Game Semantics

Andrzej S. Murawski and Nikos Tzevelekos

Day 4: The model for GroundML

Recap: game semantics for GroundML

Recall:

- we started off by introducing GroundML, a higher-order language with full ground references
- we restricted ourselves to ToyML and presented the game semantics of ToyML in terms of regular-like expressions of tagged moves
- we returned to GroundML and starting looking at its game semantics

Recap: game semantics for GroundML

Recall:

- we started off by introducing GroundML, a higher-order language with full ground references
- we restricted ourselves to ToyML and presented the game semantics of ToyML in terms of regular-like expressions of tagged moves
- we returned to GroundML and starting looking at its game semantics formally:
  - **arenas**: represent types, contain moves of the games and structure (pointers, polarity of moves, etc.)
  - **prearenas**: combinations of arenas, where the games are played
  - **plays**: sequences of moves with stores, representing computations
  - **strategies**: sets of even-length plays, representing what plays each term is ready to play

Recall GroundML

$$\frac{}{\mathrm{U},\Gamma\vdash () : \mathsf{unit}} \qquad \frac{i\in\mathbb{Z}}{\mathrm{U},\Gamma\vdash i : \mathsf{int}} \qquad \frac{(x:\theta)\in\Gamma}{\mathrm{U},\Gamma\vdash x:\theta} \qquad \frac{a\in \mathrm{U}\cap\mathbb{A}_\zeta}{\mathrm{U},\Gamma\vdash a : \mathsf{ref}\zeta}$$

$$\frac{\mathrm{U},\Gamma\vdash M : \mathsf{int} \quad \mathrm{U},\Gamma\vdash N_0 : \theta \quad \mathrm{U},\Gamma\vdash N_1 : \theta}{\mathrm{U},\Gamma\vdash \mathsf{if}\ M\ \mathsf{then}\ N_1\ \mathsf{else}\ N_0 : \theta} \qquad \frac{\mathrm{U},\Gamma\vdash M : \mathsf{int}}{\mathrm{U},\Gamma\vdash \mathsf{while}(M) : \mathsf{unit}}$$

$$\frac{\mathrm{U},\Gamma\uplus\{\,x:\theta\,\}\vdash M : \theta'}{\mathrm{U},\Gamma\vdash \lambda x^{\theta}.M : \theta\to\theta'} \qquad \frac{\mathrm{U},\Gamma\vdash M : \theta\to\theta' \quad \mathrm{U},\Gamma\vdash N:\theta}{\mathrm{U},\Gamma\vdash MN : \theta'}$$

$$\frac{\mathrm{U},\Gamma\vdash M : \theta \quad \mathrm{U},\Gamma\vdash N : \theta'}{\mathrm{U},\Gamma\vdash \langle M,N\rangle : \theta\times\theta'} \qquad \frac{\mathrm{U},\Gamma\vdash M : \theta_1\times\theta_2}{\mathrm{U},\Gamma\vdash \pi_i M : \theta_i}\ {\scriptstyle i\in\{1,2\}}$$

$$\frac{\mathrm{U},\Gamma\vdash M : \mathsf{int} \quad \mathrm{U},\Gamma\vdash N : \mathsf{int}}{\mathrm{U},\Gamma\vdash M\oplus N : \mathsf{int}} \qquad \frac{\mathrm{U},\Gamma\vdash M : \mathsf{ref}\zeta \quad \mathrm{U},\Gamma\vdash N : \mathsf{ref}\zeta}{\mathrm{U},\Gamma\vdash M = N : \mathsf{int}}$$

$$\frac{\mathrm{U},\Gamma\vdash M : \zeta}{\mathrm{U},\Gamma\vdash \mathsf{ref}(M) : \mathsf{ref}\zeta} \qquad \frac{\mathrm{U},\Gamma\vdash M : \mathsf{ref}\zeta}{\mathrm{U},\Gamma\vdash\, !M : \zeta} \qquad \frac{\mathrm{U},\Gamma\vdash M : \mathsf{ref}\zeta \quad \mathrm{U},\Gamma\vdash N : \zeta}{\mathrm{U},\Gamma\vdash M := N : \mathsf{unit}}$$

Example strategies

Let us look at some example strategies:

- $[\![y : \mathsf{int} \vdash 2 * y : \mathsf{int} \to \mathsf{int}]\!] : \mathbb{Z} \to \mathbb{Z}$
- $[\![f : \mathsf{int} \to \mathsf{int}, x : \mathsf{int} \vdash fx + 1 : \mathsf{int} \to \mathsf{int}]\!] : ((\mathbb{Z} \Rightarrow \mathbb{Z}) \otimes \mathbb{Z}) \to \mathbb{Z}$
- $[\![\vdash \lambda y^{\mathsf{int}}.\, 2 * y : \mathsf{int} \to \mathsf{int}]\!] : 1 \to (\mathbb{Z} \Rightarrow \mathbb{Z})$
- $[\![f : \mathsf{int} \to \mathsf{int} \vdash \lambda x^{\mathsf{int}}.\, fx + 1 : \mathsf{int} \to \mathsf{int}]\!] : (\mathbb{Z} \Rightarrow \mathbb{Z}) \to (\mathbb{Z} \Rightarrow \mathbb{Z})$
- $[\![\vdash \mathsf{ref}(0)]\!] : 1 \to \mathbb{A}_{\mathsf{int}}$
- $[\![x : \mathsf{ref}\,\mathsf{int} \vdash \lambda z^{\mathsf{int}}.\, x := z; x]\!] : \mathbb{A}_{\mathsf{int}} \to (\mathbb{Z} \Rightarrow \mathbb{A}_{\mathsf{int}})$
- $[\![x : \mathsf{ref}\,\mathsf{int} \vdash \lambda z^{\mathsf{ref}\,\mathsf{int}}.\, x = z]\!] : \mathbb{A}_{\mathsf{int}} \to (\mathbb{A}_{\mathsf{int}} \Rightarrow \mathbb{Z})$

Example strategies

Let us look at some example strategies:

- $[\![y : \mathsf{int} \vdash 2 * y : \mathsf{int} \to \mathsf{int}]\!] : \mathbb{Z} \to \mathbb{Z}$
- $[\![f : \mathsf{int} \to \mathsf{int}, x : \mathsf{int} \vdash f x + 1 : \mathsf{int} \to \mathsf{int}]\!] : ((\mathbb{Z} \Rightarrow \mathbb{Z}) \otimes \mathbb{Z}) \to \mathbb{Z}$
- $[\![\vdash \lambda y^{\mathsf{int}}.\, 2 * y : \mathsf{int} \to \mathsf{int}]\!] : 1 \to (\mathbb{Z} \Rightarrow \mathbb{Z})$
- $[\![f : \mathsf{int} \to \mathsf{int} \vdash \lambda x^{\mathsf{int}}.\, f x + 1 : \mathsf{int} \to \mathsf{int}]\!] : (\mathbb{Z} \Rightarrow \mathbb{Z}) \to (\mathbb{Z} \Rightarrow \mathbb{Z})$
- $[\![\vdash \mathsf{ref}(0)]\!] : 1 \to \mathbb{A}_{\mathsf{int}}$
- $[\![x : \mathsf{ref\,int} \vdash \lambda z^{\mathsf{int}}.\, x := z; x]\!] : \mathbb{A}_{\mathsf{int}} \to (\mathbb{Z} \Rightarrow \mathbb{A}_{\mathsf{int}})$
- $[\![x : \mathsf{ref\,int} \vdash \lambda z^{\mathsf{ref\,int}}.\, x = z]\!] : \mathbb{A}_{\mathsf{int}} \to (\mathbb{A}_{\mathsf{int}} \Rightarrow \mathbb{Z})$

– how did we get them??

Translating GroundML terms into strategies

So far, we have seen the static ingredients of the game model for GroundML: arenas, prearenas, moves, plays, strategies.

- With them, we can define the translation $[\![\_]\!]$ for basic terms, such as:

$$[\![\vdash \mathsf{ref}(0)]\!] = \{ \star\ a^{(a,0)} \mid a \in \mathbb{A}_{\mathsf{int}} \}$$

- To be able to translate larger terms, we need:
  - for every syntactic construct (e.g. composition, λ-abstraction, etc.)
  - to define a corresponding construction on strategies.

We start off with the most fundamental construct: strategy composition.

We first look at a few example compositions, then define composition formally.

Strategy composition

Strategy composition is the following operation:

- given strategies: $\sigma : A \to B$ and $\tau : B \to C$
- define a strategy: $\sigma;\tau : A \to C$
  that composes the behaviours of σ and τ.

Pictorially:

$$\frac{A \xrightarrow{\sigma} B \xrightarrow{\tau} C}{A \xrightarrow{\sigma;\tau} C}$$

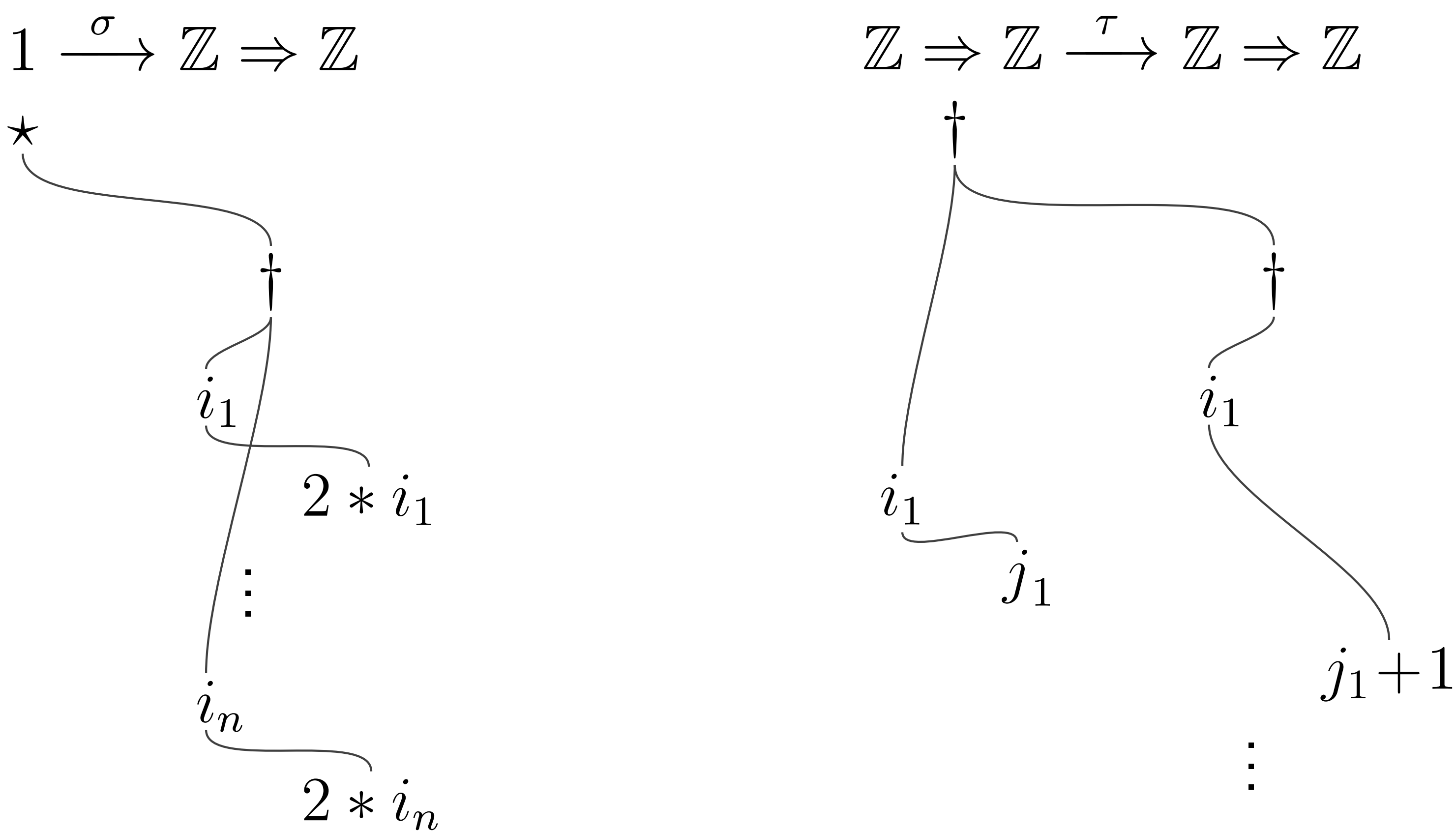
Note: we read composite strategies left-to-right, hence we write $\sigma;\tau$.
This is equivalent to writing $\tau \circ \sigma$ (in function notation).

Composition example

Recall our example terms:

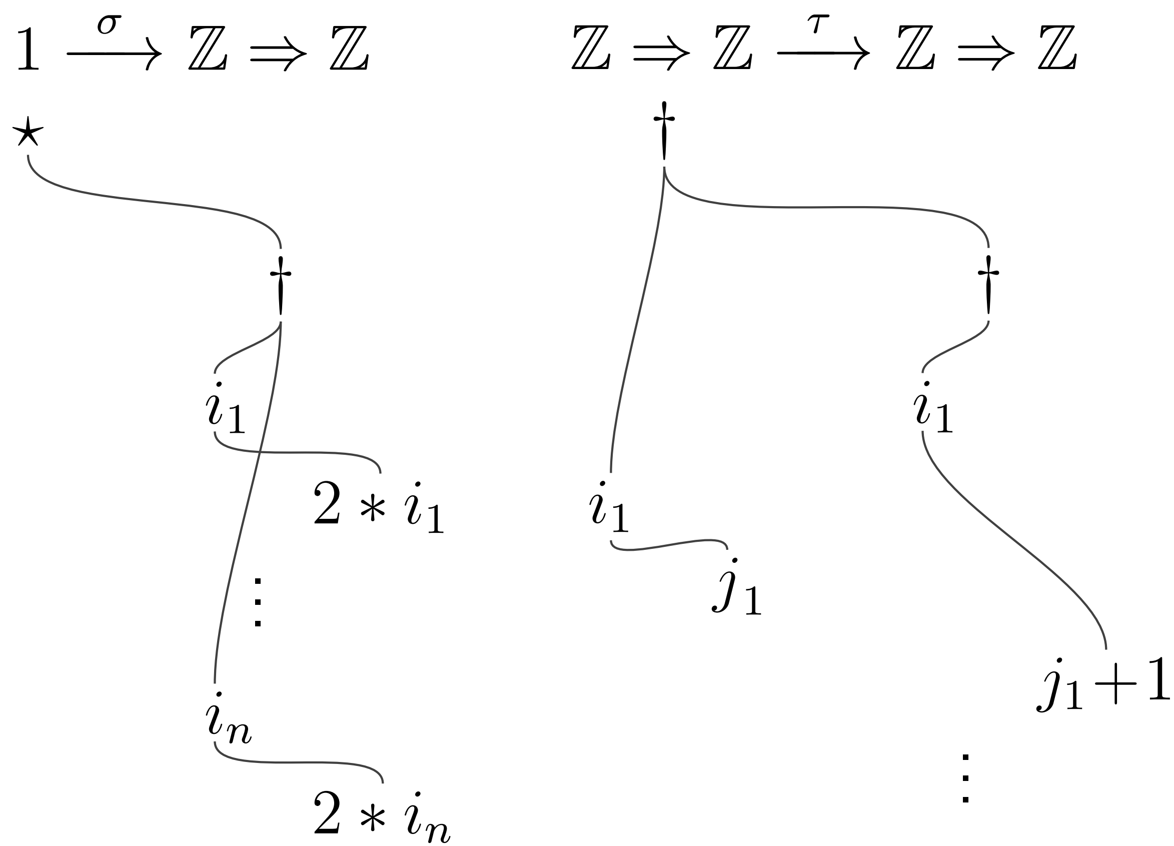
$$\frac{\vdash \lambda y^{\mathsf{int}}.\, 2 * y : \mathsf{int} \to \mathsf{int} \quad \text{and} \quad f : \mathsf{int} \to \mathsf{int} \vdash \lambda x^{\mathsf{int}}.\, f x + 1 : \mathsf{int} \to \mathsf{int}}{\vdash \mathsf{let}\ f = \lambda y^{\mathsf{int}}.\, 2 * y\ \ \mathsf{in}\ \ \lambda x^{\mathsf{int}}.\, f x + 1 : \mathsf{int} \to \mathsf{int}}$$

The corresponding strategies $\sigma = [\![\lambda y^{\mathsf{int}}.\, 2 * y]\!]$ and $\tau = [\![\lambda x^{\mathsf{int}}.\, f x + 1]\!]$:



Composition example – sync and hide

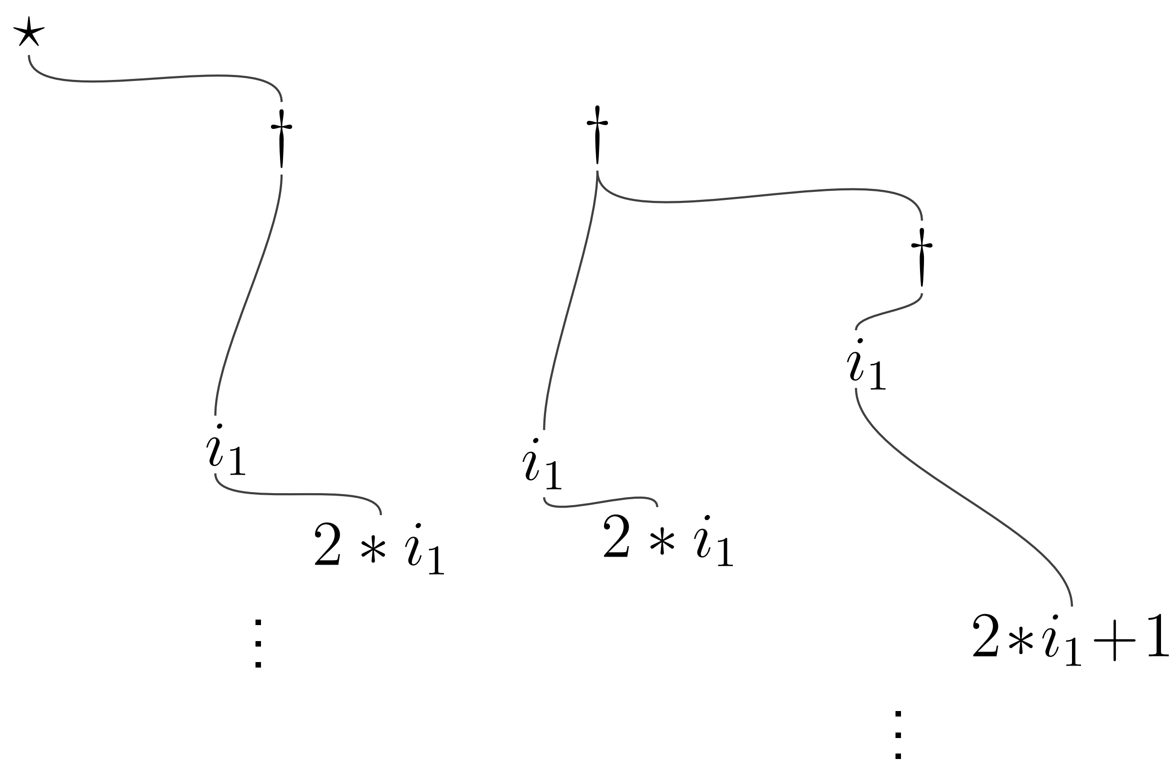
Taking $\sigma = [\![\lambda y^{\mathsf{int}}.\, 2 * y]\!]$ and $\tau = [\![\lambda x^{\mathsf{int}}.\, f x + 1]\!]$:



Composition example – sync and hide

Taking $\sigma = [\![\lambda y^{\mathsf{int}}.\, 2 * y]\!]$ and $\tau = [\![\lambda x^{\mathsf{int}}.\, f x + 1]\!]$:

$$1 \xrightarrow{\sigma} \mathbb{Z} \Rightarrow \mathbb{Z} \qquad \mathbb{Z} \Rightarrow \mathbb{Z} \xrightarrow{\tau} \mathbb{Z} \Rightarrow \mathbb{Z}$$



Composition example – sync and hide

Taking $\sigma = [\![\lambda y^{\mathsf{int}}.\, 2 * y]\!]$ and $\tau = [\![\lambda x^{\mathsf{int}}.\, f x + 1]\!]$:

$$1 \xrightarrow{\ \sigma\ } \mathbb{Z} \Rightarrow \mathbb{Z} \xrightarrow{\ \tau\ } \mathbb{Z} \Rightarrow \mathbb{Z}$$

$$\star$$

$$\dagger$$

$$\dagger$$

$$i_1$$

$$i_1$$

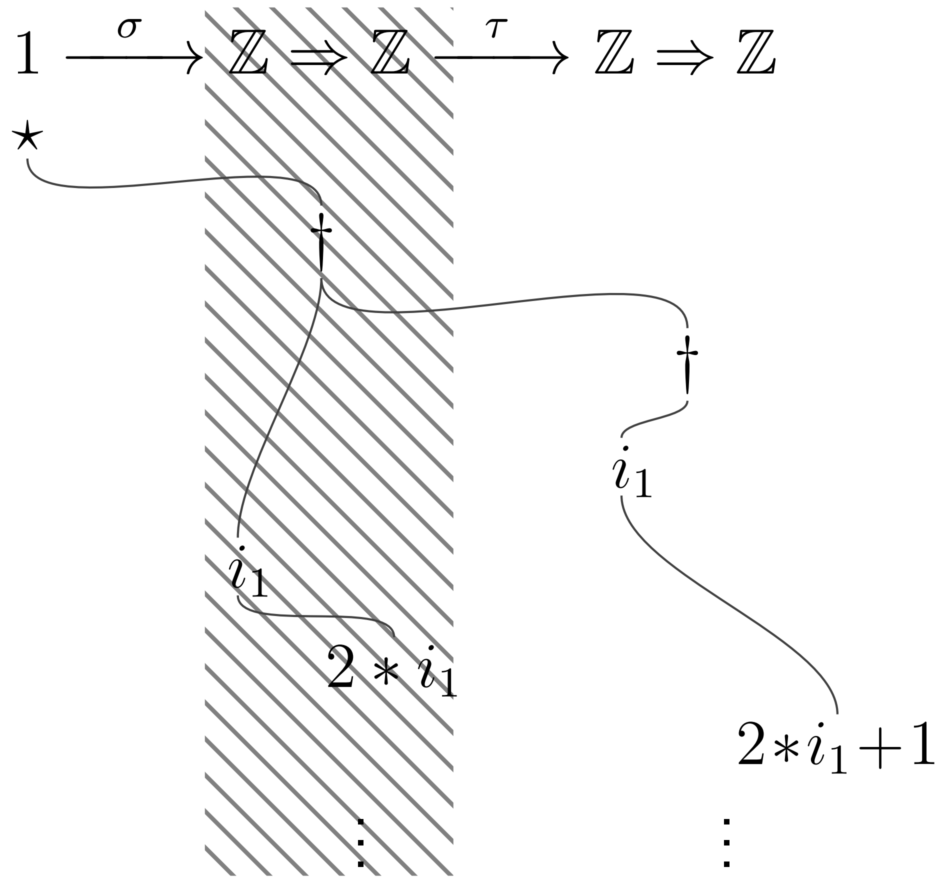
$$2 * i_1$$

$$2 * i_1 + 1$$

$$\vdots \qquad\qquad \vdots$$

Composition example – sync and hide

Taking $\sigma = [\![\lambda y^{\mathsf{int}}.\, 2 * y]\!]$ and $\tau = [\![\lambda x^{\mathsf{int}}.\, f x + 1]\!]$:



Composition example – sync and hide

Taking $\sigma = [\![\lambda y^{\mathsf{int}}.\, 2 * y]\!]$ and $\tau = [\![\lambda x^{\mathsf{int}}.\, f x + 1]\!]$:

$$
\begin{array}{lcr}
1 & \xrightarrow{\qquad\qquad \sigma;\tau \qquad\qquad} & \mathbb{Z} \Rightarrow \mathbb{Z} \\
\star & & \\
 & & \dagger \\
 & & i_1 \\
 & & 2 * i_1 + 1 \\
 & & \vdots
\end{array}
$$

Thus: $\sigma;\tau = \{\, \star \;\; \dagger \;\; i_1 \;\; (2 * i_1 + 1) \;\; i_2 \;\; (2 * i_2 + 1) \cdots \}$

and that is also $[\![\mathsf{let}\ \ f = \lambda y^{\mathsf{int}}.\, 2 * y\ \ \mathsf{in}\ \ \lambda x^{\mathsf{int}}.\, f x + 1]\!]$.

Composition example II

We look at an example with name generation:

$$\frac{\vdash \mathsf{ref}(0) : \mathsf{ref\,int} \qquad \text{and} \qquad x : \mathsf{ref\,int} \vdash \lambda z^{\mathsf{int}}.\, x := z; x : \mathsf{int} \to \mathsf{ref\,int}}{\vdash \mathsf{let}\ \ x = \mathsf{ref}(0)\ \ \mathsf{in}\ \ \lambda z^{\mathsf{int}}.\, x := z; x : \mathsf{int} \to \mathsf{ref\,int}}$$

The corresponding strategies $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{int}}.\, x := z; x]\!]$:

$$1 \xrightarrow{\sigma} \mathbb{A}_{\mathsf{int}}$$

$$\star$$

$$a^{(a,0)}$$

$$\mathbb{A}_{\mathsf{int}} \xrightarrow{\tau} \mathbb{Z} \Rightarrow \mathbb{A}_{\mathsf{int}}$$

$$a^{(a,i_0)}$$

$$\dagger^{(a,i_0)}$$

$$i_1^{(a,i'_0)}$$

$$a^{(a,i_1)}$$

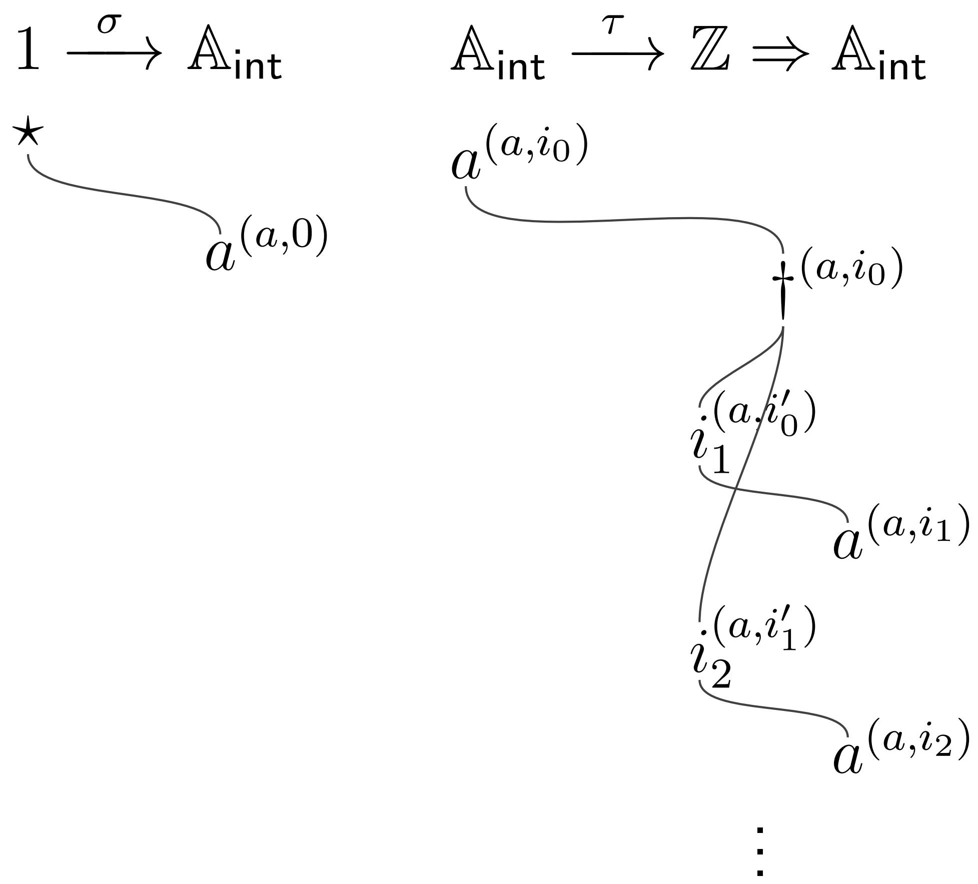
$$i_2^{(a,i'_1)}$$

$$a^{(a,i_2)}$$

$$\vdots$$

Composition example II – sync and hide

Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{int}}.\, x := z; x]\!]$:



Composition example II – sync and hide

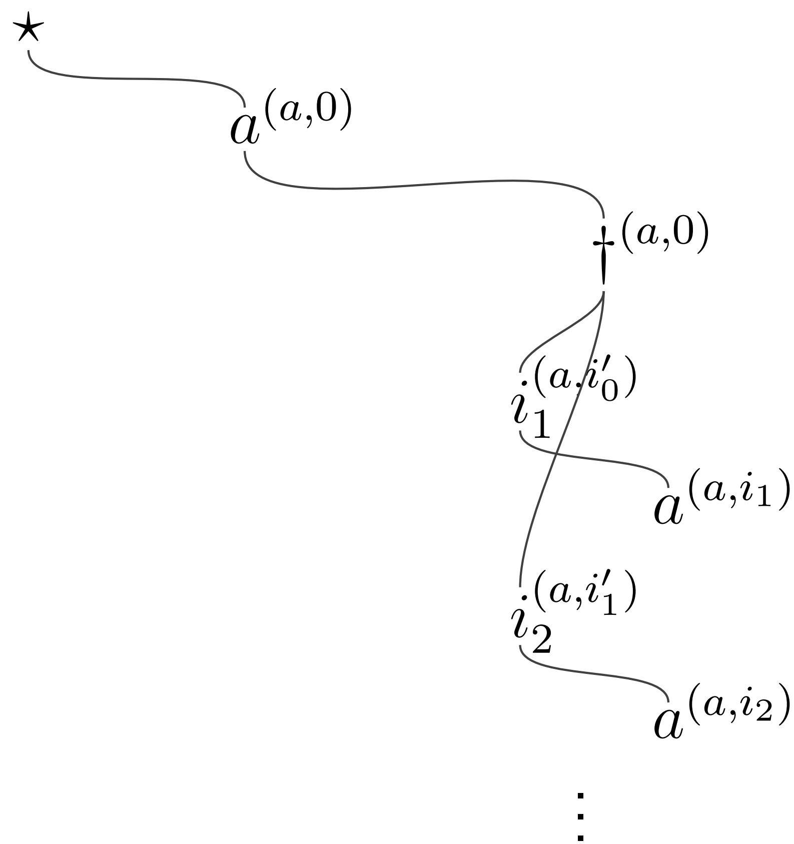
Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{int}}.\, x := z; x]\!]$:

$$
\begin{array}{lllllll}
1 & \xrightarrow{\sigma} & \mathbb{A}_{\mathsf{int}} & \qquad & \mathbb{A}_{\mathsf{int}} \xrightarrow{\tau} & \mathbb{Z} & \Rightarrow \mathbb{A}_{\mathsf{int}} \\
\star & & & & & & \\
 & & a^{(a,0)} & & & & \\
 & & & & a^{(a,0)} & & \\
 & & & & & & \dagger^{(a,0)} \\
 & & & & & i_1^{(a,i'_0)} & \\
 & & & & & & a^{(a,i_1)} \\
 & & & & & i_2^{(a,i'_1)} & \\
 & & & & & & a^{(a,i_2)} \\
 & & & & & \vdots &
\end{array}
$$

Composition example II – sync and hide

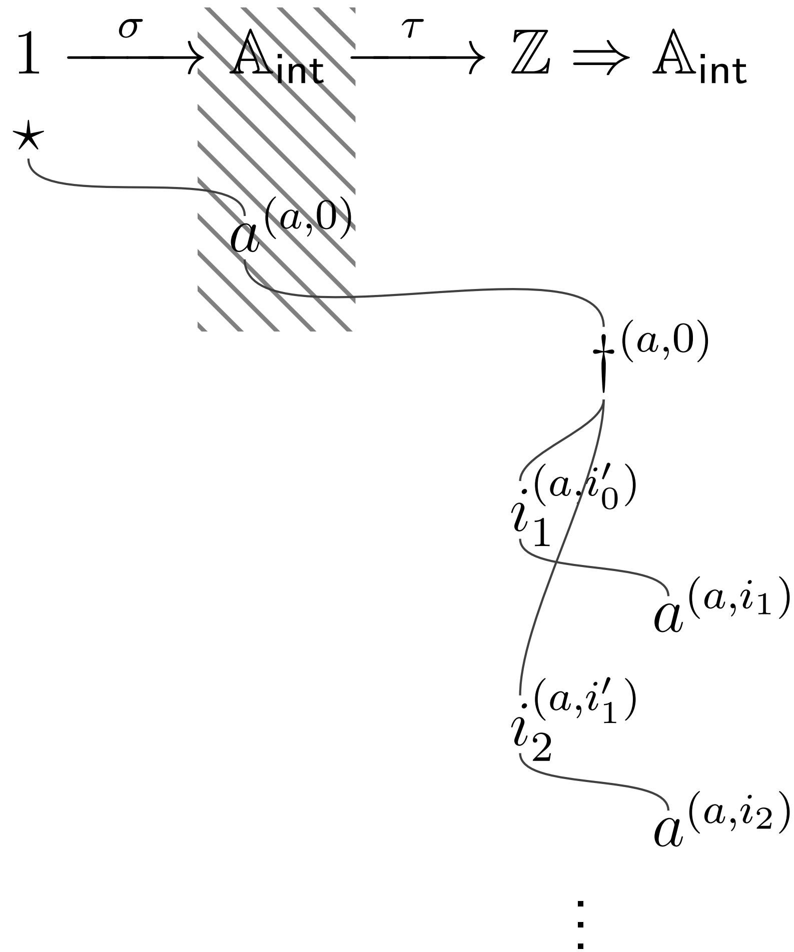
Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{int}}.\, x := z; x]\!]$:

$$1 \xrightarrow{\sigma} \mathbb{A}_{\mathsf{int}} \xrightarrow{\tau} \mathbb{Z} \Rightarrow \mathbb{A}_{\mathsf{int}}$$



Composition example II – sync and hide

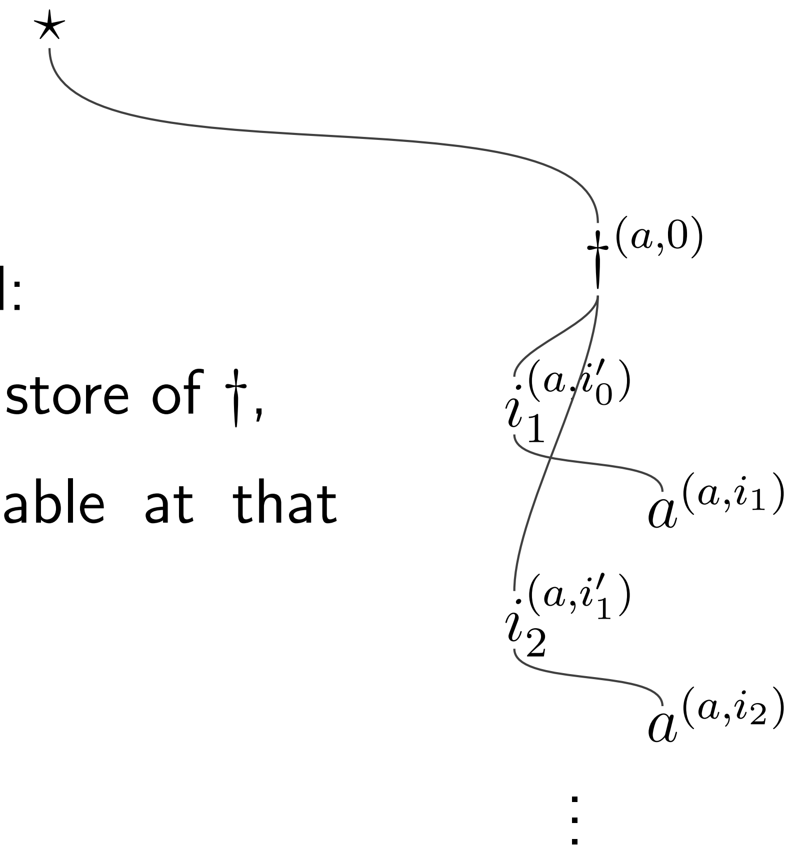
Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{int}}.\, x := z; x]\!]$:



Composition example II – sync and hide

Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{int}}.\, x := z; x]\!]$:

$$1 \xrightarrow{\sigma;\tau} \mathbb{Z} \Rightarrow \mathbb{A}_{\mathsf{int}}$$



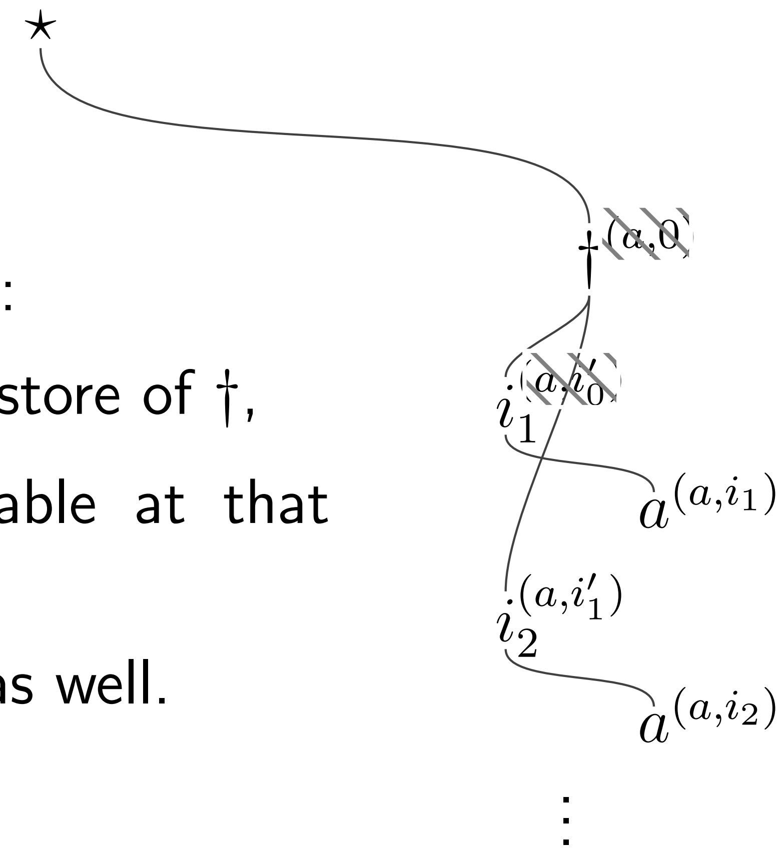
Extra care is needed:

- a appears in the store of $\dagger$,
- but is not available at that point!

Composition example II – sync and hide

Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{int}}.\, x := z; x]\!]$:

$$1 \xrightarrow{\sigma;\tau} \mathbb{Z} \Rightarrow \mathbb{A}_{\mathsf{int}}$$



Extra care is needed:

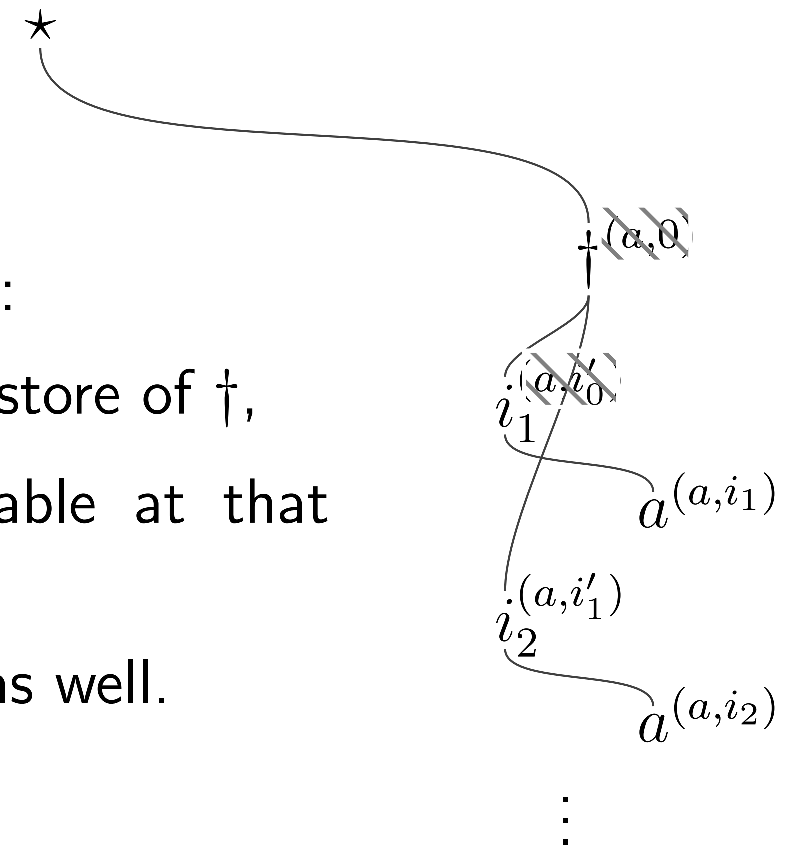
- a appears in the store of †,
- but is not available at that point!

We need to hide it as well.

Composition example II – sync and hide

Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{int}}.\, x := z; x]\!]$:

$$1 \xrightarrow{\sigma;\tau} \mathbb{Z} \Rightarrow \mathbb{A}_{\mathsf{int}}$$



Extra care is needed:

- a appears in the store of $\dagger$,
- but is not available at that point!

We need to hide it as well.

Thus: $\sigma;\tau = \{\, \star \;\; \dagger \;\; i_1 \;\; a^{(a,i_1)} \;\; i_2^{(a,i_1')} \;\; a^{(a,i_2)} \cdots \}$

and that is also $[\![\mathsf{let}\ x = \mathsf{ref}(0)\ \mathsf{in}\ \lambda z^{\mathsf{int}}.\, x := z; x]\!]$.

Composition example III

We look at another example with name generation:

$$\frac{\vdash \mathsf{ref}(0) : \mathsf{ref\,int} \qquad \mathsf{and} \qquad x : \mathsf{ref\,int} \vdash \lambda z^{\mathsf{ref\,int}}.\, x = z : \mathsf{ref\,int} \to \mathsf{int}}{\vdash \mathsf{let}\ \ x = \mathsf{ref}(0)\ \ \mathsf{in}\ \ \lambda z^{\mathsf{ref\,int}}.\, x = z : \mathsf{ref\,int} \to \mathsf{int}}$$

The corresponding strategies $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{ref\,int}}.\, x = z]\!]$:

$$1 \xrightarrow{\sigma} \mathbb{A}_{\mathsf{int}}$$

$\star$

$a^{(a,0)}$

$$\mathbb{A}_{\mathsf{int}} \xrightarrow{\tau} \mathbb{A}_{\mathsf{int}} \Rightarrow \mathbb{Z}$$

$a^{(a,i_0)}$

$\dagger^{(a,i_0)}$

a^{S_1}

1^{S_1}

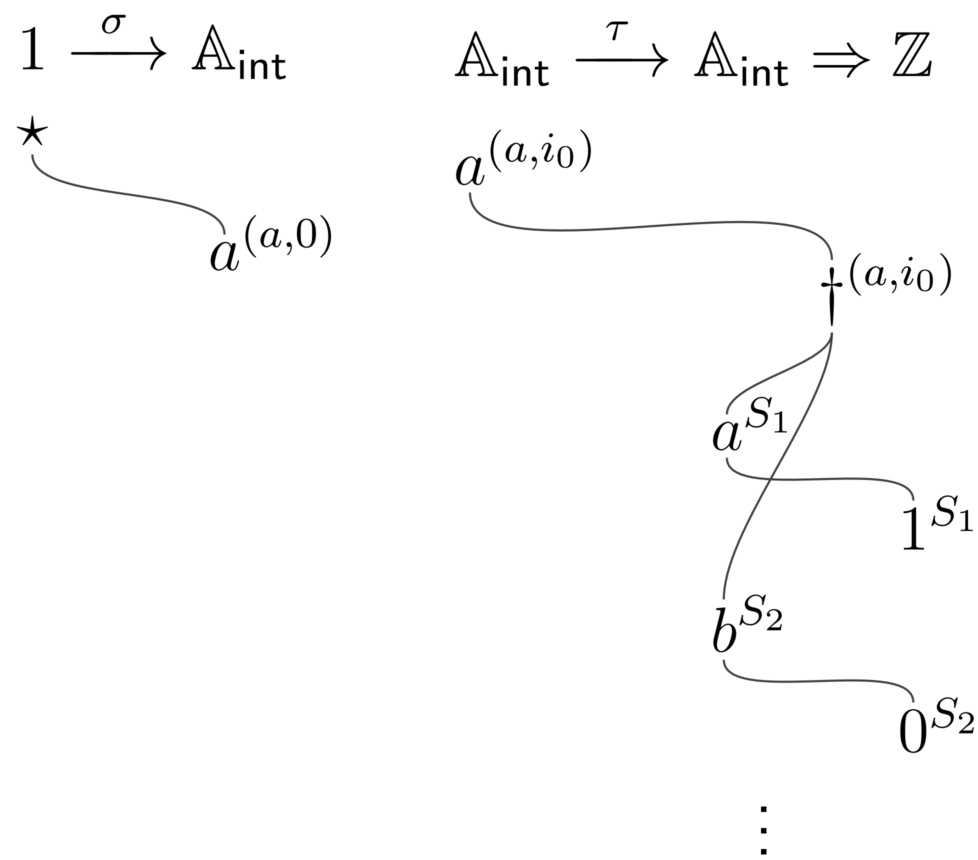
b^{S_2}

0^{S_2}

$\vdots$

Composition example III – sync and hide

Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{ref\,int}}.\, x = z]\!]$:



Composition example III – sync and hide

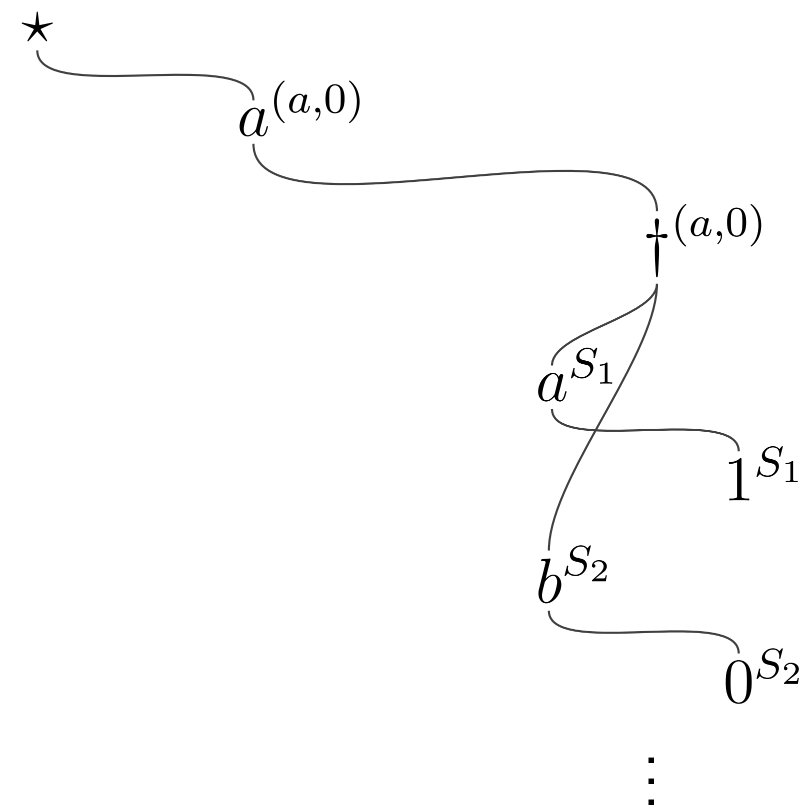
Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{ref\,int}}.\, x = z]\!]$:

$$
\begin{array}{ccccccccc}
1 & \xrightarrow{\sigma} & \mathbb{A}_{\mathsf{int}} & \qquad & \mathbb{A}_{\mathsf{int}} & \xrightarrow{\tau} & \mathbb{A}_{\mathsf{int}} & \Rightarrow & \mathbb{Z} \\
\star & & & & & & & & \\
 & & a^{(a,0)} & & a^{(a,0)} & & & & \\
 & & & & & & & \dagger^{(a,0)} & \\
 & & & & & & a^{S_1} & & \\
 & & & & & & & & 1^{S_1} \\
 & & & & & & b^{S_2} & & \\
 & & & & & & & & 0^{S_2} \\
 & & & & & & & \vdots &
\end{array}
$$

Composition example III – sync and hide

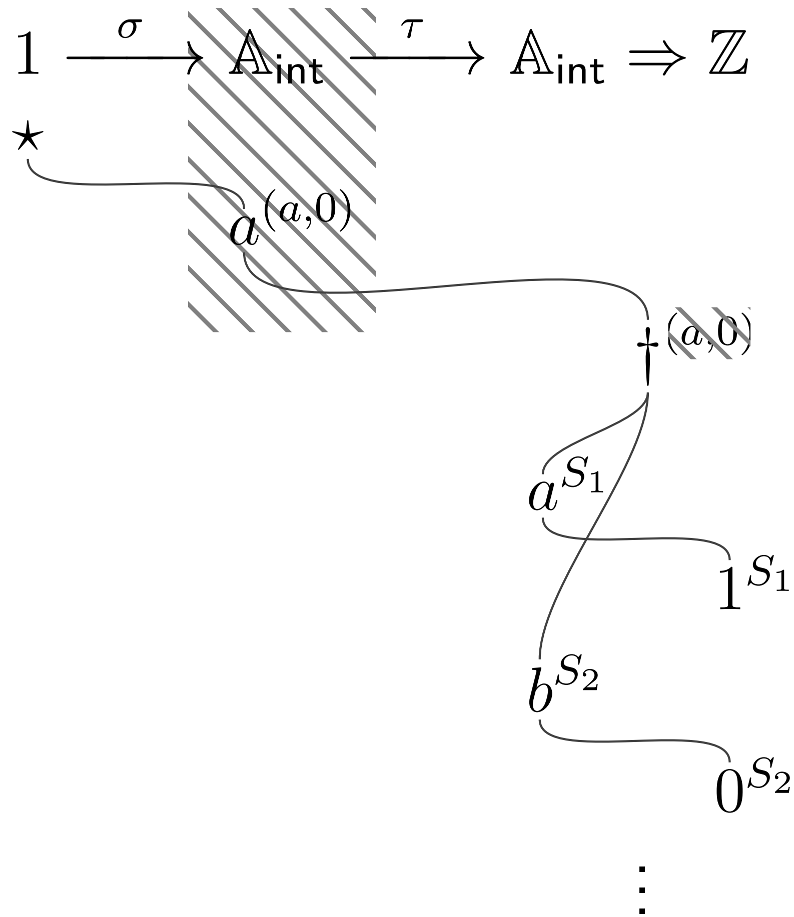
Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{ref\,int}}.\, x = z]\!]$:

$$1 \xrightarrow{\sigma} \mathbb{A}_{\mathsf{int}} \xrightarrow{\tau} \mathbb{A}_{\mathsf{int}} \Rightarrow \mathbb{Z}$$



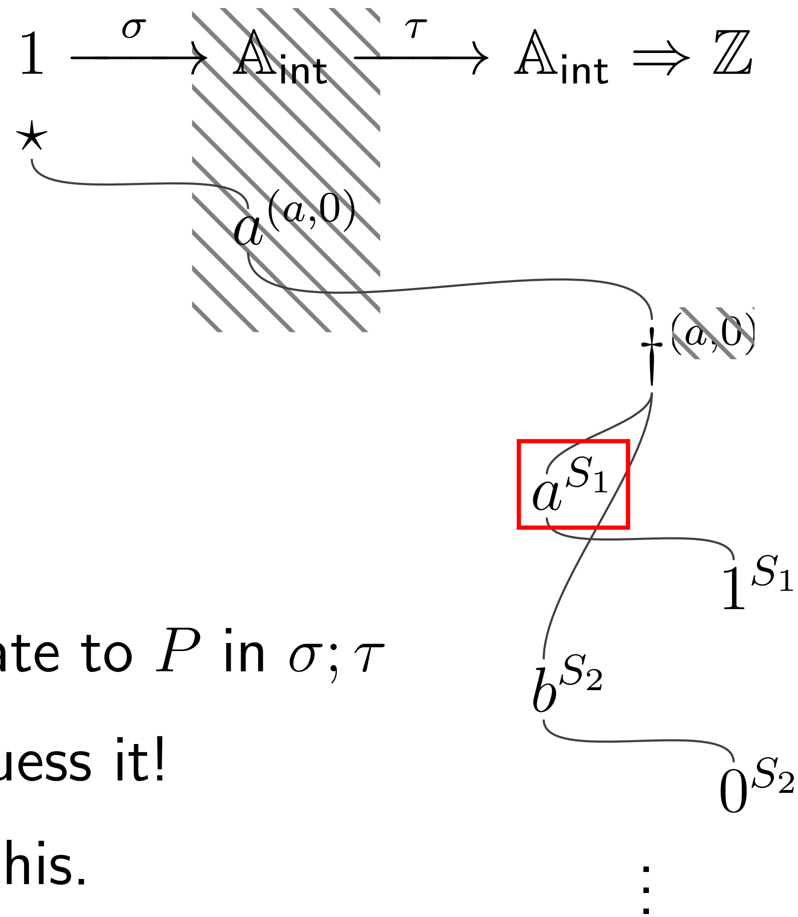
Composition example III – sync and hide

Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{ref\,int}}.\, x = z]\!]$:



Composition example III – sync and hide

Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{ref\,int}}.\, x = z]\!]$:



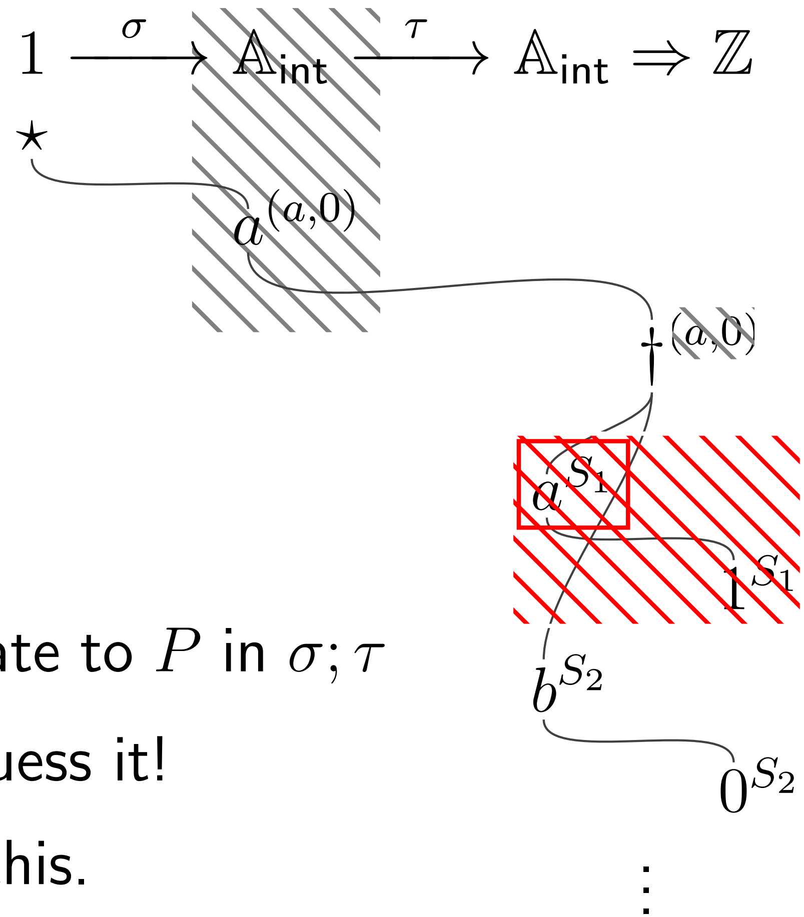
Extra care is needed:

- the name a is private to P in $\sigma;\tau$
- but O is able to guess it!

We need to disallow this.

Composition example III – sync and hide

Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{ref\,int}}.\, x = z]\!]$:



Extra care is needed:

- the name a is private to P in $\sigma;\tau$
- but O is able to guess it!

We need to disallow this.

Composition example III – sync and hide

Taking $\sigma = [\![\mathsf{ref}(0)]\!]$ and $\tau = [\![\lambda z^{\mathsf{ref\,int}}.\, x = z]\!]$:

$$1 \xrightarrow{\quad\sigma;\tau\quad} \mathbb{A}_{\mathsf{int}} \Rightarrow \mathbb{Z}$$

$$\star$$

$$\dagger \; \cancel{(a,0)}$$

$$b^{S_2}$$

$$0^{S_2}$$

$$\vdots$$

Extra care is needed:

- the name a is private to P in $\sigma;\tau$
- but O is able to guess it!

We need to disallow this.

Thus: $\sigma;\tau = \{\, \star \;\; \dagger \;\; b_1^{S_1} \;\; 0^{S_1} \;\; b_2^{S_2} \;\; 0^{S_2} \cdots \}$

and that is also $[\![\mathsf{let}\;\; x = \mathsf{ref}(0) \;\; \mathsf{in} \;\; \lambda z^{\mathsf{ref\,int}}.\, x = z]\!]$.

Composition example IV

We look at an example with name generation in both strategies:

$$\frac{f : \mathsf{ref\,int} \to \mathsf{int} \vdash f(\mathsf{ref}(0)) : \mathsf{int} \qquad \text{and} \qquad x : \mathsf{int} \vdash \mathsf{ref}(x) : \mathsf{ref\,int}}{f : \mathsf{ref\,int} \to \mathsf{int} \vdash \mathsf{let}\ x = f(\mathsf{ref}(0))\ \mathsf{in}\ \mathsf{ref}(x) : \mathsf{ref\,int}}$$

The corresponding strategies $\sigma = [\![f(\mathsf{ref}(0))]\!]$ and $\tau = [\![\mathsf{ref}(x)]\!]$:

$$\mathbb{A}_{\mathsf{int}} \Rightarrow \mathbb{Z} \xrightarrow{\sigma} \mathbb{Z}$$

$$\dagger$$

$$a^{(a,0)}$$

$$i^{(a,j)}$$

$$i^{(a,j)}$$

$$\mathbb{Z} \xrightarrow{\tau} \mathbb{A}_{\mathsf{int}}$$

$$i$$

$$a^{(a,i)}$$

Composition example IV – sync and hide

Taking $\sigma = [\![f : \mathsf{ref\,int} \to \mathsf{int} \vdash f(\mathsf{ref}(0)) : \mathsf{int}]\!]$ and $\tau = [\![\mathsf{ref}(x)]\!]$:

$$\mathbb{A}_{\mathsf{int}} \Rightarrow \mathbb{Z} \xrightarrow{\sigma} \mathbb{Z} \qquad\qquad \mathbb{Z} \xrightarrow{\tau} \mathbb{A}_{\mathsf{int}}$$

$$\dagger \qquad\qquad i$$

$$a^{(a,0)} \qquad\qquad a^{(a,i)}$$

$$i^{(a,j)}$$

$$i^{(a,j)}$$

Composition example IV – sync and hide

Taking $\sigma = [\![f : \mathsf{ref}\,\mathsf{int} \to \mathsf{int} \vdash f(\mathsf{ref}(0)) : \mathsf{int}]\!]$ and $\tau = [\![\mathsf{ref}(x)]\!]$:

$$
\begin{array}{ccccc}
\mathbb{A}_{\mathsf{int}} \Rightarrow & \mathbb{Z} \xrightarrow{\sigma} & \mathbb{Z} & \qquad \mathbb{Z} \xrightarrow{\tau} & \mathbb{A}_{\mathsf{int}} \\
& \dagger & & & \\
a^{(a,0)} & & & & \\
& i^{(a,j)} & & & \\
& & i^{(a,j)} & i & \\
& & & & a^{(a,i)}
\end{array}
$$

Composition example IV – sync and hide

Taking $\sigma = [\![f : \mathsf{ref\,int} \to \mathsf{int} \vdash f(\mathsf{ref}(0)) : \mathsf{int}]\!]$ and $\tau = [\![\mathsf{ref}(x)]\!]$:

$$\mathbb{A}_{\mathsf{int}} \Rightarrow \mathbb{Z} \xrightarrow{\sigma} \mathbb{Z} \qquad\qquad \mathbb{Z} \xrightarrow{\tau} \mathbb{A}_{\mathsf{int}}$$

$$\dagger$$

$$a^{(a,0)}$$

$$i^{(a,j)}$$

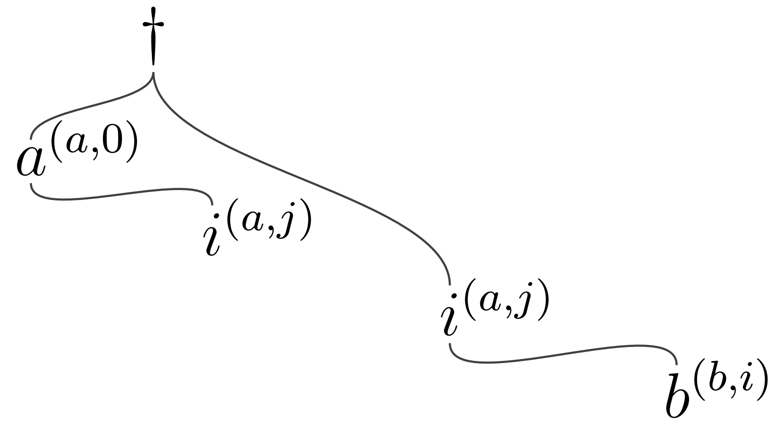
$$i^{(a,j)} \qquad\qquad i$$

$$b^{(b,i)}$$

Composition example IV – sync and hide

Taking $\sigma = [\![f : \mathsf{ref\,int} \to \mathsf{int} \vdash f(\mathsf{ref}(0)) : \mathsf{int}]\!]$ and $\tau = [\![\mathsf{ref}(x)]\!]$:

$$\mathbb{A}_{\mathsf{int}} \Rightarrow \mathbb{Z} \xrightarrow{\ \sigma\ } \mathbb{Z} \xrightarrow{\ \tau\ } \mathbb{A}_{\mathsf{int}}$$



Extra care:

- the name a should be carried over to the last move
- its value should remain the same

Composition example IV – sync and hide

Taking $\sigma = [\![f : \mathsf{ref\,int} \to \mathsf{int} \vdash f(\mathsf{ref}(0)) : \mathsf{int}]\!]$ and $\tau = [\![\mathsf{ref}(x)]\!]$:

$$\mathbb{A}_{\mathsf{int}} \Rightarrow \mathbb{Z} \xrightarrow{\ \sigma\ } \mathbb{Z} \xrightarrow{\ \tau\ } \mathbb{A}_{\mathsf{int}}$$

$$\dagger$$

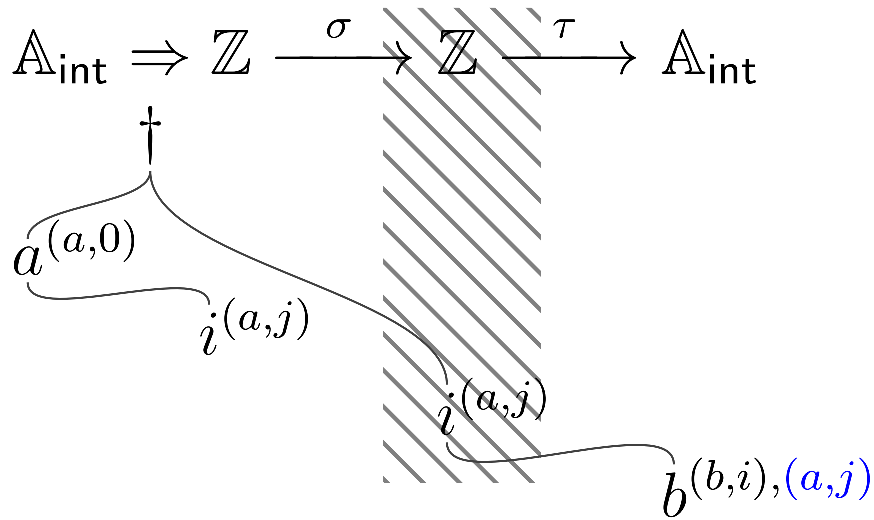
$$a^{(a,0)} \qquad i^{(a,j)} \qquad i^{(a,j)} \qquad b^{(b,i),(a,j)}$$

Extra care:

- the name a should be carried over to the last move
- its value should remain the same

Composition example IV – sync and hide

Taking $\sigma = [\![f : \mathsf{ref\,int} \to \mathsf{int} \vdash f(\mathsf{ref}(0)) : \mathsf{int}]\!]$ and $\tau = [\![\mathsf{ref}(x)]\!]$:



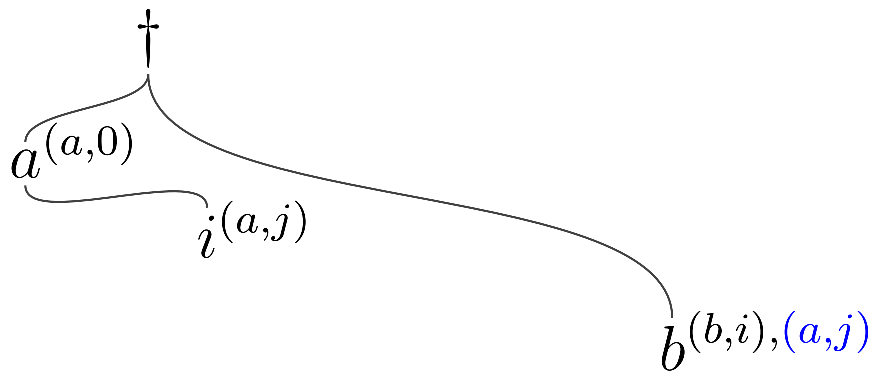
Extra care:

- the name a should be carried over to the last move
- its value should remain the same

Composition example IV – sync and hide

Taking $\sigma = \llbracket f : \mathsf{ref\,int} \to \mathsf{int} \vdash f(\mathsf{ref}(0)) : \mathsf{int} \rrbracket$ and $\tau = \llbracket \mathsf{ref}(x) \rrbracket$:

$$\mathbb{A}_{\mathsf{int}} \Rightarrow \mathbb{Z} \xrightarrow{\sigma;\tau} \mathbb{A}_{\mathsf{int}}$$



Extra care:

- the name a should be carried over to the last move
- its value should remain the same

Thus: $\sigma;\tau = \{\, \dagger \;\; a^{(a,0)} \;\; i^{(a,j)} \;\; b^{(a,j),(b,i)} \;\; \cdots \}$

and that is also $\llbracket \mathsf{let}\ x = f(\mathsf{ref}(0))\ \mathsf{in}\ \mathsf{ref}(x) \rrbracket$.

Composition formally

Thus, when composing strategies with names, we need to impose additional constraints on names and their privacy, such as:

- a name that is private in one strategy (by P) cannot be guessed by the other (by P), nor by the overall O
- any names that after composition remain in a store without being available should be removed from that store

Composition formally

Thus, when composing strategies with names, we need to impose additional constraints on names and their privacy, such as:

- a name that is private in one strategy (by P) cannot be guessed by the other (by P), nor by the overall O
- any names that after composition remain in a store without being available should be removed from that store

Formally, in order to compose strategies $\sigma : A \to B$ and $\tau : B \to C$, we define a notion of play between the three arenas A, B, C, i.e. in:

$$A \longrightarrow B \longrightarrow C$$

and impose on these plays conditions like above to ensure name privacy is ensured.

Name ownership and availability

When composing plays in $A \to B \to C$ it is important to know:

- **name ownership**: which player produced what names
  - what names did P in $A \to B$ produce
  - what names did P in $B \to C$ produce
  - what names did O in $A \to C$ produce

Name ownership and availability

When composing plays in $A \to B \to C$ it is important to know:

- **name ownership**: which player produced what names
  - what names did P in $A \to B$ produce
  - what names did P in $B \to C$ produce
  - what names did O in $A \to C$ produce
- **name availability**: what names have been revealed (and are not private to a player)

Interaction sequences

Let γ restrict stores of move sequences to available names, and $\restriction X$ restrict sequences to moves from component X. Also: $\_ \restriction_\gamma X = \gamma(\_ \restriction X)$.

A justified sequence u on $A \to B \to C$ is an **interaction sequence** if $(u \restriction_\gamma AB) \in P_{A \to B}$, $(u \restriction_\gamma BC) \in P_{B \to C}$ and:

- u is frugal, that is, $\gamma(u) = u$;
- $\mathsf{P}(u \restriction_\gamma AB) \cap \mathsf{P}(u \restriction_\gamma BC) = \emptyset$;
- $\mathsf{O}(u \restriction_\gamma AC) \cap (\mathsf{P}(u \restriction_\gamma AB) \cup \mathsf{P}(u \restriction_\gamma BC)) = \emptyset$;

Interaction sequences

Let γ restrict stores of move sequences to available names, and $\upharpoonright X$ restrict sequences to moves from component X. Also: $\_ \upharpoonright_\gamma X = \gamma(\_ \upharpoonright X)$.

A justified sequence u on $A \to B \to C$ is an **interaction sequence** if $(u \upharpoonright_\gamma AB) \in P_{A \to B}$, $(u \upharpoonright_\gamma BC) \in P_{B \to C}$ and:

- u is frugal, that is, $\gamma(u) = u$;
- $\mathsf{P}(u \upharpoonright_\gamma AB) \cap \mathsf{P}(u \upharpoonright_\gamma BC) = \emptyset$;
- $\mathsf{O}(u \upharpoonright_\gamma AC) \cap (\mathsf{P}(u \upharpoonright_\gamma AB) \cup \mathsf{P}(u \upharpoonright_\gamma BC)) = \emptyset$;
- for each $u' \sqsubseteq u$ ending in $m^S m'^{S'}$ and $a \in \mathsf{dom}(S')$ if
  - m' is a P-move in AB and $a \notin \mathsf{Av}(u' \upharpoonright AB)$,
  - or m' is a P-move in BC and $a \notin \mathsf{Av}(u' \upharpoonright BC)$,
  - or m' is an O-move in AC and $a \notin \mathsf{Av}(u' \upharpoonright AC)$,

  then $S(a) = S'(a)$.

We write $Int(ABC)$ for the set of interaction sequences on ABC.

Interaction sequence examples

$$1 \longrightarrow \mathbb{Z} \Rightarrow \mathbb{Z} \longrightarrow \mathbb{Z} \Rightarrow \mathbb{Z}$$

$\star$

$\dagger$

$\dagger$

i_1

i_1

$2 * i_1$

$2 * i_1 + 1$

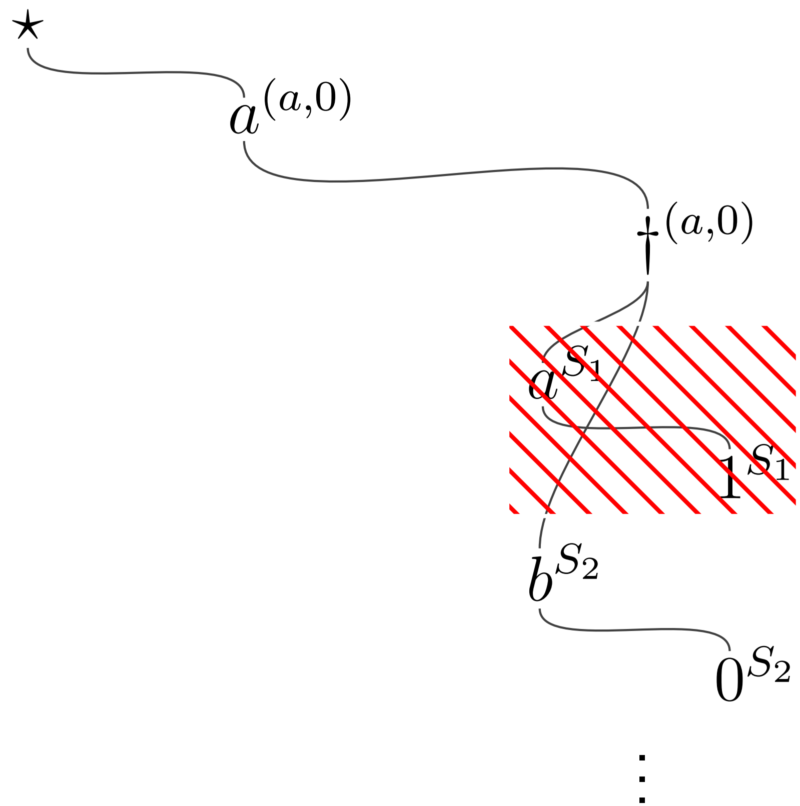
$\vdots$ $\vdots$

Interaction sequence examples

$$
\begin{array}{ccccccc}
1 & \xrightarrow{\sigma} & \mathbb{A}_{\mathsf{int}} & \xrightarrow{\tau} & \mathbb{Z} & \Rightarrow & \mathbb{A}_{\mathsf{int}} \\
\star & & & & & & \\
 & & a^{(a,0)} & & & & \\
 & & & & +^{(a,0)} & & \\
 & & & & i_1^{(a,i'_0)} & & \\
 & & & & & & a^{(a,i_1)} \\
 & & & & i_2^{(a,i'_1)} & & \\
 & & & & & & a^{(a,i_2)} \\
 & & & & & \vdots &
\end{array}
$$

Interaction sequence examples

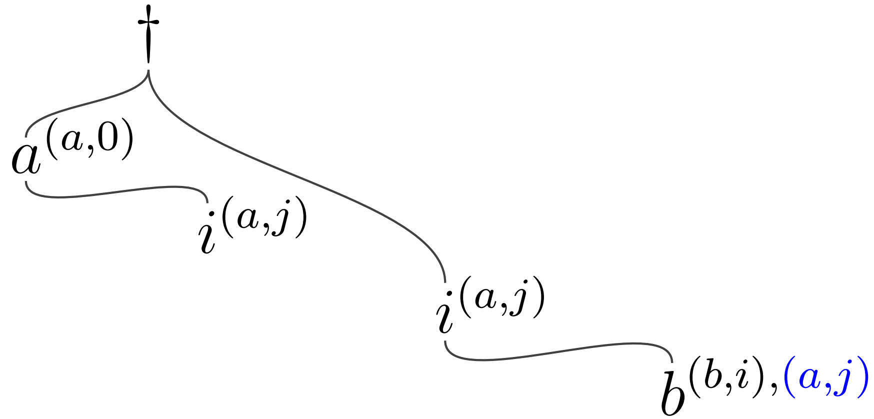
$$1 \xrightarrow{\sigma} \mathbb{A}_{\mathsf{int}} \xrightarrow{\tau} \mathbb{A}_{\mathsf{int}} \Rightarrow \mathbb{Z}$$



- $\cdots$
- $\mathsf{O}(u \upharpoonright_\gamma AC) \cap (\mathsf{P}(u \upharpoonright_\gamma AB) \cup \mathsf{P}(u \upharpoonright_\gamma BC)) = \emptyset$;

Interaction sequence examples

$$\mathbb{A}_{\mathsf{int}} \Rightarrow \mathbb{Z} \xrightarrow{\sigma} \mathbb{Z} \xrightarrow{\tau} \mathbb{A}_{\mathsf{int}}$$



- $\cdots$
- for each $u' \sqsubseteq u$ ending in $m^S m'^{S'}$ and $a \in \mathsf{dom}(S')$ if
  - m' is a P-move in AB and $a \notin \mathsf{Av}(u' \upharpoonright AB)$,
  - or m' is a P-move in BC and $a \notin \mathsf{Av}(u' \upharpoonright BC)$,
  - or m' is an O-move in AC and $a \notin \mathsf{Av}(u' \upharpoonright AC)$,

  then $S(a) = S'(a)$.

Strategy composition

Proposition. *If* $u \in Int(ABC)$ *then* $(u \restriction_\gamma AC) \in P_{A \to C}$ *.*

Thus, given $s \in P_{A \to B}$ and $t \in P_{B \to C}$, we can compose them by:

- finding some $u \in Int(ABC)$ such that
- $(u \restriction_\gamma AB) = s$ and $(u \restriction_\gamma BC) = t$.

Then, the composite of s and t is $u \restriction_\gamma AC$.

Definition. For each pair of strategies $\sigma : A \to B$ and $\tau : B \to C$, their composition $\sigma; \tau \subseteq P_{A \to C}$ is given by:

$$\sigma; \tau = \{\, u \restriction_\gamma AC \mid u \in Int(ABC) \wedge (u \restriction_\gamma AB) \in \sigma \wedge (u \restriction_\gamma BC) \in \tau \,\}\,.$$

Strategy composition results

Proposition. *$\sigma;\tau$ is a strategy in $A \to C$.*

The identity morphisms of our category of games are given by:

$$\mathrm{id}_A = \{\, s \in P_{A\to A} \mid s \upharpoonright A_l = s \upharpoonright A_r \,\}$$

where A_l above denotes the left A in $A \to A$ (and dually for A_r). This strategy behaviour, whereby P copies moves from one sub-areana A to another, is called a *copycat*.
We can immediately verify the following.

Proposition. *For any $\sigma : A \to B$, we have that $\sigma = \mathrm{id}_A;\sigma = \sigma;\mathrm{id}_B$.*

Proposition. *Given strategies $\sigma : A \to B$, $\tau : B \to C$ and $\rho : C \to D$, we have $(\sigma;\tau);\rho = \sigma;(\tau;\rho)$.*

Building the model

$$\frac{}{\mathrm{U},\Gamma \vdash () : \mathsf{unit}} \qquad \frac{i \in \mathbb{Z}}{\mathrm{U},\Gamma \vdash i : \mathsf{int}} \qquad \frac{(x:\theta) \in \Gamma}{\mathrm{U},\Gamma \vdash x : \theta} \qquad \frac{a \in \mathrm{U} \cap \mathbb{A}_\zeta}{\mathrm{U},\Gamma \vdash a : \mathsf{ref}\zeta}$$

$$\frac{\mathrm{U},\Gamma \vdash M : \mathsf{int} \quad \mathrm{U},\Gamma \vdash N_0 : \theta \quad \mathrm{U},\Gamma \vdash N_1 : \theta}{\mathrm{U},\Gamma \vdash \mathsf{if}\ M\ \mathsf{then}\ N_1\ \mathsf{else}\ N_0 : \theta} \qquad \frac{\mathrm{U},\Gamma \vdash M : \mathsf{int}}{\mathrm{U},\Gamma \vdash \mathsf{while}(M) : \mathsf{unit}}$$

$$\frac{\mathrm{U},\Gamma \uplus \{\, x:\theta \,\} \vdash M : \theta'}{\mathrm{U},\Gamma \vdash \lambda x^\theta.M : \theta \to \theta'} \qquad \frac{\mathrm{U},\Gamma \vdash M : \theta \to \theta' \quad \mathrm{U},\Gamma \vdash N : \theta}{\mathrm{U},\Gamma \vdash MN : \theta'}$$

$$\frac{\mathrm{U},\Gamma \vdash M : \theta \quad \mathrm{U},\Gamma \vdash N : \theta'}{\mathrm{U},\Gamma \vdash \langle M, N \rangle : \theta \times \theta'} \qquad \frac{\mathrm{U},\Gamma \vdash M : \theta_1 \times \theta_2}{\mathrm{U},\Gamma \vdash \pi_i M : \theta_i}\ {\scriptstyle i\in\{1,2\}}$$

$$\frac{\mathrm{U},\Gamma \vdash M : \mathsf{int} \quad \mathrm{U},\Gamma \vdash N : \mathsf{int}}{\mathrm{U},\Gamma \vdash M \oplus N : \mathsf{int}} \qquad \frac{\mathrm{U},\Gamma \vdash M : \mathsf{ref}\zeta \quad \mathrm{U},\Gamma \vdash N : \mathsf{ref}\zeta}{\mathrm{U},\Gamma \vdash M = N : \mathsf{int}}$$

$$\frac{\mathrm{U},\Gamma \vdash M : \zeta}{\mathrm{U},\Gamma \vdash \mathsf{ref}(M) : \mathsf{ref}\zeta} \qquad \frac{\mathrm{U},\Gamma \vdash M : \mathsf{ref}\zeta}{\mathrm{U},\Gamma \vdash\ !M : \zeta} \qquad \frac{\mathrm{U},\Gamma \vdash M : \mathsf{ref}\zeta \quad \mathrm{U},\Gamma \vdash N : \zeta}{\mathrm{U},\Gamma \vdash M := N : \mathsf{unit}}$$

Building the model – base cases

$$\frac{}{\mathrm{U},\Gamma \vdash () : \mathsf{unit}} \qquad \frac{i \in \mathbb{Z}}{\mathrm{U},\Gamma \vdash i : \mathsf{int}} \qquad \frac{(x:\theta) \in \Gamma}{\mathrm{U},\Gamma \vdash x : \theta} \qquad \frac{a \in \mathrm{U} \cap \mathbb{A}_\zeta}{\mathrm{U},\Gamma \vdash a : \mathsf{ref}\,\zeta}$$

$$\begin{tikzcd} [\![\mathrm{U},\Gamma]\!] \arrow[r, "[\![()]\!]"] & 1 \\ \mathrm{i}^S \arrow[r, no head, bend right] & \star^S \end{tikzcd} \qquad \begin{tikzcd} [\![\mathrm{U},\Gamma]\!] \arrow[r, "[\![i]\!]"] & \mathbb{Z} \\ \mathrm{i}^S \arrow[r, no head, bend right] & i^S \end{tikzcd} \qquad \begin{tikzcd} [\![\mathrm{U},\Gamma]\!] \arrow[r, "[\![x]\!]"] & [\![\theta]\!] \\ \mathrm{i}^S \arrow[r, no head, bend right] & \mathrm{i}_i^S \end{tikzcd} \qquad \begin{tikzcd} [\![\mathrm{U},\Gamma]\!] \arrow[r, "[\![x]\!]"] & \mathbb{A}_\zeta \\ \mathrm{i}^S \arrow[r, no head, bend right] & \mathrm{i}_i^S \end{tikzcd}$$

- in the last two rules, we assume x/a is the i-th component in U,γ.
- in the rule for x, if θ is a function type, the strategy *copycats* between i_i and (the i-th component of) i.

Building the model – pairs and projections

$$\frac{\mathrm{U}, \Gamma \vdash M : \theta \quad \mathrm{U}, \Gamma \vdash N : \theta'}{\mathrm{U}, \Gamma \vdash \langle M, N \rangle : \theta \times \theta'} \qquad \frac{\mathrm{U}, \Gamma \vdash M : \theta_1 \times \theta_2}{\mathrm{U}, \Gamma \vdash \pi_i M : \theta_i} \; {\scriptstyle i \in \{1,2\}}$$

Given $\sigma : A \to B$, $\tau : A \to C$, form their **product** strategy $\langle \sigma, \tau \rangle$ by:

$$A \xrightarrow{\langle \sigma, \tau \rangle} B \otimes C$$

i_A^S

$\vdots$ play like σ, until next move is some $\mathrm{i}_B^{S'}$

$\vdots$ play like τ (from $\mathrm{i}_A^{S'}$), until next is some $\mathrm{i}_C^{S''}$

$(\mathrm{i}_B, \mathrm{i}_C)^{S''}$

$\vdots$ $\vdots$ play like σ or τ, depending on component

Building the model – pairs and projections

$$\frac{\mathrm{U}, \Gamma \vdash M : \theta \quad \mathrm{U}, \Gamma \vdash N : \theta'}{\mathrm{U}, \Gamma \vdash \langle M, N \rangle : \theta \times \theta'} \qquad \frac{\mathrm{U}, \Gamma \vdash M : \theta_1 \times \theta_2}{\mathrm{U}, \Gamma \vdash \pi_i M : \theta_i}\ {\scriptstyle i \in \{1,2\}}$$

Given $\sigma : A \to B$, $\tau : A \to C$, form their **product** strategy $\langle \sigma, \tau \rangle$ [. . .]

$$\frac{[\![M]\!] : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta]\!] \qquad [\![N]\!] : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta']\!]}{[\![\langle M, N \rangle]\!] = \langle [\![M]\!], [\![N]\!] \rangle : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta]\!] \otimes [\![\theta']\!]}$$

Building the model – pairs and projections

$$\frac{\mathrm{U},\Gamma \vdash M : \theta \quad \mathrm{U},\Gamma \vdash N : \theta'}{\mathrm{U},\Gamma \vdash \langle M, N\rangle : \theta \times \theta'} \qquad\qquad \frac{\mathrm{U},\Gamma \vdash M : \theta_1 \times \theta_2}{\mathrm{U},\Gamma \vdash \pi_i M : \theta_i}\ {\scriptstyle i\in\{1,2\}}$$

Given $\sigma : A \to B$, $\tau : A \to C$, form their **product** strategy $\langle \sigma, \tau\rangle$ [. . .]

$$\frac{[\![M]\!] : [\![\mathrm{U},\Gamma]\!] \longrightarrow [\![\theta]\!] \qquad [\![N]\!] : [\![\mathrm{U},\Gamma]\!] \longrightarrow [\![\theta']\!]}{[\![\langle M, N\rangle]\!] = \langle [\![M]\!], [\![N]\!]\rangle : [\![\mathrm{U},\Gamma]\!] \longrightarrow [\![\theta]\!] \otimes [\![\theta']\!]}$$

On the other hand, syntactic projections are modelled using projection strategies:

$$\begin{array}{ccc} A \otimes B & \xrightarrow{\pi_1} & A \\ (\mathrm{i}_A, \mathrm{i}_B)^S & & \\ & & \mathrm{i}_A^S \quad \text{copycat from here on} \end{array}$$

Building the model – pairs and projections

$$\frac{\mathrm{U}, \Gamma \vdash M : \theta \quad \mathrm{U}, \Gamma \vdash N : \theta'}{\mathrm{U}, \Gamma \vdash \langle M, N \rangle : \theta \times \theta'} \qquad \frac{\mathrm{U}, \Gamma \vdash M : \theta_1 \times \theta_2}{\mathrm{U}, \Gamma \vdash \pi_i M : \theta_i}\ {\scriptstyle i \in \{1,2\}}$$

Given $\sigma : A \to B$, $\tau : A \to C$, form their **product** strategy $\langle \sigma, \tau \rangle$ [. . .]

$$\frac{[\![M]\!] : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta]\!] \qquad [\![N]\!] : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta']\!]}{[\![\langle M, N \rangle]\!] = \langle [\![M]\!], [\![N]\!] \rangle : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta]\!] \otimes [\![\theta']\!]}$$

On the other hand, syntactic projections are modelled using projection strategies:

$$\frac{[\![M]\!] : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta_1]\!] \otimes [\![\theta_2]\!]}{[\![\pi_i M]\!] = [\![\mathrm{U}, \Gamma]\!] \xrightarrow{[\![M]\!]} [\![\theta_1]\!] \otimes [\![\theta_2]\!] \xrightarrow{\pi_i} [\![\theta_i]\!]}$$

Building the model – basic operations

$$\frac{\mathrm{U},\Gamma \vdash M : \mathsf{int} \quad \mathrm{U},\Gamma \vdash N : \mathsf{int}}{\mathrm{U},\Gamma \vdash M \oplus N : \mathsf{int}} \qquad \frac{\mathrm{U},\Gamma \vdash M : \mathsf{ref}\zeta \quad \mathrm{U},\Gamma \vdash N : \mathsf{ref}\zeta}{\mathrm{U},\Gamma \vdash M = N : \mathsf{int}}$$

$$\frac{[\![M]\!],[\![N]\!] : [\![\mathrm{U},\Gamma]\!] \longrightarrow \mathbb{Z}}{[\![M \oplus N]\!] = [\![\mathrm{U},\Gamma]\!] \xrightarrow{\langle [\![M]\!],[\![N]\!] \rangle} \mathbb{Z} \otimes \mathbb{Z} \xrightarrow{\oplus} \mathbb{Z}}$$

$$\frac{[\![M]\!],[\![N]\!] : [\![\mathrm{U},\Gamma]\!] \longrightarrow \mathbb{A}_\zeta}{[\![M = N]\!] = [\![\mathrm{U},\Gamma]\!] \xrightarrow{\langle [\![M]\!],[\![N]\!] \rangle} \mathbb{A}_\zeta \otimes \mathbb{A}_\zeta \xrightarrow{=} \mathbb{Z}}$$

Building the model – λ-abstractions

$$\frac{\mathrm{U}, \Gamma \uplus \{\, x : \theta \,\} \vdash M : \theta'}{\mathrm{U}, \Gamma \vdash \lambda x^{\theta}.M : \theta \to \theta'} \qquad \frac{\mathrm{U}, \Gamma \vdash M : \theta \to \theta' \quad \mathrm{U}, \Gamma \vdash N : \theta}{\mathrm{U}, \Gamma \vdash MN : \theta'}$$

Given $\sigma : A \otimes B \to C$, form its **$\Lambda$-abstraction** strategy $\Lambda(\sigma)$ by:

$$A \xrightarrow{\Lambda(\sigma)} B \Rightarrow C$$

$$\begin{array}{llll}
\mathrm{i}_A^S & & & \\
 & & \dagger^S & \\
 & \mathrm{i}_B^{S'} & & \\
\vdots & & \vdots & \text{play like } \sigma \text{ (from } (\mathrm{i}_A, \mathrm{i}_B)^{S'}) \\
 & \mathrm{i'}_B^{S''} & & \\
\vdots & & \vdots & \text{play like } \sigma \text{ (from } (\mathrm{i}_A, \mathrm{i}'_B)^{S''})
\end{array}$$

Building the model – λ-abstractions

$$\frac{\mathrm{U}, \Gamma \uplus \{\, x : \theta \,\} \vdash M : \theta'}{\mathrm{U}, \Gamma \vdash \lambda x^{\theta}.M : \theta \to \theta'} \qquad \frac{\mathrm{U}, \Gamma \vdash M : \theta \to \theta' \quad \mathrm{U}, \Gamma \vdash N : \theta}{\mathrm{U}, \Gamma \vdash MN : \theta'}$$

Given $\sigma : A \otimes B \to C$, form its **$\Lambda$-abstraction** strategy $\Lambda(\sigma)$ [...]

$$\frac{\llbracket M \rrbracket : \llbracket \mathrm{U}, \Gamma \rrbracket \otimes \llbracket \theta \rrbracket \longrightarrow \llbracket \theta' \rrbracket}{\llbracket \lambda x^{\theta}.\, M \rrbracket = \Lambda(\llbracket M \rrbracket) : \llbracket \mathrm{U}, \Gamma \rrbracket \longrightarrow \llbracket \theta \rrbracket \Rightarrow \llbracket \theta' \rrbracket}$$

Building the model – λ-abstractions

$$\frac{\mathrm{U}, \Gamma \uplus \{\, x : \theta \,\} \vdash M : \theta'}{\mathrm{U}, \Gamma \vdash \lambda x^{\theta}.M : \theta \to \theta'} \qquad \frac{\mathrm{U}, \Gamma \vdash M : \theta \to \theta' \quad \mathrm{U}, \Gamma \vdash N : \theta}{\mathrm{U}, \Gamma \vdash MN : \theta'}$$

Given $\sigma : A \otimes B \to C$, form its **$\Lambda$-abstraction** strategy $\Lambda(\sigma)$ $[\dots]$

$$\frac{[\![M]\!] : [\![\mathrm{U}, \Gamma]\!] \otimes [\![\theta]\!] \longrightarrow [\![\theta']\!]}{[\![\lambda x^{\theta}.\, M]\!] = \Lambda([\![M]\!]) : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta]\!] \Rightarrow [\![\theta']\!]}$$

On the other hand, applications are modelled using evaluation strategies:

$$(A \Rightarrow B) \otimes A \xrightarrow{\mathsf{ev}_{A,B}} B$$

$(\dagger \quad , \quad \mathrm{i}_A)^S$

i_A^S copycat from here on

$\mathrm{i}_B^{S'}$

$\mathrm{i}_B^{S'}$ copycat from here on

Building the model – λ-abstractions

$$\frac{\mathrm{U}, \Gamma \uplus \{\, x : \theta \,\} \vdash M : \theta'}{\mathrm{U}, \Gamma \vdash \lambda x^{\theta}.M : \theta \to \theta'} \qquad \frac{\mathrm{U}, \Gamma \vdash M : \theta \to \theta' \quad \mathrm{U}, \Gamma \vdash N : \theta}{\mathrm{U}, \Gamma \vdash MN : \theta'}$$

Given $\sigma : A \otimes B \to C$, form its **$\Lambda$-abstraction** strategy $\Lambda(\sigma)$ [. . .]

$$\frac{[\![M]\!] : [\![\mathrm{U}, \Gamma]\!] \otimes [\![\theta]\!] \longrightarrow [\![\theta']\!]}{[\![\lambda x^{\theta}.\, M]\!] = \Lambda([\![M]\!]) : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta]\!] \Rightarrow [\![\theta']\!]}$$

On the other hand, applications are modelled using evaluation strategies:

$$\frac{[\![M]\!] : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta]\!] \Rightarrow [\![\theta']\!] \qquad [\![N]\!] : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta]\!]}{[\![MN]\!] = [\![\mathrm{U}, \Gamma]\!] \xrightarrow{\langle [\![M]\!], [\![N]\!] \rangle} ([\![\theta]\!] \Rightarrow [\![\theta']\!]) \otimes [\![\theta]\!] \xrightarrow{\mathsf{ev}} [\![\theta']\!]}$$

Building the model – references

$$\frac{\mathrm{U}, \Gamma \vdash M : \zeta}{\mathrm{U}, \Gamma \vdash \mathsf{ref}(M) : \mathsf{ref}\zeta} \qquad \frac{\mathrm{U}, \Gamma \vdash M : \mathsf{ref}\zeta}{\mathrm{U}, \Gamma \vdash\ !M : \zeta} \qquad \frac{\mathrm{U}, \Gamma \vdash M : \mathsf{ref}\zeta \quad \mathrm{U}, \Gamma \vdash N : \zeta}{\mathrm{U}, \Gamma \vdash M := N : \mathsf{unit}}$$

We rely on the following strategies for manipulating references and store:

$$\begin{array}{lll} [\![\zeta]\!] \xrightarrow{\mathsf{new}_\zeta} \mathbb{A}_\zeta & \qquad \mathbb{A}_\zeta \xrightarrow{\mathsf{get}_\zeta} \mathbb{Z} & \qquad \mathbb{A}_\zeta \otimes [\![\zeta]\!] \xrightarrow{\mathsf{set}_\zeta} 1 \\ v^S & \qquad a^S & \qquad (a, v)^S \\ \qquad\qquad a^{S,(a,v)} & \qquad\qquad S(a)^S & \qquad\qquad \star^{S[a \mapsto v]} \end{array}$$

Working-out examples

Work out step-by-step the semantics of these terms:

- $\vdash \mathsf{let}\ \ f = \lambda y^{\mathsf{int}}.\, 2 * y\ \ \mathsf{in}\ \ \lambda x^{\mathsf{int}}.\, f x + 1 : \mathsf{int} \to \mathsf{int}$

- $\vdash \mathsf{let}\ \ x = \mathsf{ref}(0)\ \ \mathsf{in}\ \ \lambda z^{\mathsf{int}}.\, x := z; x : \mathsf{int} \to \mathsf{ref}\,\mathsf{int}$

- $\vdash \lambda z^{\mathsf{int}}.\, \mathsf{let}\ \ x = \mathsf{ref}(0)\ \ \mathsf{in}\ \ x := z; x : \mathsf{int} \to \mathsf{ref}\,\mathsf{int}$

- $f : \mathsf{ref}\,\mathsf{int} \to \mathsf{int} \vdash \mathsf{let}\ \ x = f(\mathsf{ref}(0))\ \ \mathsf{in}\ \ \mathsf{ref}(x) : \mathsf{ref}\,\mathsf{int}$

Properties of the game model

Summing up, we have shown that, for each term $\mathrm{U}, \Gamma \vdash M : \theta$:

$$[\![M]\!] : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta]\!]$$

Properties of the game model

Summing up, we have shown that, for each term $\mathrm{U}, \Gamma \vdash M : \theta$:

$$[\![M]\!] : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta]\!]$$

What properties do we require? The model be:

- compositional

Properties of the game model

Summing up, we have shown that, for each term $\mathrm{U}, \Gamma \vdash M : \theta$:

$$[\![M]\!] : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta]\!]$$

What properties do we require? The model be:

- compositional
- correct wrt the operational semantics:

$$(M, S) \longrightarrow (M', S') \implies [\![\textsf{new } S \textsf{ in } M]\!] = [\![\textsf{new } S' \textsf{ in } M']\!]$$

Properties of the game model

Summing up, we have shown that, for each term $\mathrm{U}, \Gamma \vdash M : \theta$:

$$[\![M]\!] : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta]\!]$$

What properties do we require? The model be:

- compositional
- correct wrt the operational semantics:

$$(M, S) \longrightarrow (M', S') \implies [\![\textsf{new } S \textsf{ in } M]\!] = [\![\textsf{new } S' \textsf{ in } M']\!]$$

- adequate: if $[\![\vdash M : \textsf{unit}]\!] = \{\star\star\}$ then $M \Downarrow$

Properties of the game model

Summing up, we have shown that, for each term $\mathrm{U}, \Gamma \vdash M : \theta$:

$$[\![M]\!] : [\![\mathrm{U}, \Gamma]\!] \longrightarrow [\![\theta]\!]$$

What properties do we require? The model be:

- compositional
- correct wrt the operational semantics:

$$(M, S) \longrightarrow (M', S') \implies [\![\text{new } S \text{ in } M]\!] = [\![\text{new } S' \text{ in } M']\!]$$

- adequate: if $[\![\vdash M : \text{unit}]\!] = \{\star\star\}$ then $M \Downarrow$
- sound: if $[\![M]\!]$ and $[\![N]\!]$ have the same **complete** plays then $M \cong N$
- fully abstract: the converse of sound

– *why just complete plays?*

Correctness

$$
\begin{array}{rcll}
(i \oplus j, S) & \longrightarrow & (k, S) & (k = i \oplus j) \\
((\lambda x.M)V, S) & \longrightarrow & (M[V/x], S) & \\
(\pi_1 \langle V_1, V_2 \rangle, S) & \longrightarrow & (V_1, S) & \\
(\pi_2 \langle V_1, V_2 \rangle, S) & \longrightarrow & (V_2, S) & \\
(\mathsf{if}\, 0\, \mathsf{then}\, M\, \mathsf{else}\, M', S) & \longrightarrow & (M', S) & \\
(\mathsf{if}\, i\, \mathsf{then}\, M\, \mathsf{else}\, M', S) & \longrightarrow & (M, S) & (i > 0) \\
(\mathsf{while}(M), S) & \longrightarrow & (\mathsf{if}\, M\, \mathsf{then}\ \mathsf{while}(M)\ \mathsf{else}\, (), S) & \\
(a = b, S) & \longrightarrow & (0, S) & (a \neq b) \\
(a = a, S) & \longrightarrow & (1, S) & \\
(!a, S) & \longrightarrow & (S(a), S) & \\
(a := V, S) & \longrightarrow & ((), S[a \mapsto V]) & \\
(\mathsf{ref}(V), S) & \longrightarrow & (a', S[a' \mapsto V]) & (a' \notin \mathsf{dom}(S))
\end{array}
$$

$$
\frac{(M, S) \longrightarrow (M', S')}{(E[M], S) \longrightarrow (E[M'], S')}
$$

Correctness – stateless rules

$$
\begin{array}{rcll}
(i \oplus j, S) & \longrightarrow & (k, S) & (k = i \oplus j) \\
((\lambda x.M)V, S) & \longrightarrow & (M[V/x], S) & \\
(\pi_1 \langle V_1, V_2 \rangle, S) & \longrightarrow & (V_1, S) & \\
(\pi_2 \langle V_1, V_2 \rangle, S) & \longrightarrow & (V_2, S) & \\
(\text{if } 0 \text{ then } M \text{ else } M', S) & \longrightarrow & (M', S) & \\
(\text{if } i \text{ then } M \text{ else } M', S) & \longrightarrow & (M, S) & (i > 0) \\
(\text{while}(M), S) & \longrightarrow & (\text{if } M \text{ then while}(M) \text{ else } (), S) & \\
(a = b, S) & \longrightarrow & (0, S) & (a \neq b) \\
(a = a, S) & \longrightarrow & (1, S) &
\end{array}
$$

$$
\frac{(M, S) \longrightarrow (M', S')}{(E[M], S) \longrightarrow (E[M'], S')}
$$

Lemma. *For all the reductions* $(M, S) \longrightarrow (M', S')$ *above,* $[\![M]\!] = [\![M']\!]$.

For all but the last rule, this follows either directly from the definitions of the model constructs, or from the properties of products and abstractions. For the last rule, we do induction on the number of applications, relying on compositionality: if $[\![M]\!] = [\![M']\!]$ then $[\![E[M]]\!] = [\![E[M']]\!]$.

Correctness – state rules

$$\begin{array}{rcll} (!a, S) & \longrightarrow & (S(a), S) & \\ (a := V, S) & \longrightarrow & ((), S[a \mapsto V]) & \\ (\mathsf{ref}(V), S) & \longrightarrow & (a', S[a' \mapsto V]) & (a' \notin \mathsf{dom}(S)) \end{array}$$

$$\frac{(M, S) \longrightarrow (M', S')}{(E[M], S) \longrightarrow (E[M'], S')}$$

Lemma. *For $(M, S) \longrightarrow (M', S')$ as above, $[\![\textit{new } S \textit{ in } M]\!] = [\![\textit{new } S' \textit{ in } M']\!]$.*

For all but the last rule, this follows either directly from the definitions of the model constructs for manipulating state. For the last rule, we need to show that the new and the E commute.

Correctness – state rules

$$\begin{array}{rcll} (!a, S) & \longrightarrow & (S(a), S) & \\ (a := V, S) & \longrightarrow & ((), S[a \mapsto V]) & \\ (\mathsf{ref}(V), S) & \longrightarrow & (a', S[a' \mapsto V]) & (a' \notin \mathsf{dom}(S)) \end{array}$$

$$\frac{(M, S) \longrightarrow (M', S')}{(E[M], S) \longrightarrow (E[M'], S')}$$

Lemma. *For* $(M, S) \longrightarrow (M', S')$ *as above,* $[\![\textit{new } S \textit{ in } M]\!] = [\![\textit{new } S' \textit{ in } M']\!]$.

For all but the last rule, this follows either directly from the definitions of the model constructs for manipulating state. For the last rule, we need to show that the new and the E commute.

Proposition. $(M, S) \longrightarrow (M', S')$ *implies* $[\![\textit{new } S \textit{ in } M]\!] = [\![\textit{new } S' \textit{ in } M']\!]$.

Adequacy

Adequacy ensures that diverging terms have diverging semantics:

$$M\not\Downarrow \implies [\![\vdash M : \mathsf{unit}]\!] = \{\epsilon\}$$

We rely on the fact that any transition sequence with a bounded number of while unfolding is terminating. Suppose $M\not\Downarrow$ and $[\![\vdash M : \mathsf{unit}]\!] = \{\star\star\}$:

Adequacy

Adequacy ensures that diverging terms have diverging semantics:

$$M\not\Downarrow \implies [\![\vdash M : \mathsf{unit}]\!] = \{\epsilon\}$$

We rely on the fact that any transition sequence with a bounded number of while unfolding is terminating. Suppose $M\not\Downarrow$ and $[\![\vdash M : \mathsf{unit}]\!] = \{\star\star\}$:

- then, $(M, \emptyset) \longrightarrow \cdots$ has infinitely many while unfoldings
- pick some fresh x and let M_0 be obtained from M by:
  - replacing each while(N) in it with while$(x := !x + 1; N)$
  - wrapping the resulting term in let $x = \mathsf{ref}(0)$ in $[]; !x$

Adequacy

Adequacy ensures that diverging terms have diverging semantics:

$$M\not\Downarrow \implies [\![\vdash M : \mathsf{unit}]\!] = \{\epsilon\}$$

We rely on the fact that any transition sequence with a bounded number of while unfolding is terminating. Suppose $M\not\Downarrow$ and $[\![\vdash M : \mathsf{unit}]\!] = \{\star\star\}$:

- then, $(M, \emptyset) \longrightarrow \cdots$ has infinitely many while unfoldings
- pick some fresh x and let M_0 be obtained from M by:
  - replacing each while(N) in it with while$(x := !x + 1; N)$
  - wrapping the resulting term in let $x = \mathsf{ref}(0)$ in $[]; !x$
- then, because $[\![M]\!] = \{\star\star\}$, we have $[\![M_0]\!] = \{\star\, j\}$ (some $j \geq 0$)
- but $(M_0, \emptyset) \longrightarrow (M_0', (a, 0)) \longrightarrow \cdots (M_0'', S)$ with $S(a) = j + 1$
- so, by correctness, $\star\, j \in [\![\mathsf{new}\ S \text{ in } M_0'']\!]$, contradiction as in $[\![M_0'']\!]$ the value of a is non-decreasing.

Soundness

Proposition. *Given* $\Gamma \vdash M : \theta$ *and* $\Gamma \vdash N : \theta$*, if* $\mathsf{comp}([\![M]\!]) = \mathsf{comp}([\![N]\!])$ *then* $M \cong N$*.*

Proof. Suppose $M \not\cong N$. Then,

- there is C such that (WLOG) $C[M]\Downarrow$ and $C[N]\not\Downarrow$

Soundness

Proposition. *Given* $\Gamma \vdash M : \theta$ *and* $\Gamma \vdash N : \theta$, *if* $\mathsf{comp}(\llbracket M \rrbracket) = \mathsf{comp}(\llbracket N \rrbracket)$ *then* $M \cong N$*.*

Proof. Suppose $M \ncong N$. Then,

- there is C such that (WLOG) $C[M]\Downarrow$ and $C[N]\not\Downarrow$
- then, from correctness, $\llbracket C[M] \rrbracket = \llbracket () \rrbracket$ so $\star\star \in \llbracket C[M] \rrbracket$
- and, from adequacy, $\llbracket C[N] \rrbracket = \{\epsilon\}$

Soundness

Proposition. *Given* $\Gamma \vdash M : \theta$ *and* $\Gamma \vdash N : \theta$, *if* $\mathsf{comp}(\llbracket M \rrbracket) = \mathsf{comp}(\llbracket N \rrbracket)$ *then* $M \cong N$.

Proof. Suppose $M \not\cong N$. Then,

- there is C such that (WLOG) $C[M]\Downarrow$ and $C[N]\not\Downarrow$
- then, from correctness, $\llbracket C[M] \rrbracket = \llbracket () \rrbracket$ so $\star\star \in \llbracket C[M] \rrbracket$
- and, from adequacy, $\llbracket C[N] \rrbracket = \{\epsilon\}$
- now, taking $\Gamma = \{x_1 : \theta_1, \cdots, x_n : \theta_n\}$ and $f : \theta_1 \to \cdots \to \theta_n \to \theta$,

  $$\llbracket \Gamma \vdash C[M] : \mathsf{unit} \rrbracket = \llbracket C[(\lambda \vec{x}.M) x_1 \cdots x_n] \rrbracket = \vec{\Lambda}(\llbracket M \rrbracket); \llbracket C[f x_1 \cdots x_n] \rrbracket$$

  and same for N
- $\star\star \in \llbracket C[M] \rrbracket \setminus \llbracket C[N] \rrbracket$ implies that $\mathsf{comp}(\vec{\Lambda}(\llbracket M \rrbracket)) \setminus \mathsf{comp}(\vec{\Lambda}(\llbracket N \rrbracket))$ is non-empty, so $\mathsf{comp}(\llbracket M \rrbracket) \setminus \mathsf{comp}(\llbracket N \rrbracket)$ is non-empty, contradiction.

$\square$

Definability and full abstraction

Full abstraction is proven via definability: the model has no (finitary) garbage.

Proposition. *Any finitary strategy (i.e. finite up to permuting names)* $\sigma : [\![\mathrm{U}, \Gamma]\!] \to [\![\theta]\!]$ *is the translation of some* GroundML *term.*

This is proven by induction on the length of the longest play in σ, deconstructing it into smaller strategies.

Definability and full abstraction

Full abstraction is proven via definability: the model has no (finitary) garbage.

Proposition. *Any finitary strategy (i.e. finite up to permuting names)* $\sigma : [\![\mathrm{U}, \Gamma]\!] \to [\![\theta]\!]$ *is the translation of some* GroundML *term.*

This is proven by induction on the length of the longest play in σ, deconstructing it into smaller strategies.

Theorem. *Given* $\Gamma \vdash M : \theta$ *and* $\Gamma \vdash N : \theta$*,*

$$\mathsf{comp}([\![M]\!]) = \mathsf{comp}([\![N]\!]) \iff M \cong N$$

Full abstraction

Theorem. *Given* $\Gamma \vdash M, N : \theta$, $\mathsf{comp}(\llbracket M \rrbracket) = \mathsf{comp}(\llbracket N \rrbracket) \iff M \cong N$.

Proof. We need only prove the right-to-left implication. Let $\Gamma = \{x_1 : \theta_1, \cdots, x_n : \theta_n\}$ and suppose $s \in \mathsf{comp}(\llbracket M \rrbracket) \setminus \mathsf{comp}(\llbracket N \rrbracket)$.

Full abstraction

Theorem. *Given* $\Gamma \vdash M, N : \theta$, comp$(\llbracket M \rrbracket) =$ comp$(\llbracket N \rrbracket) \iff M \cong N$.

Proof. We need only prove the right-to-left implication. Let $\Gamma = \{x_1 : \theta_1, \cdots, x_n : \theta_n\}$ and suppose $s \in$ comp$(\llbracket M \rrbracket) \setminus$ comp$(\llbracket N \rrbracket)$.

- then, there is a corresponding $s' \in$ comp$(\vec{\Lambda}(\llbracket M \rrbracket)) \setminus$ comp$(\vec{\Lambda}(\llbracket N \rrbracket))$,

  a play in $1 \to (\llbracket \theta_1 \rrbracket \Rightarrow \cdots \Rightarrow \llbracket \theta_n \rrbracket \Rightarrow \llbracket \theta \rrbracket)$, say $s' = \star \ \dagger_1 \ \mathrm{i}_1^{S_1} \ \dagger_2^{S_1} \cdots \ \mathrm{i}_n^{S_n} \ s''$

Full abstraction

Theorem. *Given* $\Gamma \vdash M, N : \theta$, comp($[\![M]\!]$) = comp($[\![N]\!]$) $\iff M \cong N$.

Proof. We need only prove the right-to-left implication. Let $\Gamma = \{x_1 : \theta_1, \cdots, x_n : \theta_n\}$ and suppose $s \in$ comp($[\![M]\!]$) $\setminus$ comp($[\![N]\!]$).

- then, there is a corresponding $s' \in$ comp($\vec{\Lambda}([\![M]\!])$) $\setminus$ comp($\vec{\Lambda}([\![N]\!])$), a play in $1 \to ([\![\theta_1]\!] \Rightarrow \cdots \Rightarrow [\![\theta_n]\!] \Rightarrow [\![\theta]\!])$, say $s' = \star \dagger_1 \, \mathrm{i}_1^{S_1} \; \dagger_2^{S_1} \cdots \mathrm{i}_n^{S_n} \, s''$
- take t to be the play in $([\![\theta_1]\!] \Rightarrow \cdots \Rightarrow [\![\theta_n]\!] \Rightarrow [\![\theta]\!]) \to 1$ given by:

$$t = \dagger_1 \, \mathrm{i}_1^{S_1} \; \dagger_2^{S_1} \cdots \mathrm{i}_n^{S_n} \, s'' \star^S$$

  where S the last store in s'. By Definability, there is $f : \vec{\theta} \to \theta \vdash M'$: unit such that $[\![M']\!]$ contains just t (and prefixes)
- then $\star\star \in [\![(\lambda f.M')(\lambda \vec{x}.M)]\!]$ but $[\![(\lambda f.M')(\lambda \vec{x}.N)]\!] = \{\epsilon\}$

Full abstraction

Theorem. *Given* $\Gamma \vdash M, N : \theta$, $\mathsf{comp}(\llbracket M \rrbracket) = \mathsf{comp}(\llbracket N \rrbracket) \iff M \cong N$.

Proof. We need only prove the right-to-left implication. Let $\Gamma = \{x_1 : \theta_1, \cdots, x_n : \theta_n\}$ and suppose $s \in \mathsf{comp}(\llbracket M \rrbracket) \setminus \mathsf{comp}(\llbracket N \rrbracket)$.

- then, there is a corresponding $s' \in \mathsf{comp}(\vec{\Lambda}(\llbracket M \rrbracket)) \setminus \mathsf{comp}(\vec{\Lambda}(\llbracket N \rrbracket))$, a play in $1 \to (\llbracket \theta_1 \rrbracket \Rightarrow \cdots \Rightarrow \llbracket \theta_n \rrbracket \Rightarrow \llbracket \theta \rrbracket)$, say $s' = \star \dagger_1 \, \mathrm{i}_1^{S_1} \, \dagger_2^{S_1} \cdots \mathrm{i}_n^{S_n} \, s''$
- take t to be the play in $(\llbracket \theta_1 \rrbracket \Rightarrow \cdots \Rightarrow \llbracket \theta_n \rrbracket \Rightarrow \llbracket \theta \rrbracket) \to 1$ given by:

$$t = \dagger_1 \, \mathrm{i}_1^{S_1} \, \dagger_2^{S_1} \cdots \mathrm{i}_n^{S_n} \, s'' \star^S$$

  where S the last store in s'. By Definability, there is $f : \vec{\theta} \to \theta \vdash M' : \mathsf{unit}$ such that $\llbracket M' \rrbracket$ contains just t (and prefixes)
- then $\star\star \in \llbracket (\lambda f.M')(\lambda \vec{x}.M) \rrbracket$ but $\llbracket (\lambda f.M')(\lambda \vec{x}.N) \rrbracket = \{\epsilon\}$
- by adequacy-correctness, $(\lambda f.M')(\lambda \vec{x}.M)\Downarrow$ and $(\lambda f.M')(\lambda \vec{x}.N)\not\Downarrow$, so $M \ncong N$.

$\square$

Examples

$$M_1 \equiv \mathsf{let}\, x = \mathsf{ref}(0)\, \mathsf{in}\, \lambda y^{\mathsf{ref\,int}}.\, x = y$$

$$M_2 \equiv \lambda y^{\mathsf{ref\,int}}.\, 0$$

$$M_3 \equiv \mathsf{let}\, x = \mathsf{ref}(0)\, \mathsf{in}\, \mathsf{let}\, c = \mathsf{ref}(0)\, \mathsf{in}$$
$$f(\lambda\_.\, \mathsf{if}\, !c = 0\, \mathsf{then\, div\, else}\, x); c := 1; \lambda y^{\mathsf{ref\,int}}.\, x = y$$

$$M_4 \equiv f(\lambda\_.\, \mathsf{div}); \lambda y^{\mathsf{ref\,int}}.\, 0$$

$$M_5 \equiv \mathsf{let}\, x = \mathsf{ref}(\mathsf{ref}(0))\, \mathsf{in}$$
$$\lambda y^{\mathsf{ref\,int}}.\, \mathsf{let}\, z = !x\, \mathsf{in}\;\, \mathsf{if}\, y = z\, \mathsf{then\, div\, else}\, (x := \mathsf{ref}(0); z)$$

$$M_6 \equiv \lambda y^{\mathsf{ref\,int}}.\, \mathsf{ref}(0)$$

Exercises

1. Work out the game semantics of these terms:

   - $f : \mathsf{int} \to \mathsf{int} \vdash \mathsf{let}\, y = f(0)\, \mathsf{in}\, \lambda x^{\mathsf{int}}.\, f(x+y)+1 : \mathsf{int} \to \mathsf{int}$
   - $\vdash \lambda f^{\mathsf{int}\to\mathsf{int}}.\, \mathsf{let}\, y = f(0)\, \mathsf{in}\, \lambda x^{\mathsf{int}}.\, f(x+y)+1 : (\mathsf{int} \to \mathsf{int}) \to (\mathsf{int} \to \mathsf{int})$

   note that, in the second case, O can repeatedly play a move $\dagger$ (for f).

2. Complete the modelling of the following syntactic constructs (cf. slide 22):

$$\frac{\mathrm{U}, \Gamma \vdash M : \zeta}{\mathrm{U}, \Gamma \vdash \mathsf{ref}(M) : \mathsf{ref}\zeta} \qquad \frac{\mathrm{U}, \Gamma \vdash M : \mathsf{ref}\zeta}{\mathrm{U}, \Gamma \vdash !M : \zeta} \qquad \frac{\mathrm{U}, \Gamma \vdash M : \mathsf{ref}\zeta \quad \mathrm{U}, \Gamma \vdash N : \zeta}{\mathrm{U}, \Gamma \vdash M := N : \mathsf{unit}}$$

3. Let $\mathsf{wh} : (1 \Rightarrow \mathbb{Z}) \to 1$ be the strategy that plays in $1 \Rightarrow \mathbb{Z}$ while positive integers are returned, until 0 is returned and the strategy plays the unique move on the RHS (which we denote $\star'$):

$$\mathsf{wh} = \{\, \dagger \star i_1 \star i_2 \cdots \star i_n \star 0 \star' \mid n \geq 0 \wedge i_j > 0 \}$$

   Use the wh strategy in order to model the while loop construct:

$$\frac{\mathrm{U}, \Gamma \vdash M : \mathsf{int}}{\mathrm{U}, \Gamma \vdash \mathsf{while}(M) : \mathsf{unit}}$$