Voevodsky’s Vision for Univalent Mathematics

$h$-level 0 The Mathematics of Cantor
- Sets and structured sets

$h$-level 1 The Mathematics of Grothendieck
- Groupoids and structured groupoids
- In particular the theory of categories

$h$-level $\infty$ “Higher” Mathematics
- The study of structured homotopy types

Problem
How can we describe structures on homotopy types without recourse to a “strict” equality?
The Current State of Affairs

- Solutions in some special cases are known:
  - **Voevodsky**  Contractibility, equivalences, ...
  - **Shulman**  $\infty$-idempotents
  - **Rijke**  $\infty$-equivalence relations

- Long standing approach to the problem:
  - ▶ Construct some notion of *semi-simplicial type*
  - ▶ Use this to internalize the theory of $(\infty, 1)$-categories
  - ▶ Reduce other coherence problems to this case

- There are many other kinds of higher structures:
  - ▶ $E_n$-spaces, ring spectra, homotopy Lie algebras, ...
  - ▶ $(\infty, n)$-categories, $\infty$-double categories, ...
  - ▶ Even if these can be reduced to simplicial methods, will this be an efficient way to describe them?
  - ▶ Can we describe a natural class of higher structures *directly*?
In this talk ...

- Adapt Baez and Dolan’s operadic method of describing coherent algebraic objects to type theory
- Give an elementary definition of *cartesian polynomial monad*
- Special cases of this definition are
  1. \((\infty, 1)\)-operad
  2. \((\infty, 1)\)-category
  3. \(\infty\)-groupoid
- There is a corresponding elementary definition of an *algebra*
- Special cases of this definition are
  1. \(A_\infty\)-types, \(E_\infty\)-types, etc
  2. Type-valued diagrams on \((\infty, 1)\)-categories
  3. Corollary: simplicial types are definable in MLTT with coinduction.
Formalization

Where are we in terms of formalization?

- The formalization of the definition of monad given here is complete.  
  https://github.com/ericfinster/higher-alg
- Hence so are any of the definitions which are special cases:  
  $\infty$-operad, $\infty$-category, $\infty$-groupoid, ...
- The definition of algebra relies on a construction which is not yet  
  completely formalized (though it is sketched ...)
- Hence the complete definition of simplicial type is not yet finished.
- The “on paper” definition of algebra, however, is completely  
  transparent. I do not expect any difficulties in finishing it other than  
  the fact that it is somewhat long.
Polynomials as Multi-sorted Signatures

Definition

Fix a type $I$ of sorts. A polynomial over $I$ is the data of

1. A family of operations
   \[ \text{Op} : I \to \text{Type} \]

2. For each operation, a family of sorted parameters
   \[ \text{Param} : \{i : I\}(f : \text{Op } i) \to I \to \text{Type} \]

- For $i : I$, an element $f : \text{Op } i$ represents an operation whose output sort is $i$.
- For $f : \text{Op } i$ and $j : I$, an element $p : \text{Param } f j$ represents an input parameter of sort $j$. 
Representations of Operations

- We can think of our polynomial as a collection of *typed operation symbols*, which we might denote, for example, by
  \[ f(j, k, l) : i \]

- We can depict such an operation graphically as a corolla:

- However, we specifically allow for higher homotopy both in the operations and the parameters
Trees

A polynomial $P : \text{Poly } I$ generates an associated type of trees.

Definition

The inductive family $\text{Tr } P : I \rightarrow \text{Type}$ has constructors:

- $\text{lf} : (i : I) \rightarrow \text{Tr } P i$
- $\text{nd} : \{i : I\} \rightarrow (f : \text{Op } Pi) \rightarrow (\phi : (j : J)(p : \text{Param } f j) \rightarrow \text{Tr } P j) \rightarrow \text{Tr } P i$

We can represent trees both geometrically and algebraically

$$k(h(g(i, j), g(i, h(l)), l)) : j$$
Leaves and Nodes

For a tree $w : \text{Tr } P i$, we will need its type of leaves and type of nodes.

**Leaves**

\[
\text{Leaf} : \{i : I\} (w : \text{Tr } i) \rightarrow I \rightarrow \text{Type}
\]
\[
\text{Leaf} (\text{lf}_i j) := i = j
\]
\[
\text{Leaf} (\text{nd}(f, \phi)) j := \sum_{k:i} \sum_{p:\text{Param } f k} \text{Leaf} (\phi k p) j
\]

**Nodes**

\[
\text{Node} : \{i : I\} (w : \text{Tr } i)(j : I) \rightarrow \text{Op } j \rightarrow \text{Type}
\]
\[
\text{Node} (\text{lf}_i j g) := \bot
\]
\[
\text{Node} (\text{nd}(f, \phi)) j g := (i, f) = (j, g) \sqcup \sum_{k:i} \sum_{p:\text{Param } f k} \text{Node} (\phi k p) j g
\]
Frames

Definition

Let \( P : \text{Poly} \ I \) be a polynomial \( w : \text{Tr} \ P \ i \) a tree and \( f : \text{Op} \ P \ i \) an operation. A \textit{frame} from \( w \) to \( f \) is a family of equivalences

\[
j : I \rightarrow \text{Leaf} \ w \ j \cong \text{Param} \ P \ f \ j
\]
Polynomial Relations

Definition

A polynomial relation for $P$ is a type family

$$R : \{i : I\}(f : \text{Op } i)(w : \text{Tr } i)(\alpha : \text{Frame } w f) \rightarrow \text{Type}$$
The Slice of a Polynomial by a Relation

Definition

Let $P : \text{Poly } I$ and let $R$ be a relation on $P$. The \textit{slice of } $P$ \textit{by } $R$, denoted $P // R$, is the polynomial with sorts $\Sigma I \text{ Op}$ defined as follows:

$$\text{Op}(P // M)(i, f) := \sum_{(w: \text{Tr } P i)} \sum_{(\alpha: \text{Frame } w f)} R f w \alpha$$

$$\text{Param}(P // M)(w, \alpha, r)(j, g) := \text{Node } w g$$
Trees in the Slice Polynomial

[Diagram of trees and algebraic structures]
Flattening

\[
\text{flattening: } \{i : I\}\{f : \text{Op } i\} \rightarrow \text{Tr}(P//R)(i, f) \rightarrow \text{Tr } P\ i \\
\text{flattening-frm: } \{i : I\}\{f : \text{Op } i\}(pd : \text{Tr}(P//R)(i, f)) \\
\quad \rightarrow \text{Frame(flatten } pd)\ f \\
\text{bd-frm: } \{i : I\}\{f : \text{Op } i\}(pd : \text{Tr}(P//R)(i, f)) \\
\quad \rightarrow (j : I)(g : \text{Op } j) \rightarrow \text{Leaf}(P//R)\ pd\ g \simeq \text{Node } P\ (\text{flatten } pd)g
\]
Polynomial Magmas

Polynomials serve as our notion of higher signature. Following ideas from the categorical approach to universal algebra, we are going to encode the relations or axioms of our structure using a monadic multiplication on $P$.

**Definition**

Let $P$ be a polynomial with sorts in $I$. A polynomial magma $M$ over $P$ is

1. A function $\mu : \{i : I\} \to \text{Tr } P \, i \to \text{Op } P \, i$
2. A function $\mu_{\text{frm}} : \{i : I\}(w : \text{Tr } P \, i) \to \text{Frame } w \, (\mu \, w)$

Notice that a magma $M$ determines a polynomial relation on $P$ by using the identity type:

$$MgmRel : \text{PolyMagma } P \to \text{PolyRel } P$$

$$MgmRel \, M \, f \, w \, \alpha := (\mu \, w, \mu_{\text{frm}} \, w) = (f, \alpha)$$
Polynomial Magmas (cont’d)

Using the graphical notation we have developed, we can “picture” the multiplication $\mu$ as follows:

In algebraic notation, this corresponds to the relation

$$k(h(g(x, y)), g(u, h(v)), w) = f(x, y, u, v, w)$$
Coherent Relations

Furthermore, we can now interpret a pasting diagram $pd : \text{Tr}(P//M)(i, f)$ as a sequence of multiplications applied to subterms of flatten $pd$:

\[ \mu(g, h, h, g, k) = f \]

**But:** without further structure, there is simply no reason that this sequence of multiplications gives rise to the “obvious” relation

\[ \mu(g, h, h, g, k) = f \]
Coherent Relations

Furthermore, we can now interpret a pasting diagram $pd : \text{Tr}(P//M)(i, f)$ as a sequence of multiplications applied to subterms of flatten $pd$:

But: without further structure, there is simply no reason that this sequence of multiplications gives rise to the “obvious” relation

$$\mu(g, h, h, g, k) = f$$
**Subdivision Invariance**

**Definition**

Let $P$ be a polynomial and $R$ a relation on $P$. We say that $R$ is **subdivision invariant** if we are given a function.

$$
\psi : \{i : I\}\{f : \operatorname{Op} P \ i\} (pd : \operatorname{Tr}(P \ / \ R) (i, f))
\rightarrow R f (\text{flatten } pd) (\text{flatten-frm } pd)
$$

We write $\text{SubInvar}$ for the associated predicate on polynomial relations.

$$\text{SubInvar} : \text{PolyRel} P \rightarrow \text{Type}$$

$$\text{SubInvar} \ R \coloneqq \{i : I\}\{f : \operatorname{Op} P \ i\} (pd : \operatorname{Tr}(P \ / \ R) (i, f))
\rightarrow R f (\text{flatten } pd) (\text{flatten-frm } pd)$$
The Slice Magma

Observation

Let $P$ be a polynomial and $R$ a relation on $P$. Given a witness $\Psi$ that $R$ is subdivision invariant, the slice polynomial $P//R$ admits a magma structure given by

$$\mu(SlcMgm R) \, pd := ((\text{flatten} \, pd, \text{flatten-frm} \, pd), \Psi \, pd)$$

$$\mu_{frm}(SlcMgm R) \, pd := \text{bd-frm} \, pd$$
Example: Associativity

Let us see why, if a magma is subdivision invariant, then it is associative.

\[
\mu(\mu(g,h), \mu(h,g), k) = \mu(g,h,h,g,k)
\]
Example: Associativity

Let us see why, if a magma is subdivision invariant, then it is associative.

\[
\mu(\mu(g, h), \mu(h, g), k) = \mu(g, h, h, g, k)
\]
\[
\mu(g, h, \mu(h, g, k)) = \mu(g, h, h, g, k)
\]

Hence
\[
\mu(\mu(g, h), \mu(h, g), k) = \mu(g, h, \mu(h, g, k))
\]
Polynomial Monads

Let $P$ be a polynomial and $M$ a magma on $P$.

**Definition**

A *coherence structure* for $M$ consists of

1. A proof $\Psi : \text{SubInvar } M$
2. Coninductively, a coherence structure on $\text{SlcMgm } M \Psi$

**Definition**

A *polynomial monad* consists of

1. A polynomial $P : \text{Poly } I$
2. A magma $M : \text{PolyMagma } P$
3. A coherence structure $C$ for $M$
4. A proof that $M$ is *univalent*
Univalence for Monads

- For an operation \( f : \text{Op} i \) we define

\[
\text{Arity } f := \sum_{j : I} \text{Param } f j
\]

\[
\text{UnaryOp } M := \sum_{i : I} \sum_{f : \text{Op } i} \text{is-unary } f
\]

\[
\text{is-unary } f := \text{is-contr}(\text{Arity } f)
\]

\[
\text{id } i := \mu(\text{lf } i)
\]

- One can easily check (using \( \mu_{\text{frm}} \)) that \( \text{id } i \) is unary.

- We can think of a unary operation \( f : \text{Op } i \) as a “morphism”

\[
f : j \to i
\]

where \( j \) is the sort of its unique parameter.

- The multiplication \( \mu \) can now be used to define a composition operation

\[
\_ \circ \_ : \text{UnaryOp } \times \text{UnaryOp} \to \text{UnaryOp}
\]
Univalence for Monads (cont’d)

**Definition**

Let $M$ be a polynomial monad. A unary operation $f : j \to i$ is said to be an *isomorphism* if satisfies the bi-inverse property:

$$\text{is-iso } f := \sum_{g : i \to j} \sum_{h : i \to j} (f \circ g = \text{id}_i) \times (h \circ f = \text{id}_j)$$

Write $\text{Iso } M$ for the space of isomorphisms in $M$.

It is routine to check that for $i : I$, the operation $\text{id}_i$ is an isomorphism in this sense. Hence we have

$$\text{id-to-iso} : \{ij : I\} \to i = j \to \text{Iso } M$$

$$\text{id-to-iso}\{i\} \text{idp} = \text{id}_i$$

**Definition**

$M$ is said to be *univalent* if the above map is an equivalence.
Special Cases of Monads

- For a type $X : Type$ let
  \[
  \text{is-finite } X := \sum_{n : \mathbb{N}} \| X \simeq \text{Fin } n \|_{-1}
  \]

- Let $M$ be a polynomial monad. We define
  \[
  \text{is-\(\infty\)-operad } M := \{ i : I \}(f : \text{Op } i) \to \text{is-finite}(\text{Arity } f)
  \]
  \[
  \text{is-\(\infty\)-category } M := \{ i : I \}(f : \text{Op } i) \to \text{is-unary } f
  \]
  \[
  \text{is-\(\infty\)-groupoid } M := \text{is-\(\infty\)-category } M \times (f : \text{Op } i) \to \text{is-iso } f
  \]

- More special cases are possible:
  - A symmetric monoidal $\infty$-category is an $\infty$-operad with enough “universal” operations.
  - An $A_\infty$-type is an $\infty$-category for which the type $I$ is connected
  - etc ...
Future Directions

- Finish the definition of simplicial type
- Conjecture:
  \[ \infty\text{-groupoid} \simeq \text{Type} \]
- Loop spaces are grouplike \( A_\infty \)-types?
- Initial algebras and HIT’s
- Develop higher category theory

Thanks!