CBPV$^{\mathrm{op}}$

Max S. New

August 30, 2018

This is a small explication of my preferred variant of call-by-push-value/intuitionistic sequent calculus. The main difference with CBPV is that we dualize the presentation of "computation types" and instead think of them as "stack types". This makes the entire presentation more symmetric than CBPV.

1 Judgmental Structure

The calculus has 2 kinds of types: value types (A) and *stack* types (B). The value types classify the data in local variables whereas the stack types classify, unsurprisingly, the stack. There are three term judgments: values $\Gamma \vdash V : A$ where Γ is a cartesian context of value-typed variables, stacks $\Gamma; \alpha : B \vdash S : B'$ where there is always exactly one stack variable on the left, and finally computations $\Gamma; \alpha : B \vdash M$ where there is no formula on the right. As a logic, we view M as a refutation of its context, i.e. somehow drawing a contradiction from its premises. What I like about the logic is that it has exactly the identity and substitution rules you would expect: substitution of a value for a value variable and substitution of a stack for a stack variable.

2 Type Structure

Types and terms are in figure **??**. Note that this is a more symmetric presentation than CBPV: all values, and stacks are sequences of introduction forms and variables, whereas all terms are elimination rules (or effect invocations if we add them).

I'll just write down the rules for multiplicatives for now, the additives should be clear. First, the positive types. Like in CBPV, positive types are eliminated by a uniform pattern matching construct. β and η equations are exactly as they should be. What's most different here is that since we have op'd the stacks, the cbpv function type $A \to B$ instead appears as the stack cons type $A \oslash B$, which is just an asymmetric tensor product.

$$\frac{\Gamma \vdash V_i : A_i}{\Gamma \vdash (V_1, V_2) : A_1 \times A_2} \qquad \frac{\Gamma \vdash V : A_1 \times A_2 \qquad \Gamma, x_1 : A_1, x_2 : A_2; \alpha : B \vdash M}{\Gamma; \alpha : B \vdash \text{pm } V \text{ to } (x_1, x_2).M} \qquad \frac{\Gamma \vdash V : A \qquad \Gamma; \alpha : B' \vdash S : B}{\Gamma; \alpha : B' \vdash (V, S) : A \oslash B}$$

$$\frac{\Gamma; \alpha : B \vdash S : A \oslash B' \qquad \Gamma, x : A; \beta : B' \vdash M}{\Gamma; \alpha : B \vdash \text{pm } S \text{ to } (x, \beta).M} \qquad \text{pm } (V_1, V_2) \text{ to } (x_1, x_2).M = M[V_1/x_1][V_2/x_2]$$

$$M = \text{pm } x \text{ to } (x_1, x_2).M[(x_1, x_2)/x]$$

$$\text{pm } (V, S) \text{ to } (x, \alpha).M = M[V/x][S/\alpha]$$

$$M = \text{pm } \alpha \text{ to } (x, \beta).M[(x, \beta)/\alpha]$$

$$\frac{x : A \in \Gamma}{\Gamma \vdash x : A} \qquad \Gamma; \alpha : B \vdash \alpha : B \qquad \frac{\Gamma \vdash V : A \qquad \Gamma, x : A \vdash V' : A'}{\Gamma \vdash V'[V/x] : A'} \qquad \frac{\Gamma \vdash V : A \qquad \Gamma, x : A; \alpha : B \vdash S : B'}{\Gamma; \alpha : B \vdash S[V/x] : B'}$$

$$\frac{\Gamma \vdash V : A \qquad \Gamma, x : A; \alpha : B \vdash M}{\Gamma; \alpha : B \vdash M[V/x]} \qquad \frac{\Gamma; \alpha : B \vdash S : B' \qquad \Gamma; \beta : B' \vdash S' : B''}{\Gamma; \alpha : B \vdash S'[S/\beta] : B''} \qquad \frac{\Gamma; \alpha : B \vdash S : B' \qquad \Gamma; \beta : B' \vdash M}{\Gamma; \alpha : B \vdash M[S/\beta]}$$

Figure 1: Substitution/Identity Rules, substitution rules are admissible

$$
\begin{array}{rcl}
A & ::= & 1 \mid A \times A \mid 0_v \mid A + A \mid \neg B \\
B & ::= & A \oslash B \mid 0_s \mid B \oplus B \mid \neg A \\
V & ::= & x \mid 1 \mid (V, V) \mid \mathtt{in}_i V \mid \lambda\alpha.M \\
S & ::= & \alpha \mid (V, S) \mid \mathtt{in}_i S \mid \lambda x.M \\
M & ::= & \mathrm{pm}\ V\ \mathrm{to}\ ().M \mid \mathrm{pm}\ V\ \mathrm{to}\ (x,y).M \mid \mathrm{pm}\ V\ \mathrm{to}\ \{\} \mid \mathrm{pm}\ V\ \mathrm{to}\ \{\mathtt{in}_1 x \mapsto M_1 \mid \mathtt{in}_2 y \mapsto M_2\} \mid V\,!\,S \\
 & & \mathrm{pm}\ S\ \mathrm{to}\ (x,\alpha).M \mid \mathrm{pm}\ S\ \mathrm{to}\ \{\} \mid \mathrm{pm}\ S\ \mathrm{to}\ \{\mathtt{in}_1 \alpha \mapsto M_1 \mid \mathtt{in}_2 \beta \mapsto M_2\} \mid S\,!\,V
\end{array}
$$

Figure 2: Type, Term Structure of CBPV$^{\mathrm{op}}$

Next, the shifts U, F of CBPV appear as the only *negative* connectives in CBPVop, which following Curry-Howard we think of as negations. Since these are the negative types, I'll use λ for these constructors.

$$
\frac{\Gamma; \alpha : B \vdash M}{\Gamma \vdash \lambda\alpha.M : \neg B} \qquad \frac{\Gamma \vdash V : \neg B' \qquad \Gamma; \alpha : B \vdash S : B'}{\Gamma; \alpha : B \vdash V\,!\,S} \qquad \frac{\Gamma, x : A; \alpha : B \vdash M}{\Gamma; \alpha : B \vdash \lambda x.M : \neg A} \qquad \frac{\Gamma; \alpha : B \vdash S : \neg A \qquad \Gamma \vdash V : A}{\Gamma; \alpha : B \vdash S\,!\,V}
$$

$$
(\lambda\alpha.M)\,!\,S = M[S/\alpha]
$$

$$
V = \lambda\alpha.V\,!\,\alpha
$$

$$
(\lambda x.M)\,!\,V = M[V/x]
$$

$$
S = \lambda x.S\,!\,x
$$

So let's do a Levy-style meaning explanation for the multiplicatives and the negations.

1. A value of $A_1 \times A_2$ is a pair of a value of A_1 and a value of A_2.
2. A stack of $A \oslash B$ is a pair of a value of A and a stack of B.
3. A value of $\neg B$ is code that can be run against a stack of type B
4. A stack of $\neg A$ is code (a continuation) that can be run against a value of type A.