A λ-calculus for free response categories

Florian Hatat Tom Hirschowitz

CCS09 – 26 novembre 2009

LAMA
Laboratoire de Mathématiques
Université de Savoie

What is compilation?

Compilation starts to be pretty serious:

- High-level constructs,
- Concurrency,
- Certification.

However, compilation lacks a general theory.

- Case-by-case approach.
- Almost no reuse of results.

Drama

Compilation even lacks a definition!

A theory for compilation?

Thesis

A theory of compilation will emerge from a theory of prog. languages expliciting both:

- the dynamics,
- and composition of programs.

Existing theories for prog. languages

Usual categorical models. They:

- handle composition satisfactorily,
- elude the dynamics (β-equivalence becomes equality over morphisms).

A few 2-dimensional models, in particular cases (Melliès; Montanari; Hilken).

Game Semantics

- Deals with both aspects, but
- not yet reflected into any general structure.
- E.g.: typical algebraic framework: SMCC (or CCC).

Goal

Extract the right 2-dimensional algebraic structure from games.

First step

Here we take a first step:

- Ludics: strong notion of locality.
- We emphasise it even more, expliciting the role of topology.
- Idea present in game semantics, yet not explicited.

Outline

Context

Proof theory Graphical games for MLL: collapsing dynamics gives free (split) $\star$-autonomous categories.

Provability Games for MALL, games for LL (Hirschowitz 09).

Plan

Make a step towards proof theory for LL, with response categories/logic.

Introduction

Response categories

Graphical games

λ_r-calculus

Response categories

Response category (Selinger)

1. A category $\mathbf{C}$ with finite products,
2. with a negation functor $\neg\colon \mathbf{C}^{\mathrm{op}} \to \mathbf{C}$,
3. and a natural hom-set bijection:

$$\varphi_{A,B,C}\colon \mathrm{Hom}(A \times B, \neg C) \to \mathrm{Hom}(A, \neg(B \times C))$$

(Equivalent choice: replace $\neg$ with an exponentiable object R.)

Introduction

Response categories

Graphical games

λ_r-calculus

Associated Game — Positions

Positions are graphs:

- oriented,
- acyclic,
- labelled: formulae on edges,
- for one vertex: all output edges have the same label.

(tablo)

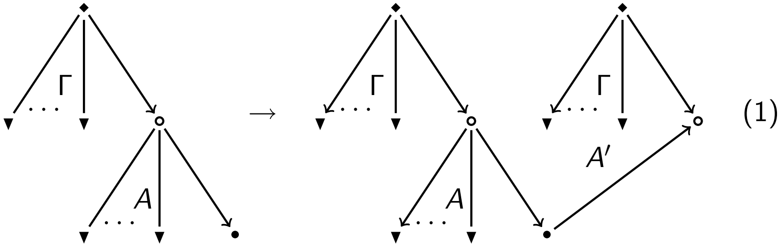
Topology:

- edges are open points,
- vertices are closed.

Associated Game — Moves

Token

Root vertex is the *active* player.



- Move: any restriction of this basic move.
- Alternated game.
- Contraction forced at each step.
- Beware of axioms.

Strategies

Plays are *sequence of moves*.

Strategies

- Prefix-closed set of plays,
- which accept any negative move,
- and are deterministic.

Winning strategy “answers” to all moves (i.e., doesn’t keep the token).

Innocence

View

For each move in a play, "follow" a player by restricting each position:

$$\begin{tikzcd}
P \arrow[r, hook] \arrow[d, "t"] & W \arrow[d, "p"] \\
U \arrow[r, hook] \arrow[d, "m'"] & X \arrow[d, "m"] \\
V \arrow[r, hook] & Y
\end{tikzcd}$$

Innocence

Any positive move is uniquely determined by the player's view.

Parallel composition

- Positions are endowed with a topology $\rightsquigarrow$ *restrict* a play/strategy over a site.
- Sheaf:
  - given a position U,
  - an open cover $(U_i)_i$ of U,
  - a "compatible" family of plays/strategies over the U_i's,
  - there exists a unique play/strategy over U which restricts to the given plays/strategies.

Descent

Cut dimension

Cut: collapse some edges into a single vertex.

Lemma (Factorisation system)

(Generalized) cuts commute with moves.

Amalgamation + descent is a composition of innocent strategies.

- Easy part: amalgamation of innocent strategies is innocent,
- Harder part: descent is innocent.

Link with response categories

How to prove that the category of innocent strategies is a free response category?

1. "Free" part easy to prove with syntactical categories (λ-calculus).
2. Relate innocent strategies to λ terms, s.t. composition of strategies is substitution of terms.

Introduction

Response categories

Graphical games

λ_r-calculus

λ_r terms

Raw terms are just usual λ terms:

$$\begin{array}{rcccccc} M, N, \ldots & ::= & x & \mid & () & \mid \\ & & \pi_i M & \mid & \langle M, N \rangle & \mid \\ & & \lambda x.M & \mid & MN & \mid \end{array} \tag{2}$$

Types:

$$A, B, \ldots \quad ::= \quad \mathbb{A} \quad \mid \quad 1 \quad \mid \quad A \times B \quad \mid \quad \neg A \tag{3}$$

Natural Deduction

Axiom as usual:

$$\Gamma, a \colon A \vdash a \colon A$$

Unit as usual:

$$\Gamma \vdash () \colon 1$$

Natural Deduction

Product as usual:

$$\frac{\Gamma \vdash M\colon B_1 \times B_2}{\Gamma \vdash \pi_i M\colon B_i}$$

$$\frac{\Gamma \vdash M\colon B \qquad \Gamma \vdash N\colon C}{\Gamma \vdash \langle M, N\rangle\colon B \times C}$$

Natural Deduction

Abstraction deserves more attention:

$$\frac{\Gamma \vdash M \colon \neg B \qquad \Gamma \vdash N \colon B}{\Gamma \vdash MN}$$

$$\frac{\Gamma, y \colon B \vdash M}{\Gamma \vdash \lambda y.M \colon \neg B}$$

β-normal η-long forms

Syntactic response category:

objects	are contexts,
morphisms	are tuples of $\beta\eta$-equivalence classes typed terms,
composition	is substitution.

Typing only normal forms

$$\Gamma, a\colon A \vdash_{\beta\eta} a \uparrow A \qquad\qquad \Gamma \vdash_{\beta\eta} () \downarrow 1$$

$$\frac{\Gamma, x\colon A \vdash_{\beta\eta} M}{\Gamma \vdash_{\beta\eta} \lambda x.M \downarrow \neg A} \qquad \frac{\Gamma \vdash_{\beta\eta} M \uparrow \neg A \qquad \Gamma \vdash_{\beta\eta} N \downarrow A}{\Gamma \vdash_{\beta\eta} MN}$$

$$\frac{\Gamma \vdash_{\beta\eta} M \uparrow A_1 \times A_2}{\Gamma \vdash_{\beta\eta} \pi_i M \downarrow A_i} \qquad \frac{\Gamma \vdash_{\beta\eta} M \downarrow A \qquad \Gamma \vdash_{\beta\eta} N \downarrow B}{\Gamma \vdash_{\beta\eta} \langle M, N\rangle \downarrow A \times B}$$

$$\frac{\Gamma \vdash_{\beta\eta} M \uparrow A}{\Gamma \vdash_{\beta\eta} M \downarrow A} \quad \text{when } A \text{ is a ground type}$$

Introducing β reduction

Make β-redexed typable:

$$\frac{\Gamma \vdash_\eta M \downarrow A}{\Gamma \vdash_\eta M \uparrow A} \ \beta \text{ redex}$$

Terms are still in η-long form.

Introducing η expansion

Allow non η-long terms:

$$\frac{\Gamma \vdash_\beta M \downarrow A}{\Gamma \vdash_\beta M \uparrow A} \quad \eta \text{ reduction}$$

Terms are still in β-normal form.

Subject reduction

Lemma

$\vdash$, $\vdash_{\beta\eta}$, $\vdash_\eta$ *and* $\vdash_\beta$ *enjoy subject reduction.*

Recovering the free response category

Negation:

$$y : \neg B \vdash \lambda x.yM : \neg A$$

Recovering the free response category

Isomorphism $\mathrm{Hom}(A \times B, \neg C) \to \mathrm{Hom}(A, \neg(B \times C))$:

$$\varphi(x \colon A \times B \vdash \lambda y^C.M \colon \neg C) := z \colon A \vdash \lambda w^{B \times C}.M[x \mapsto$$

$$\langle z, pi_1 w\rangle, y \mapsto \pi_2 w] \colon \neg(B \times C)$$

Recovering the free response category

Isomorphism $\mathrm{Hom}(A \times B, \neg C) \to \mathrm{Hom}(A, \neg(B \times C))$:

$$\varphi(x \colon A \times B \vdash \lambda y^C.M \colon \neg C) := z \colon A \vdash \lambda w^{B \times C}.M[x \mapsto$$

$$\langle z, pi_1 w \rangle, y \mapsto \pi_2 w] \colon \neg(B \times C)$$

Inverse map given by:

$$\varphi^{-1}(x \colon A \vdash \lambda y^{B \times C}.M \colon \neg(B \times C)) := z \colon A \times B \vdash \lambda w^C.M[x \mapsto$$

$$\pi_1 z, y \mapsto \langle \pi_2 z, w \rangle] \colon \neg C$$

Recovering the free response category

- This makes the syntactical category a response category.
- Evaluation map from λ-terms to any response category.

Further work

Further work

- Fill holes: today's definitions still require some tuning.
- Generalize our categorical machinery.
- Shift to an untyped setting.