Imperative Programs as Proofs via Game Semantics

Martin Churchill, Jim Laird, Guy McCusker
University of Bath

Logic in Computer Science, 21st June 2011

Curry-Howard Correspondence

The Curry-Howard isomorphism notes a striking correspondence between *proofs* and *functional programs*:

Types	Propositions
Programs	Proofs
Evaluation	Proof normalisation

- We can extend our notion of programs to include those with imperative effects...
- What are the corresponding proofs?

Overview

- We present a logic where proofs have imperative behaviour.
- The system is expressive:
  - This logic contains **first-order intuitionistic linear logic**
  - We can embed a **total imperative programming language**
- $\Rightarrow$ We can use the logical structure to specify behaviour of the imperative programs, etc...

Games Model

- The logic is based on a simple, well-studied notion of two-player game
  - Propositions $\cong$ games
  - Proofs $\cong$ (history-sensitive) winning strategies
- Longley has developed a programming language based on these games. Our work analyses its logical structure.
- We prove a strong *full completeness* result with respect to this games model

Formulas of WS1

- Fix a first-order language $\mathcal{L}$ with pairs of predicates $(\phi,\overline{\phi})$ and a variable set $\mathcal{V}$ $(=\in \phi)$
- For formulas of the logic are as follows:

$$\begin{array}{rllll}
M,\ N := & \mathbf{1} & \mid\ \bot & \mid\ \phi(\overrightarrow{x}) & \mid \\
& M \otimes N & \mid\ M \oslash N & \mid\ N \lhd P & \mid \\
& \forall x.P & \mid\ M \& N & \mid\ !N & \\
P,\ Q := & \mathbf{0} & \mid\ \top & \mid\ \overline{\phi}(\overrightarrow{x}) & \mid \\
& P \parr Q & \mid\ P \lhd Q & \mid\ P \oslash N & \mid \\
& \exists x.P & \mid\ P \oplus Q & \mid\ ?P &
\end{array}$$

- We have an involutive $(-)^{\perp}$ operation switching polarity
- We can express implication $M \multimap N = N \lhd M^{\perp}$

Formulas as Games

- Formulas denote (families of) two-player games
  - (indexed over $\mathcal{L}$-structures and valuations)
  - **Opponent** and **Player** alternately play moves according to a tree of valid plays
  - In negative formulas Opponent starts, in positive formulas Player starts

Formulas as Games

- Formulas denote (families of) two-player games
  - (indexed over $\mathcal{L}$-structures and valuations)
  - **Opponent** and **Player** alternately play moves according to a tree of valid plays
  - In negative formulas Opponent starts, in positive formulas Player starts
- Proofs of a formula denote (uniform families of) winning P-strategies on the interpretation of that formula.
  - Player must always respond to an Opponent-move
  - There is a winning condition for infinite plays
  - Player must behave "uniformly" with respect to the $\mathcal{L}$-structure

Units and Additives

- The formula $\mathbf{1}$ represents the empty game ($\vdash \mathbf{1}$).
- The formula $\bot$ represents the single-move game ($\nvdash \bot$).

- We have additive conjunction of negative formulas $M \& N$ — Opponent chooses to play in M, or in N.

(+ positive versions...)

Multiplicatives

If M and N are negative games, $M \oslash N$ denotes the *left-merge*:

- A play in $M \oslash N$ is a merge of a play in M and a play in N, that begins in M.
- Opponent starts playing in M, and may later switch between the two.

In $M \otimes N$, Opponent may start in either component.

$$M \otimes N \cong (M \oslash N) \& (N \oslash M)$$

Sequents

A sequent of WS1 is of the form $\Phi \vdash \Gamma$ where:

- Φ consists of the variables in scope and atomic assumptions
- Γ is a nonempty list of formulas of either polarity

$$\Phi \vdash M, N, P, Q$$

Comma is to be read as a left-associative $\oslash$ or $\lhd$:

$$\Phi \vdash ((M \oslash N) \lhd P) \lhd Q$$

$\Rightarrow$ First move must occur in first formula.

Rules for $\otimes$ and $\oslash$

There are *head introduction rules* for each connective.

$$\frac{\Phi \vdash A, N, \Gamma}{\Phi \vdash A \oslash N, \Gamma} \qquad \frac{\Phi \vdash M, N, \Gamma \qquad \Phi \vdash N, M, \Gamma}{\Phi \vdash M \otimes N, \Gamma}$$

Exponential

In $!N$, Opponent may open infinitely many copies of N

$$!N \cong N \oslash (N \oslash (N \oslash \dots$$

$\Rightarrow$ the *final coalgebra* of $X \mapsto N \oslash X$

$$\frac{\Phi \vdash N, !N, \Gamma}{\Phi \vdash !N, \Gamma} \qquad \frac{\Phi \vdash N, P^{\perp}, P}{\Phi \vdash !N, P}$$

Example: Boolean Storage Cell

- We can define formulas:
  - $\mathbf{B}$ corresponding to `bool`
  - $\mathbf{Bi}$ corresponding to `bool` $\rightarrow$ `1`
- $!\mathbf{var} = !(\mathbf{B} \& \mathbf{Bi})$ is a type of reusable Boolean variables (read method and write method)
- We can define a reusable Boolean cell $\vdash \mathbf{B} \multimap !\mathbf{var}$ using the anamorphism rule and a proof $p \vdash \mathbf{var}, \mathbf{B}, \mathbf{B}^{\perp}$

Example: Boolean Storage Cell

- We can define formulas:
  - $\mathbf{B}$ corresponding to `bool`
  - $\mathbf{Bi}$ corresponding to `bool` $\rightarrow$ `1`
- $!\mathbf{var} = !(\mathbf{B}\&\mathbf{Bi})$ is a type of reusable Boolean variables (read method and write method)
- We can define a reusable Boolean cell $\vdash \mathbf{B} \multimap !\mathbf{var}$ using the anamorphism rule and a proof $p \vdash \mathbf{var}, \mathbf{B}, \mathbf{B}^{\perp}$

$$\begin{tikzcd}
\mathbf{B} \arrow[r, "p"] & \mathbf{var} \oslash \mathbf{B} \arrow[r, "\mathrm{id} \oslash p"] & \mathbf{var} \oslash (\mathbf{var} \oslash \mathbf{B}) \arrow[r, "\mathrm{id} \oslash (\mathrm{id} \oslash p)"] & \ldots \arrow[d] \\
 & & & !\mathbf{var}
\end{tikzcd}$$

Boolean Cell — p

$$
\begin{array}{ccccccc}
\mathbf{B} & \multimap & (\mathbf{B} & \& & \mathbf{Bi}) & \oslash & \mathbf{B} \\
 & & r & & & & \\
r & & & & & & \\
b & & & & & & \\
 & & b & & & & \\
 & & & & & & r \\
 & & & & & & b \\
\hline
 & & & & w_b & & \\
 & & & & ok & & \\
 & & & & & & r \\
 & & & & & & b
\end{array}
$$

Boolean Cell — ana(p)

$$
\begin{array}{cccccccccccccccc}
\mathbf{B} & \multimap & (\mathbf{B} & \& & \mathbf{Bi}) & \oslash & \mathbf{B} & \multimap & (\mathbf{B} & \& & \mathbf{Bi}) & \oslash & ((\mathbf{B} & \& & \mathbf{Bi}) & \oslash \\
 & & & & & & & & & & w_b & & & & & \\
 & & & & w_b & & & & & & & & & & & \\
 & & & & ok & & & & & & & & & & & \\
 & & & & & & & & & & ok & & & & & \\
 & & & & & & & & & & & & r & & & \\
 & & & & & & r & & & & & & & & & \\
 & & & & & & b & & & & & & & & & \\
 & & & & & & & & & & & & b & & & \\
 & & & & & & & & & & & & \vdots & & &
\end{array}
$$

- We can extend this example to define a Boolean *Stack* in WS1 ($\mathbf{B} \cong \mathtt{pop}$, $\mathbf{Bi} \cong \mathtt{push}$. For the "state" we use $!\mathbf{B}$)

Atoms and Quantifiers

- A negative atom $\phi(\overrightarrow{x})$ is interpreted as $\mathbf{1}$ if satisfied, $\perp$ if not.
- At a given model $\mathcal{M}$, $\forall x.N$ is represented as the $\mathcal{M}$-fold product of N.

$$\frac{\Phi, \overline{\phi}(\overrightarrow{x}) \vdash \perp, \Gamma}{\Phi \vdash \phi(\overrightarrow{x}), \Gamma} \qquad \frac{\Phi, \overline{\phi}(\overrightarrow{x}) \vdash \top, \Gamma}{\Phi, \overline{\phi}(\overrightarrow{x}) \vdash \overline{\phi}(\overrightarrow{x}), \Gamma}$$

$$\frac{X \uplus \{x\}; \Theta \vdash N, \Gamma}{X; \Theta \vdash \forall x.N, \Gamma}\, x \notin FV(\Theta, \Gamma) \qquad \frac{X \uplus \{y\}; \Theta \vdash P[y/x], \Gamma}{X \uplus \{y\}; \Theta \vdash \exists x.P, \Gamma}$$

Proofs and Strategies

- We give semantics of proofs as (uniform, winning) strategies

Proofs and Strategies

- We give semantics of proofs as (uniform, winning) strategies

There is a partial converse...

Finitary strategy $\rightarrow$ proof denoting that strategy

- As well as the *head introduction* rules, there are rules which act to combine the tail into a single formula (reading upwards).
- We can use these to define this *full completeness* procedure

For infinite strategies, we obtain an *infinitary* core proof

Intuitionistic Linear Logic

- We can also use the anamorphism rule to derive promotion
  - $\Rightarrow$ **Embedding of Intuitionistic Linear Logic in WS1**
- There are formulas that are not provable in ILL but are provable in WS1 e.g. medial:

$$\vdash ((\alpha \otimes \beta \multimap \bot) \otimes (\gamma \otimes \delta \multimap \bot) \multimap \bot) \multimap$$
$$((\alpha \multimap \bot) \otimes (\gamma \multimap \bot) \multimap \bot) \otimes ((\beta \multimap \bot) \otimes (\delta \multimap \bot) \multimap \bot)$$

- We can also embed Polarized Linear Logic
  - And hence CBN and CBV lambda calculi
  - With Boolean cells, coroutines, ...

Uniformity for Controlling Imperative Flow

We can use uniformity of the underlying strategies to give refinements on imperative beavhiour. E.g...

- Define $\mathbf{B}' = \bot \lhd (\overline{\alpha} \oplus \overline{\beta})$, $\mathbf{Bi}' = (\alpha \& \beta) \lhd \top$.
- If α and β are false, $\mathbf{B}' = \mathbf{B}$, $\mathbf{Bi}' = \mathbf{Bi}$
- ... in which case **worm** $= \mathbf{Bi}' \oslash !\mathbf{B}'$ represents the type of a "write-once-read-many" Boolean cell.
- But since any proof must be a uniform strategy on *all* models, any proof of **worm** must act as a well-behaved Boolean cell.

Further Directions

- Enhancing the logic to be able to specify more interesting properties of more interesting programs
- Introducing propositional variables (that can represent arbitrary games — polymorphism)
- Recursive types ($\mathsf{list}(\mathbf{B}) = \mu X.\bot \lhd (\top \oplus (\top \oslash (\mathbf{B} \otimes X)))$)
- Peano axioms, programs over natural number base types and their properties

Thank You

Any questions?