A DIRECTED TYPE THEORY FOR FORMAL CATEGORY THEORY

DAN LICATA, ANDREAS NUYTS, PATRICK SCHULTZ, AND MICHAEL SHULMAN

1. Introduction

We describe a two-level theory with two kinds of contexts and types. The first level, whose types we call **categories**, is a simple linear type theory with an involutive modality, represented judgmentally by assigning variances to variables in the context. Thus, for instance, a typical term judgment would look like

$$x \overset{+}{:} A, y \overset{-}{:} B, z \overset{+}{:} C \vdash t : D$$

As usual, linearity means that all the rules maintain the invariant that each variable in the context is used exactly once in the conclusion. In particular, there are no contraction and weakening rules; but we do allow an unrestricted (and usually implicit) exchange rule. We often write Ψ for contexts of category-variables marked with variances ("**psi**gned contexts"), and we write Ψ^* for the analogous context with all variances reversed; thus $(x \overset{+}{:} A, y \overset{-}{:} B)^* = (x \overset{-}{:} A, y \overset{+}{:} B)$.

Substitution into a variable of variance "$-$" (a "contravariant variable") reverses variances. For instance, we have

$$\frac{x \overset{+}{:} A, y \overset{-}{:} B \vdash f(x,y) : C \qquad z \overset{-}{:} C \vdash g(z) : D}{x \overset{-}{:} A, y \overset{+}{:} B \vdash g(f(x,y)) : D}$$

If necessary, we may write $f(x,y)^*$ to denote the term $f(x,y)$ with all the variances of its variables reversed; of course this is not by itself a valid term, but only something that can be substituted for a contravariant variable.

The second level is a more complicated sort of linear type theory, whose types we call **types** (or sometimes **sets** or **modules**), and that depends "quadratically" on the first level (in a sense made precise below). Its basic type judgment is

$$\Psi \vdash M \ \mathsf{type}$$

That is, each type depends on some collection of category-variables, with variance.

The basic term judgment for types is

$$\Delta \mid \Gamma \vdash M$$

Here Γ is a context of type-variables and M is a type. Unlike the first level, we formulate this second level as a sequent calculus, and instead of including terms in the judgments we regard the derivations themselves as the (normal-form) terms. Later on we may introduce an equational theory, a term calculus, a third layer of logic with equality, or other techniques. This sequent calculus is also structurally linear, with no contraction or weakening, but with unrestricted exchange.

The interesting thing happens in the context Δ, which is a list of category-variables *without* variances on which the types in Γ and M can depend. This dependence is "quadratic" in the following sense: all the rules maintain the invariant that each variable in Δ appears *twice* in Γ and M, and that the two occurrences have opposite variance (where the variance of Γ is flipped relative to M). In fact, to be more precise, the actual dependence of Γ and M is on a category-context *with* variances that is obtained from Δ by splitting each variable into two with opposite variance. The implementation will, of course, use de Brujin indices; when writing judgments with named variables, if $x : A$ is a variable in Δ we write $\hat{x} \overset{+}{:} A$ and $\check{x} \overset{-}{:} A$ for the two corresponding signed variables that must appear in Γ and M (exactly once each).

If Ψ is a signed context, we write Ψ^0 for the unsigned context obtained by discarding the variances, and also any hats or checks on the variables. And if the variables in Ψ have hats and checks (which technically means that it was obtained as part of a splitting an unsigned context), we write $\Psi^\dagger$ for the signed context obtained by reversing the variances *and* interchanging hats and checks. In particular, we always have $\Psi^0 = (\Psi^\dagger)^0$.

With this notation, we can express the variable dependence conditions as follows:

$$\Delta \mid \Gamma \vdash N \text{ requires as preconditions that } \begin{array}{c} \Delta = \Psi^0 \\ \Psi, \Psi^\dagger \cong \Psi_1, \Psi_2 \\ \Psi_1^* \vdash \Gamma\ \mathsf{ctx} \\ \Psi_2 \vdash M\ \mathsf{type} \end{array}$$

As above, the notation Ψ_1^* means that we reverse the variances of the variables in Ψ_1 (without changing the hats and checks). For example, if $\Delta = (x : A)$ then we could have $\Psi = (\hat{x} \overset{+}{:} A)$, so that $\Psi^\dagger = (\check{x} \overset{-}{:} A)$; we could then take $\Psi_1 = (\check{x} \overset{-}{:} A)$ and $\Psi_2 = (\hat{x} \overset{+}{:} A)$, so that $\Psi_1^* = (\check{x} \overset{+}{:} A)$ and so a well-formed judgement could have $\check{x} \overset{+}{:} A \vdash \Gamma$ ctx and $\hat{x} \overset{+}{:} A \vdash A$ type. Note that that $\check{x}$ is allowed to occur covariantly in Γ, which counts as a negative occurrence overall, because Γ is itself in a contravariant position.

To express rules more readably, we introduce the following combined judgment

$$\frac{\Psi_1^* \vdash \Gamma\ \mathsf{ctx} \qquad \Psi_2 \vdash N\ \mathsf{type}}{\Psi_1, \Psi_2 \vdash \Gamma\ \mathsf{ctx}, N\ \mathsf{type}} \tag{1}$$

As always, concatenation of contexts is implicitly done up to permutation of variables. Thus, the preconditions for $\Delta \mid \Gamma \vdash N$ to make sense can alternatively be written as $\Delta = \Psi^0$ where $\Psi, \Psi^\dagger \vdash \Gamma\ \mathsf{ctx}, N\ \mathsf{type}$.

If $\hat{x}$ appears in M and $\check{x}$ appears in Γ, as above, then our judgment might look like this:

$$x : A \mid N(\check{x}) \vdash M(\hat{x})$$

In this case we are simply talking about a morphism of types "over the category A" — in semantic terms, a natural transformation. The dual case when $\check{x}$ appears in M and $\hat{x}$ appears in Γ simply represents a natural transformation between *contravariant* functors rather than between covariant ones. With this case in mind, one might say that we resolve the problem of "dependent linear type theory" by stipulating that both the context and the conclusion of a "linearly dependent judgment" must *separately* depend "linearly" on the same variables.

However, it is the additional freedom to allow $\hat{x}$ and $\check{x}$ to *both* appear in M and neither in Γ, or vice versa, that gives us the ability to express formal versions of nontrivial facts about category theory. Semantically, these judgments correspond to what are sometimes called *extraordinary natural transformations*, or simply *extranatural transformations*. For example, each category A will have a type of morphisms that is contravariant in its first variable and covariant in its second:

$$x \overset{-}{:} A, y \overset{+}{:} A \vdash \hom_A(x, y) \ \mathsf{type}$$

To express the *composition* of such morphisms then requires an "extranaturality judgment":

$$x : A, y : A, z : A \mid \hom_A(\hat{x}, \check{y}), \hom_A(\hat{y}, \check{z}) \vdash \hom_A(\check{x}, \hat{z})$$

The *identity* morphism judgment is similarly extranatural on the other side:

$$x : A \mid \cdot \vdash \hom_A(\check{x}, \hat{x})$$

If a is a category-term in context Ψ, we will write $a^\dagger$ for the result of interchanging hats and checks in a. This is "in context $\Psi^\dagger$" but of course it is not well-typed on its own; like a^*,its purpose is to be substituted into a contravariant variable. See, for instance, the substitution rule below.

2. Structural and admissible rules

2.1. Basic rules. The basic structural rules for context formation are unsurprising, given our linearity restriction:

$$\frac{}{\cdot \vdash \cdot \ \mathsf{ctx}} \qquad \frac{\Psi_1 \vdash \Gamma \ \mathsf{ctx} \qquad \Psi_2 \vdash M \ \mathsf{type}}{\Psi_1, \Psi_2 \vdash \Gamma, M \ \mathsf{ctx}}$$

For the moment, we are hoping to describe a cut-free sequent calculus with an admissible cut rule, and also admissible substitution for category-variables. We discuss the cut rule in its own section, since it is somewhat complicated. But we can state the intended identity rule at this point:

$$\frac{\Psi \vdash M \ \mathsf{type}}{\Psi^0 \mid M[\Psi^\dagger/\Psi] \vdash M}$$

For example, we might have

$$\frac{x \overset{+}{:} A, y \overset{-}{:} B, z \overset{+}{:} C \vdash M(x, y, z) \ \mathsf{type}}{x : A, y : B, z : C \mid M(\check{x}, \hat{y}, \check{z}) \vdash M(\hat{x}, \check{y}, \hat{z})}$$

The substitution rule for category-variables (which we also intend to be admissible) will be

$$\frac{\Psi \vdash a : A \qquad \Delta, x : A \mid \Gamma \vdash M}{\Delta, \Psi^0 \mid \Gamma[a/\hat{x}, a^\dagger/\check{x}] \vdash M[a/\hat{x}, a^\dagger/\check{x}]}$$

Of course, by quadraticality, each of $\hat{x}$ and $\check{x}$ appears exactly once in Γ and M combined. We may abbreviate a substitution $M[a/\hat{x}, a^\dagger/\check{x}]$ by $M[a/x]$.

2.2. Cut rule. The basic cut rule looks like this:

$$\frac{\begin{array}{c}\Delta_1 \mid \Gamma_1 \vdash M' \qquad \Delta_2 \mid \Gamma_2, M \vdash N \\ M^\dagger \text{ loop-free-unifies with } M' \text{ reducing } \Delta_1, \Delta_2 \text{ to } \Delta \text{ by } \theta\end{array}}{\Delta \mid \Gamma_1[\theta, \theta^\dagger], \Gamma_2[\theta, \theta^\dagger] \vdash N[\theta, \theta^\dagger]}$$

As usual, we assume at the outset that the variables in Δ_1 and Δ_2 are disjoint. Then θ is a substitution for some of the signed variables in Δ_1, Δ_2, assigning each of them to a term depending on signed versions of other variables in Δ_1, Δ_2 (i.e. with hats and/or checks), that is supposed to be produced by "loop-free unification" (to be described). As usual, we write $M^\dagger$ and $\theta^\dagger$ for the results of reversing all hats and checks (here a purely syntactic operation); and Δ is the obtained by omitting from Δ_1, Δ_2 those variables that are substituted for in θ.

The short description of loop-free unification is that it is like ordinary first-order unification, except that

(1) after deciding on a substitution for a variable, we alter the remaining equations not just by that substitution but also by its dual (reversing hats and checks); and
(2) an "identity" equation such as $\hat{x} \doteq \hat{x}$ is a failure rather than simply deleted (the loop-free condition).

To explain this, we begin with simple examples. The simplest case is when neither M' nor M contains any duplicated variables. For instance, we can cut like this:

$$\frac{x : A \mid \Gamma_1 \vdash M(\hat{x}) \qquad y : A \mid \Gamma_2, M(\check{y}) \vdash N}{x : A \mid \Gamma_1, \Gamma_2[\hat{x}/\hat{y}] \vdash N[\hat{x}/\hat{y}]}$$

where unifying $M(\hat{x})$ with $M(\check{y})^\dagger = M(\hat{y})$ produces $\hat{y} \mapsto \hat{x}$. (If it instead produced $\hat{x} \mapsto \hat{y}$, then the conclusion would be the α-variant $y : A \mid \Gamma_1[\check{y}/\check{x}], \Gamma_2 \vdash N$.) We can also cut like this:

$$\frac{x : A \mid \Gamma_1 \vdash M(f(\hat{x})) \qquad y : B \mid \Gamma_2, M(\hat{y}) \vdash N}{x : A \mid \Gamma_1, \Gamma_2[f(\hat{x})/\check{y}] \vdash N[f(\hat{x})/\check{y}]}$$

Here the unification produces $\check{y} \mapsto f(\hat{x})$, and it must be that f is contravariant, i.e. that $u \mathrel{\bar{:}} A \vdash f(u) : B$. It may also happen that there are variable duplications in the type being cut. For instance, we can cut:

$$\frac{x : A, y : A \mid \Gamma_1 \vdash M(\check{x}, \hat{y}) \qquad z : A \mid \Gamma_2, M(\hat{z}, \check{z}) \vdash N}{z : A \mid \Gamma_1[\hat{z}/\hat{x}, \check{z}/\check{y}], \Gamma_2 \vdash N}$$

Here we have unified $M(\check{x}, \hat{y})$ with $M(\check{z}, \hat{z})$ to get $(\check{x} \mapsto \check{z}, \hat{y} \mapsto \hat{z})$. A different path of unification could start by choosing $\check{z} \mapsto \check{x}$, and then the dual of this equation would alter the remaining equation $\hat{y} \doteq \hat{z}$ to $\hat{y} \doteq \hat{x}$. This would result in an eventual substitution such as $(\check{z} \mapsto \check{x}, \hat{y} \mapsto \hat{x})$, giving the α-variant conclusion $x : A \mid \Gamma_1[\check{x}/\check{y}], \Gamma_2 \vdash N$.

A more complicated allowable cut is

$$\frac{x : A, y : A \mid \Gamma_1 \vdash M(\hat{x}, \check{y}, \hat{y}) \qquad z : A, w : A \mid \Gamma_2, M(\check{z}, \hat{z}, \check{w}) \vdash N}{z : A \mid \Gamma_1[\check{z}/\check{x},], \Gamma_2[\hat{z}/\hat{w}] \vdash N[\hat{z}/\hat{w}]}$$

Here unifying $M(\hat{x}, \check{y}, \hat{y})$ with $M(\hat{z}, \check{z}, \hat{w})$ has produced $(\hat{x} \mapsto \hat{z}, \check{y} \mapsto \check{z}, \hat{w} \mapsto \hat{z})$. Finally, here is an example where the loop-free condition fails:

$$\frac{x : A \mid \Gamma_1 \vdash M(\check{x}, \hat{x}) \qquad y : A \mid \Gamma_2, M(\hat{y}, \check{y}) \vdash N}{???}$$

Here the unification of $M(\check{x}, \hat{x})$ with $M(\check{y}, \hat{y})$ may begin by choosing $\hat{x} \mapsto \hat{y}$, but then the remaining equation is substituted to become $\check{y} \doteq \check{y}$, which fails the loop-free condition.

3. Type formers

Now we are ready to start giving the rules for some type formers. Since we want substitution into category-variables to be admissible, many rules must incorporate a substitution; but they are easier to understand before that substitution has been incorporated. Thus, we often give both versions. In the version without substitutions, we also generally omit the judgments of well-formedness of the types and contexts. Note that the version with substitutions included can usually be derived fairly automatically from the other.

3.1. **Morphism types.** We begin with the morphism types. The formation and right rules are straightforward given the expected variance:

$$\frac{A\ \mathsf{cat}}{x \overset{-}{:} A, y \overset{+}{:} A \vdash \hom_A(x,y)\ \mathsf{type}} \qquad \frac{A\ \mathsf{cat}}{x : A \mid \cdot \vdash \hom_A(\check{x},\hat{x})}$$

When we incorporate substitutions, these become:

$$\frac{A\ \mathsf{cat} \qquad \Psi_1 \vdash a : A \qquad \Psi_2 \vdash b : A}{\Psi_1^*, \Psi_2 \vdash \hom_A(a,b)\ \mathsf{type}} \qquad \frac{\Psi \vdash a : A}{\Psi^0 \mid \cdot \vdash \hom_A(a^\dagger, a)}$$

There are three versions of the left rule. The first "two-sided" one is a "directed" analogue of the elimination rule for equality due to Lawvere and Martin-Löf:

$$\frac{\Delta, x : A \mid \Gamma[\hat{x}/\hat{y}] \vdash M[\hat{x}/\hat{y}]}{\Delta, x : A, y : A \mid \Gamma, \hom_A(\hat{x},\check{y}) \vdash M}$$

$$\frac{\begin{array}{c}\Psi, \Psi^\dagger, \hat{x} \overset{+}{:} A, \check{x} \overset{-}{:} A \vdash \Gamma\ \mathsf{ctx}, M\ \mathsf{type} \\ \Psi_a \vdash a : A \qquad \Psi_b \vdash b : A \qquad \Psi^0, x : A \mid \Gamma \vdash M\end{array}}{\Psi^0, \Psi_a^0, \Psi_b^0 \mid \Gamma[a^\dagger/\check{x}, b/\hat{x}], \hom_A(a, b^\dagger) \vdash M[a^\dagger/\check{x}, b/\hat{x}]}$$

Note that although we write substitutions in both Γ and M, the linearity means that each variable such as $\hat{x}$ or $\check{x}$ can only occur in one of the two.

For instance, we can use this rule to define composition of morphisms:

$$\frac{\dfrac{\dfrac{x \overset{+}{:} A \vdash x : A}{x : A \mid \cdot \vdash \hom_A(\check{x},\hat{x})}}{x : A, y : A \mid \hom_A(\hat{x},\check{y}) \vdash \hom_A(\check{x},\hat{y})}}{x : A, y : A, z : A \mid \hom_A(\hat{x},\check{y}), \hom_A(\hat{y},\check{z}) \vdash \hom_A(\check{x},\hat{z})} \tag{2}$$

Starting from the bottom we have the left rule on y, z, then the left rule again on x, y, then the right rule on x. Note that $x : A, y : A \mid \hom_A(\hat{x},\check{y}) \vdash \hom_A(\check{x},\hat{y})$ is an instance of the identity rule, so if we had that rule postulated we could stop there; but if identity is to be admissible then it would reduce to the above complete derivation.

Similarly, we have the functorial action of any judgment $x \overset{+}{:} A \vdash f(x) : B$:

$$\frac{\dfrac{x \overset{+}{:} A \vdash f(x) : B}{x : A \mid \cdot \vdash \hom_B(f(\check{x}), f(\hat{x}))}}{x : A, y : A \mid \hom_A(\hat{x},\check{y}) \vdash \hom_B(f(\check{x}), f(\hat{y}))} \tag{3}$$

And the action of morphisms on types (directed transport):

$$\dfrac{\dfrac{\Delta, u \overset{\dagger}{:} A \vdash M \text{ type}}{\Delta, x : A \mid M[\check{x}/u] \vdash M[\hat{x}/u]}}{\Delta, x : A, y : A \mid M[\check{x}/u], \hom_A(\hat{x}, \check{y}) \vdash M[\hat{y}/u]} \tag{4}$$

We also consider "one-sided" left rules for morphism types (directed analogues of the Paulin-Mohring rule for identity types):

$$\dfrac{\Delta \mid \Gamma[b^\dagger/\check{x}] \vdash M[b^\dagger/\check{x}]}{\Delta, x : A \mid \Gamma, \hom_A(\hat{x}, b^\dagger) \vdash M} \qquad \dfrac{\Delta \mid \Gamma[a/\hat{x}] \vdash M[a/\hat{x}]}{\Delta, x : A \mid \Gamma, \hom_A(a, \check{x}) \vdash M}$$

And with substitutions:

$$\dfrac{\Psi, \Psi^\dagger, \Psi_b, u \overset{-}{:} A \vdash \Gamma \text{ ctx}, M \text{ type} \qquad \Psi_a \vdash a : A \qquad \Psi^0, \Psi_b^0 \mid \Gamma[b^\dagger/u] \vdash M[b^\dagger/u]}{\Psi^0, \Psi_b^0, \Psi_a^0 \mid \Gamma[a^\dagger/u], \hom_A(a, b^\dagger) \vdash M[a^\dagger/u]}$$

$$\dfrac{\Psi, \Psi^\dagger, \Psi_a^\dagger, v \overset{\dagger}{:} A \vdash \Gamma \text{ ctx}, M \text{ type} \qquad \Psi_b \vdash b : A \qquad \Psi^0, \Psi_a^0 \mid \Gamma[a/v] \vdash M[a/v]}{\Psi^0, \Psi_a^0, \Psi_b^0 \mid \Gamma[b/v], \hom_A(a, b^\dagger) \vdash M[b/v]}$$

Note that these can be regarded as the two Yoneda lemmas, one for covariant functors and one for contravariant functors.

Either one-sided rule makes the two-sided rule admissible, just as the Paulin-Mohring rule implies the Martin-Löf one. For instance, here is a derivation of the two-sided rule from the second one-sided rule, using substitution for category-variables:

$$\dfrac{\Psi_b \vdash b : A \qquad \dfrac{\Psi_a \vdash a : A \qquad \Psi^0, x : A \mid \Gamma \vdash M}{\Psi^0, \Psi_a^0 \mid \Gamma[a^\dagger/\check{x}][a/\hat{x}] \vdash M[a^\dagger/\check{x}][a/\hat{x}]}}{\Psi^0, \Psi_a^0, \Psi_b^0 \mid \Gamma[a^\dagger/\check{x}][b/\hat{x}], \hom_A(a, b^\dagger) \vdash M[a^\dagger/\check{x}][b/\hat{x}]}$$

Conversely, here is a proof of the second one-sided rule when the variable is on the right (i.e. v occurs in M rather than Γ):

$$\dfrac{\Psi^0, \Psi_a^0 \mid \Gamma \vdash M[a/v] \qquad \dfrac{\Psi_b \vdash b : A \qquad \dfrac{\dfrac{}{\Psi_M^0, x : A \mid M[\check{x}/v] \vdash M[\hat{x}/v]}}{\Psi_M^0, x : A, y : A \mid M[\check{x}/v], \hom_A(\hat{x}, \check{y}) \vdash M[\hat{y}/v]}}{\Psi_M^0, x : A, \Psi_b^0 \mid M[\check{x}/v], \hom_A(\hat{x}, b^\dagger) \vdash M[b/v]}}{\Psi^0, \Psi_a^0, \Psi_b^0 \mid \Gamma, \hom_A(a, b^\dagger) \vdash M[b/v]}$$

We've left the division of Ψ into Ψ_M and a Ψ_Γ implicit; the unification at the cut gives $\hat{x} \mapsto a$ and matches up all the other variables in Ψ_M with their counterparts in Ψ, so that Ψ_M disappears in the final conclusion.

(Can we do something analogous when v appears in Γ? At least if we assume Γ is a single type, we can. I (Patrick) think this could probably be made to work in general.)

$$\dfrac{\dfrac{\Psi_b \vdash b : A \qquad \dfrac{\dfrac{}{\Psi_N^0, x : A \mid N[\hat{x}/v] \vdash N[\check{x}/v]}}{\Psi_N^0, x : A, y : A \mid N[\hat{y}/v], \hom_A(\hat{x}, \check{y}) \vdash N[\check{x}/v]}}{\Psi_N^0, x : A, \Psi_b^0 \mid N[b/v], \hom_A(\hat{x}, b^\dagger) \vdash N[\check{x}/v]} \qquad \Psi^0, \Psi_a^0 \mid N[a/v] \vdash M}{\Psi^0, \Psi_a^0, \Psi_b^0 \mid N[b/v], \hom_A(a, b^\dagger) \vdash M}$$

Finally, note that with morphism types, we can make the categories into a 2-category: the morphisms are judgments $x : A \vdash f(x) : B$, and the 2-cells from f to g are the judgments $x : A \mid \cdot \vdash \hom_B(f(\check{x}), g(\hat{x}))$. The identity 2-cell is the

$\hom_B$-right rule, while the composite of 2-cells (along a morphism) is obtained by cutting with (**??**). Prewhiskering of a 2-cell is a substitution, while postwhiskering is a cut with (**??**).

3.2. **Tensor.** The rules for the plain (or "non-binding") tensor are very similar to those in ordinary linear logic.

$$\frac{\Psi_M \vdash M \text{ type} \qquad \Psi_N \vdash N \text{ type}}{\Psi_M, \Psi_N \vdash M \otimes N \text{ type}}$$

$$\frac{\Psi, \Psi^\dagger \vdash (\Gamma, M, N) \text{ ctx}, C \text{ type} \qquad \Psi^0 \mid \Gamma, M, N \vdash C}{\Psi^0 \mid \Gamma, M \otimes N \vdash C}$$

$$\frac{\begin{array}{cc}\Psi_M, \Psi_M^\dagger \vdash \Gamma_M \text{ ctx}, M \text{ type} & \Psi_N, \Psi_N^\dagger \vdash \Gamma_N \text{ ctx}, N \text{ type} \\ \Psi_M^0 \mid \Gamma_M \vdash M & \Psi_N^0 \mid \Gamma_N \vdash N\end{array}}{\Psi_M^0, \Psi_N^0 \mid \Gamma_M, \Gamma_N \vdash M \otimes N}$$

3.3. **Coend type.** The coend type is significantly trickier than either of the examples we've considered so far. It can be thought of as a sort of "quadratic Σ-type" that binds a category-variable x in both covariant and contravariant positions at the same time. This intuition is sufficient for the formation and left rules.

$$\frac{\Psi, \hat{x} \overset{+}{:} A, \check{x} \overset{-}{:} A \vdash M \text{ type}}{\Psi \vdash \int^{x:A} M \text{ type}}$$

$$\frac{\Psi, \Psi^\dagger, \hat{x} \overset{+}{:} A, \check{x} \overset{-}{:} A \vdash (\Gamma, M) \text{ ctx}, C \text{ type} \qquad \Psi^0, x : A \mid \Gamma, M \vdash C}{\Psi^0 \mid \Gamma, \int^{x:A} M \vdash C}$$

However, the right rule is trickier to state correctly. In order to get the necessary generality, we need to allow the right rule to "decouple" the two occurrences of x that are bound in the coend. Our first approximation to the rule might therefore be:

$$\frac{\Delta, x : A, y : A \mid \Gamma[\check{x}/\check{z}, \hat{y}/\hat{z}] \vdash M[\check{x}/\check{z}, \hat{x}/\hat{w}, \check{y}/\check{w}, \hat{y}/\hat{z}]}{\Delta, z : A \mid \Gamma \vdash \int^{w:A} M} \tag{5}$$

That is, to construct an element of $\int^{w:A} M$, we choose some other variable z that may appear in M or in the context, and we alter the pairings

$$\begin{array}{ccc} \hat{w} & \leftrightarrow & \check{w} \\ & & \\ \check{z} & \leftrightarrow & \hat{z} \end{array} \quad \text{to become instead} \quad \begin{array}{cc} \hat{x} & \check{y} \\ \updownarrow & \updownarrow \\ \check{x} & \hat{y} \end{array}$$

The simplest case of this is when $\hat{z}$ and $\check{z}$ both occur in the context Γ. In this case, it can be used to derive a functoriality result for coends:

$$\frac{\dfrac{x : A, y : A \mid M(\hat{y}, \check{x}) \vdash N(\check{y}, \hat{x})}{z : A \mid M(\hat{z}, \check{z}) \vdash \int^{w:A} N(\check{w}, \hat{w})}}{\cdot \mid \int^{z:A} M(\hat{z}, \check{z}) \vdash \int^{w:A} N(\check{w}, \hat{w})}$$

(Hmm, should the *bound* occurrences of z on the final line still have their variances reversed because of being in contravariant position in the context? That seems wrong...) The case when $\hat{z}$ or $\check{z}$ appears in M instead is required, for instance, to derive the backwards direction of the co-Yoneda lemma. The forwards direction is easy using the left rules for tensor and coend:

$$\dfrac{\dfrac{\dfrac{}{x : A \mid M(\check{x}) \vdash M(\hat{x})}}{x : A, y : A \mid M(\check{y}) \otimes \hom_A(\hat{y}, \check{x}) \vdash M(\hat{x})}}{x : A \mid \int^{y:A} M(\check{y}) \otimes \hom_A(\hat{y}, \check{x}) \vdash M(\hat{x})}$$

But the backwards direction requires the right rule for both, and in the case of the coend we have $\hat{z}$ appearing in the type being coended rather than in the context:

$$\dfrac{\dfrac{\dfrac{}{x : A \mid M(\check{x}) \vdash M(\hat{x})} \qquad \dfrac{}{y : A \mid \cdot \vdash \hom_A(\check{y}, \hat{y})}}{x : A, y : A \mid M(\check{x}) \vdash M(\hat{x}) \otimes \hom_A(\check{y}, \hat{y})}}{z : A \mid M(\check{z}) \vdash \int^{w:A} M(\hat{w}) \otimes \hom_A(\check{w}, \hat{z})}$$

To prove that these are inverses requires either extensive manipulation of cut-eliminations or a term calculus, so we postpone it until later.

However, (**??**) is insufficient (even before we worry about adding substitutions). For one thing, it doesn't seem to suffice to prove a more general functoriality of coends in which the category also varies along a functor $f : C \to A$; we would like to argue as follows:

$$\dfrac{\dfrac{x : C, y : C \mid M(\hat{y}, \check{x}) \vdash N(f(\check{y}), f(\hat{x}))}{z : C \vdash M(\hat{z}, \check{z}) \vdash \int^{w:A} N(\check{w}, \hat{w})}}{\cdot \vdash \int^{z:C} M(\hat{z}, \check{z}) \vdash \int^{w:A} N(\check{w}, \hat{w})}$$

but the first step is not justified by the right rule we have now, since $\check{w}$ and $\hat{w}$ have been replaced not by $\check{y}$ and $\hat{x}$ but by functions of them. A rule that allows this is

(6)
$$\dfrac{u \overset{+}{:} C \vdash a : A \qquad \Delta, x : C, y : C \mid \Gamma[\check{x}/\check{z}, \hat{y}/\hat{z}] \vdash M[\check{x}/\check{z}, a[\hat{x}/u]/\hat{w}, a[\check{y}/u]/\check{w}, \hat{y}/\hat{z}]}{\Delta, z : C \mid \Gamma \vdash \int^{w:A} M}$$

Of course, in general we want to allow a to depend on more than one variable, in which case C needs to be replaced everywhere by a whole context.

(7)
$$\dfrac{\begin{array}{c}\Psi_1, \Psi_1^\dagger, \Psi_2, \Psi_2^\dagger \cong \Psi_\Gamma, \Psi_M \qquad \Psi_\Gamma^* \vdash \Gamma \;\mathsf{ctx} \qquad \Psi_M, \hat{w} \overset{+}{:} A, \check{w} \overset{-}{:} A \vdash M \;\mathsf{type} \\ \Psi_2 \vdash a : A \qquad \Psi_x \cong \Psi_y \cong \Psi_2 \\ \Psi_1^0, \Psi_x^0, \Psi_y^0 \mid \Gamma[\Psi_x^\dagger/\Psi_2^\dagger, \Psi_y/\Psi_2] \vdash M[\Psi_x^\dagger/\Psi_2^\dagger, a[\Psi_x/\Psi_2]/\hat{w}, a[\Psi_y/\Psi_2]^\dagger/\check{w}, \Psi_y/\Psi_2]\end{array}}{\Psi_1^0, \Psi_2^0 \mid \Gamma \vdash \int^{w:A} M}$$

(We are unable to use the single-judgment version (**??**) of the preconditions, since we need to ensure that $\hat{w}$ and $\check{w}$ appear in M rather than in Γ.) Note that an extra generalization has also happened automatically: unlike $\hat{z}$ and $\check{z}$, each of which can appear in only one of Γ and M, the variables in Ψ_2 and $\Psi_2^\dagger$ could each be distributed with some in Γ and some in M. Furthermore, this rule now includes substitutions already, so we don't need to incorporate them by a further step.

3.4. **Binding tensor types.** The binding tensor $M \otimes_{x:A} N$ can be defined as $\int^{x:A} M \otimes N$. Its derived rules, with their derivations, are:

$$\dfrac{\dfrac{\Psi_M, \hat{x} \overset{+}{:} A \vdash M \text{ type} \qquad \Psi_N, \check{x} \overset{-}{:} A \vdash N \text{ type}}{\Psi_M, \Psi_N, \hat{x} \overset{+}{:} A, \check{x} \overset{-}{:} A \vdash M \otimes N \text{ type}}}{\Psi_M, \Psi_N \vdash \int^{x:A} M \otimes N \text{ type}} \qquad \dfrac{\dfrac{\Psi^0, x : A \mid \Gamma, M, N \vdash C}{\Psi^0, x : A \mid \Gamma, M \otimes N \vdash C}}{\Psi^0 \mid \Gamma, \int^{x:A} M \otimes N \vdash C}$$

$$\dfrac{\dfrac{\Psi^0_M, \Psi^0_2 \mid \Gamma_M \vdash M[a/\hat{w}] \qquad \Psi^0_N, \Psi^0_2 \mid \Gamma_N \vdash N[a^\dagger/\check{w}]}{\Psi^0_M, \Psi^0_N, \Psi^0_x, \Psi^0_y \mid \Gamma_M[\Psi^\dagger_x/\Psi^\dagger_2], \Gamma_N[\Psi_y/\Psi_2] \vdash (M \otimes N)[\Psi^\dagger_x/\Psi^\dagger_2, a[\Psi_x/\Psi_2]/\hat{w}, a[\Psi_y/\Psi_2]^\dagger/\check{w}, \Psi_y/\Psi_2]}}{\Psi^0_M, \Psi^0_N, \Psi^0_2 \mid \Gamma_M, \Gamma_N \vdash \int^{w:A} M \otimes N}$$

(In the right rule, we have renamed Ψ_x and Ψ_y back to Ψ_2 on the top line, for readability.)

More generally, for a context $\Phi = (x_1 : A_1, \ldots, x_n : A_n)$ we can write $M \otimes_\Phi N$ for $\int^{x_1:A_1} \cdots \int^{x_n:A_n} M \otimes N$. Its derived rules should be the following:

$$\dfrac{\Psi_M, \Phi \vdash M \text{ type} \qquad \Psi_N, \Phi^\dagger \vdash N \text{ type}}{\Psi_M, \Psi_N \vdash M \otimes_\Phi N \text{ type}} \qquad \dfrac{\Psi^0, \Phi^0 \mid \Gamma, M, N \vdash C}{\Psi^0 \mid \Gamma, M \otimes_\Phi N \vdash C}$$

$$\dfrac{\Psi_2 \vdash \alpha : \Phi \qquad \Psi^0_M, \Psi^0_2 \mid \Gamma_M \vdash M[\alpha/\Phi] \qquad \Psi^0_N, \Psi^0_2 \mid \Gamma_N \vdash N[\alpha^\dagger/\Phi^\dagger]}{\Psi^0_M, \Psi^0_N, \Psi^0_2 \mid \Gamma_M, \Gamma_N \vdash M \otimes_\Phi N}$$

In the other direction, the non-binding tensor is of course recoverable from the binding one by tensoring over the empty context. Less obviously, the coend should be derivable by tensoring over $(\hat{x} \overset{+}{:} A, \check{x} \overset{-}{:} A)$ with $\hom_A(\check{x}, \hat{x})$, but we should write out the details.

3.5. **End type.** The end type is dual to the coend type. In particular, its formation rule looks the same, and its right rule looks like the left rule for the coend type.

$$\dfrac{\Psi, \hat{x} \overset{+}{:} A, \check{x} \overset{-}{:} A \vdash M \text{ type}}{\Psi \vdash \int_{x:A} M \text{ type}}$$

$$\dfrac{\Psi, \Psi^\dagger, \hat{x} \overset{+}{:} A, \check{x} \overset{-}{:} A \vdash \Gamma \text{ ctx}, M \text{ type} \qquad \Psi^0, x : A \mid \Gamma \vdash M}{\Psi^0 \mid \Gamma \vdash \int_{x:A} M}$$

Its left rule similarly looks like the right rule for the coend; first without substitutions:

$$\dfrac{\Delta, x : A, y : A \mid \Gamma[\check{x}/\check{z}, \hat{y}/\hat{z}], M[\check{x}/\check{z}, \hat{x}/\hat{w}, \check{x}/\check{w}, \hat{y}/\hat{z}] \vdash C[\check{x}/\check{z}, \hat{y}/\hat{z}]}{\Delta, z : A \mid \Gamma, \int_{w:A} M \vdash C}$$

and then with them:

$$\dfrac{\begin{array}{c}\Psi_1, \Psi^\dagger_1, \Psi_2, \Psi^\dagger_2 \cong \Psi_\Gamma, \Psi_M \qquad \Psi^*_\Gamma \vdash \Gamma \text{ ctx} \qquad \Psi_M, \hat{w} \overset{+}{:} A, \check{w} \overset{-}{:} A \vdash M \text{ type}\\ \Psi_2 \vdash a : A \qquad \Psi_x \cong \Psi_y \cong \Psi_2\\ \Psi^0_1, \Psi^0_x, \Psi^0_y \mid \Gamma[\Psi^\dagger_x/\Psi^\dagger_2, \Psi_y/\Psi_2], M[\Psi^\dagger_x/\Psi^\dagger_2, a[\Psi_x/\Psi_2]/\hat{w}, a[\Psi_y/\Psi_2]^\dagger/\check{w}, \Psi_y/\Psi_2] \vdash N[\Psi^\dagger_x/\Psi^\dagger_2, \Psi_y/\Psi_2]\end{array}}{\Psi^0_1, \Psi^0_2 \mid \Gamma, \int_{w:A} M \vdash N}$$

3.6. **Function types.** The non-binding function type acts mostly like a linear function type, except that it reverses the variance of category-variables in its domain.

$$\frac{\Psi_M \vdash M\ \mathsf{type} \qquad \Psi_N \vdash N\ \mathsf{type}}{\Psi_M^*, \Psi_N \vdash M \multimap N\ \mathsf{type}} \qquad \frac{\Psi, \Psi^\dagger \vdash (\Gamma, M)\ \mathsf{ctx}, N\ \mathsf{type} \qquad \Psi^0 \mid \Gamma, M \vdash N}{\Psi^0 \mid \Gamma \vdash M \multimap N}$$

$$\frac{\begin{array}{c}\Psi_1, \Psi_1^\dagger \vdash \Gamma_1\ \mathsf{ctx}, M\ \mathsf{type} \qquad \Psi_2, \Psi_2^\dagger \vdash (\Gamma_2, N)\ \mathsf{ctx}, C\ \mathsf{type} \\ \Psi_1^0 \mid \Gamma_1 \vdash M \qquad \Psi_2^0 \mid \Gamma_2, N \vdash C\end{array}}{\Psi_1^0, \Psi_2^0 \mid \Gamma_1, \Gamma_2, M \multimap N \vdash C}$$

4. Category formers

4.1. **Opposites.**

4.2. **Tensor products.**

4.3. **Exponentials.**

4.4. **Collages.**

4.5. **The type classifier.**

5. Without terms

We can do some category theory without a term calculus. All we need is the fact that the left/right rules for each type former express a universal property. In other words, for a positive type like $\hom_A$, tensor, and coend:

- Applying the left rule, then cutting with the right rule, gives back what we started with. This should be essentially the principal cut corresponding to this type former in the proof of cut admissibility.
- Cutting with a right rule, then applying the left rule, also gives back what we started with. Equivalently, two sequents containing such a type on the left are equal if they become equal upon cutting with the right rule. This should dually follow from the proof of admissibility for identity.

The situation with negative types like hom and end is the same but with the roles of right and left switched.

(We also need the fact that cut is associative, unital, and commutes with things like substitution.)

5.1. **The co-Yoneda lemma.** Recall that in **??** we defined the sequents going in both directions of the co-Yoneda lemma:

$$\frac{\dfrac{\dfrac{}{x : A \mid M(\check{x}) \vdash M(\hat{x})}}{x : A, y : A \mid M(\check{y}) \otimes \hom_A(\hat{y}, \check{x}) \vdash M(\hat{x})}}{x : A \mid \int^{y:A} M(\check{y}) \otimes \hom_A(\hat{y}, \check{x}) \vdash M(\hat{x})}$$

$$\frac{\dfrac{\dfrac{}{x : A \mid M(\check{x}) \vdash M(\hat{x})} \qquad \dfrac{}{y : A \mid \cdot \vdash \hom_A(\check{y}, \hat{y})}}{x : A, y : A \mid M(\check{x}) \vdash M(\hat{x}) \otimes \hom_A(\check{y}, \hat{y})}}{z : A \mid M(\check{z}) \vdash \int^{w:A} M(\hat{w}) \otimes \hom_A(\check{w}, \hat{z})}$$

If we cut these together in the order

$$M(\check{z}) \vdash \int^{w:A} M(\hat{w}) \otimes \hom_A(\check{w}, \hat{z}) \vdash M(\hat{z})$$

we have a right-then-left for the coend, and upon reducing that we have a right-then-left for the tensor, which reduces back to the identity. And if we cut them in the other order

$$\int^{w:A} M(\hat{w}) \otimes \hom_A(\check{w}, \hat{z}) \vdash M(\hat{z}) \vdash \int^{w:A} M(\hat{w}) \otimes \hom_A(\check{w}, \hat{z})$$

then we can test its equality with the identity by cutting with the two right rules on the left that form the map $M(\check{z}) \vdash \int^{w:A} M(\hat{w}) \otimes \hom_A(\check{w}, \hat{z})$; thus this direction follows from the other.

6. Terms

For now, the rules for terms are written without substitutions incorporated. We just take care that for variables mentioned explicitly in the contexts of the conclusion, it would make sense to substitute any reasonable term for them wherever they occur in the term being defined.

6.1. Morphism types.

$$\frac{}{x : A \mid \cdot \vdash \mathrm{id}(x) : \hom_A(\check{x}, \hat{x})}$$

$$\frac{\Delta, z : A \mid \Gamma[\hat{z}/\hat{y}, \check{z}/\check{x}] \vdash m : M[\hat{z}/\hat{y}, \check{z}/\check{x}]}{\Delta, x : A, y : A \mid \Gamma, f : \hom_A(\hat{x}, \check{y}) \vdash J(x.y.M, z, m; x, y, f) : M}\,(\text{2-SIDED})$$

$$J(x.y.M, z, m; x, x, \mathrm{id}(x)) \equiv m \qquad J(x.y.M, z, m[z/x, z/y, \mathrm{id}(z)/f]; x, y, f) \equiv m$$

We will show that the following one-sided elimination rules are actually admissable.

$$\frac{\Delta \mid \Gamma \vdash m : M[a/\hat{x}]}{\Delta, x : A \mid \Gamma, f : \hom_A(a, \check{x}) \vdash \hat{J}(m; x, f) : M}\,(\text{COVARIANT})$$

$$\frac{\Delta \mid \Gamma \vdash m : M[b/\check{x}]}{\Delta, x : A \mid \Gamma, f : \hom_A(\hat{x}, b) \vdash \check{J}(m; x, f) : M}\,(\text{CONTRAVARIANT})$$

$$\hat{J}(m; a, \mathrm{id}(a)) \equiv m \qquad \hat{J}(m[a/x, \mathrm{id}(a)/f]; x, f) \equiv m$$

$$\check{J}(m; b, \mathrm{id}(b)) \equiv m \qquad \check{J}(m[b/x, \mathrm{id}(b)/f]; x, f) \equiv m$$

The derivation of the covariant rule is as follows:

$$\frac{\Delta \mid \Gamma \vdash t : M[a/\hat{x}] \qquad \dfrac{x : A \mid m : M \vdash m : M}{x : A, y : A \mid m : M, f : \hom A(\hat{x}, \check{y}) \vdash J(x.y.M, x, m; x, y, f) : M[\hat{y}/\hat{x}]}}{\Delta, y : A \mid \Gamma, f : \hom_A(a, \check{y}) \vdash J(x.y.M, a, t; a, y, f) : M[\hat{y}/\hat{x}]}$$

Thus we define $\hat{J}(m; x, f) \equiv J(x.y.M, a, m; a, x, f)$. The computation rules are easy to check.

6.2. **The rest.** Tensor types:

$$\frac{\Delta_M \mid \Gamma_M \vdash t_m : M \qquad \Delta_N \mid \Gamma_N \vdash t_n : N}{\Delta_M, \Delta_N \mid \Gamma_M, \Gamma_N \vdash t_m \otimes t_n : M \otimes N}$$

$$\frac{\Delta \mid \Gamma, m : M, n : N \vdash c : C}{\Delta \mid \Gamma, t : M \otimes N \vdash (\mathsf{let}\ m \otimes n \coloneqq t\ \mathsf{in}\ c) : C}$$

$(\mathsf{let}\ m \otimes n \coloneqq t_m \otimes t_n\ \mathsf{in}\ c) \equiv c[t_m/m, t_n/n] \qquad (\mathsf{let}\ m \otimes n \coloneqq t\ \mathsf{in}\ c[m \otimes n/z]) \equiv c[t/z]$

Coend types:

$$\frac{\Delta, x : A, y : A \mid \Gamma[\check{x}/\check{z}, \hat{y}/\hat{z}] \vdash m : M[\check{x}/\check{z}, \hat{x}/\hat{w}, \check{y}/\check{w}, \hat{y}/\hat{z}]}{\Delta, z : A \mid \Gamma \vdash (\mathsf{mix}\ {}^{\check{x} \coloneqq \check{z}}_{\hat{y} \coloneqq \hat{z}}\ \mathsf{in}\ m) : \int^{w:A} M}$$

$$\frac{\Psi_C \vdash C\ \mathsf{type} \qquad \Delta, w : A \mid \Gamma, m : M \vdash c : C}{\Delta \mid \Gamma, t : \int^{w:A} M \vdash (\mathsf{let}\ \langle w, m \rangle \coloneqq t\ \mathsf{in}\ c) : C}$$

Note that both the left rule and the right rule bind variables! The term $(\mathsf{mix}\ {}^{\check{x} \coloneqq \check{z}}_{\hat{y} \coloneqq \hat{z}}\ \mathsf{in}\ m)$ binds $x : A$ and $y : A$ in m, while the term $(\mathsf{let}\ \langle w, m \rangle \coloneqq t\ \mathsf{in}\ c)$ binds $w : A$ and $m : M$ in c. Also there is some sort of "renaming in the context" going on as we pass across the mix, since the Γ above the line is also a substitution instance.

However, I (Mike) can't figure out what the computation rule should say, for reasons that make me wonder whether a term calculus is doomed to failure. Going back to the principal cut for the coend type, there are actually several different possible such cuts:

$$\frac{\dfrac{\Delta', x : A, y : A \mid \Gamma'[\check{x}/\check{z}, \hat{y}/\hat{z}] \vdash M[\check{x}/\check{z}, \hat{x}/\hat{w}, \check{y}/\check{w}, \hat{y}/\hat{z}]}{\Delta', z : A \mid \Gamma' \vdash \int^{w:A} M} \qquad \dfrac{\Delta, w : A \mid \Gamma, M \vdash C}{\Delta \mid \Gamma, \int^{w:A} M \vdash C}}{\Delta, \Delta', (z : A)? \mid \Gamma, \Gamma' \vdash C}$$

depending on how the occurrences of $\hat{z}$ and $\check{z}$ are distributed through Γ' and M.

- If both are in Γ', then $z \notin \Delta$, so $z : A$ remains in the context of the conclusion separate from Δ, Δ'.
- If both are in M, then we must have $z \in \Delta$, and the cut is disallowed because it forms a variable loop.
- If one is in M and one in Γ', then one must also be in Γ and we have $z \in \Delta$, so $z : A$ doesn't need to appear separately in the context of the conclusion.

In each case I can see how to reduce the cut, namely to the obvious

$$\frac{\Delta', x : A, y : A \mid \Gamma'[\check{x}/\check{z}, \hat{y}/\hat{z}] \vdash M[\check{x}/\check{z}, \hat{x}/\hat{w}, \check{y}/\check{w}, \hat{y}/\hat{z}] \qquad \Delta, w : A \mid \Gamma, M \vdash C}{\Delta, \Delta', (z : A)? \mid \Gamma, \Gamma' \vdash C}$$

but this latter cut involves nontrivial category-variable manipulation because it's "contracting a string", and I can't figure out how to represent that by substitution. In particular, the category-variables x, y appearing in one of the premises disappear in the conclusion and I don't see how to represent that disappearance in terms of "substituting" something for them. But possibly I'm just being dense.

Perhaps the solution is that the notion of "substitution" has to involve a more careful "matching up of variables" according to a loop-free string picture. See the example below.

Proof of co-Yoneda lemma, starting with one direction:

$$\dfrac{\dfrac{\dfrac{\overline{x : A \mid n : N(\check{x}) \vdash n : N(\hat{x})}}{x : A, y : A \mid n : N(\check{x}), t : \hom_A(\hat{x}, \check{y}) \vdash \hat{J}(n, y, f) : N(\hat{y})}}{x : A, y : A \mid t : N(\check{x}) \otimes \hom_A(\hat{x}, \check{y}) \vdash (\mathsf{let}\ n \otimes f \coloneqq t\ \mathsf{in}\ \hat{J}(n, y, f)) : N(\hat{y})}}{y : A \mid q : \int^{x:A} N(\check{x}) \otimes \hom_A(\hat{x}, \check{y}) \vdash (\mathsf{let}\ \langle x, t\rangle \coloneqq q\ \mathsf{in}\ (\mathsf{let}\ n \otimes f \coloneqq t\ \mathsf{in}\ \hat{J}(n, y, f))) : N(\hat{y})}$$

I think that we have to use the covariant J here, because when n depends on a variable in the context we can't "bind" it with $x.n$ in the two-sided J. Now the other direction:

$$\dfrac{\dfrac{\overline{x : A \mid n : N(\check{x}) \vdash n : N(\hat{x})} \qquad \overline{z : A \mid \cdot \vdash \mathrm{id}(z) : \hom_A(\check{z}, \hat{z})}}{x : A, z : A \mid n : N(\check{x}) \vdash n \otimes \mathrm{id}(z) : N(\hat{x}) \otimes \hom_A(\check{z}, \hat{z})}}{y : A \mid n : N(\check{y}) \vdash \left(\mathsf{mix}\ \substack{\check{x} \coloneqq \check{y} \\ \hat{z} \coloneqq \hat{y}}\ \mathsf{in}\ n \otimes \mathrm{id}(z)\right) : \int^{w:A} N(\hat{w}) \otimes \hom_A(\check{w}, \hat{y})}$$

It remains to check that they are inverse. Supposing the obvious reduction rule

$$\left(\mathsf{let}\ \langle w, m\rangle \coloneqq \left(\mathsf{mix}\ \substack{\check{x} \coloneqq \check{z} \\ \hat{y} \coloneqq \hat{z}}\ \mathsf{in}\ n\right)\ \mathsf{in}\ c\right) \equiv c[m/n],$$

and given $n' : N(\check{x})$, we have

$$\begin{aligned}
&\left(\mathsf{let}\ \langle x, t\rangle \coloneqq \left(\mathsf{mix}\ \substack{\check{x} \coloneqq \check{y} \\ \hat{z} \coloneqq \hat{y}}\ \mathsf{in}\ n' \otimes \mathrm{id}(z)\right)\ \mathsf{in}\ (\mathsf{let}\ n \otimes f \coloneqq t\ \mathsf{in}\ \hat{J}(n, y, f))\right) \\
&\equiv (\mathsf{let}\ n \otimes f \coloneqq n' \otimes \mathrm{id}(z)\ \mathsf{in}\ \hat{J}(n, y, f)) \\
&\equiv \hat{J}(n', x, \mathrm{id}(x)) \\
&\equiv n'
\end{aligned}$$

This looks good superficially, but let's think about what it actually means. In the second line, the types are something like

$$n : N(\check{w}) \qquad f : \hom_A(\hat{w}, \check{y}) \qquad n' : N(\check{x}) \qquad \mathrm{id}(z) : \hom_A(\check{z}, \hat{z})$$

$$n \otimes f : N(\check{w}) \otimes \hom_A(\hat{w}, \check{y}) \qquad n' \otimes \mathrm{id}(z) : N(\check{x}) \otimes \hom_A(\check{z}, \hat{z})$$

So the "$\mathsf{let}\ n \otimes f \coloneqq n' \otimes \mathrm{id}(z)\ \mathsf{in}$ ", which represents a principal cut of tensor-right against tensor-left, actually involves a nontrivial stringy variable matchup, with only one string even though there are 4 variables. Thus when it reduces, the variables w and z disappear.

Similarly, the third term $\hat{J}(n', x, \mathrm{id}(x))$ also represents a stringy cut, in which the variable y is getting identified with an x and reducing away. Right now it's hard for me to believe in the term calculus with so much variable manipulation being hidden.