1 Bicartesian Doubly Closed Category

Given a category $\mathcal{C}$, its presheaf category ($\widehat{\mathcal{C}} := [\mathcal{C}^{op}, Set]$) is bicartesian closed. Given a monoidal category $(\mathcal{C}, \otimes_C, I_C)$, its presheaf category is bicartesian closed and monoidal closed via the Day convolution. The monoidal product is given by:

$$(P \otimes^{Day} Q)(x) = \int^{y,z} \mathcal{C}[x, y \otimes_C z] \times P(y) \times Q(z)$$

The Day monoidal product has the universal property that any maps out of it are in bijective correspondence with a family of maps natural in x and y (Agda): [1]

$$\widehat{\mathcal{C}}[P \otimes^{Day} Q, R] \;\cong\; \widehat{\mathcal{C} \times \mathcal{C}}[P \overline{\times} Q, R \circ \otimes_C] \;\cong\; \Pi_{x,y:\, ob\ C}\ Set[P(x) \times Q(y), R(x \otimes_C y)]$$

The monoidal closed structure is given by:

$$(P \multimap\!\!\ast\, Q)(X) = \widehat{\mathcal{C}}[P, Q(X, -)]$$

With the universal property that the closed structure is right adjoint to the tensor (Agda):

$$\widehat{\mathcal{C}}[A \otimes_C B, C] \cong \widehat{\mathcal{C}}[A, B \multimap\!\!\ast\, C] \tag{1}$$

Bicartesian doubly closed categories have been used in the denotational semantics of bunched type theories [5][1][4].

2 Towards Bunched Call By Push Value with Dynamic Store

Categorical models of dynamic store use presheaf categories to model the dependence of the heap structure on a current *world* [3][6][2]. Seemingly none of these existing models attempt to combine a call by push value language with the separating type connectives, $\otimes$ and $\multimap\!\!\ast$, used in bunched type theories. Our investigation into possible models of such a language have run into some potential issues when attempting to define the computational separating function type. To illustrate this, we will start with the model for a call by push value language with dynamic store presented in chapter 7 of Levy's thesis.

2.1 Definitions

Let $(C, \otimes_C, I_C)$ be a monoidal category, the value category be $\mathcal{V} := [C^{op}, Set]$, computation category $\mathcal{C} := [C, Set]$, and use the *standard* monad for ground dynamic store with $F : \mathcal{V} \to \mathcal{C}$ as:

$$F(A)(x) := \Sigma_{y:ob\ C}\Sigma_{f:C^{op}[x,y]}A(y)$$

and $U : \mathcal{C} \to \mathcal{V}$ as :

$$U(\underline{B})(x) := \Pi_{y:ob\ C}\Pi_{f:C^{op}[x,y]}\underline{B}(y)$$

The oblique morphisms in this model are given by families of maps:

$$\mathcal{O}[A, \underline{B}] := \Pi_{x:ob\ C} Set[A(x), \underline{B}(x)]$$

we have the following isomorphims:

$$\mathcal{V}[A, U(\underline{B})] \cong \mathcal{O}[A, \underline{B}] \cong \mathcal{C}[F(A), \underline{B}]$$

And we can attempt to define a computation separating function by:

$$(A \multimap\!\!\ast\, \underline{B})(x) := \Pi_{y:ob\ C} Set[A(y), \underline{B}(x \otimes_C y)]$$

2.2 Problems with an Abstract Monoidal Category

Before committing to the category of worlds used in Levy's model, we will work with an arbitrary monoidal category $(C, \otimes_C, I_C)$.

[1] here $\overline{\times}$ is the *external* product

2.2.1 Issue 1: Universal Property of Tensor for Oblique Morphisms

Let's attempt to show the following:

$$\mathcal{O}[P \otimes Q, \underline{R}] \cong \mathcal{O}\times[P \overline{\times} Q, \underline{R} \circ \otimes_C]$$

where

$$\mathcal{O}\times[P \overline{\times} Q, \underline{R} \circ \otimes_C] := \Pi_{x,y:ob\ C} Set[P(x) \times Q(y), \underline{R}(x \otimes_C y)]$$

A problem arises when trying to define the backwards map of this isomorphims. Given $m : \mathcal{O}\times[P \overline{\times} Q, \underline{R} \circ \otimes_C]$ and $x : ob\ C$, we need to define a map $Set[(P \otimes Q)(x), \underline{R}(x)]$. This is a map out of a coequalizer [2] which we can attempt to give as a map induced from:

$$(f : C[x, y \otimes_C z], p : P(y), q : Q(z)) \mapsto ? : \underline{R}(x)$$

However, using the data we currently have, we can only construct

$$m(y)(z)(p, q) : \underline{R}(y \otimes_C z)$$

and since $\underline{R}$ is covariant in C, we can't use $\underline{R}(f) : \underline{R}(x) \to \underline{R}(y \otimes_C z)$. This is not surprising since the proof of this universal property in the value category $\mathcal{V}[P \otimes Q, R] \cong \mathcal{V}\times[P \overline{\times} Q, R \circ \otimes_C]$ uses the functorial action of R on f (see here) [3] So by swapping the variance of R (now $\underline{R}$ since it is from the computation category) this proof should break. Seemingly, this proof won't go through when we assume a generic monoidal category C. Perhaps we can recover this property if we work with a specific concrete category?

2.2.2 Issue 2: Universal Property of the Separating Function Type

This is just another perspective on the variance issue above. We'd like to show

$$\mathcal{O}[P \otimes Q, \underline{R}] \cong \mathcal{O}[P, Q \twoheadrightarrow \underline{R}]$$

Since we don't have the universal property of tensor for oblique morphisms, we can try to get at this proof via the universal property of tensor in the value category. Note that we have

$$\mathcal{O}[P \otimes Q, \underline{R}] \cong \mathcal{V}[P \otimes Q, U(\underline{R})] \cong \mathcal{V}\times[P \overline{\times} Q, U(\underline{R}) \circ \otimes_C]$$

and

$$\mathcal{O}[P, Q \twoheadrightarrow \underline{R}] \cong \mathcal{V}[P, U(Q \twoheadrightarrow \underline{R})]$$

So we can try to show

$$\mathcal{V}\times[P \overline{\times} Q, U(\underline{R}) \circ \otimes_C] \cong \mathcal{V}[P, U(Q \twoheadrightarrow \underline{R})]$$

Again, we fail to define the backwards direction of this isomorphim due to a variance issue with $\underline{R}$. Given $m : \mathcal{V}[P, U(Q \twoheadrightarrow \underline{R})]$, it suffices to construct a map $eval : \mathcal{V}\times[U(Q \twoheadrightarrow \underline{R}) \overline{\times} Q, U(\underline{R}) \circ \otimes_C]$ with components

$$(x, y)(f : U(Q \twoheadrightarrow \underline{R})(x), q : Q(y)) \mapsto ?\ :\ (U(\underline{R}))(x \otimes_C y)$$

unfolding some of the definitions, we have

$$\begin{aligned}
&f : \Pi_{z:ob\ C}\Pi_{g:C^{op}[x,z]}(\Pi_{w:ob\ C} Set[Q(w), \underline{R}(z \otimes_c w)]) \\
&? : \Pi_{z:ob\ C}\Pi_{g:C^{op}[x \otimes_C y, z]}(R(z))
\end{aligned}$$

Thus we have to define $? : \underline{R}(z)$ from the following data:

$$\begin{aligned}
&x, y, z : ob\ C \\
&q : Q(y) \\
&f : \Pi_{z:ob\ C}\Pi_{g:C^{op}[x,z]}(\Pi_{w:ob\ C} Set[Q(w), \underline{R}(z \otimes_c w)]) \\
&g : C^{op}[x \otimes_C y, z]
\end{aligned}$$

The *obvious* thing do to would be to use $f(x)(id_x)(y)(q) : \underline{R}(x \otimes_C y)$ and $\underline{R}(g)$, but the variance of $\underline{R}$ is working against us.

[2] since coends in Set can be encoded as coequalizers

[3] note the difference in variance is due to the fact this proof is for presheaves and not covariant presheaves

2.2.3 Issue 3: Problem with Action Laws

The following three isomorphims should hold:

$$I_D \twoheadrightarrow^c Q \cong Q$$
$$P \otimes_D P' \twoheadrightarrow^c Q \cong P \twoheadrightarrow^c P' \twoheadrightarrow^c Q$$
$$U(P \twoheadrightarrow^c Q) \cong P \twoheadrightarrow^v UQ$$

where $P, P' : ob\ \mathcal{V}$, $Q : ob\ \mathcal{C}$, and $\otimes_D, I_D := Yoneda(I_C) = C[-, I_C]$ are the day convolution product and its identity. The top two isomorphims come from the requirement that CBPV function types should be an action of $\mathcal{V}^{op}$ on $\mathcal{C}$. Let's see how the first isomorphism fails. Defining the forward direction by its components:

$$(x : ob\ C)(f : \Pi_{y:ob\ C} Set[C[y, I_C], Q(x \otimes_C y)]) \mapsto ?\ : Q(x)$$

we have

$$f(I_C)(id_{I_C}) : Q(x \otimes_C I_C)$$

from which we can obtain

$$Q(x \otimes_C I_C \xrightarrow{idr} x)(f(I_C)(id_{I_C}) : Q(x \otimes_C I_C)) : Q(x)$$

In attempting to define the backwards direction, we run into issues.

$$(x : ob\ C)(q : Q(x))(y : ob\ C)(f : C[y, I_C]) \mapsto ?\ : Q(x \otimes_C y)$$

We'd expect to use the functorial action of Q on morphisms of C

$$Q(g)(q) : Q(x \otimes_C y)$$

but we'd need a morphism $g : C[x, x \otimes_C y]$. Note that **if the direction of f were inverted**, we'd be able to define this by

$$g := x \xrightarrow{idr^{-1}} x \otimes_C I_C \xrightarrow{id_x \otimes_C f} x \otimes_C y$$

2.3 Problems with Concrete Models

Now we consider substituting the monoidal category $(C, \otimes_C, I_C)$ with $(FinSet^{op}_{mono}, \oplus, \emptyset)$ where $\oplus$ is given by disjoint union of sets.. This category is used to represent single sorted heap configurations.

2.3.1 Action Laws with Concrete Category

Again we attempt to show $I_D \twoheadrightarrow^c Q \cong Q$. The forward direction holds following the abstract case. It seems we can't define the backwards direction.

$$(x : ob\ C)(r : Q(x))(y : ob\ C)(! : C[\emptyset, y]) \mapsto ?\ : Q(x \uplus y)$$

again, we'd expect to use the functorial action of Q

$$Q(g)(r) : Q(x \uplus y)$$

where $g : C[x \uplus y, x]$, but we have no hope of defining this morphism! HARD STOP. Notice that even if the given map was inverted as we desired in the abstract case, that is, $C[y, \emptyset]$ instead of $C[\emptyset, y]$, then we could define this backwards isomorphim using absurd!

2.3.2 Universal Property of Tensor for Oblique Morphisms

See Agda file here. Again, we will attempt to show the following:

$$\mathcal{O}[P \otimes Q, \underline{R}] \cong \mathcal{O}\times[P \overline{\times} Q, \underline{R} \circ \oplus]$$

We reconsider the backwards direction, given

$$m : \Pi_{x,y:ob\ C} Set[P(x) \times Q(y), \underline{R}(x \uplus y)]$$

we need to construct components of the form $?_z : Set[(P \otimes Q)(z), \underline{R}(z)]$. To map out of the coequalizer, we can define a map

$$(f : x \uplus y \to z, p : P(x), q : Q(y)) \mapsto ? : \underline{R}(z)$$

Attempt 1

We can promote p and q to the larger world z.

$$\begin{aligned}
g &: x \to z = f \circ inl \\
h &: y \to z = f \circ inr \\
p' &: P(z) = P(g)(p) \\
q' &: Q(z) = Q(h)(q)
\end{aligned}$$

We can then use m at (z, z)

$$m(z)(z)(p', q') : \underline{R}(z \uplus z)$$

and **arbitrarily** restrict the resulting element of $\underline{R}(z \uplus z)$ to $\underline{R}(z)$ using $\underline{R}(inl)$ or $\underline{R}(inr)$. This does satisfy the coequalizer requirement. To define a map out of the day convolution product, we can define a map on the underlying *diagram* and then prove a coequalizer condition. Given:

$$\begin{aligned}
&y\ z\ y'\ z' : ob\ C \\
&f : C[y', y] \\
&g : C[z', z] \\
&(h : C[x, y' \otimes_c z'], p : P(y), q : Q(z))
\end{aligned}$$

Any map, m, out of the diagram must satisfy

$$m(h; (f \otimes_c g), p, q) = m(h, P(f)(p), Q(g)(q))$$

Concretely, in this case:

$$\begin{aligned}
&R(inl)(m\ x\ x(P(inl; f \otimes_c g; h)(p), Q(inr; f \otimes_c g, h)(q))) \\
&= \\
&R(inl)(m\ x\ x(P(f; inl; h)(p), Q(g; inr; h)(q)))
\end{aligned}$$

However, the section and retraction of the isomorphim looks less promising. It seems we are missing some kind of naturality condition.

Section:

Given:

$b : \mathcal{O}\times[P \overline{\times} Q, \underline{R} \circ \oplus]$

$x\ y : ob\ FinSet_{mono}$

$p : P(x)$

$q : Q(y)$

Show:

$R(inl)(b(x \uplus y)(x \uplus y)(P(inl)(p), Q(inr)(q))) = b\ x\ y\ (p, q)$

Retraction:

Given:

$b : \mathcal{O}[P \otimes Q, \underline{R}]$

$x\ y\ z : ob\ FinSet_{mono}$

$f : FinSet_{mono}[y \uplus z, x]$

$p : P(y)$

$q : Q(z)$

Show:

$R(inl)(b(x \uplus y)[(id, P(inl; f)(p), Q(inr; f)(q))]) = b\ x\ [(f, p, q)]$

Attempt 2
We reconsider the backwards direction, given

$$m : \Pi_{x,y:ob\ C} Set[P(x) \times Q(y), \underline{R}(x \uplus y)]$$

we need to construct components of the form $?_z : Set[(P \otimes Q)(z), \underline{R}(z)]$. To map out of the coequalizer, we can define a map

$$(f : x \uplus y \to z, p : P(x), q : Q(y)) \mapsto ? : \underline{R}(z)$$

We can recognize that since f is injective and the domain is a disjoint union, z is partitioned into three parts

$$\begin{aligned} z_x &: \text{the range of f restricted to x} \\ z_y &: \text{the range of f restricted to y} \\ z_{miss} &: z - (z_x \uplus z_y) \\ \text{where } z &\cong z_x \uplus z_y \uplus z_{miss} \end{aligned}$$

Thus we can **arbitrarily** promote p to $p' : P(z_x \uplus z_{miss})$ or q to $q' : Q(z_y \uplus z_{miss})$

$$\begin{aligned} &m(z_x \uplus z_{miss})(z_y)(p', q) \\ &m(z_x)(z_y \uplus z_{miss})(p, q') \end{aligned}$$

Choose the first option, check the coequalizer condition.

$$\begin{aligned} &p : P(y), q : Q(z) \\ &f : y \to y' \\ &g : z \to z' \\ &m(x_y \uplus x_{miss})(x_z)(P(y \to (x_y \uplus x_{miss}))(p), q) = m(x_{y'} \uplus x_{miss})(x_{z'})(P(), Q(g)(q)) \end{aligned}$$

References

[1] Biering, B. On the Logic of Bunched Implications.

[2] Kammar, O., Levy, P. B., Moss, S. K., and Staton, S. A monad for full ground reference cells. In *2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)* (June 2017), pp. 1–12.

[3] Levy, P. *Call-By-Push-Value: A Functional/Imperative Synthesis*. 01 2004.

[4] O'Hearn, P. On bunched typing. *Journal of Functional Programming 13*, 4 (July 2003), 747–796.

[5] Pym, D. J. *The Semantics and Proof Theory of the Logic of Bunched Implications*, vol. 26 of *Applied Logic Series*. Springer Netherlands, Dordrecht, 2002.

[6] Sterling, J., Gratzer, D., and Birkedal, L. Denotational semantics of general store and polymorphism, Apr. 2023. arXiv:2210.02169 [cs].