Functorial Polymorphism

Dinaturality

Philip Scott

February 5, 2023

References

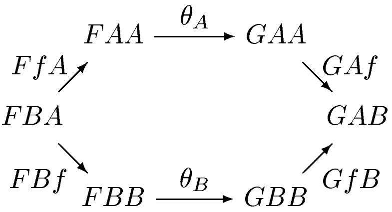
- [BFSS] E.S. Bainbridge, P.J. Freyd, A. Scedrov, P. J. Scott, Functorial Polymorphism, *Theoretical Comp. Science* **70** (1990), 35-64
- [B-S] R. F. Blute, P.J.Scott, Linear Läuchli Semantics, APAL (1996), 101-142.
- E J. Dubuc and R. Street, Dinatural Transformations, in: *Reports of the Midwest Category Seminar IV*, Springer Lecture Notes in Mathematics 137 (1973),126-138
- [GSS] J-Y Girard, A. Scedrov, P. J. Scott, Normal Forms and Cut-free proofs as Natural Transformations., (1992), Logic from Computer Science, Springer-Verlag.
- S. Mac Lane *Categories for the Working Mathematician*, 2nd Ed.,1998, Springer (Chap. IX: Special Limits: Diagonal Naturality, Ends, Co-Ends), 218-230

Diagonal Naturality (Dubuc & Street, 1973)

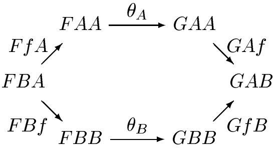
Consider functors of the form $F : (\mathcal{C}^{op})^n \times \mathcal{C}^n \longrightarrow \mathcal{D}$. For exposition, we often take $\mathcal{D} = \mathcal{C}$. Official notation $F(\boldsymbol{A};\boldsymbol{B})$, often written as FAB, if clear.

Definition (Dinaturality)

Consider $F, G : (\mathcal{C}^{op})^n \times \mathcal{C}^n \longrightarrow \mathcal{C}$. A *dinatural transformation* $\theta : F \to G$ is a family of $\mathcal{C}$-arrows $\theta = \{\theta_A : FAA \to GAA \mid A \in \mathcal{C}^n\}$ satisfying: for any n-tuple $f : A \to B \in \mathcal{C}^n$:

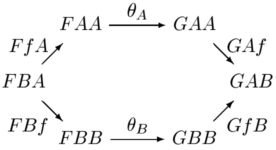


Special Cases of Dinats

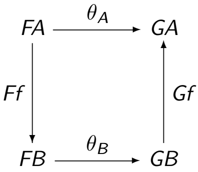


1. Suppose $F, G : \mathcal{C} \to \mathcal{C}$ are covariant functors, construed as functors $\mathcal{C}^{op} \times \mathcal{C} \to \mathcal{C}$ *dummy in the first (contravariant) argument*, e,g. $F(A; B) := F(B)$. Then the NE-oblique arrows $FfA = id_{FA}$, $GfB = id_{GB}$, since there's no contravariance. *So θ reduces to an ordinary natural transformation $F \xrightarrow{\theta} G$.*

Special Cases of Dinats



1. Suppose $F, G : \mathcal{C} \to \mathcal{C}$ are covariant functors, construed as functors $\mathcal{C}^{op} \times \mathcal{C} \to \mathcal{C}$ *dummy in the first (contravariant) argument*, e,g. $F(A; B) := F(B)$. Then the NE-oblique arrows $FfA = id_{FA}$, $GfB = id_{GB}$, since there's no contravariance. *So θ reduces to an ordinary natural transformation $F \xrightarrow{\theta} G$.*
2. If F is covariant and G is contravariant (dummy in the missing arguments) we get following shape (for $f : A \to B$):

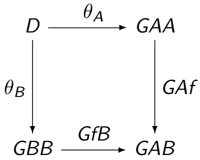


.

Exercise:
What is shape if F contra, G covariant?

Special Cases of Dinats II

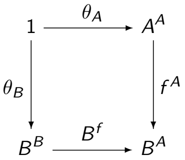
3. (Wedges) A *wedge* or *extranatural transformation* is a dinatural transformation from F to G where one of F or G is a constant functor. For example, suppose $F = K_D$, the constant functor with value D. Then the dinatural hexagon becomes a family $\{\theta_A : D \to GAA \mid A \in \mathcal{C}\}$ satisfying, for $f : A \to B$:



(note: $\mathcal{C}$ can be $\mathcal{C}^n$, so we get n-tuples of objects & arrows).

Examples of Dinats I

1. *Polymorphic Identity*: Consider a ccc $\mathcal{C}$ and let K_1 be the constant functor on 1. Consider a wedge $\theta : K_1 \rightarrow (\ )^{(\ )}$ where $\theta_A : 1 \rightarrow A^A$ is the "name" of the identity (given by the bijection $\dfrac{1 \times A \xrightarrow{\pi_2} A}{1 \xrightarrow{\theta_A} A^A}$). In λ-calculus, $\theta_A = \lambda x : A.\, x$.
   Dinaturality says:

   This amounts to saying: $f \circ id_A = id_B \circ f$, which is true!
   (There is an "external" version (Mac Lane): consider $\mathcal{C}^{op} \times \mathcal{C} \xrightarrow{hom} Set$ and a dinat $\theta : K_1 \rightarrow hom$.)

Important: θ_A is a family of *uniform* algorithms: the identity!

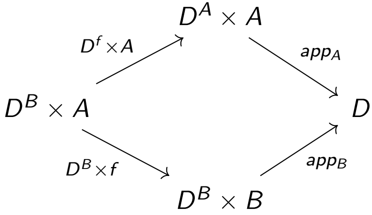
Examples of Dinats II

2. *Polymorphic Church Numerals*: Consider a ccc $\mathcal{C}$ (e.g. Sets) and a uniform family $\boldsymbol{n} : (\ )^{(\ )} \longrightarrow (\ )^{(\ )}$, where $\boldsymbol{n}_A : A^A \to A^A$ is given by (on generic arrows $f \in A^A$), $\boldsymbol{n}_A(f) = f^n = f \circ \cdots \circ f$ (n times). This makes sense in any ccc. The reader can compute that the dinaturality condition becomes: for any $f : B^A, g : A^B$, $f \circ (g \circ f)^n = (f \circ g)^n \circ f$, which is always true (a special instance of associativity).

   Question([BFSS]): In Sets, does this characterize the dinats $(\ )^{(\ )} \longrightarrow (\ )^{(\ )}$? No. There is a proper class of such dinats. Let k be any cardinal number, define $\theta_k(A) : A^A \to A^A$ by: $\theta_k(h) = h$, if card $(fix(h) = k)$; otherwise, $\theta_k(h) = id_A$. A detailed calculation shows this is a dinat. But there is a proper class of cardinals k !

Examples of Dinats III

3. *Simple Application(Mac Lane)*:
   Consider a ccc $\mathcal{C}$ (e.g. Sets) and a fixed object D. Consider the dinatural transformation $app : D^{(\ )} \times (\ ) \to K_D$, where $app_A : D^A \times A \to D$. The hexagon condition reduces to the following (co)wedge: for any $f : A \to B$,

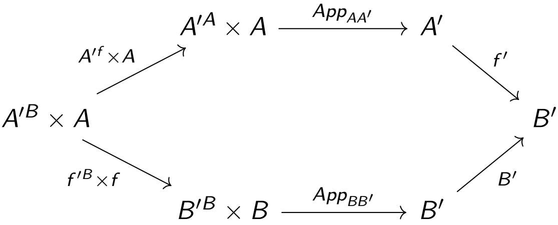


   For $g : D^B$, $a : A$, this says: $(g \circ f)(a) = g(f(a))$, always true in a ccc.

Note: key example of dinaturality: the variable A in app_A occurs both contra-variantly and co-variantly.

Examples of Dinats IV

4. *Generalized Application*. For a ccc $\mathcal{C}$ consider the application $App_{AA'} : A'^{A} \times A \to A'$. Then for $f : A \to B$, $f' : A' \to B'$, we have a dinatural hexagon:

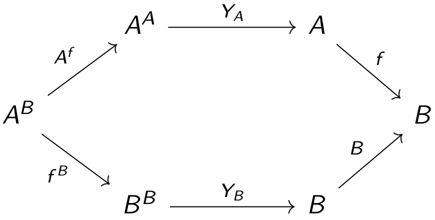


For $g : A'^{B}$, $a : A$, this says $f'((g \circ f)(a)) = (f' \circ g)(f(a))$. This is always true in a ccc.

Note: there is a precise sense in which we have a dinatural transformation $App : F \to G$ between functors $F, G : (\mathcal{C}^2)^{op} \times \mathcal{C}^2 \to \mathcal{C}$, with the dinat family $App_{AA'} : F(AA'; AA') \to G(AA'; AA')$. See the BFSS paper.

Examples of Dinats IV

5. *Fixed Point Combinators.* In many ccc's $\mathcal{C}$ used in theoretical CS (Continuous Lattices, ω-CPO, etc) we have *least fixed point combinators*. These give dinaturals $Y : (\ )^{(\ )} \longrightarrow id$, where $Y_A(f) =$ the least fixed point of f. In general, $Y = \{Y_A : A^A \to A\}$ satisfies, for any $f : A \to B$,



This says if $g : A^B$, $f(Y_A(g \circ f)) = Y_B(f \circ g)$. In particular, setting $A = B$ and $g = id_A$, this confirms $f(Y_A(f)) = Y_A(f)$, so Y is a *fixed point combinator at each object A*.

Examples of Dinats V

6. *Traced Symmetric Monoidal Categories*. We'll just recall this briefly. Joyal-Street-Verity (1996) introduced an abstract trace on an smc $\mathcal{C}$, where
$\mathtt{Tr}^U_{X,Y} : \mathcal{C}(X \otimes U, Y \otimes U) \rightarrow \mathcal{C}(X, Y)$

satisfies:

(1) **Natural** in X, $\mathtt{Tr}^U_{X,Y}(f)g = \mathtt{Tr}^U_{X',Y}(f(g \otimes 1_U))$, where $f : X \otimes U \rightarrow Y \otimes U$, $g : X' \rightarrow X$,

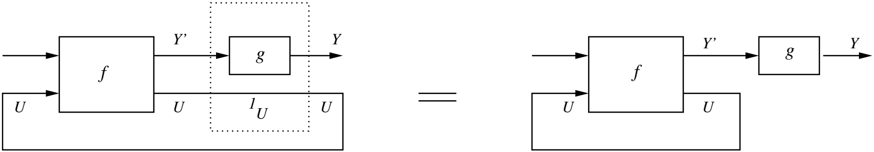
(2) **Natural** in Y, $g\mathtt{Tr}^U_{X,Y}(f) = \mathtt{Tr}^U_{X,Y'}((g \otimes 1_U)f)$, where $f : X \otimes U \rightarrow Y \otimes U$, $g : Y \rightarrow Y'$,

(3) **Dinatural** in U, $\mathtt{Tr}^U_{X,Y}((1_Y \otimes g)f) = \mathtt{Tr}^{U'}_{X,Y}(f(1_X \otimes g))$, where $f : X \otimes U \rightarrow Y \otimes U'$, $g : U' \rightarrow U$,

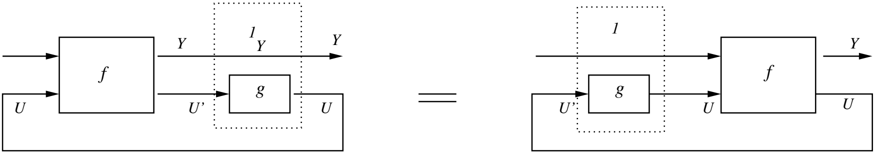
and also (4) Vanishing I, II, (5) Superposing, (6)Yanking.

Let's illustrate the graphical calculus of (2)-(3):

Examples of Dinats VI: naturality of trace graphically



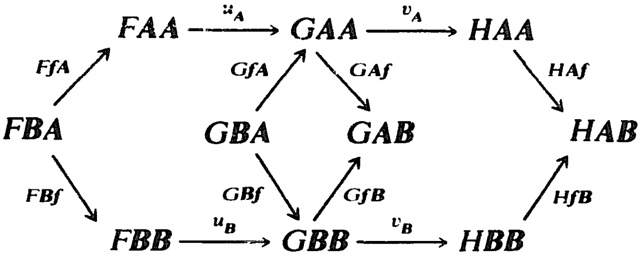
Naturality in Y



Dinaturality in U

So what's wrong with Dinats?

In general, they don't compose!



But if the middle diamond is a weak pullback (or weak pushout) they do compose (exercise!) More possibilities:

- Dinats compose with nats.
- Dinats wrt restricted morphisms f can compose (e.g. f iso)
- Question: do extranaturals (wedges or co-wedges) compose?

Some constructions on multivariant functors

Products: Given functors $F = F(\boldsymbol{A}; \boldsymbol{B}) : (\mathcal{C}^{op})^n \times \mathcal{C}^n \to \mathcal{C}$ and $G = G(\boldsymbol{A}; \boldsymbol{B}) : (\mathcal{C}^{op})^n \times \mathcal{C}^n \to \mathcal{C}$, define their product *pointwise*:

$$(F \times G)(\boldsymbol{A}; \boldsymbol{B}) = F(\boldsymbol{A}; \boldsymbol{B}) \times G(\boldsymbol{A}; \boldsymbol{B}) \ .$$

Twisted Exponential: Given Functors $F = F(\boldsymbol{A}; \boldsymbol{B})$, and $G = G(\boldsymbol{A}; \boldsymbol{B})$, define $F \Rightarrow G : (\mathcal{C}^{op})^n \times \mathcal{C}^n \to \mathcal{C}$ as follows:

$$(F \Rightarrow G)(\boldsymbol{A}; \boldsymbol{B}) = F(\boldsymbol{B}; \boldsymbol{A}) \Rightarrow G(\boldsymbol{A}; \boldsymbol{B})$$

In certain CCC's, the above constructions may lead to a compositional model: e.g. in BFSS, we studied the original Realizability Topos.

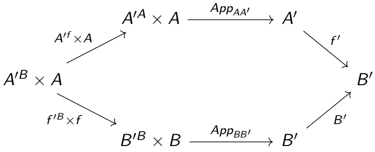
In general, as Peter Freyd said, dinats form a cartesian closed non-category!

Generalized Application, again

Covariant Projection: Define $P_i^n : (\mathcal{C}^{op})^n \times \mathcal{C}^n \to \mathcal{C}$ by:

$$P_i^n(\boldsymbol{A}; \boldsymbol{B}) = B_i, \text{where } \boldsymbol{B} = B_1 \ldots B_{\boldsymbol{n}}.$$

Recall: generalized application



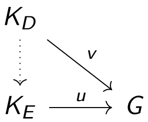
Tricky calculation:

$$\boldsymbol{app}_{P_1^2, P_2^2} : (P_1^2 \Rightarrow P_2^2) \times P_1^2 \longrightarrow P_2^2$$

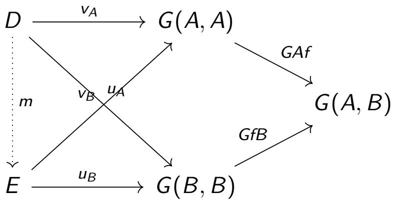
determines a dinatural transformation agreeing with generalized application, e.g. $(\boldsymbol{app}_{P_1^2, P_2^2})_{AA'} = App_{AA'} : A'^A \times A \longrightarrow A'$.

Ends and Co-ends (Yoneda)

Given multivariant functor G, we seek a *universal wedge* into G. This is an object E with wedge $K_E \xrightarrow{u} G$ s.t. for any other object D and wedge $K_D \xrightarrow{v} G$, there is a unique map $D \xrightarrow{m} E$ such that



that is



We write $E = \int_A GAA$. It is called the *end* of G.

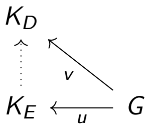
In many concrete categories, when defined, it's a limit:

$\int_A GAA = \{g \in \prod_A GAA \mid GAf(g_A) = GfB(g_B) \text{ for all } f : A \to B\}$

In functorial polymorphism, it is a kind of "parametric" universal quantifier: $\forall A. GAA$

Co-Ends (Yoneda)

The dual of end is co-end, denoted $\int^A GAA$. It is a solution to the co-universal problem:



Again, in appropriate concrete categories, it's a colimit:

$\int^A GAA = \uplus\{GAA \mid A \in \mathcal{C}^n\}/\sim$, where

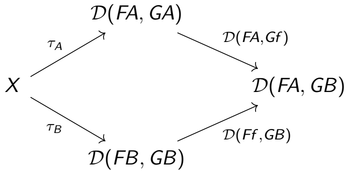
$\sim$ is the smallest equivalence relation on $\uplus\{GAA \mid A \in \mathcal{C}^n\}$ satisfying: given (x, A) and (y, B), with $x \in GAA$, $y \in GBB$

$$
\begin{array}{lll}
(x, A) \sim (y, B) \quad \text{iff} & \text{there exists} & z \in GBA \text{ and } A \xrightarrow{f} B \\
 & \text{such that} & x = (GfA)(z) \;\&\; y = (GBf)(z)
\end{array}
$$

Naturality Formulas

Let $F, G : \mathcal{C} \to \mathcal{D}$. Consider the hom functor $\mathcal{D}(F-, G-) : (\mathcal{C}^{op}) \times \mathcal{C} \to Set$. For any set X, the wedge $\tau : K_X \to \mathcal{D}(F-, G-)$, satisfies: for any $f : A \to B$

(†)



$\forall x \in X, Gf\tau_{A,x} = \tau_{B,x}Ff.$

- Claim: this says $\forall x \in X$, $\tau_{-,x} \in \boldsymbol{Nat}(F, G)$.

- Consider the family of maps $\omega = \{\omega_A : Nat(F, G) \to \mathcal{D}(FA, GA) \mid A \in \mathcal{C}\}$

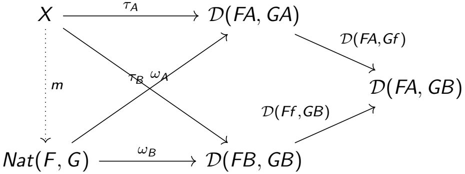
given by $\omega_A(\theta) = \theta_A : FA \to GA$, for a natural transformation $\theta \in \boldsymbol{Nat}(F, G)$. Claim: $\omega : K_{\boldsymbol{Nat}(F,G)} \to \mathcal{D}(F-, G-)$ determines a wedge (as in (†)) .

Naturality Formulas II

Summarizing the previous discussion:

Proposition (Mac Lane)

Let $\mathbf{Nat}(F,G) \xrightarrow{\omega_A} \mathcal{D}(FA,GA)$ *assign* $\theta \mapsto \theta_A$ *on nat* θ. *The function* $m := x \mapsto \tau_{-,x} : X \to \mathbf{Nat}(F,G)$ *is the unique map making* ω *a universal wedge:*



Hence $\mathbf{Nat}(F,G) \cong \int_A \mathcal{D}(FA,GA)$

A similar result holds for dinats: if $F,G : \mathcal{C}^{op} \times \mathcal{C} \to \mathcal{D}$, we get

$\mathbf{Dinat}(F,G) \cong \int_A \mathcal{D}(F(A;A), G(A;A))$.

Functorial Polymorphism

In the papers [GSS] and [BFSS], we consider some compositional models for dinaturals arising from logic.
(i) In [GSS], we looked at cut-free proofs as determining a dinatural modelling in a cartesian-closed setting.
(ii) In [BFSS] we found a "parametric" model of Girard's System $\mathcal{F}$, in which dinats compose, based on certain dinats in the original Realizability topos and $\forall\alpha$ is interpreted as an end.
(iii) In the Blute-Scott (et.al.) papers, we were interested in proving certain concrete categorical models $\mathit{Dinat}(\mathcal{C})$ for multiplicative linear logic admitted a full interpretation, i.e. the unique free functor $F \to \mathit{Dinat}(\mathcal{C})$ is full, where F is the free (syntactically generated) category of a fragment of LL.

Functorial Polymorphism: the paper [GSS]

Consider (free) cartesian closed categories (equivalently, simply typed λ-calculus or a simple typed functional programming language). Question: Are there "unexpected" equations between terms (beyond simple β, η equality)? Answer: Yes, dinatural equations!

Example

Consider a simple closed term $r : \alpha \times \alpha \Rightarrow \alpha \times \alpha$. What can we say about r? For each type A, consider $r_A : A \times A \Rightarrow A \times A$. Suppose $f : A \Rightarrow B$ is a closed term. Then we show:

$$(f \times f) \circ r_A = r_B \circ (f \times f)$$

This says $r : F \rightarrow F$ is a natural transformation, where $F(-) = (-) \times (-)$.

Functorial Polymorphism: the paper [GSS] II

More generally, the previous example r seems to depend on $f : A \Rightarrow B$. So consider the more general term:

$$m_{\alpha,\beta} : (\alpha \Rightarrow \beta) \Rightarrow (\alpha \times \alpha \Rightarrow \beta \times \beta)$$

What could it be? First guess: m maps $f \mapsto f \times f$ for $f : A \Rightarrow B$.

In fact we show: for any types A, B, and any closed term $f : A \Rightarrow B$,

$$m_{AB}(f) = (f \times f) \circ m_{AA}(1_A) = m_{BB}(1_B) \circ (f \times f)$$

So any instantiation of $m_{\alpha,\beta}$ really is $f \mapsto f \times f$ *up to an endomorphism of the domain or codomain.*

Functorial Polymorphism: the paper [GSS] III

Let $\mathcal{C}$ be a ccc. For each type expression $\sigma(\alpha_1 \ldots, \alpha_n)$ with type variables α_i, define its interpretation $||\sigma|| : (\mathcal{C}^{op})^n \times \mathcal{C}^n \to \mathcal{C}$ as:

1. If $\sigma = \alpha_i$, $||\sigma||(\mathbf{A};\mathbf{B}) = B_i$, the projection onto the ith component of $\mathbf{B}$.
2. If σ = C, a ground constant interpreted as an object of $\mathcal{C}$, then $||\sigma||(\mathbf{A};\mathbf{B}) = K_C$, the constant functor with value C.
3. (Product) If $\sigma = \tau_1 \times \tau_2$, then $||\sigma||(\mathbf{A};\mathbf{B}) = ||\tau_1||(\mathbf{A};\mathbf{B}) \times ||\tau_2||(\mathbf{A};\mathbf{B})$.
4. (Twisted Exponential) If $\sigma = \tau_1 \Rightarrow \tau_2$ then $||\sigma||(\mathbf{A};\mathbf{B}) = ||\tau_1||(\mathbf{B};\mathbf{A}) \Rightarrow ||\tau_2||(\mathbf{A};\mathbf{B})$.

Now consider a term $t : \tau$ with typing $x_1 : \sigma_1, \ldots, x_k : \sigma_k \rhd t : \tau$. We interpret this as a dinat family

$$||t|| = \{||t||\mathbf{A} : (||\sigma_1|| \times \ldots \times ||\sigma_k||)\mathbf{AA} \to ||\tau||\mathbf{AA} \mid \mathbf{A} \in \mathcal{C}^n\}$$

where $\mathbf{A} = \mathrm{A}_1, \ldots, \mathrm{A}_n$ and $\mathrm{A}_i = ||\alpha_i||$, by induction:

Functorial Polymorphism: the paper [GSS] IV

The interpretations of terms is as follows:

1. If $t : \sigma_i$ is the variable x_i, then $\|t\| : \|\sigma_1\| \times \cdots \times \|\sigma_k\| \longrightarrow \|\sigma_i\|$ is the ith projection.

2. If $t : \sigma \Rightarrow \tau$ is $\lambda x : \sigma.r$, where $\|r\| : \|\sigma_1\| \times \cdots \times \|\sigma_k\| \times \|\sigma\| \longrightarrow \|\tau\|$, then $\|t\| : \|\sigma_1\| \times \cdots \times \|\sigma_k\| \longrightarrow \|\sigma \Rightarrow \tau\|$ is defined by: $\|t\|\mathbf{A} = (\|r\|\mathbf{A})^*$, the curryfication (or exponential transpose) of the arrow $\|r\|\mathbf{A} : \|\sigma_1\|\mathbf{AA} \times \cdots \times \|\sigma_k\|\mathbf{AA} \times \|\sigma\|\mathbf{AA} \longrightarrow \|\tau\|\ \mathbf{AA}$ in the ccc $\mathcal{C}$ (cf. [19], p.61).

3. If $t : \tau$ is ua, where $u : \sigma \Rightarrow \tau$ and $a : \sigma$, then $\|t\|$ is $ev\circ <\|u\|, \|a\|>$, where ev denotes the appropriate evaluation map and $\circ$ denotes composition in $\mathcal{C}$.

4. If $t : \tau_1 \times \tau_2$ is $t =< t_1, t_2 >$ and $\|t_i\| : \|\sigma_1\| \times \cdots \times \|\sigma_k\| \longrightarrow \|\tau_i\|$ for $i = 1, 2$, then $\|t\|\mathbf{A} =< \|t_1\|\mathbf{A}, \|t_2\|\mathbf{A} >$.

5. If $t : \tau_i$ is $\Pi_i(t')$ ($i = 1, 2$) where $\|t'\| : \|\sigma_1\| \times \cdots \times \|\sigma_k\| \longrightarrow \|\ \tau_1 \times \tau_2\ \|$ then $\|\Pi_i(t')\|$ is $\Pi_i \circ \|t'\|$, where Π_i is the ith projection.

Functorial Polymorphism: the paper [GSS] V

Theorem (GSS, Thm. 2.2)

Let $\mathcal{L}$ be simply typed lambda calculus with type variables, let $\mathcal{C}$ be any ccc, and let $\alpha_1, \ldots, \alpha_n$ be a list of type variables. Then any type $\tau(\alpha_1, \ldots, \alpha_n)$ with the indicated type variables induces a functor $||\tau|| : (\mathcal{C}^{op})^n \times \mathcal{C}^n \to \mathcal{C}$. Furthermore, any legal typing judgement $x_1 : \sigma_1, \ldots, x_k : \sigma_k \triangleright t : \tau$ induces a family of lambda terms $||t|| = \{||t||\mathbf{A} \mid \mathbf{A} \in \mathcal{C}^n\}$ which is actually a dinatural transformation $||t|| : ||\sigma_1|| \times \cdots \times ||\sigma_k|| \to ||\tau||$. Beta-eta equal lambda terms give the same dinatural family.

Functors and dinatural transformations induced by lambda terms (i.e. arrows in a free ccc) will be called *definable*. The point is that under type and term-substitution, definable dinats actually do compose. This leads to the following theorem:

Functorial Polymorphism: the paper [GSS] VI

Theorem (GSS, Thm. 2.3)

Let $\mathcal{C}$ be any ccc. Then for each n the definable multivariate functors $(\mathcal{C}^{op})^n \times \mathcal{C}^n \to \mathcal{C}$ and definable dinatural transformations between them form a ccc. The cartesian closed structure on definable functors is given by products and twisted exponentials. In fact, they form an indexed category (hyperdoctrine, without the quantifiers) of ccc's, with base = natural numbers, and the fibre over n the definable multivariate n-ary functors, and maps the definable dinats between them.

Functorial Polymorphism: the paper [GSS] VIII.

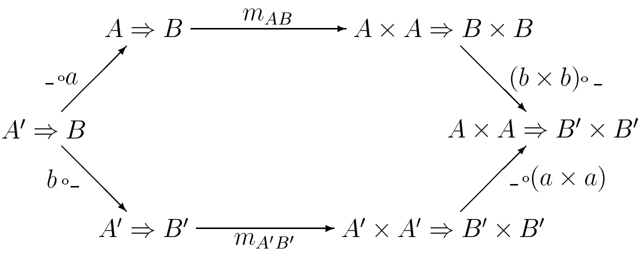
Recall the typing judgement:

$$x : \alpha \Rightarrow \beta \ \triangleright\ m(x) : \alpha \times \alpha \Rightarrow \beta \times \beta$$

There are 2 variables, so this is a dinat $(\mathcal{C}^{op})^2 \times \mathcal{C}^2 \xrightarrow{m} \mathcal{C}$.
Variables are covariant projections, so this becomes:

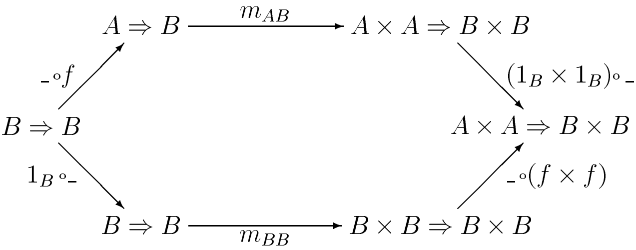
$$(P_1 \Rightarrow P_2) \xrightarrow{m} (P_1 \times P_1) \Rightarrow (P_2 \times P_2)$$

Calculating the LHS at $(A'B'; AB)$ and RHS at $(AB; A'B')$ for arrows $a : A \to A'$ and $b : B \to B'$, we have



Functorial Polymorphism: the paper [GSS] VIII.

Letting $A' = B' = B$, $b = 1_B$, $a = f$ leads to



Chasing $1_B : B \Rightarrow B$ on LHS around the diagram, we obtain:

$$m_{AB}(f) = m_{BB}(1_B) \circ (f \times f)$$

and similarly for the dual situation of an endomorphism of the domain.

Functorial Polymorphism: the paper [GSS] VII

Idea of the proof of the main theorem. Subtle problems arise!

1. Definable dinats between definable functors are uniformly given by (instantiations of) a single lambda term. But lambda terms are closed under usual substitution.
2. The types themselves $\sigma(\alpha_1, \ldots, \alpha_n)$ also permit substitutions of arbitrary formulas B_i for the α_i.
3. Dinats, in general, don't compose. But definables do. Why?
4. Idea: we will represent the definable dinats in a Gentzen Sequent Calculus (for $\Rightarrow, \times$) which admits Cut-Elimination. In a cut-free system, we can essentially ignore using the cut-rule for composing dinats. But this is *very* subtle: substitution doesn't disappear, but is hidden in the basic rules. We show this by a non-trivial interpretation of sequent calculus into ND, then looking at the inherited Curry-Howard lambda-term assignment of cut-free proofs from within ND.

[GSS]: Intuitionist Sequent Calculus for $\{\Rightarrow, \wedge\}$

Axiom: $A \vdash A$

Cut: $$\frac{\Gamma \vdash A \quad \Delta, A \vdash B}{\Gamma, \Delta \vdash B}$$

Structural:

Exchange $$\frac{\Gamma, A, B, \Delta \vdash C}{\Gamma, B, A, \Delta \vdash C}$$

Contraction $$\frac{\Gamma, A, A \vdash C}{\Gamma, A \vdash C}$$

Weakening $$\frac{\Gamma \vdash C}{\Gamma, A \vdash C}$$

Logical:

$\wedge \mathrm{L}_i$ $$\frac{\Gamma, A_i \vdash C}{\Gamma, A_1 \wedge A_2 \vdash C}, \ i = 1, 2.$$ $\wedge$R $$\frac{\Gamma \vdash A \quad \Delta \vdash B}{\Gamma, \Delta \vdash A \wedge B}$$

$\Rightarrow$L $$\frac{\Gamma \vdash A \quad \Delta, B \vdash C}{\Gamma \ , \Delta, A \Rightarrow B \vdash C}$$ $\Rightarrow$ R $$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B}$$

[GSS]: Inherited λ-terms (within ND) for sequent proofs

Axiom : $$x : A \ \triangleright\ x : A$$

Cut : $$\frac{\vec{x} : \Gamma \ \triangleright\ t[\vec{x}] : A \qquad \vec{y} : \Delta, z : A \ \triangleright\ f[\vec{y}, z] : B}{\vec{x} : \Gamma, \vec{y} : \Delta \ \triangleright\ f[\vec{y}, t[\vec{x}]/z] : B}$$

Structural :

Exchange $$\frac{\vec{x} : \Gamma, y : A, z : B, \vec{w} : \Delta \ \triangleright\ t[\vec{x}, y, z, \vec{w}] : C}{\vec{x} : \Gamma, z : B, y : A, \vec{w} : \Delta \ \triangleright\ t^e[\vec{x}, z, y, \vec{w}] : C}$$
where $t^e[\vec{x}, z, y, \vec{w}] = t[\vec{x}, y, z, \vec{w}]$

Contraction $$\frac{\vec{x} : \Gamma, y : A, z : A \ \triangleright\ t[\vec{x}, y, z] : C}{\vec{x} : \Gamma, y : A \ \triangleright\ t^c[\vec{x}, y] : C}$$
where $t^c[\vec{x}, y] = t[\vec{x}, y, y/z]$.

Weakening $$\frac{\vec{x} : \Gamma \ \triangleright\ t[\vec{x}] : C}{\vec{x} : \Gamma, y : A \ \triangleright\ t^w[\vec{x}, y] : C}$$
where $t^w[\vec{x}, y] = t[\vec{x}]$.

[GSS]: Inherited λ-terms (within ND) for sequent proofs II

Logical :

$\wedge L_i,\ {\scriptstyle i=1,2}$ $$\frac{\vec{x}:\Gamma, y_i:A_i \ \rhd\ t[\vec{x},y_i]:C}{\vec{x}:\Gamma, z:A_1\wedge A_2 \ \rhd\ t^{\wedge}[\vec{x},z]:C}$$
where $t^{\wedge}[\vec{x},z] = t[\vec{x},\pi_i(z)/y_i]$

$\wedge R$ $$\frac{\vec{x}:\Gamma \ \rhd\ s[\vec{x}]:A \qquad \vec{y}:\Delta \ \rhd\ t[\vec{y}]:B}{\vec{x}:\Gamma,\vec{y}:\Delta \ \rhd\ <s[\vec{x}],t[\vec{y}]>: A\wedge B}$$

$\Rightarrow L$ $$\frac{\vec{x}:\Gamma \ \rhd\ f[\vec{x}]:A \qquad \vec{y}:\Delta, z:B \ \rhd\ g[\vec{y},z]:C}{\vec{x}:\Gamma\ ,\vec{y}:\Delta, u:A\Rightarrow B \ \rhd\ g[\vec{y},uf[\vec{x}]/z]:C}$$
where , if $u : A \Rightarrow B$ and $a : A$ then $ua : B$ denotes application of u to argument a.

$\Rightarrow R$ $$\frac{\vec{x}:\Gamma, y:A \ \rhd\ t[\vec{x},y]:B}{\vec{x}:\Gamma \ \rhd\ \lambda y:A.t[\vec{x},y]:A\Rightarrow B}$$

[GSS]: Conclusion of [GSS]

The "real" meaning of Cut-elimination ([GSS], p.15):

> *All general instances of term substitution arising from the cut rule are already derivable (up to equality of terms) from the special instances of substitution used in the other rules.*

We obtain, as a Corollary of the main proof in [GSS], the following:

• Cut-free proofs are represented by closed terms in normal form, thus arrows in $\mathcal{C}$.

• Any arrow $m : A \rightarrow B$ (qua closed normal form of type $A \Rightarrow B$) induces a dinatural transformation between definable functors $||A|| \rightarrow ||B||$. These normal forms compose by substitution.

• m provably satisfies the dinaturality equations.

Aside: An example of an end calculation

Recall last time we mentioned Mac Lane's Proposition: $\boldsymbol{Nat}(F, G) \cong \int_A \mathcal{D}(FA, GA)$. We illustrate this:

Consider a monoid $\mathcal{M}$. A (left) *$\mathcal{M}$-set* is a set X with a left action $\mathcal{M} \times X \xrightarrow{\lambda} X$ satisfying well-known axioms; equivalently, it's a set X and a monoid hom $\mathcal{M} \to (End(X), \circ)$. A *morphism of $\mathcal{M}$-sets* $(X, \lambda) \to (Y, \mu)$ is an equivariant map.

The category of $\mathcal{M}$-sets and homs can be thought of as the functor category $Set^{\mathcal{M}}$, where $\mathcal{M}$ is a category with one object.

Example: By cartesian-closedness, the monoid multiplication $\mathcal{M} \times \mathcal{M} \to \mathcal{M}$ induces an action $\mathcal{M} \xrightarrow{\lambda^L} (Endo(\mathcal{M}), \circ)$ called *left representation*, defined by $m \mapsto (n \mapsto mn)$. We write $\mathcal{M}^L = (\mathcal{M}, \lambda^L)$ for this $\mathcal{M}$-set.

Aside: An example of an end calculation II

- Consider the forgetful functor $U : Set^{\mathcal{M}} \to Set$, mapping $(X, \lambda) \mapsto X$. **Claim:** U is representable: $U \cong Hom_{Set^{\mathcal{M}}}(\mathcal{M}^L, -)$.

Exercise 1: Prove U is representable. Hence,

$$\begin{aligned} Nat(U,U) &\cong Nat(Hom_{Set^{\mathcal{M}}}(\mathcal{M}^L,-), Hom_{Set^{\mathcal{M}}}(\mathcal{M}^L,-)) \\ &\cong Hom_{Set^{\mathcal{M}}}(\mathcal{M}^L, \mathcal{M}^L) \quad \text{by Yoneda's Lemma} \\ &\cong \mathcal{M} \quad \text{by } \textbf{Exercise 2} \text{ below; (actually a } \textit{monoid} \text{ iso).} \end{aligned}$$

Exercise 2: Prove $Hom_{Set^{\mathcal{M}}}(\mathcal{M}^L, (S, \lambda)) \cong S$ by mapping $f \mapsto f(e)$, for f equivariant.

Hence, by Mac Lane, $\int_{C \in Set^{\mathcal{M}}} (U(C), U(C)) \cong Nat(U,U) \cong \mathcal{M}$.

- For any category $\mathcal{C}$, $Nat(Id_{\mathcal{C}}, Id_{\mathcal{C}}) \cong \int_{A \in \mathcal{C}} Hom_{\mathcal{C}}(A, A)$. Related to Picard Group of a category, etc.

- See also Tannakian Duality on nLab.

Girard's System $\mathcal{F}$: Polymorphic λ-calculus

Consider all the formulas of second-order intuitionistic propositional logic built out of propositional variables using $\forall, \Rightarrow$.

$$A, B ::= \mathit{Prop.Vbls} \mid A \Rightarrow B \mid \forall\alpha.A$$

Examples: $(\alpha \Rightarrow \alpha)$, $\forall\alpha.(\alpha \Rightarrow \alpha)$, $\forall\alpha.(\alpha \Rightarrow \beta)$, etc. We will call these formulas **Types**, or **Polymorphic Types** to be precise. Notice, the first type has free α, the second has no free type variables, the last has β free and α bound.

We write $\Gamma \vdash B$ to say hypotheses Γ entail B, where $\Gamma = \{A_1 \ldots, A_n\}$ is a finite set of hypotheses.

Girard's System $\mathcal{F}$: Polymorphic λ-calculus II

1. $\Gamma, A \vdash A$

2. $$\frac{\Gamma \vdash A \quad \Delta, A \vdash B}{\Gamma, \Delta \vdash B}$$

3. $$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B}$$

4. $$\frac{\Gamma \vdash A(\alpha)}{\Gamma \vdash \forall \alpha A(\alpha)} \ \forall\text{-I}$$
   $\alpha \notin FTV(\Gamma)$

5. $$\frac{\Gamma \vdash \forall \alpha A(\alpha)}{\Gamma \vdash A[\alpha := B]} \ \forall\text{-E}$$
   for any B.

Remarks: Notice in #5, the formula B is of arbitrary complexity–it can even contain $\forall \alpha A(\alpha)$ as a subformula! So Rule #5 definitely increases complexity.

We now discuss Girard's second-order (polymorphic) λ-calculus and the assignment of proof terms (à la Curry-Howard) to second-order propositional calculus.

Girard's System $\mathcal{F}$ and Polymorphic λ-calculus IV

$$(var) \quad \frac{}{\Gamma, x{:}A \vdash x : A}$$

$$(app) \quad \frac{\Gamma \vdash M : A \to B \qquad \Gamma \vdash N : A}{\Gamma \vdash MN : B}$$

$$(abs) \quad \frac{\Gamma, x{:}A \vdash M : B}{\Gamma \vdash \lambda x^A.M : A \to B}$$

$$(typeapp) \quad \frac{\Gamma \vdash M : \forall\alpha.A}{\Gamma \vdash MB : A[B/\alpha]}$$

$$(typeabs) \quad \frac{\Gamma \vdash M : A \qquad \alpha \notin FTV(\Gamma)}{\Gamma \vdash \Lambda\alpha.M : \forall\alpha.A}$$

Equations: β and η rules for both lambdas, that is:

(i) $(\lambda x : A.\varphi)a \;=_\beta\; \varphi[x := a]$ and $\lambda x : A(fx) \;=_\eta\; f$, where $x \notin FV(f)$.

(ii) $(\Lambda\alpha.M)[B] \;=_{\beta^2}\; M[\alpha := B]$ and $\Lambda\alpha.(M[\alpha]) \;=_{\eta^2}\; M$, where $\alpha \notin FTV(M)$.

The power of Girard's System $\mathcal{F}$, II

Consider $(\lambda x{:}\alpha.x) : \alpha \Rightarrow \alpha$, i.e. id_α. In fact, the type-indexed family of identity functions $\{id_\alpha\}_\alpha$ of type $\forall\alpha.(\alpha \Rightarrow \alpha)$ suggests a *dinatural transformation*. This led to Functorial Polymorphism...

Continuing with the power of 2nd-order logic, here's a chart (from Selinger's notes) for defining most of the usual logical connectives:

$$\begin{aligned}
A \wedge B &\iff \forall\alpha.(A \to B \to \alpha) \to \alpha, &(1)\\
A \vee B &\iff \forall\alpha.(A \to \alpha) \to (B \to \alpha) \to \alpha, &(2)\\
\neg A &\iff \forall\alpha.A \to \alpha, &(3)\\
\top &\iff \forall\alpha.\alpha \to \alpha, &(4)\\
\bot &\iff \forall\alpha.\alpha, &(5)\\
\exists\beta.A &\iff \forall\alpha.(\forall\beta.(A \to \alpha)) \to \alpha. &(6)
\end{aligned}$$

The power of Girard's System $\mathcal{F}$, II

Consider $(\lambda x : \alpha . x) : \alpha \Rightarrow \alpha$, i.e. id_α. In fact, the type-indexed family of identity functions $\{id_\alpha\}_\alpha$ of type $\forall \alpha . (\alpha \Rightarrow \alpha)$ suggests a *dinatural transformation*. This led to Functorial Polymorphism...

Continuing with the power of 2nd-order logic, here's a chart (from Selinger's notes) for defining most of the usual logical connectives:

$$
\begin{aligned}
A \wedge B &\iff \forall \alpha . (A \to B \to \alpha) \to \alpha, && (1)\\
A \vee B &\iff \forall \alpha . (A \to \alpha) \to (B \to \alpha) \to \alpha, && (2)\\
\neg A &\iff \forall \alpha . A \to \alpha, && (3)\\
\top &\iff \forall \alpha . \alpha \to \alpha, && (4)\\
\bot &\iff \forall \alpha . \alpha, && (5)\\
\exists \beta . A &\iff \forall \alpha . (\forall \beta . (A \to \alpha)) \to \alpha. && (6)
\end{aligned}
$$

There are no Set-theoretical models of Girard's System $\mathcal{F}$. **Goal**: find a model based on definable functors and some class of dinatural transformations over a base category $\mathcal{C}$.

PER models [BFSS]

A *partial equivalence relation (per)* E on a set A is a symmetric transitive relation on A. The *domain of a per* $E = dom_E = \{a \in A \mid aEa\}$.

So, E is an e.r. on its domain, hence partitions the domain into disjoint classes. We'll consider $Per(\mathbb{N})$ as a category, as follows:

(i) Objects of $Per(\mathbb{N})$ are pers on $\mathbb{N}$. (i.e. partitions of subsets of $\mathbb{N}$).

(ii) Arrows $E \to E'$ are (named by) partial recursive functions f preserving the partition, i.e. whenever nEm, then $f(n)\downarrow, f(m)\downarrow$ and $f(n)E'f(m)$. Equality of arrows: given $f, g : E \to E'$, $f = g$ if for all $m, n \in dom_E$, nEm implies $f(n)\downarrow, g(m)\downarrow$ and $f(n)E'g(m)$. In particular, f and g name the same morphism if for all $n \in dom_E$, $f(n)E'g(n)$.

PER models [BFSS] II

Theorem

$Per(\mathbb{N})$ forms a ccc.

Proof Sketch: uses some elementary recursive function theory. $\mathbf{1}$ is any per with a unique equivalence class. Given $A, B \in Per(\mathbb{N})$, $\langle m, n\rangle(A \times B)\langle p, q\rangle$ iff mAp and nBq, where $\langle\ ,\ \rangle : \mathbb{N}^2 \to \mathbb{N}$ is a chosen recursive bijection. $m(B^A)n$ iff m and n are codes of equal morphisms $A \to B$. The fundamental bijection $C \times A \to B \cong C \to B^A$ uses the S-m-n theorem of Recursion Theory.

A fundamental subcategory of $Per(\mathbb{N})$ is $\boldsymbol{I}$, *the inclusions*. So $f : E \to E'$ if f is (named by) the identity map $id_{\mathbb{N}}$. The maps are not intuitively "inclusions", since they may not be monos, indeed they may be surjections set-theoretically!

PER models [BFSS] III

Proposition (BFSS)

Every morphism in $Per(\mathbb{N})$ *may be decomposed into an iso, followed by an* (I)*-map, followed by an iso.*

Aside: A subcategory $\widetilde{\mathbb{N}}$ of $Per(\mathbb{N})$ was introduced by Lambek and Scott in a paper "An exactification of the monoid of primitive recursive functions". Objects of $\widetilde{\mathbb{N}}$ are *recursively enumerable* pers.

There is a somewhat similar Proposition for $\widetilde{\mathbb{N}}$ to the above.

Proposition (L-S)

In the category $\widetilde{\mathbb{N}}$ *of r.e. pers, every map can be factored as* (I)*-map, followed by an iso, followed by an* (I)*-map. The first* (I)*-map is a canonical surjection, and the last* (I)*-map is a canonical injection.*

PER models and Realizable Functors

Definition

A realizable *functor* $F : Per(\mathbb{N}) \to Per(\mathbb{N})$ *takes* $\mathbf{I}$ *to* $\mathbf{I}$ *and is such that there is an endomap* Φ *of partial recursive functions such that if* $E \xrightarrow{f} E'$ *then* $F(E) \xrightarrow{F(f)} F(E')$*, where* $F(f)$ *is named by* $\Phi(f)$*.*

Any functor definable in Girard's System $\mathcal{F}$ is realizable. Realizable functors are closed under all usual operations: pointwise products, twisted exponentials, substitution.

PER models [BFSS] Realizable Dinats

Definition

Let $\mathcal{C} = Per(\mathbb{N})$. For realizable functors $F, G : (\mathcal{C}^{op})^n \times \mathcal{C}^n \to \mathcal{C}$, a family $u = \{u_{\boldsymbol{A}} : F\boldsymbol{AA} \to G\boldsymbol{AA} \mid \boldsymbol{A} \in \mathcal{C}^n\}$ (not necessarily dinat) is *realizable* if there is a single partial recursive function φ that names each component $u_{\boldsymbol{A}}$.

Theorem

On $Per(\mathbb{N})$, realizable dinatural transformations (between realizable functors) compose.

Corollary

For each n, the realizable functors $F, G : (Per(\mathbb{N})^{op})^n \times Per(\mathbb{N})^n \to Per(\mathbb{N})$ and realizable dinats form a ccc.

PER models: horizontal and vertical merging of dinats

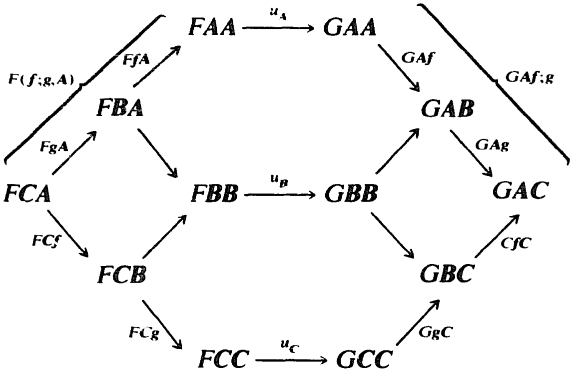
- Dinats wrt isos compose (i.e. we can horizontally merge dinat families restricted to isos), as the middle “diamond” is a pullback!

- Realizable dinats wrt I-maps horizontally compose...basically, it’s the same partial recursive function at each level, pre and post composed with an identity inclusion. Compose them horizontally.

In $Per(\mathbb{N})$, a map f is a composition of iso; inclusion; iso. So to get a general dinat (with respect to an arbitrary map f), begin with a 3×2 array of 6 composable dinat hexagons. These 6 hexagons are in 3 horizontal rows of two composable dinats each. The first row consists of two composable dinats wrt iso maps. The second row are two composable dinats wrt to "inclusion" maps. And the third row is again two composable dinats wrt to isos. After composing each row, the 3 remaining hexagons are vertically merged pairwise into a big dinat for arbitrary f represented as iso;inc;iso (see next page).

PER models [BFSS] (pairwise vertical merging)

Pairwise vertical merging of dinats into one bigger dinat wrt composite maps. So three dinat hexagons can be vertically composed wrt 3 composite maps $f;g;h$. Apply this to arbitrary maps in $Per(\mathbb{N})$ factored as iso; inclusion; iso to get dinats wrt arbitrary maps.



PER models [BFSS] System $\mathcal{F}$ modelling

We continue with the functorial polymorphism program: what about modelling $\forall\alpha.\tau$ in Girard's System $\mathcal{F}$? That's what polymorphism is about!

Proposition

In $Per(\mathbb{N})$, Realizable ends exist, i.e, the per $\int_{\boldsymbol{A}} G\boldsymbol{AA}$ exists.

Idea of proof: Intersections of pers exist. So take the intersection of all pers $G\boldsymbol{AA}$ then take the subper E corresponding to the dinaturality condition. Officially, mEn iff for any $\boldsymbol{f} : \boldsymbol{A} \rightarrow \boldsymbol{A'}$ $G\boldsymbol{AA'}$ relates $G(\boldsymbol{A}, \boldsymbol{f})(m)$ to $G(\boldsymbol{f}, \boldsymbol{A'})(n)$. For details, see [BFSS], Thm. 2.11.

PER models [BFSS] System $\mathcal{F}$ modelling II

To model Girard's $\forall\alpha$, we use realizable ends:

$$||\forall\alpha_i.\tau|| = \int_Q ||\tau||\boldsymbol{A}[A_i := Q]\boldsymbol{B}[B_i := Q]$$

E.g. $||\forall\alpha(\alpha \times (\alpha \Rightarrow \beta))||(A_1A_2; B_1B_2) =$

$\int_Q ||\alpha \times (\alpha \Rightarrow \beta))||(QA_2, QB_2) =$

$\int_Q Q \times (Q \Rightarrow B_2)$

PER models [BFSS] : System $\mathcal{F}$ Soundness

Given a derivable typing judgement $\vec{x_i} : \sigma_i \triangleright t : \tau$ in System $\mathcal{F}$, we can consider the term $e_t = \mathit{Erase}(t)$ which erases all the type information from t. It is an untyped lambda term, and can denote a partial recursive function (see [BFSS], Appendix A.1).

Theorem (Soundness of Realizable Dinats)

Given a derivable typing judgement in System $\mathcal{F}$, $\vec{x_i} : \sigma_i \triangleright t : \tau$ with free type variables among $\alpha_1, \ldots, \alpha_k$. Then e_t names a realizable dinat $||\sigma_1 \times \cdots \times \sigma_n|| \to ||\tau|| : (\mathcal{C}^{op})^k \times \mathcal{C}^k \to \mathcal{C}$, where $\mathcal{C} = Per(\mathbb{N})$. Provably β, η equal System $\mathcal{F}$ terms name the same dinat family, up to equality of dinats.

Some relations of dinats to linear logic

Many of the previous ideas work in the monoidal/linear logic world. For example let $\mathcal{C}$ be symmetric monoidal closed. If $F, G : (\mathcal{C}^{op})^n \times \mathcal{C}^n \to \mathcal{C}$, we define the following operations on multivariant functors:

- $(F \otimes G)(\boldsymbol{A}; \boldsymbol{B}) = F(\boldsymbol{A}; \boldsymbol{B}) \otimes G(\boldsymbol{A}; \boldsymbol{B})$
- $(F \multimap G)(\boldsymbol{A}; \boldsymbol{B}) = F(\boldsymbol{B}; \boldsymbol{A}) \multimap G(\boldsymbol{A}; \boldsymbol{B})$

If $\mathcal{C}$ is actually *-autonomous (in the sense of Barr), we can lift the basic interpretation of $\{\multimap, \otimes\}$ to formulas of multiplicative linear logic (MLL), $\{\otimes, (-)^{\perp}\}$. To the inductive clauses for definable functors, add

- If $\varphi(\alpha_1, \ldots, \alpha_n) \equiv (\alpha_i)^{\perp}$, define $[\![\, \varphi \,]\!](\boldsymbol{A}; \boldsymbol{B}) = (A_i)^*$, the linear negation of the ith *contravariant* projection.

See work of Blute-Scott (e.g. Linear Läuchli Semantics) on Full Completeness and earlier papers of Blute.

Day's Construction: A bit of History I

In 1970, SLN 137 published an important issue "Reports of the Midwest Category Seminar IV". There we see two important papers, which form the basis of this seminar:

(i) Brian Day: On Closed Categories of Functors,

(ii) Eduardo Dubuc and Ross Street: Dinatural Transformations

Day convolution was defined in this paper, of course using the co-end calculus, arising in (ii). I will leave the abstract treatment to other speakers. But I will now mention some concrete versions of Day's ideas.

Day's Construction: A bit of History II

The use of Day's tensor on functor categories $Set^{\mathcal{C}}$ ($\mathcal{C}$ monoidal) was popularized in CS in the period 1997-2000 by various researchers (e.g. Marcelo Fiore, Ian Stark, Moggi, Sangiorgi, et.al.) who were trying to find fully-abstract models of π-calculus (a language by Robin Milner for concurrency) as well as an early theory of "names" and "fresh names" .

My student Guy Beaulieu wrote an MSc thesis in 2002, entitled "Finding Presheaf Models for the Finite π-calculus", based on Day's tensor. The detailed calculations are in his thesis, available online: https://ruor.uottawa.ca/handle/10393/6206

Day's Construction on Functor Categories: concretely

Given the category of Sets with its usual ccc structure $(Set, \times, 1, \Rightarrow)$ and if $(\mathcal{C}, \otimes, I)$ is monoidal (not necessarily closed), we can define a structure $(Set^{\mathcal{C}}, \otimes_{day}, I_{Day}, \multimap_{Day})$ on the functor category. Given $S, T \in Set^{\mathcal{C}}$, and $s, X, Y \in \mathcal{C}$, define

$$\begin{aligned}
(S \otimes_{Day} T)(s) &= \int^{X} SX \times \int^{Y} TY \times \mathcal{C}(X \otimes Y, s) \\
&= \int^{X,Y} SX \times TY \times \mathcal{C}(X \otimes Y, s) \\
&= \Sigma_{X,Y} SX \times TY \times \mathcal{C}(X \otimes Y, s)/\sim \\
&\quad \text{(a quotient by an e.r.)} \\
I_{Day} &= \mathcal{C}(I, -) \\
(S \multimap_{Day} T)(s) &= Hom_{Set^{\mathcal{C}}}(S, T(s \otimes -))
\end{aligned}$$

Day's Construction on Functor Categories II

Guy Beaulieu points out the following results:

Proposition

$(- \otimes_{Day} S)$ is left adjoint to $(S \multimap_{Day} -)$, i.e.

$$Hom(B \otimes_{Day} S, T) \cong Hom(B, S \multimap_{Day} T)$$

Proposition

If $\mathcal{C}$ is symmetric monoidal, $\otimes_{Day}$ is symmetric

Also, if $Set^{\mathcal{C}}$ has its usual cartesian closed structure, then $Set^{\mathcal{C}}$ has two monoidal closed structures; moreover it has finite coproducts, so is often called a bicartesian closed category (and is a model of the O'Hearn-Pym "Bunched Logic")

Day's Construction on Functor Categories III

Given $S, T \in Set^{\mathcal{C}}$, and $s, X, Y \in \mathcal{C}$, recall

$$\begin{aligned}(S \otimes_{Day} T)(s) &= \int^{X} SX \times \int^{Y} TY \times \mathcal{C}(X \otimes Y, s) \\ &= \int^{X,Y} SX \times TY \times \mathcal{C}(X \otimes Y, s) \\ &= \Sigma_{X,Y} SX \times TY \times \mathcal{C}(X \otimes Y, s)/\sim\end{aligned}$$

where the equivalence relation $\sim$ is generated from :

$$(x \in SX, y \in TY, \eta : X \otimes Y \to s) \quad \sim$$
$$(x' \in SX', y' \in TY', \eta' : X' \otimes Y' \to s')$$

iff $\exists f : X' \to X, \exists g : Y' \to Y$ s.t. $Sf(x') = x$, $Tg(y') = y$ and $\eta(f \otimes g) = \eta'$.

Day's Construction on Functor Categories IV

Exercise: Define $\otimes_{Day}$, $\multimap_{Day}$ on maps.

Beaulieu shows: for $S \in Set^{\mathcal{C}}$, that $S \otimes_{Day} I_{Day} \cong S$.

Exercise: Check out the calculations with the notation here (there are some typos in Beaulieu!)