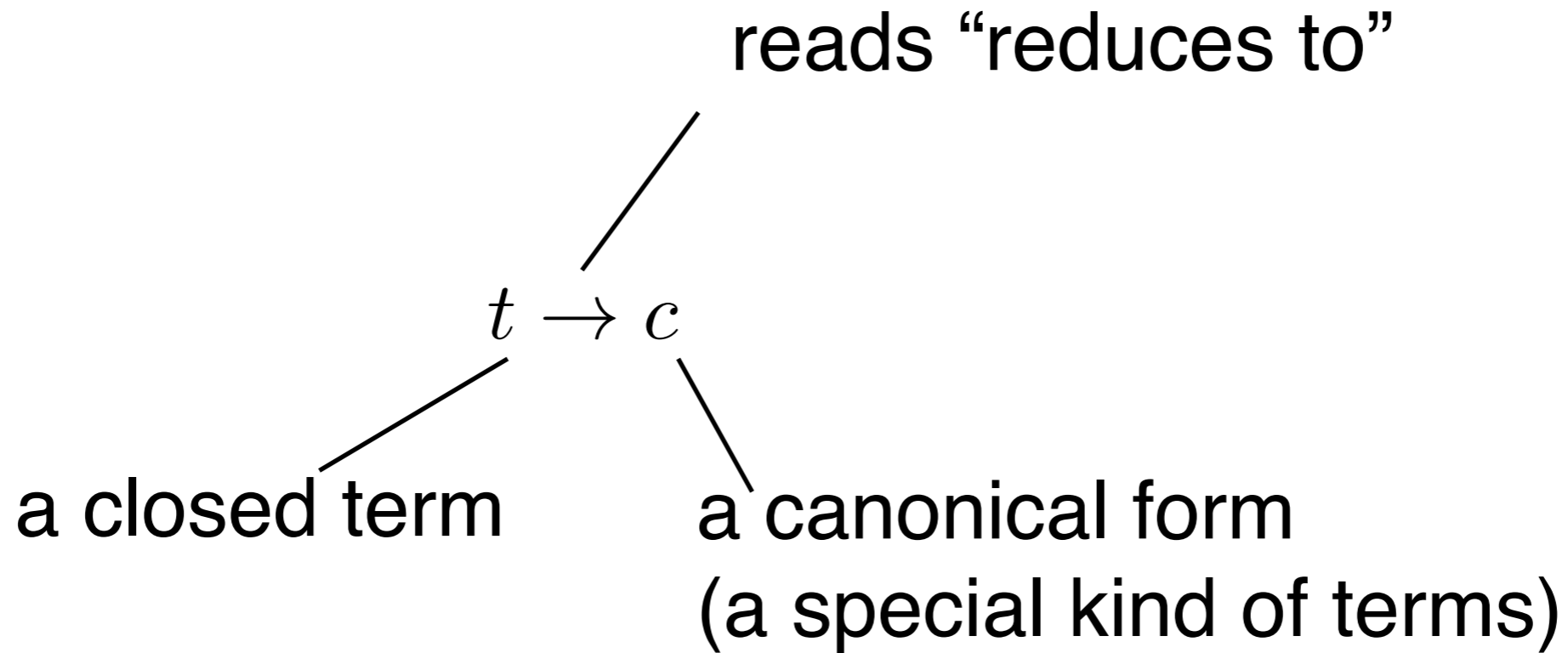
$$t ::= x \mid n \mid t_0 \text{ op } t_1 \mid \text{if } t \text{ then } t_0 \text{ else } t_1$$
$$\mid (t_0, t_1) \mid \text{fst}(t) \mid \text{snd}(t)$$
$$\mid \lambda x. t \mid t_0 t_1$$
$$\mid \text{rec } x. t$$
$$\tau ::= \text{int} \mid \tau_0 * \tau_1 \mid \tau_0 \rightarrow \tau_1$$
$$\frac{}{x : \widehat{x}} \quad \frac{}{n : \text{int}} \quad \frac{t_0 : \text{int} \quad t_1 : \text{int}}{t_0 \text{ op } t_1 : \text{int}} \quad \frac{t : \text{int} \quad t_0 : \tau \quad t_1 : \tau}{\text{if } t \text{ then } t_0 \text{ else } t_1 : \tau}$$
$$\frac{t_0 : \tau_0 \quad t_1 : \tau_1}{(t_0, t_1) : \tau_0 * \tau_1} \quad \frac{t : \tau_0 * \tau_1}{\text{fst}(t) : \tau_0} \quad \frac{t : \tau_0 * \tau_1}{\text{snd}(t) : \tau_1}$$
$$\frac{x : \tau_0 \quad t : \tau_1}{\lambda x. t : \tau_0 \rightarrow \tau_1} \quad \frac{t_1 : \tau_0 \rightarrow \tau_1 \quad t_0 : \tau_0}{t_1 t_0 : \tau_1}$$
$$\frac{x : \tau \quad t : \tau}{\text{rec } x. t : \tau}$$

we assign semantics
only to terms that are:
well-formed and closed

$$t : \tau$$
$$\text{fv}(t) = \emptyset$$

Canonical forms

Statements



Big step operational semantics

computation of canonical form
(by term manipulation)

Canonical forms

set of canonical forms $C_\tau \subseteq T_\tau$
with type τ

(laziness)
not required to be
in canonical forms

$$\frac{\frac{n \in C_{int}}{\text{---}} \quad \frac{t_0 : \tau_0 \quad t_1 : \tau_1 \quad t_0, t_1 \text{ closed}}{\text{---}}}{(t_0, t_1) \in C_{\tau_0 * \tau_1}}$$

t not necessarily
a closed term

$$\frac{\frac{\lambda x. t : \tau_0 \rightarrow \tau_1 \quad \lambda x. t \text{ closed}}{\text{---}}}{\lambda x. t \in C_{\tau_0 \rightarrow \tau_1}}$$

Canonical forms?

$$\frac{}{n \in C_{int}} \quad \frac{t_0 : \tau_0 \quad t_1 : \tau_1 \quad t_0, t_1 \text{ closed}}{(t_0, t_1) \in C_{\tau_0 * \tau_1}}$$

$$\frac{\lambda x. t : \tau_0 \rightarrow \tau_1 \quad \lambda x. t \text{ closed}}{\lambda x. t \in C_{\tau_0 \rightarrow \tau_1}}$$

$1 + 2 \times 3$ ❌

if 0 then 0 else 0 ❌

$(1, 2)$ ✅

$\lambda x. 1$ ✅

$(1 + 2, 2 - 1)$ ✅

$\lambda x. 1 + 2 \times 3$ ✅

fst(1, 2) ❌

$\lambda x. \text{fst}(1, 2)$ ✅

HOFΛ

Lazy operational semantics

Operational semantics: axioms

$$\frac{c \in C_{\tau}}{c \rightarrow c}$$

i.e., expanding the various cases

$$\frac{}{n \rightarrow n} \quad \frac{t_0 : \tau_0 \quad t_1 : \tau_1 \quad t_0, t_1 \text{ closed}}{(t_0, t_1) \rightarrow (t_0, t_1)} \quad \frac{\lambda x. t : \tau_0 \rightarrow \tau_1 \quad \lambda x. t \text{ closed}}{\lambda x. t \rightarrow \lambda x. t}$$

integers, pairs and abstractions
are already in canonical form

Lazy op semantics

$\frac{}{n \rightarrow n}$	$\frac{t_0 : \tau_0 \quad t_1 : \tau_1 \quad t_0, t_1 \text{ closed}}{(t_0, t_1) \rightarrow (t_0, t_1)}$	$\frac{\lambda x. t : \tau_0 \rightarrow \tau_1 \quad \lambda x. t \text{ closed}}{\lambda x. t \rightarrow \lambda x. t}$
$\frac{t_0 \rightarrow n_0 \quad t_1 \rightarrow n_1}{t_0 \text{ op } t_1 \rightarrow n_0 \text{ op } n_1}$	$\frac{t \rightarrow 0 \quad t_0 \rightarrow c_0}{\mathbf{if } t \text{ then } t_0 \text{ else } t_1 \rightarrow c_0}$	$\frac{t \rightarrow (t_0, t_1) \quad t_0 \rightarrow c_0}{\mathbf{fst}(t) \rightarrow c_0}$
$\frac{t[\mathbf{rec } x. t / x] \rightarrow c}{\mathbf{rec } x. t \rightarrow c}$	$\frac{t \rightarrow n \quad n \neq 0 \quad t_1 \rightarrow c_1}{\mathbf{if } t \text{ then } t_0 \text{ else } t_1 \rightarrow c_1}$	$\frac{t \rightarrow (t_0, t_1) \quad t_1 \rightarrow c_1}{\mathbf{snd}(t) \rightarrow c_1}$
	$\frac{t_1 \rightarrow \lambda x. t'_1 \quad t'_1[t_0 / x] \rightarrow c}{(t_1 t_0) \rightarrow c}$	<p>(lazy)</p>

Type system (remind)

$$\frac{}{x : \hat{x}} \quad \frac{}{n : int} \quad \frac{t_0 : int \quad t_1 : int}{t_0 \text{ op } t_1 : int} \quad \frac{t : int \quad t_0 : \tau \quad t_1 : \tau}{\text{if } t \text{ then } t_0 \text{ else } t_1 : \tau}$$

$$\frac{t_0 : \tau_0 \quad t_1 : \tau_1}{(t_0, t_1) : \tau_0 * \tau_1}$$

$$\frac{t : \tau_0 * \tau_1}{\text{fst}(t) : \tau_0}$$

$$\frac{t : \tau_0 * \tau_1}{\text{snd}(t) : \tau_1}$$

$$\frac{x : \tau_0 \quad t : \tau_1}{\lambda x. t : \tau_0 \rightarrow \tau_1}$$

$$\frac{t_1 : \tau_0 \rightarrow \tau_1 \quad t_0 : \tau_0}{t_1 t_0 : \tau_1}$$

$$\frac{x : \tau \quad t : \tau}{\text{rec } x. t : \tau}$$

Example

$$t \triangleq \lambda x. \underbrace{x + 1}_{\text{int}} : \underbrace{\text{int} \rightarrow \text{int}}_{\text{int} \rightarrow \text{int}}$$

$$\lambda x. x + 1 \rightarrow c \quad \swarrow_{c = \lambda x. x + 1} \quad \square$$

Example

$$t \triangleq \underbrace{(\underbrace{\lambda x. x + 1}_{int \rightarrow int}, \underbrace{\underbrace{1}_{int} + \underbrace{2}_{int}}_{int})}_{(int \rightarrow int) * int} : (int \rightarrow int) * int$$

$$(\lambda x. x + 1, 1 + 2) \rightarrow c \quad \swarrow c = (\lambda x. x + 1, 1 + 2) \quad \square$$

laziness:

no need to evaluate 1+2

Example

$$t \triangleq \lambda x. \text{if } \text{fst}(x) \text{ then } 1 \text{ else } \text{snd}(x) : (int * int) \rightarrow int$$

$\underbrace{\hspace{10em}}_{(int * int) \rightarrow int}$

$\underbrace{\hspace{10em}}_{int}$

$\underbrace{\hspace{10em}}_{int = \tau_1}$

$\underbrace{\hspace{10em}}_{int * \tau_1}$

$\underbrace{\hspace{10em}}_{int * \tau_1}$

$\underbrace{\hspace{10em}}_{int}$

$\underbrace{\hspace{10em}}_{int * int}$

Example (ctd)

$t \triangleq \lambda x. \text{if fst}(x) \text{ then } 1 \text{ else snd}(x)$

$t (1, 2) \rightarrow c \quad \swarrow \quad t \rightarrow \lambda x'. t' , t' [^{(1,2)} / x'] \rightarrow c$

$\swarrow_{x'=x, t'=\text{if} \dots (\text{if fst}(x) \text{ then } 1 \text{ else snd}(x)) [^{(1,2)} / x]} \rightarrow c$
 $= \text{if fst}(1, 2) \text{ then } 1 \text{ else snd}(1, 2) \rightarrow c$

$\swarrow \quad \text{fst}(1, 2) \rightarrow n , n \neq 0 , \text{snd}(1, 2) \rightarrow c$

$\swarrow \quad (1, 2) \rightarrow (n_1, n_2) , n_1 \rightarrow n , n \neq 0 , \text{snd}(1, 2) \rightarrow c$

$\swarrow_{n_1=1, n_2=2, n=1}^* \quad \text{snd}(1, 2) \rightarrow c$

$\swarrow \quad (1, 2) \rightarrow (n_3, n_4) , n_4 \rightarrow c$

$\swarrow_{n_3=1, n_4=2, c=2}^* \quad \square$

$t (1, 2) \rightarrow 2$

Example

$$t \triangleq \mathbf{rec} \underbrace{x}_{\tau}. \underbrace{x}_{\tau} : \tau$$

$$\begin{aligned} \mathbf{rec} x. x \rightarrow c & \swarrow x[\mathbf{rec} x. x / x] \rightarrow c \\ & = \mathbf{rec} x. x \rightarrow c \end{aligned}$$

same goal from which we started
no other option to explore:
divergence!

Example

$$fact \triangleq \mathbf{rec} f. \lambda x. \mathbf{if} x \mathbf{then} 1 \mathbf{else} x \times (f (x - 1))$$

$$\begin{aligned} fact \rightarrow c & \swarrow (\lambda x. \mathbf{if} x \mathbf{then} 1 \mathbf{else} x \times (f(x - 1))) \left[\frac{fact}{f} \right] \rightarrow c \\ & = \lambda x. \mathbf{if} x \mathbf{then} 1 \mathbf{else} x \times \overbrace{((\mathbf{rec} f. \dots)(x - 1))} \\ & \swarrow c = \lambda x. \mathbf{if} x \mathbf{then} 1 \mathbf{else} x \times (fact(x - 1)) \quad \square \end{aligned}$$

Example

$fact \triangleq \mathbf{rec} f. \lambda x. \mathbf{if} x \mathbf{then} 1 \mathbf{else} x \times (f (x - 1))$

$(fact\ 1) \rightarrow c \quad \swarrow \quad fact \rightarrow \lambda x'. t' , t' [^1/x'] \rightarrow c$

$\swarrow_{x'=x, t'=\mathbf{if} \dots}^*$ $(\mathbf{if} x \mathbf{then} 1 \mathbf{else} x \times (fact (x - 1))) [^1/x] \rightarrow c$
 $= \mathbf{if} 1 \mathbf{then} 1 \mathbf{else} 1 \times (fact (1 - 1)) \rightarrow c$

\swarrow $1 \rightarrow n , n \neq 0 , 1 \times (fact (1 - 1)) \rightarrow c$

$\swarrow_{n=1, c=n_1 \times n_2}^*$ $1 \rightarrow n_1 , (fact (1 - 1)) \rightarrow n_2$ **laziness**

$\swarrow_{n_1=1}$ $fact \rightarrow \lambda x''. t'' , t'' [^{1-1}/x''] \rightarrow n_2$ **evident here**

$\swarrow_{x''=x, t''=\mathbf{if} \dots}^*$ $(\mathbf{if} x \mathbf{then} 1 \mathbf{else} x \times (fact (x - 1))) [^{1-1}/x] \rightarrow n_2$
 $= \mathbf{if} 1 - 1 \mathbf{then} 1 \mathbf{else} (1 - 1) \times (fact ((1 - 1) - 1)) \rightarrow n_2$

\swarrow $1 - 1 \rightarrow 0 , 1 \rightarrow n_2$

$\swarrow_{n_2=1}^*$ \square $c = n_1 \times n_2 = 1 \times 1 = 1$

HOFΛ

Eager operational semantics

Lazy vs Eager

$$\frac{t_1 \rightarrow \lambda x. t'_1 \quad t'_1[t_0/x] \rightarrow c}{(t_1 t_0) \rightarrow c} \quad (\text{lazy})$$

$$\frac{t_1 \rightarrow \lambda x. t'_1 \quad t_0 \rightarrow c_0 \quad t'_1[c_0/x] \rightarrow c}{(t_1 t_0) \rightarrow c} \quad (\text{eager})$$

Lazy vs Eager

$$t \triangleq (\lambda x. 1) (\mathbf{rec} y. y) : int$$

$$x : \tau$$

$$y : \tau$$

$$t \rightarrow c \quad \swarrow \quad \lambda x. 1 \rightarrow \lambda x'. t' , t' [\mathbf{rec} y. y / x'] \rightarrow c$$

lazy

$$\swarrow_{x'=x, t'=1} \quad 1 [\mathbf{rec} y. y / x] \rightarrow c$$
$$= 1 \rightarrow c$$

$$\swarrow_{c=1} \quad \square$$

$$t \rightarrow c \quad \swarrow \quad \lambda x. 1 \rightarrow \lambda x'. t' , \mathbf{rec} y. y \rightarrow c' , t' [c' / x'] \rightarrow c$$

eager

$$\swarrow_{x'=x, t'=1} \quad \mathbf{rec} y. y \rightarrow c' , 1 [c' / x] \rightarrow c$$

$$\swarrow \quad \mathbf{rec} y. y \rightarrow c' , 1 [c' / x] \rightarrow c$$

divergence!

Lazy vs Eager

$$t \triangleq (\lambda x. x + x) (1 \times 2) : int$$

$x : int$

$$t \rightarrow c \quad \swarrow \quad \lambda x. x + x \rightarrow \lambda x'. t' , t' [1 \times 2 / x'] \rightarrow c$$

lazy $\swarrow_{x'=x, t'=x+x} (x + x) [1 \times 2 / x] \rightarrow c$
 $= (1 \times 2) + (1 \times 2) \rightarrow c$

evaluated
twice

$$\swarrow_{c=c_1 \pm c_2} \boxed{(1 \times 2) \rightarrow c_1 , (1 \times 2) \rightarrow c_2}$$

$$\swarrow_{c_1=2, c_2=2}^* \square$$

$$c = c_1 \pm c_2 = 2 \pm 2 = 4$$

$$t \rightarrow c \quad \swarrow \quad \lambda x. x + x \rightarrow \lambda x'. t' , 1 \times 2 \rightarrow c' , t' [c' / x'] \rightarrow c$$

eager $\swarrow_{x'=x, t'=x+x} 1 \times 2 \rightarrow c' , (x + x) [c' / x] \rightarrow c$

$$\swarrow_{c'=2}^* (x + x) [2 / x] \rightarrow c$$
$$= 2 + 2 \rightarrow c$$


$$\swarrow_{c=4}^* \square$$

HOFL

Properties of operational semantics

Termination

termination?

$\forall t. \exists c. t \rightarrow c?$ 

rec $x. x$

Determinacy?

determinacy?

$\forall t. \forall c_1, c_2. t \rightarrow c_1 \wedge t \rightarrow c_2 \Rightarrow c_1 = c_2$?



$P(t \rightarrow c) \triangleq \forall c_1. t \rightarrow c_1 \Rightarrow c_1 = c$

by rule induction (try by yourself)

Subject reduction

(statically assigned types do not change at runtime)

subject reduction? $\forall t. \forall c. \forall \tau. t \rightarrow c \wedge t : \tau \Rightarrow c : \tau$? 

$$P(t \rightarrow c) \triangleq \forall \tau. t : \tau \Rightarrow c : \tau$$

by rule induction (try by yourself)

Congruence?

$$t_1 \equiv_{\text{op}} t_2 \quad \text{iff} \quad \forall c. (t_1 \rightarrow c \Leftrightarrow t_2 \rightarrow c)$$

is it a congruence? 

$$2 \equiv_{\text{op}} 1 + 1$$

$$\lambda x. 2 \not\equiv_{\text{op}} \lambda x. 1 + 1$$

$$\lambda x. 2, \lambda x. 1 + 1 \in C_{\tau \rightarrow \text{int}}$$

$$\lambda x. 2 \rightarrow \lambda x. 2$$

$$\lambda x. 1 + 1 \rightarrow \lambda x. 1 + 1$$