Computational Higher Type Theory and Practice

Robert Harper

Carlo Angiuli, Evan Cavallo, Favonia, Yue Niu, Jon Sterling

MURI Meeting March 6, 2020

Acknowledgements

Thanks to many, including

- Collaborators: Guillaume Brunerie, Dan Licata, Anders Mörtberg, Todd Wilson.
- Colleagues: Jeremy Avigad, Steve Awodey, Marc Bezem, Ulrik Buchholtz, Thierry Coquand, Simon Huber.

Supported by AFOSR MURI FA9550-15-1-0053 and AFOSR FA9550-19-1-0216, Tristan Nguyen, PM.

A Reminiscence

From: **Vladimir Voevodsky** vladimir@ias.edu
Subject: a visit
Date: December 21, 2009 at 17:49
To: rwh@cs.cmu.edu
Cc: Vladimir Aleksandrovich Voevodsky vladimir@ias.edu

Dear Robert,

over the last few months I have been developing some ideas which I originally started to think about in 2005/6 and which I referred to, rather vaguely, as "homotopy lambda calculus".

The larger goal of what I am trying to do is to develop new foundations for mathematics which would be formally equi-consistent with Zermelo Fraenkel theory and at the same time would better reflect the realities of mathematical thinking.

On the more concrete level many of my ideas can be expressed as a construction of a new class of models of LF (I understand LF here as defined in "The groupoid interpretation of type theory" where contexts and signatures are not distinguished). These models satisfy some strong axioms including an appropriately defined "equivalence axiom" which creates members of the identity types from "weak equivalences" between types. Another important fact is that the equivalence classes of models of this type form an ordered set (class) i.e. for any two models which are not equivalent one is strictly stronger (satisfies more axioms) than another.

I have talked about these ideas to some German people (including Hofmann and Streicher) and would very much like to talk about it with people at CMU.
Would it be possible to arrange for me to visit CMU some time in a near future and if so then when would be a better time?

Very best,
Vladimir.

Homotopy Type Theory

Steve's and Vladimir's work inspired many developments in type theory!

- Types as higher-dimensional spaces, identifications as paths.
- Univalence principle: paths in the universe are equivalences.
- Higher Inductive types: specified by generators and more generators.

Book HoTT [2013] axiomatized these ideas in ITT:

- Univalence: $\mathsf{idToEquiv} : \mathsf{Id}_{\mathcal{U}}(A, B) \simeq \mathsf{Equiv}(A, B)$.
- Higher Inductives: Spheres, truncations, suspensions, ...

Initial Explorations

Dan Licata: Dependently Typed Programming with Domain-Specific Logics, February 2011.

- Chapter 7: 2-Dimensional Directed Type Theory.
- Higher structure at the judgmental level: $\Gamma \vdash \alpha :: M \Rightarrow M' : A$.
- Early development of cubical structure of types.

Directed case will be discussed Sunday!

Initial Explorations

Favonia: Higher-Dimensional Types in the Mechanization of Homotopy Type Theory, February 2017.

- Mechanizing homotopy theory in HoTT using Agda.
- Rezk: “a most excellent proof” of Blakers-Massey.
- Covering spaces, Seifert-van Kampen, cohomology theory.

Honored with the 2017 SCS Dissertation Award.

And is now on faculty at University of Minnesota.

Computational Content of HoTT

Book HoTT has rich mathematical content, but what is its computational content?

- $\mathsf{J}_{a,b,c.C}[a.Q](\mathsf{refl}_A(M)) \equiv Q[M/a]$.
- $\mathsf{J}_{a,b,c.C}[a.Q](\mathsf{ua}(A,B,E)) \equiv \mathbf{???}$.
- $\mathsf{J}_{a,b,c.C}[a.Q](\mathsf{loop}) \equiv \mathbf{???}$.

Basic conflict: type-dependent vs. type-independent identifications.

- J works independently of type (hence only reflexivity).
- Yet coercion is type sensitive (send functions to functions, etc.)

The Rise of Cubical Thought

Judgmental account of paths within and among types.

- Types merely internalize prior judgmental structure.
- Univalence and HIT's provide paths.
- Kan conditions ensure enough paths.

The Rise of Cubical Thought

Judgmental account of paths within and among types.

- Types merely internalize prior judgmental structure.
- Univalence and HIT's provide paths.
- Kan conditions ensure enough paths.

Cubical structure emerges naturally.

- Bezem, Coquand, Huber: face maps, degeneracies.
- Awodey; Licata, Brunerie; Coquand; Angiuli, Favonia, and H: diagonals (Cartesian cubes).
- Cohen, Coquand, Huber, Mörtberg: de Morgan algebra.

Cartesian Computational Cubical Type Theory

Carlo Angiuli: Computational Semantics of Cartesian Cubical Type Theory, September 2019.

- Cubes given by structural dimension context $\Psi = x_1, \ldots, x_n$.
- Cubical PL: $M \longmapsto_\Psi M'$, $M\ \mathsf{val}_\Psi$ (no coherence yet).
- Types specify how programs behave, including types themselves.
- Novel algorithmic treatment of univalence and Kan operations.

Thm: If $M \in \mathsf{Bool}$, then M evaluates to true or false.

Honored with the 2019 SCS Dissertation Award, nominated for ACM Award.

(And we hope he'll be gone next year!)

CCCTT Overview

Judgments express exact equality of types and terms at any dimension:

- Type equality: $\Gamma \gg A \doteq A'$ type $[\Psi]$.
- Member equality: $\Gamma \gg M \doteq M' \in A$ $[\Psi]$.

Defined in terms of evaluation behavior, subject to coherence requirements wrt dimension substitution.

- Extensional behavior for functions.
- Based on computability relations (cf. current work on computability families).

Kan composition admits diagonal constraints:

$$\mathsf{hcom}(M; \ldots x = y \hookrightarrow z.N \ldots)$$

CCCTT Overview

Universes are defined using induction-recursion, as in NuPRL.

- Univalence types $\mathsf{V}_x(A, B, E)$: extend a type line by an equivalence.
- Composition types $\mathsf{fcom}^{r \rightsquigarrow r'}(A; \vec{T})$, where $r \neq r'$.

Inductive Jdentity types $\mathsf{Id}_A(M, N)$.

- Least reflexive family.
- J computes only on reflexivity.

Path types internalize lines.

- $\mathsf{Path}_{x.A}(M, N)$: lines in A between $M \in A\langle 0/x \rangle$ and $N \in A\langle 1/x \rangle$.
- Equivalent to identity, supports J only up to a path.

Cartesian Cubical Relational Parametricity

Evan Cavallo: Higher Inductive Types and Parametricity in Cubical Type Theory, December 2020 (expected).

- Formulation of a rich class of higher inductive types in CCCTT.
- Internalized, proof-relevant form of parametricity.
- Application to path structure of HIT's.

Reynolds's Parametricity

Parametricity is central to PL theory:

- Uniformity of polymorphic functions (universal type quantification).
- Representation independence for abstract types (existential).

Exploit relational action of type constructors to derive equations.

- $R \to S \subseteq (A \to B)^2$ defined by $(R \to S)(f, f')$ iff $R(x, x')$ implies $S(f(x), f'(x'))$.
- $\forall X.R \subseteq (\forall X.A)^2$ quantifies over candidates aka similarities.

Thm: If $M : A$, then $M \doteq M \in A$ (obeys parametricity).

Famously, if $M : \forall X.X \to X$, then $M = \Lambda X.\lambda x : X.x$.
(Idea: fix x_0, pick $R(x, y)$ to mean $x = y = x_0$.)

Internalizing and Generalizing Parametricity

Reynolds's parametricity is an external property of a language.

- Takes place in an ambient theory of (evidence-free) relations.
- Derives (evidence-free) relations.

Bernardy, et al. developed internal parametricity in DTT.

- Relations are expressed within the type theory (as types).
- Can we extend this to the higher-dimensional case?

Cubical Parametricity Overview

Per Reynolds, if $F \in X : \mathcal{U} \to X \to X$, then it should act on relations between types.

Analogous to action of F on equivalences between types:

- Types vary in path dimensions to express equivalence.
- Types vary in bridge dimensions to express similarity.

Bridge dimensions $\Phi = u_1, \ldots, u_n$ index relational invariance:

- $\Gamma \gg A \doteq A'$ type $[\Phi \mid \Psi]$.
- $\Gamma \gg M \doteq M' \in A\ [\Phi \mid \Psi]$.

Bridges merely span two elements of disparate types.

- Express their "similarity" for deriving invariance properties.
- No notion of coercion along bridges.

Cubical Parametricity Overview

In cubical type theory functions respect paths:

$$\mathsf{Path}_{x.A \to B}(F, F') \simeq a : A \to \mathsf{Path}_{x.B}(F(a), F'(a'))$$

In parametric cubical type theory they should respect bridges:

$$\mathsf{Bridge}_{u.A \to B}(F, F') \simeq$$

$$a_0 : A\langle 0/u\rangle \to a_1 : A\langle 1/u\rangle \to \mathsf{Bridge}_{u.A}(a_0, a_1) \to \mathsf{Bridge}_{u.B}(F(a_0), F(a_1))$$

To achieve this, bridge dimensions are substructural, a la BCH cubes: no contraction.

Cubical Parametricity Overview

Bridge types internal bridges:

- $\mathsf{Bridge}_{u.A}(M, N)$ analogous to $\mathsf{Path}_{x.A}(M, N)$.
- May take end points, but no coercion!

Gel types internalize relations:

- $\mathsf{Gel}_u(A, B, R)$ analogous to $\mathsf{V}_x(A, B, E)$.
- Elements include evidence that end points are related by R.

Relativity: bridges in the universe are relations:

$$\mathsf{Bridge}_{\mathcal{U}}(A, B) \simeq (A \times B) \to \mathcal{U}.$$

Formalisms and Implementation

Two roles for formalisms:

- Provide convenient access to the truth in an implementation.
- Provide axiomatic freedom of multiple interpretations.

But what is a convenient formalism for CCCTT?

- Book HoTT works, but is not computationally adequate.
- RedPRL: turned out to be very inconvenient (*pace* NuPRL).
- RedTT: current candidate for implementation.

(Cf. cubical Agda for de Morgan cubes.)

A Formalism for RedTT

Formal judgments use a consolidated dimension context Ψ:

- Dimension variables: $x : \mathbb{I}$.
- Dimension constraints: $r = r' : \mathbb{I}$.
- Entailment: $\Psi \vdash r = r' : \mathbb{I}$.

Formation and equality in a context:

- $\Psi \mid \Gamma \vdash A \ \mathsf{type}_i$, type formation at universe level i.
- $\Psi \mid \Gamma \vdash A = A' \ \mathsf{type}_i$, silent equality of types.
- $\Psi \mid \Gamma \vdash M : A$, member of a type.
- $\Psi \mid \Gamma \vdash M = M' : A$, silent equality of members.

A Formalism for RedTT

Typing respects silent equality, silently:

$$\frac{\Psi \mid \Gamma \vdash M : A \qquad \Psi \mid \Gamma \vdash A = A' \ \mathsf{type}_i}{\Psi \mid \Gamma \vdash M : A'}\ \textsc{Eq}$$

Typing judgments track faces/diagonals:

$$\frac{\Psi \vdash r, r' : \mathbb{I} \qquad \Psi, x \mid \Gamma \vdash A \ \mathsf{type}_i \qquad \Psi \mid \Gamma \vdash M : A\langle r/x\rangle}{\Psi \mid \Gamma \vdash \mathsf{coe}_{x.A}^{r \rightsquigarrow r'}(M) : A\langle r'/x\rangle\ [r = r' \hookrightarrow M]}\ \textsc{Coe}$$

(See Carlo's talk for more details and demonstration.)

Theory of Implementation

Jon Sterling: Objective Metatheory of Cubical Type Theories, proposal expected Spring 2020.

- RedTT relies on decidability of silent equality of open terms.
- How to develop a decision method?

Decide equality using normalization by evaluation.

- Type-directed, hereditary head normalization.
- Account for dimensional equalities as well as $\beta\eta$-like principles.

Objective Metatheory

Syntax-invariant meta-theory based on algebraic formulation of type theory.

- Equational logical framework avoids handling "raw" terms.
- Formalism presents initial model.

Glueing: generalize computability to be proof-relevant:

- Evidence for computability lies over equivalence class of terms.
- Emphasize structure over property.
- Direct read-off of canonical forms.

Related and Ongoing Work

- ABCFHL: Orton-Pitts topos-theoretic interpretation of CCTT.
- Sterling, Angiuli, Gratzer: XTT, "thin" cubes for extensional function equality.
- Cavallo, Mörtberg, Swan: relating Cartesian to de Morgan cubical type theories.
- Angiuli, Cavallo, Favonia, Licata, Sterling, Voysey, and H: implementing RedTT.
- Angiuli, Sterling: general glueing for type theories; normalization for CCTT.
- Cavallo and H: modality for bridge-independence.
- Niu and H: types for complexity properties of programs.
- Sterling and H: abstraction for program modules via glueing.

Agenda

Today's talks:

- Evan Cavallo: higher-dimensional parametricity.
- Carlo Angiuli: RedTT formalism and demonstration.
- Jon Sterling: objective metatheory of formalisms.

(Dan, Matt, and Mitchell will speak Sunday.)

A Few Self-References (Others by Reference!)

Carlo Angiuli, Kuen-Bang Hou (Favonia), and Robert Harper. Cartesian cubical computational type theory: Constructive reasoning with paths and equalities. In Dan Ghica and Achim Jung, editors, *27th EACSL Annual Conference on Computer Science Logic (CSL 2018)*, volume 119 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 6:1–6:17, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-088-0. doi: 10.4230/LIPIcs.CSL.2018.6.

Carlo Angiuli, Guillaume Brunerie, Thierry Coquand, Kuen-Bang Hou (Favonia), Robert Harper, and Daniel R. Licata. Syntax and models of cartesian cubical type theory. Preprint, February 2019. URL `https://github.com/dlicata335/cart-cube`.

Evan Cavallo and Robert Harper. Higher inductive types in cubical computational type theory. *PACMPL*, 3(POPL):1:1–1:27, 2019. doi: 10.1145/3290314.

Evan Cavallo and Robert Harper. Internal parametricity for cubical type theory. In Maribel Fernández and Anca Muscholl, editors, *28th EACSL Annual Conference on Computer Science Logic, CSL 2020, January 13-16, 2020, Barcelona, Spain*, volume 152 of *LIPIcs*, pages 13:1–13:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi: 10.4230/LIPIcs.CSL.2020.13.

Jonathan Sterling, Carlo Angiuli, and Daniel Gratzer. Cubical syntax for reflection-free extensional equality. In Herman Geuvers, editor, *Proceedings of the 4th International Conference on Formal Structures for Computation and Deduction (FSCD 2019)*, volume 131 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 31:1–31:25. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-107-8. doi: 10.4230/LIPIcs.FSCD.2019.31. URL `http://drops.dagstuhl.de/opus/volltexte/2019/10538`.