On dependently typed first-order logic

Erik Palmgren
Stockholm University
www.math.su.se

Logic Seminar
Stockholm, September 18, 2013

Introduction

A dependently typed (or sorted) system of first-order logic, FOLDS, was introduced and studied by Makkai (1995, 1998, 2013).

The purpose of FOLDS is to provide a natural logical system for formalizing (higher) category theory.

FOLDS can be considered as a so-called logic enriched type theory (Maietti-Sambin 2005, Gambino-Aczel 2006) where the underlying type theory is very rudimentary.

We present a system like FOLDS which is based on Cartmell's (1986) generalized algebraic theories but without equality of types. Unlike Makkai we allow function symbols.

Single-sorted vs multi-sorted first-order logic

The standard classical (or intutionistic) first-order logic assumes one non-empty (inhabited) domain of quantification. The latter restriction is due to the formulation of the existence introduction rule

$$\frac{\vdash \top[t/x]}{\vdash \exists x.\top}\ (\exists I) \tag{1}$$

(we may let $t = x$)
In many-sorted logics, especially those used in the categorical logic, one specifies the possible free variables of involved formulas

$$\frac{x_1 : A_1, \ldots, x_n : A_n; \Delta \vdash \psi[t/x]}{x_1 : A_1, \ldots, x_n : A_n; \Delta \vdash (\exists x : A)\psi}\ (\exists I)$$

if $FV(t) \subseteq \{x_1, \ldots, x_n\}$ and t has sort A. The derivation (1) is not possible for $n = 0$.

Many-sorted logic can be formulated (sequent-style) using judgements of the form

$$\Gamma; \Delta \vdash \phi$$

where

$$\Gamma = x_1 : A_1, \ldots, x_n : A_n$$

is a sequence of variables x_i with associated sort A_i, and where

$$\Delta = \phi_1, \ldots, \phi_m$$

is a sequence of formulas assumed to be true, and with their free variables listed in Γ, the formula proved true ϕ has also its free variables in Γ.

In standard many-sorted logic the sorts A_i belong to a given set of sorts and the order of the variables x_i does not matter

$$x_1 : A_1, \dots, x_n : A_n; \Delta \vdash \phi$$

We may generalize this situation by letting the sort or types be given by a type theory. This is the idea of *logic enriched type theory* (Gambino and Aczel 2006).

We introduce three new judgement forms

$A \text{ type}$ — A is a type

$a : A$ — a is an element (term) of type A

$\phi \text{ formula}$

... and their hypothetical versions

$$x_1 : A_1, \ldots, x_n : A_n \Longrightarrow A \text{ type}$$

$$x_1 : A_1, \ldots, x_n : A_n \Longrightarrow a : A$$

$$x_1 : A_1, \ldots, x_n : A_n \Longrightarrow \phi \text{ formula}$$

We will consider a type theory in which it is possible to introduce a finite number of dependent types and dependent function.

A representative example is the language of a category (cf. Cartmell 1986, Makkai 1995).

1. $\langle\rangle \Longrightarrow \mathsf{Ob}\ \mathrm{type}$
2. $x : \mathsf{Ob}, y : \mathsf{Ob} \Longrightarrow \mathsf{Hom}(x, y)\ \mathrm{type}$
3. $x : \mathsf{Ob} \Longrightarrow 1_x : \mathsf{Hom}(x, x)$
4. $x : \mathsf{Ob}, y : \mathsf{Ob}, z : \mathsf{Ob}, g : \mathsf{Hom}(y, z), f : \mathsf{Hom}(x, y) \Longrightarrow$ $g \circ_{x,y,z} f : \mathsf{Hom}(x, z)$

This is will be considered as a dependent signature declaring the constants Ob, Hom, 1 and $\circ$.
A complication is that we have to know that constants are declared in legal contexts, e.g. in (4) above we have to know its context is legal relative to (1) – (3).

Example 1: E-category

Signature as above but add an equality predicate only on Hom-types

$$x, y : \mathrm{Ob}, f, g : \mathrm{Hom}(x, y) \Longrightarrow f =_{x,y} g \text{ formula.}$$

Axioms in DTFOL/type theory

$$\begin{aligned}
&x, y : \mathrm{Ob}, f : \mathrm{Hom}(x, y); \vdash f =_{x,y} f.\\
&x, y : \mathrm{Ob}, f, g : \mathrm{Hom}(x, y); f =_{x,y} g \vdash g =_{x,y} .\\
&x, y, z : \mathrm{Ob}, f, g, h : \mathrm{Hom}(x, y); f =_{x,y} g, g =_{x,y} h \vdash f =_{x,y} h.\\
&x, y, z : \mathrm{Ob}, f, h : \mathrm{Hom}(x, y), g, k : \mathrm{Hom}(y, z);\\
&\qquad\qquad f =_{x,y} h, g =_{y,z} k \vdash g \circ_{x,y,z} f =_{x,z} k \circ_{x,y,z} h
\end{aligned}$$

$$\begin{aligned}
&x, y : \mathrm{Ob}, f : \mathrm{Hom}(x, y); \vdash 1_y \circ_{x,y,y} f =_{x,y} f.\\
&x, y : \mathrm{Ob}, f : \mathrm{Hom}(x, y); \vdash f \circ_{x,x,y} 1_x =_{x,y} f.\\
&x, y, z, w : \mathrm{Ob}, f : \mathrm{Hom}(x, y), g : \mathrm{Hom}(y, z), h : \mathrm{Hom}(z, w);\\
&\qquad\qquad \vdash h \circ_{x,z,w} (g \circ_{x,y,z} f) =_{x,w} (h \circ_{y,z,w} g) \circ_{x,y,w} f.
\end{aligned}$$

Example 2: E-bicategory

Signature for a bicategory (objects, arrows, transformations):

1. $\langle\rangle \Longrightarrow \mathsf{O}\ \mathrm{type}$
2. $x, y : \mathsf{O} \Longrightarrow \mathsf{A}(x, y)\ \mathrm{type}$
3. $x : \mathsf{O} \Longrightarrow 1_x : \mathsf{A}(x, x)$
4. $x, y, z : \mathsf{O}, g : \mathsf{A}(y, z), f : \mathsf{A}(x, y) \Longrightarrow g \circ_{x,y,z} f : \mathsf{A}(x, z)$.
5. $x, y : \mathsf{O}, f, g : \mathsf{A}(x, y) \Longrightarrow \mathsf{T}(x, y, f, g)\ \mathrm{type}$
6. $x, y : \mathsf{O}, f : \mathsf{A}(x, y) \Longrightarrow i_f : \mathsf{T}(x, y, f, f)\ \mathrm{type}$
7. $x, y : \mathsf{O}, f, g, h : \mathsf{A}(x, y), \beta : \mathsf{T}(x, y, g, h), \alpha : \mathsf{T}(x, y, f, g) \Longrightarrow$ $\beta \cdot_{x,y,f,g,h} \alpha : \mathsf{T}(x, y, f, h)$
8. $x, y, z : \mathsf{O}, f, h : \mathsf{A}(x, y), g, k : \mathsf{A}(y, z), \beta : \mathsf{T}(g, k), \alpha :$ $\mathsf{T}(f, h) \Longrightarrow \beta *_{x,y,z,f,h,g,k} \alpha : \mathsf{T}(x, z, g \circ_{x,y,z} f, k \circ_{x,y,z} h)$
9. $x, y, z, w : \mathsf{O}, f : \mathsf{A}(x, y), g : \mathsf{A}(y, z), h : \mathsf{A}(z, w) \Longrightarrow$ $\mathsf{a}_{x,y,z,w,f,g,h} : \mathsf{T}(x, w, (h \circ_{y,z,w} g) \circ_{x,y,w} f, h \circ_{x,z,w} (g \circ_{x,y,z} f))$
10. $x, y : \mathsf{O}, f : \mathsf{A}(x, y) \Longrightarrow \mathsf{l}_f : \mathsf{T}(x, y, 1_y \circ_{x,y,y} f, f)$
11. $x, y : \mathsf{O}, f : \mathsf{A}(x, y) \Longrightarrow \mathsf{r}_f : \mathsf{T}(x, y, f \circ_{x,x,y} 1_x, f)$

Further we add only congruence relations on the transformations:

$$x, y : \mathsf{O}, f, g : \mathsf{A}(x, y), \alpha, \beta : \mathsf{T}(x, y, f, g) \Longrightarrow \alpha =_{x,y,f,g} \beta\ \mathrm{formula}.$$

Depently typed first-order logic: Pre-syntax

We assume that F and T are two disjoint sets of symbols equipped with arities $0, 1, 2, \ldots$.
Pre-terms: formed using F as function symbols and variables in the usual way
Pre-types: $S(t_1, \ldots, t_n)$ where $S \in T$ is a type symbol, and $t_1, \ldots, t_n$ are pre-terms, $n = \mathrm{ar}(S)$
Pre-contexts: $\langle x_1 : A_1, \ldots, x_n : A_n \rangle$, where $x_1, \ldots, x_n$ are distinct variables, $A_1, \ldots, A_n$ are pre-types and $FV(A_i) \subseteq \{x_1, \ldots, x_{i-1}\}$.

Pre-syntax (cont.)

Pre-declarations: Either
of types:

$$x_1 : A_1, \ldots, x_n : A_n \Longrightarrow S(x_1, \ldots, x_n) \ \mathrm{type}$$

encoded as a pair (Γ, S) where $\Gamma = x_1 : A_1, \ldots, x_n : A_n$ pre-context, $S \in T$ and $\mathsf{ar}(S) = |\Gamma|$,
or of functions:

$$x_1 : A_1, \ldots, x_n : A_n \Longrightarrow f(x_1, \ldots, x_n) : U$$

endcoded as a triple (Γ, f, U) where $\Gamma = x_1 : A_1, \ldots, x_n : A_n$ pre-context, U pre-type, $FV(U) \subseteq FV(\Gamma)$, $\mathsf{ar}(f) = |\Gamma|$.

Pre-signature: a finite sequence $[D_1, \ldots, D_n]$ of pre-declarations.

Legal syntax — syntactic judgements

Contexts, types, elements and signatures are generated by simultaneous induction given by rules (R1) – (R9)

$$\frac{\Sigma \text{ signature}}{\langle\rangle \text{ context } [\Sigma]} \text{ (R1)}$$

$$\frac{\Sigma \text{ signature} \quad \Gamma \text{ context } [\Sigma] \quad \Gamma \Longrightarrow A \text{ type } [\Sigma] \quad x \in \mathit{Var} \setminus \mathrm{FV}(\Gamma)}{\Gamma, x : A \text{ context } [\Sigma]} \text{ (R2)}$$

$$\frac{\Sigma \text{ signature} \quad x_1 : A_1 \ldots, x_n : A_n \text{ context } [\Sigma]}{x_1 : A_1 \ldots, x_n : A_n \Longrightarrow A_i \text{ type } [\Sigma]} \text{ (R3)}$$

$$\frac{\Sigma \text{ sig.} \quad x_1 : A_1 \ldots, x_n : A_n \text{ cont. } [\Sigma] \quad x_1 : A_1 \ldots, x_n : A_n \Longrightarrow A_i \text{ type } [\Sigma]}{x_1 : A_1 \ldots, x_n : A_n \Longrightarrow x_i : A_i \ [\Sigma]}$$

To formulate the next two rules we introduce the notion of context map. Let Δ and $\Gamma = x_1 : A_1, \ldots, x_n : A_n$ be two contexts relative to Σ. A sequence $(a_1, \ldots, a_n)$ of preterms is called a *context map* $\Delta \longrightarrow \Gamma$ if

$$\begin{aligned}
&\Delta \Longrightarrow A_1 \text{ type } [\Sigma]\\
&\Delta \Longrightarrow a_1 : A_1 \ [\Sigma]\\
&\Delta \Longrightarrow A_2[a_1/x_1] \text{ type } [\Sigma]\\
&\Delta \Longrightarrow a_2 : A_2[a_1/x_1] \ [\Sigma]\\
&\quad\vdots\\
&\Delta \Longrightarrow A_n[a_1 \ldots, a_{n-1}/x_1, \ldots, x_{n-1}] \text{ type } [\Sigma]\\
&\Delta \Longrightarrow a_n : A_n[a_1 \ldots, a_{n-1}/x_1, \ldots, x_{n-1}] \ [\Sigma]
\end{aligned}$$

We write $(a_1, \ldots, a_n) : \Delta \longrightarrow \Gamma \ [\Sigma]$ for the conjunction of these judgements.

The sequence $(a_1, \ldots, a_n)$ can be plugged into correct function and type declarations according to the next rules.

$$\frac{\begin{array}{ll} (a_1, \ldots, a_n) : \Delta \longrightarrow \langle x_1 : A_1, \ldots, x_n : A_n \rangle \ [\Sigma] & \Sigma \text{ signature} \\ (\langle x_1 : A_1 \ldots, x_n : A_n \rangle, S) \text{ decl. in } \Sigma & \Delta \text{ context } [\Sigma] \end{array}}{\Delta \Longrightarrow S(a_1, \ldots, a_n) \text{ type } [\Sigma]} \quad \text{(R5)}$$

$$\frac{\begin{array}{ll} (a_1, \ldots, a_n) : \Delta \longrightarrow \langle x_1 : A_1, \ldots, x_n : A_n \rangle \ [\Sigma] & \\ (\langle x_1 : A_1 \ldots, x_n : A_n \rangle, f, U) \text{ decl. in } \Sigma & \Sigma \text{ signature} \\ \Delta \Longrightarrow U[a_1, \ldots, a_n / x_1, \ldots, x_n] \text{ type } [\Sigma] & \Delta \text{ context } [\Sigma] \end{array}}{\Delta \Longrightarrow f(a_1, \ldots, a_n) : U[a_1, \ldots, a_n / x_1, \ldots, x_n] \ [\Sigma]} \quad \text{(R6)}$$

The final three rules are concerned with the formation of correct signatures, i.e. sequences of function and type declarations.

$$\frac{}{[]\ \text{signature}}\ (\text{R7})$$

$$\frac{\Sigma\ \text{sig.} \quad \Gamma\ \text{con.}\ [\Sigma] \quad S \in T \quad S\ \mathsf{not\ decl.\ in}\ \Sigma \quad |\Gamma| = \text{ar}(S)}{[\Sigma, (\Gamma, S)]\ \text{sig.}}\ (\text{R8})$$

$$\frac{\Sigma\ \text{sig.} \quad \Gamma\ \text{con.}\ [\Sigma] \quad \Gamma \Longrightarrow U\ \text{type}\ [\Sigma] \quad f \in F\ \mathsf{not\ decl.\ in}\ \Sigma \quad |\Gamma| = \text{ar}(f}{[\Sigma, (\Gamma, f, U)]\ \text{sig.}}$$

The necessary substitution and weakening lemmas can be proved by induction on the derivations:

Substitution lemma Let $(s_1, \ldots, s_n) : \Theta \longrightarrow \Gamma$ be a context map, where $\Gamma = x_1 : A_1, \ldots, x_n : A_n$.

(a) If $\Gamma \Longrightarrow B \text{ type}$, then $\Theta \Longrightarrow B[s_1, \ldots, s_n/x_1, \ldots, x_n] \text{ type.}$

(b) If $\Gamma \Longrightarrow b : B$, then
$\Theta \Longrightarrow b[s_1, \ldots, s_n/x_1, \ldots, x_n] : B[s_1, \ldots, s_n/x_1, \ldots, x_n] \text{ type}$

Weakening lemma Suppose that $\Gamma \Longrightarrow B \text{ type.}$ Let y be a variable not in $\mathrm{FV}(\Gamma, \Theta)$.

(a) If $\Gamma, \Theta \text{ context}$, then $\Gamma, y : B, \Theta \text{ context}$

(b) If $\Gamma, \Theta \Longrightarrow A \text{ type}$, then $\Gamma, y : B, \Theta \Longrightarrow A \text{ type}$

(c) If $\Gamma, \Theta \Longrightarrow a : A$, then $\Gamma, y : B, \Theta \Longrightarrow a : A$

First-order formulas with dependent sorts

Similar to ordinary many-sorted first-order logic.

Predicate symbols are given by a set P of symbols with an arity.

Given a term signature Σ, we may to any

$$x_1 : A_1, \dots, x_n : A_n \ \mathrm{context}\ [\Sigma]$$

assign a new n-ary predicate symbol $R \in P$. This gives a predicate declaration

$$(\langle x_1 : A_1, \dots, x_n : A_n \rangle, R).$$

Thus a signature for first-order logic with dependent sorts consists of a term signature Σ and a sequence of predicate declarations $\Pi = [E_1, \dots, E_m]$, where all predicates declared are distinct. Given this we can form the set of formulas in each variable context Γ over $[\Sigma; \Pi]$:

- For each predicate declaration (Δ, R) in Π, we assume the rule

$$\frac{\Gamma \text{ context } [\Sigma] \quad (a_1, \ldots, a_n) : \Gamma \longrightarrow \Delta}{\Gamma \Rightarrow R(a_1, \ldots, a_n) \text{ formula } [\Sigma; \Pi]}$$

- $$\frac{\Gamma \text{ context } [\Sigma]}{\Gamma \Rightarrow \bot \text{ formula } [\Sigma; \Pi]} \qquad \frac{\Gamma \text{ context } [\Sigma]}{\Gamma \Rightarrow \top \text{ formula } [\Sigma; \Pi]}$$

- $$\frac{\Gamma \Rightarrow \phi \text{ formula} \quad \Gamma \Rightarrow \psi \text{ formula } [\Sigma; \Pi]}{\Gamma \Rightarrow (\phi \wedge \psi) \text{ formula } [\Sigma; \Pi]}$$

$$\frac{\Gamma \Rightarrow \phi \text{ formula} \quad \Gamma \Rightarrow \psi \text{ formula } [\Sigma; \Pi]}{\Gamma \Rightarrow (\phi \vee \psi) \text{ formula } [\Sigma; \Pi]}$$

$$\frac{\Gamma \Rightarrow \phi \text{ formula} \quad \Gamma \Rightarrow \psi \text{ formula } [\Sigma; \Pi]}{\Gamma \Rightarrow (\phi \rightarrow \psi) \text{ formula } [\Sigma; \Pi]}$$

- $$\frac{\Gamma, x : A \Rightarrow \phi \text{ formula } [\Sigma; \Pi]}{\Gamma \Rightarrow (\forall x : A)\phi \text{ formula } [\Sigma; \Pi]} \qquad \frac{\Gamma, x : A \Rightarrow \phi \text{ formula } [\Sigma; \Pi]}{\Gamma \Rightarrow (\exists x : A)\phi \text{ formula } [\Sigma; \Pi]}$$

Logic

Assumption rule:

$$\frac{\Gamma \Rightarrow \phi_1, \ldots, \phi_n \text{ formulas}}{\Gamma; \phi_1, \ldots, \phi_m \vdash \phi_i}$$

Propositional rules:

$$\frac{\Gamma \Rightarrow \phi \text{ formula} \quad \Gamma; \Delta \vdash \bot}{\Gamma; \Delta \vdash \phi}\ (\bot E) \qquad \frac{\Gamma \Rightarrow \Delta \text{ formulas}}{\Gamma; \Delta \vdash \top}\ (\top I)$$

$$\frac{\Gamma; \Delta \vdash \phi \quad \Gamma; \Delta \vdash \psi}{\Gamma; \Delta \vdash \phi \wedge \psi}\ (\wedge I) \qquad \frac{\Gamma; \Delta \vdash \phi \wedge \psi}{\Gamma; \Delta \vdash \phi}\ (\wedge E_1) \quad \frac{\Gamma; \Delta \vdash \phi \wedge \psi}{\Gamma; \Delta \vdash \psi}\ (\wedge E_2)$$

$$\frac{\Gamma; \Delta, \phi \vdash \psi}{\Gamma; \Delta \vdash \phi \to \psi}\ (\to I) \qquad \frac{\Gamma; \Delta \vdash \phi \to \psi \quad \Gamma; \Delta \vdash \phi}{\Gamma; \Delta \vdash \psi}\ (\to E)$$

$$\frac{\Gamma; \Delta, \vdash \phi \quad \Gamma \Longrightarrow \psi \text{ formula}}{\Gamma; \Delta \vdash \phi \vee \psi} \ (\vee I_1) \qquad \frac{\Gamma; \Delta, \vdash \psi \quad \Gamma \Longrightarrow \phi \text{ formula}}{\Gamma; \Delta \vdash \phi \vee \psi} \ (\vee I_2)$$

$$\frac{\Gamma; \Delta \vdash \phi \vee \psi \quad \Gamma; \Delta, \phi \vdash \theta \quad \Gamma; \Delta, \psi \vdash \theta}{\Gamma; \Delta \vdash \theta} \ (\vee E)$$

Quantifier rules: (usual variable conditions in blue)

$$\frac{\Gamma, x : A; \Delta \vdash \psi \quad \Gamma \Rightarrow \Delta \text{ formulas}}{\Gamma; \Delta \vdash (\forall x : A)\psi}\ (\forall I)$$

$$\frac{\Gamma; \Delta \vdash (\forall x : A)\psi \quad \Gamma \Rightarrow t : A}{\Gamma; \Delta \vdash \psi[t/x]}\ (\forall E)$$

$$\frac{\Gamma \Rightarrow t : A \quad \Gamma, x : A \Rightarrow \psi \text{ formula} \quad \Gamma; \Delta \vdash \psi[t/x]}{\Gamma; \Delta \vdash (\exists x : A)\psi}\ (\exists I)$$

$$\frac{\Gamma; \Delta \vdash (\exists x : A)\psi \quad \Gamma, x : A; \Delta, \psi \vdash \phi \quad \Gamma \Rightarrow \Delta, \phi \text{ formulas}}{\Gamma; \Delta \vdash \phi}\ (\exists E)$$

We assume capture free substitution and α-equivalence, so that substitutions into quantified formulas are always possible.

Partial functions - in classical logic

In classical (non-dependent, many-sorted,) logic a functional relation, can be extended to a function.
Suppose

$$x =_A u, y =_B z, \phi(x, y) \vdash \phi(u, z)$$

and

$$\phi(x, y), \phi(x, z) \vdash y =_B z$$

Suppose $*$ is some constant in B. We can introduce a total function symbol $f : A \to B$ with defining axiom

$$f(x) =_B y \leftrightarrow \phi(x, y) \vee y =_B * \wedge \neg(\exists y : B)\, \phi(x, y).$$

In intuitionistic logic this is not possible, unless the domain of definition of the relation ϕ is decidable.

Partial functions - in dependently typed FOL

Suppose again

$$x =_A u, y =_B z, \phi(x, y) \vdash \phi(u, z)$$

and

$$\phi(x, y), \phi(x, z) \vdash y =_B z$$

Introduce a type D for the domain of definition of ϕ

$$x : A \Rightarrow D(x) \ \mathrm{type}$$

and an axiom

$$x : A, y : B; \phi(x, y) \vdash (\exists p : D(x))\top$$

and a dependent function symbol f

$$x : A, p : D(x) \Rightarrow f(x, p) : B$$

with axiom

$$x : A, p : D(x); \vdash \phi(x, f(x, p))$$

This works out as it should ...

Local propositions-as-types

Consider a fixed signature. Suppose $\Gamma = x_1 : A_1, \ldots, x_n : A_n$ is a context and that ϕ is a formula in that context.

Add a new dependent type to the signature

$$\Gamma \Longrightarrow F(x_1, \ldots, x_n) \ \mathrm{type}$$

Then add two axioms over the extended signature

$$\Gamma, p : F(x_1, \ldots, x_n); \vdash \phi$$

$$\Gamma; \phi \vdash (\exists p : F(x_1, \ldots, x_n))\top$$

Truth of $\phi (= \phi(x_1, \ldots, x_n))$ has thus been encoded as inhabitedness of $F(x_1, \ldots, x_n)$.

Standard semantics in Martin-Löf type theory

A signature Σ, Π is interpreted by a sequence of constant declarations. These will be postulates of type theory.
The standard semantics of a judgement in dependently typed first-order logic

$$x_1 : A_1, \ldots, x_n : A_n; \phi_1, \ldots, \phi_m \vdash \psi$$

will be the type-theoretic judgement

$$x_1 : A_1, \ldots, x_n : A_n, \phi_1 \text{ true}, \ldots, \phi_m \text{ true} \Longrightarrow \psi \text{ true}$$

which can be paraphrased as the existence of a term q such that

$$x_1 : A_1, \ldots, x_n : A_n, p_1 : \phi_1, \ldots, p_m : \phi_m \Longrightarrow q : \psi.$$

This is interpretation is straightforward in Coq and Agda and should be easily supported by these proof assistants.

As we consider the standard semantics to be M-L type theory (MLTT) the notion of function will be the (intensional) functions or operations of that theory. The axiom of choice is valid in MLTT with this notion of function, since it does not require the functions to respect prescribed equivalence relations. In E-categories we may for example have functions that chose pullbacks and projections from the arrow data.

$$\begin{tikzcd}
\mathrm{P}(X, Y, Z, f, g) \arrow[r, "\mathrm{p}_1(x,y,z,f,g)"] \arrow[d, "\mathrm{p}_2(x,y,z,f,g)"'] & X \arrow[d, "f"] \\
Y \arrow[r, "g"'] & Z
\end{tikzcd}$$

But it is not necessary that these functions respect equalities. Functions that respect equalities will be called extensional functions.

CETCS - constructive version of Lawvere's ETCS

The Elementary Theory of the Category of Sets (ETCS) is a classical first-order axiomatization of the category of sets and function (Lawvere 1963). The set-theoretic constructions possible are basically those of Zermelo's set theory Z. In modern categorical terms ETCS is a well-pointed boolean topos with the axiom of choice. S. MacLane has argued (not quite successfully) such theories are enough for mathematical practice.

Unlike set theory its theorems are invariant under set-isomorphism

$$\phi(A) \text{ and } A \cong B \text{ implies } \phi(B) \qquad (A, B \text{ are sets})$$

CETCS is a constructive version of ETCS suitable for formalizing elementary parts of Bishop's constructive mathematics (P. 2012).

CETCS

Signature: to Σ_{Cat} we add declarations for terminal object and pullbacks

1. $\langle\rangle \Longrightarrow 1 : \mathsf{Ob}$
2. $x, y, z : \mathsf{Ob}, f : \mathsf{Hom}(x, z), g : \mathsf{Hom}(y, z) \Longrightarrow \mathsf{P}(x, y, z, f, g) : \mathsf{Ob}$
3. $-" - \Longrightarrow \mathsf{p}_1(x, y, z, f, g) : \mathrm{Hom}(\mathsf{P}(x, y, z, f, g), x)$
4. $-" - \Longrightarrow \mathsf{p}_2(x, y, z, f, g) : \mathrm{Hom}(\mathsf{P}(x, y, z, f, g), y)$

and similarly for initial object and pushouts. Further we can add the Π-construction

5. $x, y, z : \mathsf{Ob}, f : \mathsf{Hom}(x, y), g : \mathsf{Hom}(z, x) \Longrightarrow \Pi(x, y, z, f, g) : \mathsf{Ob}$
6. $-" - \Longrightarrow \pi(x, y, z, f, g) : \mathsf{Hom}(\Pi(x, y, z, f, g), y)$
7. $-" - \Longrightarrow \mathrm{ev}(x, y, z, f, g) : \mathsf{Hom}(\mathsf{P}(x, \Pi(x, y, z, f, g), y, f, \pi(x, y, z, f, g)), z)$

That the object 1 is terminal is formalized as

$$\begin{aligned}
&X : \mathrm{Ob}, f, g : \mathrm{Hom}(X, 1); \vdash f =_{X,1} g \\
&X : \mathrm{Ob}; \vdash (\exists f : \mathrm{Hom}(X, 1))\top
\end{aligned} \tag{2}$$

That pullbacks exists are given by (dropping some subscripts and derivable arguments)

$$\begin{aligned}
&X, Y, Z : \mathrm{Ob}, f : \mathrm{Hom}(X, Z), g : \mathrm{Hom}(Y, Z); \vdash \\
&\qquad f \circ \mathsf{p}_1(f, g) = g \circ \mathsf{p}_2(f, g) \\
&X, Y, Z, W : \mathrm{Ob}, f : \mathrm{Hom}(X, Z), g : \mathrm{Hom}(Y, Z), h : \mathrm{Hom}(W, X), k : \mathrm{Hor} \\
&\qquad f \circ h = g \circ k \vdash \\
&\qquad (\exists t : \mathrm{Hom}(W, \mathsf{P}(f, g))) \\
&\qquad\qquad\qquad \mathsf{p}_1(f, g) \circ t =_{w,x} h \wedge \mathsf{p}_2(f, g) \circ t = k. \\
&X, Y, Z, W : \mathrm{Ob}, f : \mathrm{Hom}(X, Z), g : \mathrm{Hom}(Y, Z), t, s : \mathrm{Hom}(w, \mathsf{P}(f, g))); \\
&\qquad \mathsf{p}_1(f, g) \circ t = \mathsf{p}_1(f, g) \circ s, \\
&\qquad \mathsf{p}_2(f, g) \circ t = \mathsf{p}_2(f, g) \circ s \vdash t = s
\end{aligned} \tag{3}$$

A morphism $x : 1 \longrightarrow X$ is called an *element* of X. We write this as $x \in X$. For $x, x' \in X$ we write $x =_{1,X} x'$ as $x =_X x'$. A morphism $f : X \longrightarrow Y$ is called *surjective* if for every $y \in Y$, there is $x \in X$ such that $f \circ x =_Y y$. The following axiom states that 1 is a strong generator of the category:

(G) Every surjective mono $X \longrightarrow Y$ is an isomorphism.

We consider as in (P. 2012) a sequence of maps $\alpha_1 : P \longrightarrow X_1, \ldots, \alpha_1 : P \longrightarrow X_n$ that are jointly monic as a relation between $X_1, \ldots, X_n$ and for elements $x_1 \in X_1, \ldots, x_n \in X_n$ we write

$$(x_1, \ldots, x_n) \,\epsilon\, (\alpha_1, \ldots, \alpha_n)$$

if there is $p \in P$ with $\alpha_1 \circ p =_{X_1} x_1, \ldots, \alpha_n \circ p =_{X_n} x_n$. A relation $\alpha_1 : P \longrightarrow X_1, \alpha_2 : P \longrightarrow X_2$ is called a *partial function from X_1 to X_2* if α is mono.

Existence of dependent products Π is formulated as follows

For any mappings $Y \xrightarrow{g} X \xrightarrow{f} I$ we have a commutative diagram

$$\begin{tikzcd}
Y \arrow[dr, "g"'] & \mathsf{P}(f, \pi(f, g)) \arrow[l, "\mathrm{ev}(f,g)"'] \arrow[r, "\mathsf{p}_1"] \arrow[d, "\mathsf{p}_2"'] & \Pi(f, g) \arrow[d, "\pi(f,g)"] \\
 & X \arrow[r, "f"'] & I
\end{tikzcd} \tag{4}$$

where for any for any element $i \in I$ and any partial function $\psi =_{\mathrm{def}} (\xi : R \longrightarrow X, \upsilon : R \longrightarrow Y)$ satisfying (a) and (b):

(a) for all $x \in X$, $y \in Y$, $(x, y) \,\epsilon\, \psi$ implies $gy =_X x$ and $fx =_I i$,

(b) if $fx =_I i$, then there is $y \in Y$ with $(x, y) \,\epsilon\, \psi$

there is a unique $s \in \Pi(f, g)$ s.t. $\pi(f, g) \circ s =_I i$ and for all $x \in X$, $y \in Y$,

$$(s, x, y) \,\epsilon\, \alpha \iff (x, y) \,\epsilon\, \psi. \tag{5}$$

Here $\alpha =_{\mathrm{def}} (\mathsf{p}_1, \mathsf{p}_2, \mathrm{ev}(f, g))$.

Further axioms include: Natural numbers object (for recursion and induction), non-triviality of binary sums.
and optionally a presentation axiom with new constants

8. $x : \mathsf{Ob} \Longrightarrow \mathsf{Pre}(x) : \mathsf{Ob}$
9. $x : \mathsf{Ob} \Longrightarrow \mathsf{pre}_x : \mathsf{Hom}(\mathsf{Pre}(x), x)$

(PA) For every object x, $\mathsf{Pre}(x)$ is projective and $\mathsf{pre}_x : \mathsf{Pre}(x) \longrightarrow x$ is surjective.

This says that every object is a quotient of an object on which the axiom of choice is valid.

Axiomatizing the category of small categories

An early proposal:

F.W. Lawvere: The category of categories as a foundation for mathematics, In: *La Jolla conference on categorical algebra,* Springer-Verlag, 1966, pp. 1-20.

Is there a constructive version of this theory ?

A start:

O.Wilander: An E-bicategory of E-categories exemplifying a type-theoretic approach to bicategories. In: O.Wilander, *On constructive sets and partial structures.* PhD Thesis, Uppsala University 2011.

Project: carry out such a axiomatization using dependently typed FOL, with E-categories in type theory rather than categories in sets as intended model.

General semantics of dependently typed FOL

The type system we introduced may be interpreted by a Category with Attributes. This is a category $\mathcal{E}$ (contexts and context maps) with a contravariant functor $\mathsf{Ty} : \mathcal{E}^{\mathrm{op}} \longrightarrow \mathrm{Set}$ (the possible types in a context and the action on them by substitution). There is moreover an operation . that extends contexts $\Gamma \in \mathrm{Ob}(\mathcal{E})$ by a type $B \in \mathrm{Ty}(\Gamma)$: $\Gamma.B \in \mathrm{Ob}(\mathcal{E})$. The extended context has a projection down to the orginal context

$$\mathsf{p}_{\Gamma,B} : \Gamma.B \longrightarrow \Gamma.$$

The logic is then interpreted using another contravariant functor $\mathcal{E}^{\mathrm{op}} \longrightarrow \mathrm{Heyting}$ similarly to a hyper doctrine or tripos.

In detail:

- A category with attributes $\mathcal{E}$ consists of a category with terminal object, a functor $\mathrm{Ty} : \mathcal{E}^{op} \longrightarrow \mathrm{Sets}$, for each $\Gamma \in \mathrm{Ob}(\mathcal{E})$ and each $B \in \mathrm{Ty}(\Gamma)$ an object $\Gamma.B$ and a morphism in $\mathcal{E}$

  $$\mathrm{p}_{\Gamma,B} : \Gamma.B \longrightarrow \Gamma$$

  and for each $\sigma : \Delta \longrightarrow \Gamma$ and morphism $\mathsf{q}_{\Delta,\Gamma,B}(\sigma) : \Delta \cdot (\mathsf{Ty}(\sigma)(B)) \longrightarrow \Gamma.B$ such that

  $$\begin{tikzcd}
  \Delta \cdot (\mathsf{Ty}(\sigma)(B)) \arrow[r, "\mathsf{q}_{\Delta,\Gamma,B}(\sigma)"] \arrow[d, "\mathsf{p}_{\Delta,\mathsf{Ty}(\sigma)(B)}"'] & \Gamma \cdot B \arrow[d, "\mathsf{p}_{\Gamma,B}"] \\
  \Delta \arrow[r, "\sigma"'] & \Gamma
  \end{tikzcd}$$

  is a pullback square in $\mathcal{E}$.

The following functoriality conditions should hold:
For $\Delta = \Gamma$ and $\sigma = 1_\Gamma$

$$\mathsf{q}_{\Gamma,\Gamma,B}(1_\Gamma) = 1_{\Gamma.B} :$$

For $\tau : \Theta \longrightarrow \Delta$, $\sigma : \Delta \longrightarrow \Gamma$

$$\mathsf{q}_{\Theta,\Gamma,B}(\sigma \circ \tau) = \mathsf{q}_{\Delta,\Gamma,B}(\sigma) \circ \mathsf{q}_{\Theta,\Delta,\mathsf{Ty}(\sigma)(B)}(\tau)$$

(and consequently the corresponding domains and codomains should be equal).

The logical part :

- A functor $P : \mathcal{E}^{op} \longrightarrow \mathrm{Heyting}$ into the category of Heyting-algebras with maps preserving all propositional logic operations ($\wedge$, $\vee$, $\rightarrow$,$\top$,$\bot$)
- For any $\Gamma \in \mathrm{Ob}(\mathcal{E})$ and $S \in \mathrm{Ty}(\Gamma)$ monotone operations

$$\forall_{\Gamma,S}, \exists_{\Gamma,S} : P(\Gamma.S) \longrightarrow P(\Gamma)$$

  such that for $Q \in P(\Gamma)$, $R \in P(\Gamma.S)$,

  - $Q \leq \forall_{\Gamma,S}(R) \iff P(\mathsf{p}_{\Gamma,S})(Q) \leq R$
  - $\exists_{\Gamma,S}(R) \leq Q \iff R \leq P(\mathsf{p}_{\Gamma,S})(Q)$.

- (Beck-Chevalley) For the pullback square

$$\begin{tikzcd}
\Delta.S\{\sigma\} \arrow[r, "\mathsf{q}_{\Delta,\Gamma,S}(\sigma)"] \arrow[d, "\mathsf{p}_{\Delta,S\{\sigma\}}"'] & \Gamma.S \arrow[d, "\mathsf{p}_{\Gamma,S}"] \\
\Delta \arrow[r, "\sigma"'] & \Gamma
\end{tikzcd} \tag{6}$$

  we have for $R \in P(\Gamma.S)$,

  - $P(\sigma)(\forall_{\Gamma,S}(R)) = \forall_{\Delta,S\{\sigma\}}(P(q_{\Delta,\Gamma,S})(\sigma)(R))$,
  - $P(\sigma)(\exists_{\Gamma,S}(R)) = \exists_{\Delta,S\{\sigma\}}(P(q_{\Delta,\Gamma,S})(\sigma)(R))$.

The Lindenbaum-Tarski model

We indicate how to construct a universal Heyting-algebra model.

Theorem. The contexts and context maps for a fixed signature form a category with attributes.

Proof. Let Σ be a fixed signature. Let $\Gamma = x_1 : A_1, \ldots, x_n : A_n$ be a context with respect to the signature. By rules (R3) and (R3) we have for all $i = 1, \ldots, n$

$$\Gamma \Longrightarrow A_i \text{ type}$$

and

$$\Gamma \Longrightarrow x_i : A_i$$

Now trivially, $A_i = A_i[x_1, \ldots, x_{i-1}/x_1, \ldots, x_{i-1}]$ so

$$1_\Gamma =_{\text{def}} (x_1, \ldots, x_n) : \Gamma \longrightarrow \Gamma$$

is a context map. This will be the identity.

Suppose that $\Delta = y_1 : B_1, \ldots, y_m : B_m$ and $\Theta = z_1 : C_1, \ldots, z_k : C_k$ are contexts and that

$$\sigma = (s_1, \ldots, s_m) : \Gamma \longrightarrow \Delta \text{ and } \tau = (t_1, \ldots, t_k) : \Delta \longrightarrow \Theta$$

are context maps. The vector of terms

$$\tau \circ \sigma =_{\mathrm{def}} \big(t_1[s_1, \ldots, s_m/y_1, \ldots, y_m], \ldots, t_k[s_1, \ldots, s_m/y_1, \ldots, y_m]\big)$$

is a context map $\Gamma \longrightarrow \Theta$, the composition of τ and σ.

It is straightforward to show that this gives a category where the empty context $\langle\rangle$ is terminal.

The following square is a pullback diagram for any variable $v \notin FV(\Delta)$, and $u \notin FV(\Gamma)$:

$$\begin{tikzcd}
\Delta, v : S[s_1, \ldots, s_n/x_1, \ldots, x_n] \arrow[r, "(s_1,\ldots,s_n,v)"] \arrow[d, "(y_1,\ldots,y_m)"'] & \Gamma, u : S \arrow[d, "(x_1,\ldots,x_n)"] \\
\Delta \arrow[r, "(s_1,\ldots,s_n)"'] & \Gamma
\end{tikzcd}$$

Assume that the variables form a set V with decidable equality, and that there is a function fresh such that for any list of variables $x_1, \ldots, x_n$,

$$\mathrm{fresh}(x_1, \ldots, x_n) \in V \setminus \{x_1, \ldots, x_n\}.$$

For a context $\Gamma = x_1 : A_1, \ldots, x_n : A_n$ write

$$\mathrm{fresh}(\Gamma) = \mathrm{fresh}(x_1, \ldots, x_n).$$

Then this becomes the required pullback diagram of the CwA:

$$\begin{tikzcd}
\Delta, \mathrm{fresh}(\Delta) : S[s_1, \ldots, s_n/x_1, \ldots, x_n] \arrow[r, "(s_1, \ldots, s_n, \mathrm{fresh}(\Gamma))"] \arrow[d, "(y_1, \ldots, y_m)"'] & \Gamma, \mathrm{fresh}(\Delta) : S \arrow[d, "(x_1, \ldots, x_n)"] \\
\Delta \arrow[r, "(s_1, \ldots, s_n)"'] & \Gamma
\end{tikzcd}$$

Let $\mathcal{C}_{\Sigma}$ be the category of contexts and context maps relative to a signature Σ. We define a functor $\mathsf{Ty} : \mathcal{C}_{\Sigma}^{\mathrm{op}} \longrightarrow \mathrm{Set}$ which assigns types to contexts

$$\mathsf{Ty}(\Gamma) = \{S \in \mathrm{pretype} : \Gamma \Longrightarrow S \ \mathrm{type}\}$$

and for a context map $\sigma : \Delta \longrightarrow \Gamma$, let

$$\mathsf{Ty}(\sigma)(S) = S[\sigma/\Gamma].$$

Here

$$S[\sigma/\Gamma] = S[s_1, \ldots, s_n/x_1, \ldots, x_n]$$

Let $\mathcal{S} = [\Sigma, \Pi]$ be a fixed signature.

Theorem. Let Γ be a context in the signature. Then the set of formulas in the context

$$P(\Gamma) = \{\phi : \Gamma \Longrightarrow \phi\}$$

is a Heyting pre-algebra ordered ($\leq$) by derivability

$$\phi \leq \psi \Longleftrightarrow_{\mathrm{def}} \Gamma; \phi \vdash \psi.$$

Substitution is an operation that preserves order and the propositional connectives.
For a context map $f : \Delta \longrightarrow \Gamma$ define $P(f) : P(\Gamma) \longrightarrow P(\Delta)$ by

$$P(f)(\phi) = \phi[f/\Gamma]$$

Theorem. For a context map $\sigma : \Delta \longrightarrow \Gamma$, $P(\sigma)$ is a morphism of pre-Heyting algebras, i.e. it preserves the order and the operations $\wedge$, $\vee$, $\rightarrow$, $\top$ and $\bot$. Moreover this assignment is pseudo-functorial in the sense that for another context map $\tau : \Theta \longrightarrow \Delta$

$$P(\tau \circ \sigma)(\phi) \equiv P(\tau)(P(\sigma)(\phi))$$

and

$$P(1_\Gamma)(\phi) \equiv \phi.$$

Here $\phi \equiv \psi$ means $\phi \leq \psi$ and $\psi \leq \phi$.

Suppose Γ $\mathrm{context}$ and $\Gamma \Rightarrow S : \mathrm{type}$. We have the projection context map $\mathsf{p}_{\Gamma,S}; \Gamma.S \longrightarrow \Gamma$. Define $\forall_{\Gamma,S}, \exists_{\Gamma,S} : P(\Gamma.A) \longrightarrow P(\Gamma)$ by

$$\forall_{\Gamma,S}(\phi) = (\forall\, \mathsf{fr}(\Gamma) : S)(\phi)$$

and

$$\exists_{\Gamma,S}(\phi) = (\exists\, \mathsf{fr}(\Gamma) : S)(\phi).$$

Theorem. Suppose Γ $\mathrm{context}$ and $\Gamma \Rightarrow S : \mathrm{type}$. Then

(a) $\forall_{\Gamma,S}$ and $\exists_{\Gamma,S}$ are monotone

(b) For $Q \in P(\Gamma)$, $R \in P(\Gamma.S)$,

$$Q \leq \forall_{\Gamma,S}(R) \iff P(\mathsf{p}_{\Gamma,S})(Q) \leq R$$

(c) For $Q \in P(\Gamma)$, $R \in P(\Gamma.S)$,

$$\exists_{\Gamma,S}(R) \leq Q \iff R \leq P(\mathsf{p}_{\Gamma,S})(Q).$$

Theorem. (Beck-Chevalley condition) Suppose that Γ is a context and S is a type in context Γ. Let $\sigma : \Delta \longrightarrow \Gamma$ be a context map. Then for $\phi \in P(\Gamma.S)$

(a) $P(\sigma)(\forall_{\Gamma,S}(\phi)) \equiv \forall_{\Delta,\mathrm{Ty}(\sigma)(S)}(P(q_{\Delta,\Gamma,S}(\sigma)(\phi)))$

(b) $P(\sigma)(\exists_{\Gamma,S}(\phi)) \equiv \exists_{\Delta,\mathrm{Ty}(\sigma)(S)}(P(q_{\Delta,\Gamma,S}(\sigma)(\phi)))$

References

N. Gambino and P. Aczel. The generalized type-theoretic interpretation of constructive set theory. *Journal of Symbolic Logic* 71(2006), 67 – 103.

J. Cartmell. Generalized algebraic theories and contextual categories. *Annals of Pure and Applied Logic,* 32(1986), 209 – 243.

T. Coquand, P. Dybjer, E. Palmgren and A. Setzer. *Type-theoretic foundation of constructive mathematics.* Notes distributed at TYPES Summer School, Göteborg, August 2005.

P.T. Johnstone. *Sketches of an Elephant: A Topos Theory Compendium.* Vol 1 and 2. Oxford University Press 2002.

M. Hofmann. Syntax and semantics of dependent types. In: *Semantics and Logics of Computation.* Cambridge University Press 1997.

References (cont.)

M.E. Maietti and G. Sambin. Towards a minimalist foundation for constructive mathematics. In: *From Sets and Types to Topology and Analysis* (eds. L. Crosilla and P. Schuster) Oxford University Press 2005, 91 – 114.

M.E. Maietti. Modular correspondence between dependent type theories and categories including pretopoi and topoi. *Mathematical Structures in Computer Science* 15(2005), 1089 –1149.

M. Makkai. First-order logic with dependent sorts, with applications to category theory. Preprint 1995. Available from the author's webpages.

M. Makkai. Towards a categorical foundation of mathematics. In: *Logic Colloquium '95* (eds. J.A. Makowsky and E.V. Ravve) Lecture Notes in Logic, vol. 11, Association for Symbolic Logic 1998, 153 – 190.

M. Makkai. The theory of abstract sets based on first-order logic with depend types. Preprint 2013. Available from the author's webpages.

References (cont.)

P. Martin-Löf (1984). *Intuitionistic Type Theory.* Notes by Giovanni Sambin of a series of lectures given in Padua, June 1980. Bibliopolis.

E. Palmgren and S. Vickers. Partial Horn logic and cartesian categories. *Annals of Pure and Applied Logic,* 145(2007), 314 – 355.

E. Palmgren. Constructivist and structuralist foundations: Bishop's and Lawvere's theories of sets. *Annals of Pure and Applied Logic,* 163(2012), 1384 – 1399.

O. Wilander. *On Constructive Sets and Partial Structures.* PhD dissertation in Mathematics. Uppsala University 2011.