Harper 4

Spse $a, b : A, c : Eq_A(a,b) \gg c$ type
If $a : A \gg P \in C[a, a, * / a, b, c]$
and $M \in Eq_A(M_1, M_2)$
then $P[M/a] \in C[M_1, M_2, M / a, b, c]$

Equality is least reflexive relation

Pf $M = *$ if $\in Eq_A(M_1, M_2)$ and $M_1 \doteq M_2 \in A$

$$C[M_1, M_2, M] \doteq C[M_1, M_1, *]$$

$$P[M_1]$$

Fact The only element of $Eq_A(M_1, M_2)$ is $*$.

if $M \in Eq_A(M_1, M_2)$ then $M \doteq * \in \ldots$

$* \in Eq_{Eq_A(M_1, M_2)}(M, *)$.

"uniqueness of equality $\begin{Bmatrix} \text{evidence} \\ \text{proof} \end{Bmatrix}$"

Examining formalisms ("abstract implementation") for type theory

- inductively defined by rules for deriving judgments of the following forms

$$\begin{matrix} \Gamma \vdash M : A & \Gamma \vdash M \equiv M' : A \\ \Gamma \vdash A \text{ type} & \Gamma \vdash A \equiv A' \end{matrix} \Bigg\} \begin{matrix} \text{definitional equivalence} \\ \text{"calculational" equivalence} \end{matrix}$$

$$\frac{\Gamma, a : A_1 \vdash M : A_2}{\Gamma \vdash \lambda a : A_1 . M : a : A_1 \to A_2}$$

$$\frac{\Gamma \vdash M_1 : a : A_2 \to A}{\Gamma \vdash M_1 M_2 : A[M_2/a]}$$

$$\frac{\Gamma, a : A_1 \vdash M_1 : A_2 \qquad \Gamma \vdash M_2 : A_1}{\Gamma \vdash (\lambda a : A_1 . M_1) M_2 \equiv M_1[M_2/a] : A_2[M_2/a]}$$ "β rule" for $\to$

$$\langle M_1, M_2 \rangle \cdot 1 \equiv M_1 \qquad \langle M \cdot 1, M \cdot 2 \rangle \equiv M$$
$$\cdot 2 \equiv M_2' \qquad \eta$$
$$\beta$$

Criteria

1. decidability

→ 2. canonicity (jumping-off point)

Idea formalism is "just" a means of deriving truths (about computations)

props-as-types / proofs-as-programs
derivations

extraction

1. define notion of erasure of formal terms, types
$|M|$ $|A|$

e.g.) $|\lambda a{:}A.\,M| = \lambda a.|M|$

(more or less trivial)

2. Soundness (w/ erasure)

if $\Gamma \vdash M : A$ then $|\Gamma| \gg |M| \in |A|$

then $|\Gamma| \gg |M| \in |A|$

if $\Gamma \vdash M_1 = M_2$ then $|\Gamma| \gg |M_1| \doteq |M_2| \in |A|$

$\Gamma \vdash A$ type $|\Gamma| \gg |A|$ type

$\Gamma \vdash A_1 \equiv A_2$ $|\Gamma| \gg |A_1| \doteq |A_2|$

"fundamental theorem"
of logical relations

(typical relations)
if you will

Corollary If $\overbrace{M : Bool}^{\text{closed (no } \Gamma)}$ then

$|M| \Downarrow true$ or $|M| \Downarrow false$

Cor logical consistentcy

(true doesn't eval to false)

Soundness tells us that we can do proof extraction, that proofs have a computational content

→ would like to internalize computation as definitional equivalence

Canonicity If $M : Bool$, then $M \equiv true : \overset{Bool}{\cancel{ok}}$
or $M \equiv false : Bool$

(separate thm.) M-L 1972(-75)

"Internal completeness" property

How to formalize equality (not just calculation)

eg) $a : Nat \gg a + a \doteq 2 \times a \in Nat$

$\lambda a. a + a \doteq \lambda a. 2 \times a \in Nat \to Nat$

1. ETT - equality reflection
uniqueness

expressive - not decidable

2. ITT (Martin-Löf)

formalizes Id_A as the least reflexive

$$\frac{\Gamma \vdash A\ \mathrm{type} \quad \Gamma \vdash M_1 : A \quad \Gamma \vdash M_2 : A}{\Gamma \vdash \mathrm{Id}_A(M_1, M_2)\ \mathrm{type}}$$

$$\frac{\Gamma \vdash M : A}{\Gamma \vdash \mathrm{refl}_A(M) : \mathrm{Id}_A(M, M)}$$

$$\left\{ \frac{\begin{array}{l}\Gamma, a, b : A, c : \mathrm{Id}_A(a,b) \vdash C\ \mathrm{type} \\ \Gamma \vdash P : \mathrm{Id}_A(M,N) \\ \Gamma, a : A \vdash Q : C[a, a, \mathrm{refl}_A(M)]\end{array}}{\Gamma \vdash J(a,b,c.C)(a.Q)(P) : C[M,N,P]} \right.$$

$$J(a,b,c.C)(a.Q)(\mathrm{refl}_A(M)) \equiv Q[M/a]$$

<u>Soundness</u>

Idea
$$\left.\begin{array}{l} |\mathrm{Id}_A(M,N)| \triangleq \mathrm{Eq}_{|A|}(|M|, |N|) \\ |\mathrm{refl}_A(M)| \triangleq * \\ |J \ldots| \triangleq |Q|[|M|/a] \end{array}\right\}$$ ITT has a computational meaning

Thm of Martin-Löf

✓ $a : Nat \vdash \underline{\ \vee\ } : Id_{Nat}(a + a, 2 \times a)$ ✓

not the case:

✗ $Id_{Nat \to Nat}(\lambda a. a+a, \lambda. 2 \times a)$!

not definitionally equal $\not\equiv$

failure of function extensionality!

<u>What to do?</u>

A common approach is to add an axiom (??)

$$\text{FunExt} : (a : A_1 \to Id_{A_2}(ap(M, a), ap(N, a))) \to Id_{a:A_1 \to A_2}(M, N))$$

$$\text{FunExt}(H) : Id_{a:A_1 \to A_2}(M, N)$$

↳ premise

Weakness is strength (Strength in weakness)
(Hofmann + Streicher)

ITT cannot prove (internally) that there is only one identification!

"groupoid model"

group model w/ evidence
binary relation

Semantically OK (FunExt)
but it $\boxed{\text{ruins canonicity}}$

$$J\,(a,b,c.\,C)\,(a.\,Q)\,(\mathrm{FunExt}(H)) \equiv \underline{??}$$

YET it is valid under erasure.

$$|\mathrm{FunExt}(H)| \triangleq *.$$

way out using OTT
Altenkirch, McBride

Idea along similar lines called $\boxed{\text{univalence}}$
by Voevodsky

Motivation: it is common to informally
"identify structures up to isomorphism"

eg) $A \times B \cong B \times A$

by swap is a bijection

what does it mean to identify these?

Add axiom (univalence axiom) UA

$$\underbrace{UA(\text{"swap"})}_{\substack{\text{data}\\\text{matters}}} : \mathrm{Id}_{\mathcal{U}}(A \times B,\ B \times A)$$

$\mathcal{U}$ ← universe (a type of types)

For specific choices of A, B
there can be many such isomorphisms.

eg. $Bool \simeq Bool$

by: 1) id
2) ~~swap~~ not

1. Id is a mechanism for interchangeability
stipulate that equivalence types
are identical and therefore exchangeable
(don't want to manually put swap
everywhere in code)

Voev. Simplicial sets model validates this

ruins canonicity:

1. J ___ (Fun Ext) $\equiv$?

2. J ___ $(UA(\epsilon)) \equiv$?

case 1. has a computational interpretation

case 2. what would be a computational
interpretation? (of univalence)

it ought to exist (want sprinkle in swaps when you need them)

Answer: CCH 11 18
AFH 17, 18 } Coquand
Angeli Favonio Harper

Roberto Di Cosmo – Isomorphism of types