Extensional Semantics of Program Behaviour

Jim Laird

University of Sussex,UK

Extensional vs. Intensional Semantics

Extensional denotational semantics (domain theory) is:

- Intuitively appealing, tractable.
- A well developed, *quantitative* theory.

Intensional (game) semantics is:

- Suitable for describing "real" programming languages: references, exceptions, concurrent threads ...
- *Qualitative* — precise descriptions of computational effects.

Wouldn't it be nice to have both ?

Three parts of this talk:

- Bistable biorders — a cartesian closed category of *observably sequential functionals*. A *universal* model of $\Lambda^{\top}_{\bot}$ (simply-typed λ-calculus with constants $\top$ and $\bot$).
- Locally boolean orders — a model of linear logic, equivalent to a category of games and strategies (sequential algorithms).
- A locally boolean model of higher-order stateful computation (Idealized Algol) based on new linear connectives.

Bistable Biorders

A *bistable biorder* D is a set $|D|$ with two partial orders — the *extensional* order $\sqsubseteq_D$ and the *bistable* order $\leq$, such that:

- $a, b \in (|D|, \leq)$ are bounded above if and only if they are bounded below (written $a \updownarrow b$),
- if $a \updownarrow b$ then there exist $a \wedge b, a \vee b \in |D|$ which are the glb and lub of a and b with respect to both orders,
- if $a \updownarrow b$, $a \updownarrow c$ and $b \updownarrow c$, then $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$.

D is *pointed* if it has least element $\perp$ and a greatest element $\top$, such that $\perp \leq \top$,

A CCC of bistable orders

f is *bistable* if for all $x, y \in |D|$, if $x \updownarrow y$ then $f(x) \updownarrow f(y)$, $f(x \wedge y) = f(x) \wedge f(y)$ and $f(x \vee y) = f(x) \vee f(y)$.

Proposition The category of bistable orders and monotone bistable functions is cartesian closed.

PROOF: We define the bistable order:
$f \leq_{D \Rightarrow E} g$ if for all $x, y \in |D|$ such that $x \leq_D y$, $f(y) \updownarrow g(x)$ and $f(x) = f(y) \wedge g(x)$ and $g(y) = f(y) \vee g(x)$. $\square$

So, in particular we have a model of $\Lambda_{\bot}^{\top}$ — the simply-typed λ-calculus over Sierpinski space $\Sigma = \top, \bot$.

Lemma If A_1, A_2 are bounded, and $f : A_1 \times A_2 \to \Sigma$ is bi-strict, monotone and bistable, then f is left-bi-strict or right-bi-strict.

Proof: Suppose f is strict. Since $\langle \bot, \top \rangle \updownarrow \langle \top, \bot \rangle$, we have
$f(\langle \bot, \top \rangle) \wedge f(\langle \top, \bot \rangle) = f(\langle \bot, \top \rangle \wedge \langle \top, \bot \rangle) = f(\bot) = \bot$, and
$f(\langle \bot, \top \rangle) \vee f(\langle \top, \bot \rangle) = f(\langle \bot, \top \rangle \vee \langle \top, \bot \rangle) = f(\top) = \top$.

Hence either $f(\langle \bot, \top \rangle) = \bot$ and $f(\langle \top, \bot \rangle) = \top$ (i.e. f is left-strict) or $f(\langle \bot, \top \rangle) = \top$ and $f(\langle \top, \bot \rangle) = \bot$ (i.e. f is right-strict). $\square$

Corollary The bistable semantics of $\Lambda_\bot^\top$ is universal for any type of the form $\Sigma^i \Rightarrow \Sigma^j$.

Universality

Definition A definable retraction from σ to τ is a pair of terms $\mathtt{in} : \sigma \Rightarrow \tau$ and $\mathtt{proj} : \tau \Rightarrow \sigma$ such that $[\![x : \sigma \vdash \mathtt{proj}\ (\mathtt{in}\ x) : \sigma]\!] = \mathtt{id}_{[\![\sigma]\!]}$.

For example, $(\Sigma \Rightarrow \Sigma) \Rightarrow \Sigma \trianglelefteq (\Sigma \Rightarrow \Sigma) \times \Sigma$ via the terms $\lambda f.\langle \lambda x.(f\ \lambda y.x), f\ \lambda y.y\rangle$ and $\lambda x.\lambda g.(\pi_1(x)\ (g\ \pi_2(x)))$

Lemma If universality holds at type τ, and $\sigma \trianglelefteq \tau$, then universality holds at type σ.

Proof: Suppose $\mathtt{in} : \sigma \trianglelefteq \tau : \mathtt{proj}$. Given an element $e \in [\![\sigma]\!]$, we have a term $M : \tau$ such that $[\![M]\!] = e; [\![x \vdash \mathtt{in}\ x]\!]$ and thus $[\![\mathtt{proj}\ M]\!] = e; [\![x \vdash \mathtt{proj}\ (\mathtt{in}\ x)]\!] = e$. $\square$

Lemma For any $i, j \geq 0$, $\Sigma^i \Rightarrow \Sigma^j \Rightarrow \Sigma \trianglelefteq \Sigma^{i+j} \Rightarrow \Sigma^{(2i)^j}$.

Proof: By induction on j.

For the induction case,

$(\Sigma^i \Rightarrow \Sigma^{j+1}) \Rightarrow \Sigma \cong (\Sigma^i \Rightarrow \Sigma^j) \Rightarrow ((\Sigma^i \Rightarrow \Sigma) \Rightarrow \Sigma)$

$\trianglelefteq (\Sigma^i \Rightarrow \Sigma^j) \Rightarrow (\Sigma \Rightarrow \Sigma) \times \Sigma^i$

$\cong (\Sigma \Rightarrow (\Sigma^i \Rightarrow \Sigma^j) \Rightarrow \Sigma) \times ((\Sigma^i \Rightarrow \Sigma^j) \Rightarrow \Sigma)^i$

$\trianglelefteq (\Sigma \Rightarrow \Sigma^{i+j} \Rightarrow \Sigma^{(2i)^j}) \times (\Sigma^{i+j} \Rightarrow \Sigma^{(2i)^j})^i$ by induction hypothesis

$\trianglelefteq (\Sigma^{i+j+1} \Rightarrow \Sigma^{(2i)^j})^i \times (\Sigma^{i+j+1} \Rightarrow \Sigma^{(2i)^j})^i$

$\trianglelefteq \Sigma^{i+j+1} \Rightarrow \Sigma^{(2i)^j.2i} \cong \Sigma^{i+j+1} \Rightarrow \Sigma^{(2i)^{j+1}}$. $\square$

Lemma For any type τ there exists $i, j \in \mathbb{N}$ such that τ is a definable retract of $\Sigma^i \Rightarrow \Sigma^j$.

PROOF: By induction on type structure. If $\tau = \sigma_1 \Rightarrow \sigma_2$, then $\sigma_1 \trianglelefteq \Sigma^{i_1} \Rightarrow \Sigma^{j_1}$ and $\sigma_2 \trianglelefteq \Sigma^{i_2} \Rightarrow \Sigma^{j_2}$.

Hence $\tau \trianglelefteq (\Sigma^{i_1} \Rightarrow \Sigma^{j_1}) \Rightarrow (\Sigma^{i_2} \Rightarrow \Sigma^{j_2})$
$\cong (\Sigma^{i_2} \Rightarrow (\Sigma^{i_1} \Rightarrow \Sigma^{j_1}) \Rightarrow \Sigma)^{j_2}$
$\trianglelefteq (\Sigma^{i_2} \Rightarrow (\Sigma^{i_1+j_1} \Rightarrow \Sigma^{(2i_1)^{j_1}})^{j_2} \cong \Sigma^{i_1+j_1+i_2} \Rightarrow \Sigma^{(2i_1)^{j_1} \cdot j_2}$. $\square$

Corollary The bistable model of $\Lambda^{\top}_{\bot}$ is universal.

Locally Boolean Orders

A locally boolean order $A = (|A|, \sqsubseteq, \neg)$ is a partial order with an endofunction $\neg : P \to P$ such that:

- $\neg\neg x = x$
- if $x \sqsubseteq y$, then $\neg y \sqsubseteq \neg x$.
- for each x, $\{x, \neg x\}$ has a least upper bound, $x^\top$.
- if $x \sqsubseteq y^\top$ and $y \sqsubseteq x^\top$ $(x \uparrow y)$ then $x \wedge y$ and $x \vee y$ exist, and $x \wedge y \uparrow x$. If $x \uparrow z$ and $y \uparrow z$ then $x \wedge (y \vee z) = (x \wedge y) \vee x \wedge z)$ and $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$.

Proposition Every LBO determines a bistable biorder.

PROOF: Set $x \leq_B y$ if $x \sqsubseteq y$ and $x^\top = y^\top$. $\square$

Moreover:

- locally boolean orders and bistable functions form a CCC (set $(\neg f)(x) = \neg f(\neg x)$)
- we can define the *stable* order: $x \leq_s y$ if $x \sqsubseteq y$ and $x \uparrow y$.

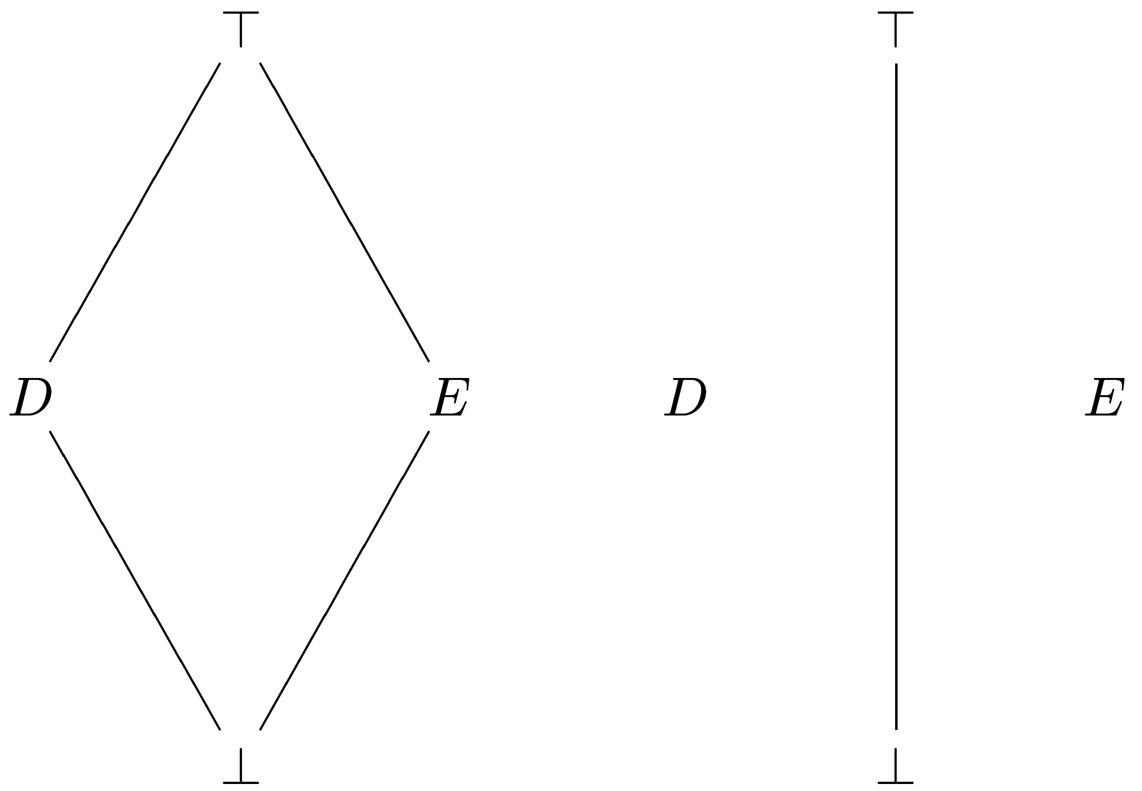
Locally Boolean Domains

A *prime* p is an element such that if $x \uparrow y$ or $x \downarrow y$, and $p \sqsubseteq x \vee y$, then $p \sqsubseteq x \vee y$ implies $p \sqsubseteq x$ or $p \sqsubseteq y$.

A *Locally Boolean Domain* is a locally boolean order such that:

- Every element is a stably bounded supremum of compact primes.
- $x \sqsubseteq y$ if and only if for all $p \leq_s x$ there exists $q \leq_s y$ such that $p \sqsubseteq q$.

A bistable function of LBDs is *affine* if it preserves stably bounded lubs — and hence is determined by its action on primes.



Extensional and bistable orders in the separated sum $D \oplus E$

Prenex Normal Forms

Theorem Every (pointed) locally boolean domain A is isomorphic to one of the form $\Pi_{i \in I} \Sigma_{j \in J_i} A_{ij}$.

Proof: If $\top$ is prime, then $A \cong \Sigma_{m \in M} A_m$, where M is the set of minimal non-$\bot$ elements, and $A_m = \{x \in A \mid m \sqsubseteq x \sqsubset \top\}$
If $\top$ is not prime, then $A \cong \Pi_{p \in P} A_p$, where P is the set of maximal primes, and $A_p = \{x \in A \mid x \sqsubseteq p\}$. $\square$

Duality

For locally boolean domains A, B, $A \multimap B$ is the LBD of affine functions. $A \multimap_s B$ is the LBD of strict affine functions.

Proposition For any locally boolean domain, $(A \multimap \Sigma) \multimap_s \Sigma \cong A$.
For any *pointed* locally boolean domain, $(A \multimap_s \Sigma) \multimap \Sigma \cong A$.

Proof: We can use prenex normal forms — e.g.
$(\Pi_{i \in I} \Sigma_{j \in J_i} A_{ij} \multimap_s \Sigma) \multimap \Sigma$
$\cong (\coprod_{i \in I} \Pi_{j \in J_i} (A_{ij} \multimap_s \Sigma)) \multimap \Sigma$
$\cong \Pi_{i \in I} \Sigma_{j \in J_i} ((A_{ij} \multimap_s \Sigma) \multimap \Sigma)$
$\cong \Pi_{i \in I} \Sigma_{j \in J_i} A_{ij}$. $\square$

The connectives $\times, \multimap, \multimap_s$, are sufficient to express all of the connectives of intuitionistic (affine) linear logic:

- $A \otimes B = (A \multimap (B \multimap \Sigma)) \multimap_s \Sigma$,
- $A \oplus B = ((A \multimap \Sigma) \times (B \multimap \Sigma)) \multimap_s \Sigma$,
- $!A = (A \Rightarrow \Sigma) \multimap_s \Sigma$,

And for *pointed* M, N:

- $N \invamp M = (N \multimap_s \Sigma) \multimap M$
- $?N \cong (N \multimap_s \Sigma) \multimap \Sigma$.

Games and Locally Boolean Domains

A game A is given by:

- Disjoint sets of *Opponent* and *Player* moves,
- A set of *alternating* sequences, or plays (Opponent starts).

Proposition Every game is isomorphic to one of the form $\Pi_{i \in I} \Sigma_{j \in J_i} A_{ij}$.

PROOF: I is the set of starting moves, J_i is the set of moves which can follow i, and A_{ij} contains plays s such that $ij \cdot s$ is a play of A. $\Box$

By the pnf theorem we have a correspondence between games and locally boolean domains, which extends to the linear connectives — e.g.
$\Pi_{i\in I}\Sigma_{j\in J_i}A_{ij} \otimes \Pi_{k\in K}\Sigma_{l\in K_l}B_{kl} \cong$
$(\Pi_{i\in I}\Sigma_{j\in J_i}A_{ij} \otimes B) \times (\Pi_{k\in K}\Sigma_{l\in K_l}A \otimes B_{kl})$, and

$!\Pi_{i\in I}\Sigma_{j\in J}A_{ij} \cong \Pi_{k\in I}\Sigma_{l\in J_k}!(A_{kl} \times \Pi_{i\in I-\{k\}}\Sigma_{j\in J_i}A_{ij})$

Moreover (by sequentiality of affine and bistable maps) this is an equivalence of categories

Modelling State

The problem: procedures in Idealized Algol do not correspond to functions from values to values.
— e.g. λ`x` : `bool`.`If x then x else not x` is not equivalent to λ`x` : `bool`.`x`.

The solution: procedures in Idealized Algol denote functions from *streams* of values to values. (Formally, this corresponds to the construction of a new ! on our linear category).

But how to interpret contraction (merging of streams) ? — e.g. from `If x then y else not y` to `If x then x else not x`.

Sequoidal Categories

A sequoidal category consists of:

- a symmetric monoidal category $(\mathcal{C}, I, \otimes)$, with a specified symmetric monoidal subcategory C_S of *pointed* objects and *strict* maps
- a bifunctor $\_ \oslash \_ : \mathcal{C} \times \mathcal{C}_S \to \mathcal{C}_S$
- a natural transformation $\mathsf{wk} : \_ \otimes \_ \to \_ \oslash \_$
- isomorphisms $I \oslash A \cong A$ and $A \oslash I \cong I$ and $A \oslash (B \oslash C)$ $cong (A \otimes B) \oslash C$

satisfying some coherence equations.

A sequoidal *closed* category is a sequoidal category $(\mathcal{C}, \mathcal{C}_S, I, \otimes, \oslash)$ with a functor $\_ \multimap \_ : \mathcal{C}^{OP} \times \mathcal{C} \to \mathcal{C}$ which restricts to a functor from $\mathcal{C}^{OP} \times \mathcal{C}_S$ to $\mathcal{C}_S$ such that for each A:

- $A \otimes \_$ is left adjoint to $A \multimap \_$ (so that $(\mathcal{C}, I, \otimes, \multimap)$ is a SMCC),
- $A \oslash \_ : \mathcal{C}_S \to \mathcal{C}_S$ is left adjoint to $A \multimap \_ : \mathcal{C}_S \to \mathcal{C}_S$
- The co-units and units of the adjunctions commute with wk.

In the category of $\mathcal{LBD}$ of locally boolean domains, we define $A \oslash B = (A \multimap (B \multimap_s \Sigma)) \multimap \Sigma$.

Proposition $(\mathcal{LBD}, \mathcal{PLBA}_{\mathcal{S}}, \otimes, \oslash)$ is sequoidal closed.

A sequoidal category (with finite products) is *sequentially decomposable* if:

- $A \otimes B \cong A \oslash B \times B \oslash A$
- $A \oslash (B \times C) \cong (A \oslash B) \times (A \oslash C)$.

Proposition $\mathcal{LBD}$ is sequentially decomposable

PROOF: To show e.g. $(A \multimap (B \multimap \Sigma)) \multimap_s \Sigma \cong (A \multimap (B \multimap_s \Sigma)) \multimap \Sigma \times (B \multimap (A \multimap_s \Sigma)) \multimap \Sigma$ we use sequentiality of bistable maps. $\square$

Modelling Idealized Algol in a sequoidal category:

- We model the stateful ! as a *minimal invariant* (least fixpoint) such that $= !A \cong !A \oslash A$, using sequential decomposability to derive the contraction and promotion operations as least fixpoints.
- We model `new` variable declaration by deriving a morphism $\mathsf{cell} : I \to !(\mathtt{nat} \times \mathsf{unit}^{\omega})$ as least fixpoint, again using sequential decomposability to derive its *history-sensitive* behaviour.

Further Work

- Full completeness for models of (polarized) MALL, PCF, Idealized Algol, ... — reproducing intensional constraints such as innocence and well-bracketing using logical relations, linear continuation passing, etc.
- Completing the intensional hierarchy — using non-determinism (may and must testing interpreted via the duality between stability and co-stability) to model concurrent threads. Also, higher-order references, exceptions, ...
- An intensional model of the fully category of bistable biorders, also of the sequential fragments of stable and co-stable biorders (non-deterministic sequential functional computation).