Effects Handlers and Bicomodules

David I. Spivak (joint with Owen Lynch)

2022/12/12 at Topos Berkeley Seminar

I. Introduction

   A. CT & Functional vs. Imperative programming
   B. **Poly** in CS: DB, DTT, ADT, Lenses, FSA, with Kris: rewriting protocols
   C. Today: imperative programming
   D. Story of Owen-David collaboration
   E. Plan
      1. Left Kan extensions in **Poly**
      2. Selection categories
      3. Effects handlers

II. Left Kan extensions in **Poly**

   A. Recall definition of $(\mathbf{Poly}, y, \triangleleft)$
   B. Pointwise left Kan as adjoint to composition
   C. Proof that $\begin{bmatrix} q \\ p \end{bmatrix} = \sum_{I \in p(1)} y^{q(p[I])}$ (preserve coproducts, yoneda)[1]

III. Selection categories

   A. $\begin{bmatrix} p \\ p \end{bmatrix}$ has a comonoid structure for any $p : \mathbf{Poly}$[2]
   B. For any category c, also $\begin{bmatrix} p \\ p \triangleleft c \end{bmatrix}$ has a comonoid (category) structure
   C. Examples: $\begin{bmatrix} y^A \\ y^A \triangleleft c \end{bmatrix}, \begin{bmatrix} \text{list} \\ \text{list} \end{bmatrix}, \begin{bmatrix} Ay \\ Ay \triangleleft c \end{bmatrix}$.
   D. Why I called it selection categories ("reproduction")

IV. Effects handlers

   A. Math

[1] Credit to Josh Meyers for showing me that **Poly** has adjoint to composition, to Todd Trimble for noting that it's a Left Kan extension in **Cat**.

[2] Credit to David Jaz Myers.

1. TFAE for $m : \mathbf{Poly}$, $c, d : Ob(\mathbf{Comon}(\mathbf{Poly})) = Ob(\mathbf{Cat})$
   a. $m \triangleleft d \to c \triangleleft m$ satisfying counit and coassociative laws
   b. $\begin{bmatrix} m \\ m \triangleleft d \end{bmatrix} \nrightarrow c$ cofunctor
   c. bicomodule $c \xleftarrow{m \triangleleft d} d$
2. Every bicomodule $c \leftarrow 0$ and $c \leftarrow y$ are of this form.
3. The composite $c \xleftarrow{m \triangleleft d} d \xleftarrow{n \triangleleft e} e$ is $c \xleftarrow{m \triangleleft n \triangleleft e} e$.
4. Handler bicomodules form a double subcat of $\mathbf{Cat}^\sharp$
   a. Same verticals
   b. Handler horizontals
   c. It has conjoints because every cofunctor is a handler (with $m = y$).

B. Elementary effects handlers
1. If $c = \mathfrak{c}_p$ then we can identify cofunctors $\begin{bmatrix} m \\ m \triangleleft d \end{bmatrix} \nrightarrow c$ with poly maps $\begin{bmatrix} m \\ m \triangleleft d \end{bmatrix} \to p$.
2. If $d = \mathfrak{c}_q$ then we have projection $d \to q$, so we get a handler from any *elementary handler*, $m \triangleleft q \to p \triangleleft m$.

C. Example:
1. $m = \{0,1\}^{\mathbb{Z}}, \quad p = \{Go\}y + \{0,1\}^{\mathbb{Z}}, \quad q = \{\leftarrow, \rightarrow\} \times \{0,1\}y^{0,1}$
2. Tape handling is given by a very particular map $m \triangleleft q \to p \triangleleft m$
3. Get $\mathfrak{c}_p \leftarrow \mathfrak{c}_q$
   a. That is, a functor $p - \mathbf{Coalg} \to q - \mathbf{Coalg}$
   b. Think of it as a UI for Turing Machines

V. Conclusion

A. This connects imperative programming, dynamical systems, and data migration