GAME SEMANTICS (DAY 2)

Andrzej Murawski
UNIVERSITY OF OXFORD

Nikos Tzevelekos
QUEEN MARY
UNIVERSITY OF LONDON

GAME SEMANTICS

★ Two players: **O** (System) and **P** (Program)

★ Types are interpreted by games.

★ Programs are interpreted as strategies for P.

★ No winners or losers.

★ The dialogue is the central object of study.

GROUND ML

$$\theta \;::=\; \zeta \mid \theta \times \theta \mid \theta \to \theta$$

$$\zeta \;::=\; \mathsf{unit} \mid \mathsf{int} \mid \mathsf{ref}\,\zeta$$

TYPING RULES I

$$\mathbb{A} = \biguplus_\zeta \mathbb{A}_\zeta$$

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

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

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

TYPING RULES 2

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

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

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

OPERATIONAL SEMANTICS (AUXILIARY NOTATION)

$$\text{Stores} \;=\; \{\, S : \mathbb{A} \rightharpoonup (\{\,\star\,\} \cup \mathbb{Z} \cup \mathbb{A}) \mid S \text{ finite and legal} \,\}$$

$$(S[a \mapsto x])(a') = \begin{cases} S(a') & \text{if } a' \in \mathsf{dom}(S) \setminus \{a\} \\ x & \text{if } a' = a \\ \text{undefined} & \text{otherwise} \end{cases}$$

$$V \;::=\; () \mid i \mid x \mid a \mid \langle V, V \rangle \mid \lambda x^{\theta}.M$$

OPERATIONAL SEMANTICS

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

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

EVALUATION CONTEXTS

$$
\begin{aligned}
E \;::=\;& [\,] \mid E \oplus M \mid V \oplus E \mid \mathsf{if}\, E \,\mathsf{then}\, M \,\mathsf{else}\, M \mid E M \mid V E \mid \langle E, M \rangle \\
& \mid \langle V, E \rangle \mid \pi_i E \mid \mathsf{ref}(E) \mid E = M \mid x = E \mid !E \mid E := M \mid V := E
\end{aligned}
$$

For any term $\vdash M : \mathsf{unit}$, we write $M \Downarrow$ if

$$(\emptyset, M) \longrightarrow\!\!\!\!\to (S, ())$$

for some store S.

SHORTHANDS

- $\mathsf{let}\, x = M \,\mathsf{in}\, N$ stands for the term $(\lambda x^{\theta}.N)M$
- $M; N$ stands for $\mathsf{let}\, x = M \,\mathsf{in}\, N$, where x does not occur in N
- *while* M *do* N can be coded as

$$\mathsf{if}\, M \,\mathsf{then}\, \mathsf{while}((N; M))\, \mathsf{else}\, ()$$

- We can define divergent terms of type θ by

$$\mathsf{div}_{\theta} \;\equiv\; \mathsf{while}(1); M_{\theta},$$

  where M_{θ} is an arbitrary term of type θ.

CONTEXTS

$$
\begin{aligned}
C \;::=\;& [\,] \mid \mathsf{if}\ C\ \mathsf{then}\ M\ \mathsf{else}\ M \mid \mathsf{if}\ M\ \mathsf{then}\ C\ \mathsf{else}\ M \mid \mathsf{if}\ M\ \mathsf{then}\ M\ \mathsf{else}\ C \\
& \mid \mathsf{while}(C) \mid \lambda x^{\theta}.C \mid MC \mid CM \mid \langle C, M\rangle \mid \langle M, C\rangle \mid \pi_i C \mid C \oplus M \\
& \mid M \oplus C \mid C = M \mid M = C \mid \mathsf{ref}(C) \mid !C \mid C := M \mid M := C
\end{aligned}
$$

EQUIVALENCE

$\Gamma \vdash M_1 : \theta$ and $\Gamma \vdash M_2 : \theta$ are ***equivalent*** (written $\Gamma \vdash M_1 \cong M_2$) if, for any context C such that $\vdash C[M_1], C[M_2] : \mathsf{unit}$,

$$C[M_1]\Downarrow \text{ if and only if } C[M_2]\Downarrow.$$

EQUIVALENCE?

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

$$\mathsf{gen}' \equiv \mathsf{let}\, x = \mathsf{ref}(0)\,\mathsf{in}\,\lambda z^{\mathsf{int}}.(x := z; x) \quad : \quad \mathsf{int} \to \mathsf{ref\ int}$$

$$C \equiv (\lambda f^{\mathsf{int} \to \mathsf{ref\ int}}.\,\mathsf{if}\,(f0 = f0)\,\mathsf{then}\,()\,\mathsf{else\ div})\,[\,]$$

EQUIVALENCE?

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

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

FULL ABSTRACTION

$$[\![M_1]\!] = [\![M_2]\!] \quad \text{if and only if} \quad M_1 \cong M_2$$

COMPOSITIONAL INTERPRETATION

- Types interpreted by games between O and P.
- Terms interpreted by strategies for P.
- Each syntactic construct interpreted through special strategies, constructions on strategies and composition.
- This is elegant but may obscure intuitions. We shall start with a more direct interpretation.

TOYML

The types of ToyML are generated according to the following grammar.

$$\beta ::= \mathsf{unit} \mid \mathsf{int} \mid \mathsf{ref\ int}$$

$$\theta ::= \beta \mid \beta \to \beta$$

TOYML

$$\frac{}{\Gamma \vdash () : \mathsf{unit}} \qquad \frac{i \in \mathbb{Z}}{\Gamma \vdash i : \mathsf{int}} \qquad \frac{(x : \theta) \in \Gamma}{\Gamma \vdash x : \theta} \qquad \frac{\Gamma \vdash M : \mathsf{int}}{\Gamma \vdash \mathsf{while}(M) : \mathsf{unit}}$$

$$\frac{\Gamma \vdash M : \mathsf{int} \quad \Gamma \vdash N : \mathsf{int}}{\Gamma \vdash M \oplus N : \mathsf{int}} \qquad \frac{\Gamma \vdash M : \mathsf{int} \qquad \Gamma \vdash N, N' : \theta}{\Gamma \vdash \mathsf{if}\ M\ \mathsf{then}\ N\ \mathsf{else}\ N' : \theta}$$

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

$$\frac{\Gamma \vdash M : \mathsf{int}}{\Gamma \vdash \mathsf{ref}(M) : \mathsf{ref\ int}} \qquad \frac{\Gamma \vdash M : \mathsf{ref\ int} \quad \Gamma \vdash N : \mathsf{ref\ int}}{\Gamma \vdash M = N : \mathsf{int}}$$

$$\frac{\Gamma \vdash M : \mathsf{ref\ int}}{\Gamma \vdash !M : \mathsf{int}} \qquad \frac{\Gamma \vdash M : \mathsf{ref\ int} \quad \Gamma \vdash N : \mathsf{int}}{\Gamma \vdash M := N : \mathsf{unit}}$$

EXAMPLES

$\vdash 1 : \mathsf{int}$

$$\begin{array}{cc} \star & (\mathsf{r}_{\downarrow}, 1) \\ O & P \end{array}$$

$x : \mathsf{int} \vdash x + 1 : \mathsf{int}$

$$\begin{array}{cc} i & (\mathsf{r}_{\downarrow}, i + 1) \\ O & P \end{array}$$

EXAMPLES

$$x : \mathsf{int}, f : \mathsf{int} \to \mathsf{int} \vdash fx + fx : \mathsf{int}$$

$$\begin{array}{cccccc} (i, \dagger) & (\mathsf{c}_f, i) & (\mathsf{r}_f, j) & (\mathsf{c}_f, i) & (\mathsf{r}_f, j') & (\mathsf{r}_\downarrow, j + j') \\ O & P & O & P & O & P \end{array}$$

$$f : \mathsf{int} \to \mathsf{int}, g : \mathsf{int} \to \mathsf{int} \vdash f(g(0)) + 1 : \mathsf{int}$$

$$\begin{array}{cccccc} (\dagger, \dagger) & (\mathsf{c}_g, 0) & (\mathsf{r}_g, i) & (\mathsf{c}_f, i) & (\mathsf{r}_f, j) & (\mathsf{r}_\downarrow, j + 1) \\ O & P & O & P & O & P \end{array}$$

EXAMPLES

$$\vdash \lambda x^{\mathsf{int}}.x + 1 : \mathsf{int} \to \mathsf{int}$$

$$\begin{array}{ccccccc} \star & (\mathsf{r}_\downarrow, \dagger) & (\mathsf{c}, i_1) & (\mathsf{r}, i_1 + 1) & (\mathsf{c}, i_2) & (\mathsf{r}, i_2 + 1) & \cdots \\ O & P & O & P & O & P & \end{array}$$

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

$$\begin{array}{ccccccccc} \dagger & (\mathsf{c}_f, 0) & (\mathsf{r}_f, j) & (\mathsf{r}_\downarrow, \dagger) & (\mathsf{c}, i_1) & (\mathsf{c}_f, i_1 + j) & (\mathsf{r}_f, j_1) & (\mathsf{r}, j_1 + 1) & \cdots \\ O & P & O & P & O & P & O & P & \end{array}$$

EXAMPLES

$$x : \mathsf{ref\ int} \vdash\ !x + 1 : \mathsf{int}$$

$$\begin{array}{cc} a^{(a,i)} & (\mathsf{r}_{\downarrow}, i+1)^{(a,i)} \\ O & P \end{array}$$

$$x : \mathsf{ref\ int}, f : \mathsf{int} \to \mathsf{int} \vdash f(!x)+!x : \mathsf{int}$$

$$\begin{array}{cccc} (a, \dagger)^{(a,i)} & (\mathsf{c}_f, i)^{(a,i)} & (\mathsf{r}_f, j)^{(a,i')} & (\mathsf{r}_{\downarrow}, j + i')^{(a,i')} \\ O & P & O & P \end{array}$$

EXAMPLES

$$\vdash \lambda x^{\mathsf{int}}.\mathsf{ref}(x) : \mathsf{int} \to \mathsf{ref\ int}$$

$$\begin{array}{cccc} \star & (\mathsf{r}_{\downarrow}, \dagger) & (\mathsf{c}, i_1) & (\mathsf{r}, a_1)^{(a_1,i_1)} \\ O & P & O & P \end{array}$$

$$\begin{array}{ccccc} (\mathsf{c}, i_2)^{(a_1,i_1')} & (\mathsf{r}, a_2)^{(a_1,i_1')(a_2,i_2)} & (\mathsf{c}, i_3)^{(a_1,i_1'')(a_2,i_2')} & (\mathsf{r}, a_3)^{(a_1,i_1'')(a_2,i_2')(a_3,i_3)} & \ldots \\ O & P & O & P & \end{array}$$

$$a_i \neq a_j$$

LET'S PLAY!

- Dialogue between the environment (O) and the program (P).
- Technically, sequences of moves that involve names drawn from an infinite set (stable under name invariance, i.e. nominal sets).
- Moves are accompanied by evolving stores.
- We focus on complete plays (all questions answered, all calls have returns), as these characterize contextual equivalence.

SEMANTIC VALUES (USED BY PLAYERS)

$$
\begin{aligned}
\mathcal{V}_{\mathsf{unit}} &= \{\star\} \\
\mathcal{V}_{\mathsf{int}} &= \mathbb{Z} \\
\mathcal{V}_{\mathsf{ref\ int}} &= \mathbb{A}_{\mathsf{int}} \\
\mathcal{V}_{\beta\to\beta'} &= \{\dagger\}
\end{aligned}
$$

MOVES

Let $\Gamma = \{x_1 : \theta_1, \cdots, x_m : \theta_m\}$ and $\Gamma \vdash M : \theta$ be a ToyML typing judgment. The set $M_{\Gamma\vdash\theta}$ of ***moves associated with*** Γ ***and*** θ is defined to be

$$M_{\Gamma\vdash\theta} = I_\Gamma \cup M_\theta \cup \bigcup_{1\leq i\leq m} M_{x_i}$$

$$M_{\Gamma\vdash\theta} = I_\Gamma \cup M_\theta \cup \bigcup_{1\le i\le m} M_{x_i}$$

- I_Γ is the set of *initial moves* given by

$$I_\Gamma = \{\, (\ell_1, \cdots, \ell_m) \mid \ell_i \in \mathcal{V}_{\theta_i},\ 1 \le i \le m \,\};$$

- M_θ is the set of *output moves* defined by
  - $M_\theta = \{\, (\mathsf{r}_\downarrow, \ell) \mid \ell \in \mathcal{V}_\theta \,\}$ if θ is a base type,
  - $M_\theta = \{\, (\mathsf{r}_\downarrow, \dagger) \,\} \cup \{\, (\mathsf{c}, \ell) \mid \ell \in \mathcal{V}_{\theta'} \,\} \cup \{\, (\mathsf{r}, \ell) \mid \ell \in \mathcal{V}_{\theta''} \,\}$ if $\theta = \theta' \to \theta''$;

- M_{x_i} is the set of *variable moves*, taken to be empty if θ_i is a base type and, if $\theta_i = \theta_i' \to \theta_i''$, equal to

$$\{\, (\mathsf{c}_{x_i}, \ell) \mid \ell \in \mathcal{V}_{\theta_i'} \,\} \cup \{\, (\mathsf{r}_{x_i}, \ell) \mid \ell \in \mathcal{V}_{\theta_i''} \,\}.$$

MOVES SUMMARY

- Each non-initial move consists of a pair (t, ℓ) of a *tag* and a *(semantic) value*.
- For each function-type identifier x in Γ, we have introduced tags c_x and r_x. They can be viewed as calls and returns related to that identifier. The accompanying value in e.g. a move (c_x, ℓ) corresponds to the value that identifier is called with.
- Similarly, $\mathsf{r}_\downarrow$ can be taken to correspond to the fact that our modelled term was successfully evaluated, and, if θ is a function type, c and r refer respectively to calling the corresponding value and obtaining a result.

NOTATION FOR MOVES

$$M_{\Gamma\vdash\theta} = I_\Gamma \cup M_\theta \cup \bigcup_{1\leq i\leq m} M_{x_i}$$

- I_Γ is the set of *initial moves* given by

$$I_\Gamma = \{\,(\ell_1, \cdots, \ell_m) \mid \ell_i \in \mathcal{V}_{\theta_i},\ 1 \leq i \leq m\,\};$$

Moves are ranged over by m and variants. We shall use i to range over I_Γ, and we shall often write i_{x_i} for ℓ_i.

MOVE OWNERSHIP

- O-moves (context)

  initial moves and those with tags r_x, c

- P-moves (program)

  those with tags $\mathsf{r}_\downarrow, \mathsf{c}_x, \mathsf{r}$

Using ownership of moves, we can extend the definition to *names* saying that a name a is owned by the owner of the first move m in which it occurs.

SEQUENCES AND PLAYS

- We aim to define a notion of complete play next.
- It is intended to model successful interactions between a program and some environment.
- We begin with the underlying **sequences** of moves.
- Sequences augmented with stores yield **plays**.

COMPLETE SEQUENCES

A ***complete sequence*** over $\Gamma \vdash \theta$ is a (possibly empty) sequence of moves $\mathrm{i}\,(t_1, \ell_1) \cdots (t_k, \ell_k)$ such that the sequence $t_1 \cdots t_k$ of tags matches the grammar:

$$X\,\mathsf{r}_{\downarrow}\,(\mathsf{c}\,X\,\mathsf{r})^* \quad \text{where} \quad X \;=\; \left(\sum\nolimits_{(x\,:\,\theta' \to \theta'') \in \Gamma} (\mathsf{c}_x\,\mathsf{r}_x)\right)^* .$$

We assume that $X\mathsf{r}_{\downarrow}(\mathsf{c}X\mathsf{r})^*$ degenerates to $X\mathsf{r}_{\downarrow}$ when c, r are not available in $M_{\Gamma\vdash\theta}$, i.e. θ is a base type.

COMPLETE PLAYS

A ***complete play*** over $\Gamma \vdash \theta$ is a sequence $m_1^{S_1} \cdots m_k^{S_k}$ of moves-with-store satisfying the conditions below.

- $m_1 \cdots m_k$ is a complete sequence over $\Gamma \vdash \theta$.
- For any $1 \leq i \leq k$, $\mathsf{dom}(S_i) = \nu(m_1 \cdots m_i)$.

$\nu(x)$ stands for the set of elements of $\mathbb{A}$ (names) that occur in x.

INTERPRETATION

- Next we shall discuss how to assign, to any ToyML term $\Gamma \vdash M : \theta$, a set of complete plays over $\Gamma \vdash \theta$. We shall write $(\!|\, \Gamma \vdash M : \theta \,|\!)$ for that set.
- This constitutes a very direct account of the game-semantic interpretation of GroundML (to follow), specialised to ToyML.
- The complete-play interpretation is guaranteed to yield the following result.

Theorem (Full Abstraction)
Let $\Gamma \vdash M_1, M_2 : \theta$ be ToyML terms. Then $\Gamma \vdash M_1 \cong M_2$ if and only if $(\!|\, \Gamma \vdash M_1 \,|\!) = (\!|\, \Gamma \vdash M_2 \,|\!)$.

INITIAL CASES

$$() \mid i \mid x \mid x \oplus y \mid \mathsf{ref}(x) \mid x = y \mid$$

$$!x \mid x := y \mid \mathsf{if}\ x\ \mathsf{then}\ N\ \mathsf{else}\ N'$$

SKIP AND INTEGERS

Skip command (())

$(\!|\Gamma \vdash () : \mathsf{unit}|\!)$ is defined to contain all complete plays over $\Gamma \vdash \mathsf{unit}$ that have the shape $\mathrm{i}^S(\mathsf{r}_\downarrow, \star)^S$. Here P simply responds with the move $(\mathsf{r}_\downarrow, \star)$ without modifying the store.

Integer constant (i)

The defining complete plays for $(\!|\Gamma \vdash i : \mathsf{int}|\!)$ have the shape $\mathrm{i}^S(\mathsf{r}_\downarrow, i)^S$. This follows the same pattern as above, except that the value is i.

VARIABLES

- The complete plays in $\llparenthesis\, \Gamma \vdash x : \beta \,\rrparenthesis$ all have the form $\mathrm{i}^S(\mathsf{r}_\downarrow, \mathrm{i}_x)^S$.
- For $\llparenthesis\, \Gamma \vdash x : \beta \to \beta' \,\rrparenthesis$, the complete plays must have the form
$$\mathrm{i}^S(\mathsf{r}_\downarrow, \dagger)^S X_1 \cdots X_k$$
where $k \geq 0$ and
$$X_i = (\mathsf{c}, \ell_i)^{S_i}(\mathsf{c}_x, \ell_i)^{S_i}(\mathsf{r}_x, \ell_i')^{S_i'}(\mathsf{r}, \ell_i')^{S_i'}$$
for all $1 \leq i \leq k$.

P never changes the stores played by O. In contrast, O is allowed to modify the stores insofar as the definition of complete play allows, i.e. the integer values in S_i' may be different from the corresponding values in S_i.

Arithmetic operations ($x \oplus y$)

$(\!| \Gamma \vdash x \oplus y : \mathsf{int} |\!)$ is given by complete plays of the form

$$\mathrm{i}^S(\mathsf{r}_\downarrow, \mathrm{i}_x \oplus \mathrm{i}_y).$$

Reference creation ($\mathsf{ref}(x)$)

$(\!| \Gamma \vdash \mathsf{ref}(x) : \mathsf{ref\ int} |\!)$ is defined by complete plays of the form

$$\mathrm{i}^S(\mathsf{r}_\downarrow, a)^{S[a \mapsto \mathrm{i}_x]}$$

with $a \in \mathbb{A}_{\mathsf{int}} \setminus \mathsf{dom}(S)$.