Abstracting away from cell complexes

Michael Shulman\textsuperscript{1} \quad Peter LeFanu Lumsdaine\textsuperscript{2}

\textsuperscript{1}University of San Diego

\textsuperscript{2}Stockholm University

March 12, 2016
Replacing big messy cell complexes with smaller and simpler but more abstract ones

Michael Shulman\textsuperscript{1} \quad Peter LeFanu Lumsdaine\textsuperscript{2}

\textsuperscript{1}University of San Diego

\textsuperscript{2}Stockholm University

March 12, 2016
Outline

1. Cell complexes
2. Presentations of 1-monads
3. Algebraic model category theory
4. Presentations of homotopical monads
5. More abstract cell complexes
6. Higher Inductive Types
Small simple cell complexes

- Spheres
- Tori
- Projective space
- Manifolds
- …
Big messy cell complexes, I

Postnikov towers

The $n^{\text{th}}$ Postnikov section of $X$ is obtained from $X$ by

- gluing on enough $(n + 2)$-cells to kill $\pi_{n+1}(X)$, then
- gluing on enough $(n + 3)$-cells to kill $\pi_{n+2}$ of the result, then
- gluing on enough $(n + 4)$-cells to kill $\pi_{n+3}$ of the result,
- and so on.

Note: Gluing on a $k$-cell is the same as taking a pushout
Big messy cell complexes, II

Localization

The localization of $X$ at a map $f : S \to T$ is obtained by:

- Replacing $f$ by a cofibration,
- Taking its pushout product with all the boundary inclusions $S^n \hookrightarrow D^{n+1},$
- For each resulting map $\hat{f}_n : A_n \to B_n$, taking one pushout

\[
\begin{array}{ccc}
A_n & \rightarrow & X \\
\downarrow & & \downarrow \\
B_n & \rightarrow & \text{for each map } A_n \to X,
\end{array}
\]

- Repeating the previous step, perhaps \textit{transfinitely often},
- \ldots until we’re done.
Big messy cell complexes, III

...and it doesn’t get any easier from there.
...and it doesn’t get any easier from there.

Can we package up this machinery better so we don’t have to think about it?
Outline

1. Cell complexes
2. Presentations of 1-monads
3. Algebraic model category theory
4. Presentations of homotopical monads
5. More abstract cell complexes
6. Higher Inductive Types
The 1-categorical case

The 1-categorical case


**Theorem (Kelly)**

Let \( A \) be a cocomplete category with two cocomplete factorization systems \((\mathcal{E}, \mathcal{M})\) and \((\mathcal{E}', \mathcal{M}')\), let \( A \) be \( \mathcal{E} \)- and \( \mathcal{E}' \)-cowellpowered, let \( S \) be a well-pointed endofunctor, and for some regular cardinal \( \alpha \) let \( S \) preserve the \( \mathcal{E} \)-tightness of \((\mathcal{M}, \alpha)\)-cones. Then \( S \)-Alg is constructively reflective in \( A \).
The 1-categorical case, really now

Theorem (Kelly?)

Let $C$ be a locally presentable category. Then:

- Every accessible endofunctor of $C$ generates an algebraically-free monad.
- Every small diagram of accessible monads on $C$ has an algebraic colimit.
The 1-categorical case, really now

Theorem (Kelly?)

Let $C$ be a locally presentable category. Then:

- Every accessible endofunctor of $C$ generates an algebraically-free monad.
- Every small diagram of accessible monads on $C$ has an algebraic colimit.

- What does this mean?
- What is it good for?
Review about monads

- Every monad $T$ has a category of algebras $T$-Alg.
- The forgetful functor $U_T : T$-Alg $\to \mathcal{C}$ has a left adjoint $F_T$.
- $T = U_T F_T$
Review about monads

- Every monad $T$ has a category of algebras $T$-Alg.
- The forgetful functor $U_T : T$-Alg $\to \mathcal{C}$ has a left adjoint $F_T$.
- $T = U_T F_T$
- The assignation $T \mapsto T$-Alg is a fully faithful embedding $\text{Monads}^{\text{op}} \hookrightarrow \text{Cat}_{/\mathcal{C}}$.

i.e. we have

$$\text{Monads}(T_1, T_2) \cong \text{Cat}_{/\mathcal{C}}(T_2\text{-Alg}, T_1\text{-Alg})$$

$$\left[\phi : T_1 \to T_2\right] \mapsto \left[(T_2X \to X) \mapsto (T_1X \xrightarrow{\phi} T_2X \to X)\right]$$
Review about monads

- Every monad \( T \) has a category of algebras \( T\)-Alg.
- The forgetful functor \( U_T : T\)-Alg \( \to \mathcal{C} \) has a left adjoint \( F_T \).
- \( T = U_T F_T \)
- The assignation \( T \mapsto T\)-Alg is a fully faithful embedding

\[
\text{Monads}^{\text{op}} \hookrightarrow \text{Cat}_{/\mathcal{C}}.
\]

i.e. we have

\[
\text{Monads}(T_1, T_2) \cong \text{Cat}_{/\mathcal{C}}(T_2\text{-Alg}, T_1\text{-Alg})
\]

\[
\begin{bmatrix}
T_1 & T_1 \eta_2 & T_1 T_2 & \text{act}_{G T_2} & T_2
\end{bmatrix}
\]

\[
\begin{bmatrix}
T_2\text{-Alg} & G & T_1\text{-Alg}
\end{bmatrix}
\]

\[
\begin{bmatrix}
U_{T_2} & U_{T_1}
\end{bmatrix}
\]

\[
\begin{bmatrix}
\mathcal{C}
\end{bmatrix}
\]
Definition

Every monad has an underlying endofunctor; this defines a functor

\[ \text{monads on } \mathcal{C} \longrightarrow \text{endofunctors on } \mathcal{C}. \]

A free monad on an endofunctor \( S \) is the value at \( S \) of a (partially defined) left adjoint to this:

\[ \text{Monads}(\overline{S}, T) \cong \text{Endofrs}(S, T) \]
### Algebraically-free monads

**Definition**

For an endofunctor $S$, an $S$-algebra is an object $X$ equipped with a map $SX \to X$.

**Definition**

A monad $\overline{S}$ is **algebraically-free** on $S$ if we have an equivalence of categories over $\mathcal{C}$:

$$\overline{S} \text{ monad-algebras } \xrightarrow{\sim} S \text{ endofunctor-algebras}$$

**Theorem (Kelly?)**

*Every algebraically-free monad is free, and the converse holds if $\mathcal{C}$ is locally small and complete.*
Examples

\( S(X) = A \)

- \( S \)-algebras are objects \( X \) with a map \( A \rightarrow X \).
- \( \bar{S}(X) = X + A \).
Examples

\( S(X) = A \)

- \( S \)-algebras are objects \( X \) with a map \( A \to X \).
- \( \bar{S}(X) = X + A \).

\( S(X) = X \times X \)

- \( S \)-algebras are “magmas”: objects \( X \) with a binary operation \( X \times X \to X \).
- \( \bar{S}(X) \) is the free magma on \( X \); in \textbf{Set} its elements are bracketed words \(((xy)z)(zy)\).
### Examples

<table>
<thead>
<tr>
<th><strong>S(X) = A</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• S-algebras are objects $X$ with a map $A \to X$.</td>
</tr>
<tr>
<td>• $\bar{S}(X) = X + A$.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>S(X) = X × X</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• S-algebras are “magmas”: objects $X$ with a binary operation $X \times X \to X$.</td>
</tr>
<tr>
<td>• $\bar{S}(X)$ is the free magma on $X$; in <strong>Set</strong> its elements are bracketed words $((xy)z)(zy)$.</td>
</tr>
</tbody>
</table>

**Note:** can be built as an infinite “cell complex”

$$X \to X \sqcup (X \times X) \to \cdots$$
Examples, II

\[ S(X) = X \times X + 1 \]

- \( S \)-algebras are “pointed magmas”.

\[ S(\emptyset) = N \]
Examples, II

\[ S(X) = X \times X + 1 \]

- \( S \)-algebras are “pointed magmas”.

\[ S(X) = A \times X + 1 \]

- \( \overline{S}(\emptyset) \) is the “list object” on \( A \), generated by \( \text{nil} : 1 \to X \) and \( \text{cons} : A \times X \to X \).
- If \( A = 1 \), then \( \overline{S}(\emptyset) = \mathbb{N} \).
Algebraically-free monads

Theorem (Kelly?)

*Every algebraically-free monad is free, and the converse holds if \( \mathcal{C} \) is locally small and complete.*

Proof.

If \( \overline{S} \) is algebraically-free on \( S \), then

\[
\text{Endofrs}(S, T) \cong \text{Cat}_c(T\text{-Alg}_{\text{monad}}, S\text{-Alg}_{\text{endofr}}) \\
\cong \text{Cat}_c(T\text{-Alg}_{\text{monad}}, \overline{S}\text{-Alg}_{\text{monad}}) \\
\cong \text{Monads}(\overline{S}, T)
\]
## Algebraically-free monads

<table>
<thead>
<tr>
<th>Theorem (Kelly?)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Every algebraically-free monad is free, and the converse holds if $\mathcal{C}$ is locally small and complete.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Proof.</th>
</tr>
</thead>
<tbody>
<tr>
<td>If $\mathcal{C}$ is loc sm and complete, $X \in \mathcal{C}$ has an endomorphism monad $\langle X, X \rangle = \text{Ran}_{X: 1 \to \mathcal{C}}(X : 1 \to \mathcal{C})$ such that</td>
</tr>
<tr>
<td>• For a monad $T$,</td>
</tr>
<tr>
<td>$\left( \text{T-algebra structures on } X \right) \leftrightarrow \left( \text{Monad maps } T \to \langle X, X \rangle \right)$.</td>
</tr>
<tr>
<td>• For an endofunctor $S$,</td>
</tr>
<tr>
<td>$\left( \text{S-algebra structures on } X \right) \leftrightarrow \left( \text{Endofr maps } S \to \langle X, X \rangle \right)$.</td>
</tr>
</tbody>
</table>
Algebraic colimits of monads

**Definition**

An **algebraic colimit** of a diagram $D : J \to \text{Monads}$ is a monad $T$ with an equivalence of categories over $\mathcal{C}$:

$$T\text{-Alg} \xrightarrow{\sim} \lim_{j \in J} D_j\text{-Alg}$$

This is a limit in $\text{Cat}/\mathcal{C}$, so it means that

$T$-algebra structures on $X$ $\iff$ compatible families of $D_j$-algebra structures on $X$.

**Theorem**

*Every algebraic colimit is a colimit in the category of monads, and the converse holds if $\mathcal{C}$ is locally small and complete.*
Example 0: coproducts

Example

The algebraically-initial monad has, as category of algebras, the terminal object of $\text{Cat}/C$, namely $C$ itself. Thus, it is the identity monad $\text{Id}$. 
Example 0: coproducts

Example

The algebraically-initial monad has, as category of algebras, the terminal object of $\text{Cat}/C$, namely $C$ itself. Thus, it is the identity monad $\text{Id}$.

Example

An algebra structure for the algebraic coproduct $T_1 \sqcup T_2$ consists of unrelated $T_1$-algebra and $T_2$-algebra structures.
Example 1: algebra

Recall a **semigroup** is a magma whose operation is associative.

1. Let $S_2(X) = X \times X$; so $S_2$-algebras are magmas and $S_2 X$ is the free magma on $X$.

2. Let $S_3(X) = X \times X \times X$, so $S_3$-algebras are sets equipped with a ternary operation.

3. A magma $X$ has two induced ternary operations $x(yz)$ and $(xy)z$. This yields two functors

$$S_2\text{-Alg} \Rightarrow S_3\text{-Alg}$$

whose **equalizer** is the category of semigroups.

4. Thus, the **algebraic coequalizer** of the corresponding two monad morphisms $S_3 \Rightarrow S_2$ is the monad for semigroups.
Presentations of monads

Definition

A **presentation** of a group (or other algebraic structure) is a coequalizer of maps between free groups:

\[ F\langle R \rangle \rightrightarrows F\langle X \rangle \rightarrow G \]
Presentations of monads

**Definition**

A *presentation* of a group (or other algebraic structure) is a coequalizer of maps between free groups:

\[ F\langle R \rangle \rightrightarrows F\langle X \rangle \to G \]

**Definition**

A *generalized presentation* of an object is an iterated colimit of diagrams of free objects.

**Definition**

A *generalized presentation* of a monad is an (algebraic) iterated colimit of diagrams of (algebraically-)free monads.
Example 2: localization

Let \( f : A \to B \) be a fixed morphism.

1. Let \( S_f(X) = \mathcal{C}(A, X) \cdot B \) and \( S_A(X) = \mathcal{C}(A, X) \cdot A \) and \( S_B(X) = \mathcal{C}(B, X) \cdot B \).

2. An \( S_f \)-algebra \( X \) is equipped with a map \( \mathcal{C}(A, X) \cdot B \to X \), or equivalently \( \mathcal{C}(A, X) \to \mathcal{C}(B, X) \).

3. An \( S_f \)-algebra has two \( S_A \)-algebra structures: "evaluation" and the composite \( \mathcal{C}(A, X) \cdot A \xrightarrow{f} \mathcal{C}(A, X) \cdot B \to X \).

4. These coincide iff the \( S_f \)-algebra structure \( \mathcal{C}(A, X) \to \mathcal{C}(B, X) \) is a right inverse to \( (\mathcal{C}(B, X) \to \mathcal{C}(A, X)) \).

5. Similarly, an \( S_f \)-algebra has two \( S_B \)-algebra structures, which coincide iff the \( S_f \)-algebra structure is a left inverse to \( (\mathcal{C}(B, X) \to \mathcal{C}(A, X)) \).
Example 2: localization

6 The joint equalizer of the two parallel pairs

\[
\begin{array}{c}
\overline{S}_A\text{-Alg} \\
\downarrow \\
\overline{S}_f\text{-Alg} \\
\downarrow \\
\overline{S}_B\text{-Alg}
\end{array}
\]

consists of \(X\) for which \((- \circ f) : C(B, X) \to C(A, X)\) has a (necc. unique) two-sided inverse — i.e. the \(f\)-local objects.

7 Thus, the joint algebraic coequalizer of the pairs

\[
\begin{array}{c}
\overline{S}_A \\
\downarrow \\
\overline{S}_f \\
\downarrow \\
\overline{S}_B
\end{array} \xrightarrow{f} \begin{array}{c}
\overline{S}_f \\
\overline{S}_B
\end{array}
\]

is the \(f\)-localization (the “free \(f\)-local object” monad).
Suppose instead of one morphism $f : A \rightarrow B$ we have a set of them, $\{f_i : A_i \rightarrow B_i\}_{i \in I}$.

1. We could generalize the construction of $L_f$ to $L\{f_i\}$.
2. Or we could simply take the algebraic coproduct $\coprod_{i \in I} L_{f_i}$, whose algebras are equipped with (unrelated) $L_{f_i}$-algebra structures, hence $f_i$-local for all $i$. 
Outline

1. Cell complexes
2. Presentations of 1-monads
3. Algebraic model category theory
4. Presentations of homotopical monads
5. More abstract cell complexes
6. Higher Inductive Types
Assumption

$C$ is a presheaf category with a right proper simplicial model structure in which the cofibrations are the monomorphisms.

(For most of what follows, this is much more than necessary.)

Examples:

- Simplicial sets
- The injective model structure on simplicial presheaves
- Any right proper localization of the latter
- Any locally presentable, locally cartesian closed $\infty$-category has such a presentation (Cisinski, Gepner–Kock)
Let $\mathcal{J}$ be the generating acyclic cofibrations. Copying only the first half of the localization construction, we obtain a monad $R$ whose algebras are algebraically fibrant: objects $X$ equipped with chosen lifts against all $\mathcal{J}$-maps.

**Theorem (Garner)**

Each unit map $\eta_X : X \to RX$ is an acyclic cofibration.

Thus, $R$ is a “fibrant replacement monad”.
More generally, we have a monad $R$ on $\mathcal{C} \rightarrow$ whose algebras are algebraic fibrations: maps $g : Y \rightarrow X$ equipped with chosen lifts against all $\mathcal{J}$-maps.

**Theorem (Garner)**

*For any $g : Y \rightarrow X$, the unit $\eta_g : g \rightarrow Rg$ looks like*

\[
\begin{array}{ccc}
Y & \longrightarrow & Eg \\
\downarrow & & \downarrow \\
X & \longrightarrow & X
\end{array}
\]

*and $Y \rightarrow Eg$ is an acyclic cofibration.*

(In fact, we have a whole “algebraic weak factorization system”: cofibrant replacement is also a comonad.)
Composing algebraic fibrations

**Theorem (Garner)**

If $g : Y \to X$ and $f : Z \to Y$ are algebraic fibrations, then $gf$ is naturally an algebraic fibration, and the square

\[
\begin{array}{ccc}
Z & \xrightarrow{f} & Y \\
\downarrow{gf} & & \downarrow{g} \\
X & = & X
\end{array}
\]

is a morphism of $\mathbb{R}$-algebras.
Composing algebraic fibrations

**Theorem (Garner)**

If $g : Y \to X$ and $f : Z \to Y$ are algebraic fibrations, then $gf$ is naturally an algebraic fibration, and the square

$$
\begin{array}{ccc}
Z & \xrightarrow{f} & Y \\
gf & \downarrow & \downarrow g \\
X & \cong & X \\
\end{array}
$$

is a morphism of $R$-algebras.

**Corollary**

If $f : Z \to Y$ is an algebraic fibration and $Y$ is algebraically fibrant, then $Z$ is algebraically fibrant and $f$ is a map of $R$-algebras.
Outline

1. Cell complexes
2. Presentations of 1-monads
3. Algebraic model category theory
4. Presentations of homotopical monads
5. More abstract cell complexes
6. Higher Inductive Types
Homotopical localization

We can repeat the 1-categorical construction, with mapping spaces and homotopies instead of hom-sets and equalities.

1. \( S_f(X) = \text{Map}(A, X) \otimes B \), with \( \text{Map}(A, X) \) the simplicial mapping space and \( \otimes \) the simplicial tensor.

2. Instead of the coequalizer of \( \overline{S_A} \rightrightarrows \overline{S_f} \) we take the pushout of

\[
\begin{array}{ccc}
2 \otimes S_1 & \longrightarrow & S_A \sqcup S_A \\
\downarrow & & \downarrow \\
\Delta^1 \otimes S_A & \longrightarrow & P_f
\end{array}
\]

A \( P_f \)-algebra is equipped with a right homotopy inverse to \(( - \circ f ) : \text{Map}(B, X) \to \text{Map}(A, X)\).
Instead of a joint coequalizer/pushout using a single copy of $S_f$, we take the coproduct of $P_f$ and its dual.

- The algebras are equipped with both a left and a right homotopy inverse to $(- \circ f)$, perhaps different.
- The existence of such is still equivalent to $(- \circ f)$ being a homotopy equivalence.
- The space of (left inverse, right inverse) pairs is contractible (if nonempty); the space of two-sided homotopy inverses is not.

Let’s call the resulting monad $\tilde{L}_f$. 
**Ensuring fibrancy**

**BUT!!!**

- $\tilde{L}_f$ does not produce fibrant objects.
- $\text{Map}(A, X)$ has the wrong homotopy type if $X$ is not fibrant! So non-fibrant $\tilde{L}_f$-algebras need not even have $f$-local homotopy type.
- In particular, $\tilde{L}_f X$ need not have $f$-local homotopy type.
Ensuring fibrancy

BUT!!!

- $\tilde{L}_f$ does not produce fibrant objects.
- $\text{Map}(A, X)$ has the wrong homotopy type if $X$ is not fibrant! So non-fibrant $\tilde{L}_f$-algebras need not even have $f$-local homotopy type.
- In particular, $\tilde{L}_f X$ need not have $f$-local homotopy type.

Simple answer

Take the algebraic coproduct $L_f = \tilde{L}_f \sqcup R$ with the fibrant replacement monad.

- $L_f$-algebras are $\tilde{L}_f$-algebras with an unrelated $R$-structure.
- In particular, they are fibrant, hence $f$-local; including $L_f X$. 
So $L_f X$ is $f$-local; but is it the $f$-localization?

- $L_f X$ is strictly initial in the category of “algebraically $f$-local”, algebraically fibrant objects under $X$ (and maps that preserve the algebraic structure).
- The $f$-localization is supposed to be homotopy initial in the category of $f$-local, fibrant objects under $X$ (and all maps).
Acyclically cofibrant algebras

**Theorem**

*If* $p : Y \to L_f X$ *is a (non-algebraic) fibration where* $Y$ *is* $f$-*local, then any section of* $p$ *over* $X$ *extends to a section of* $p$: 

\[
\begin{array}{ccc}
Y & \xrightarrow{p} & L_f X \\
\downarrow & & \downarrow \\
X & \rightarrow & L_f X
\end{array}
\quad \text{or} \quad
\begin{array}{ccc}
X & \rightarrow & Y \\
\downarrow & & \downarrow \\
L_f X & \rightarrow & L_f X
\end{array}
\]
Acyclically cofibrant algebras

If $p : Y \to L_fX$ is a (non-algebraic) fibration where $Y$ is $f$-local, then any section of $p$ over $X$ extends to a section of $p$.

Proof.

1. Since $Y$ is $f$-local and $p$ is a fibration, we can choose an $\tilde{L}_f$-algebra structure on $Y$ making $p$ an $\tilde{L}_f$-algebra map.

2. Choose an arbitrary $R$-algebra structure on $p$.

3. By composition, $Y$ becomes an $R$-algebra, hence an $L_f = (\tilde{L}_f \sqcup R)$-algebra, and $p$ an $L_f$-algebra map.

4. Since $L_fX$ is the free $L_f$-algebra on $X$, the given section $X \to Y$ induces an $L_f$-algebra map $L_fX \to Y$.

5. Since the composite $L_fX \to Y \to L_fX$ is again an $L_f$-algebra map, by uniqueness it is the identity.
Homotopy initiality

Theorem

$L_f X$ is the $f$-localization of $X$.

Proof.

1. We will show $\text{Map}(L_f X, Z) \to \text{Map}(X, Z)$ is an acyclic fibration for any $f$-local fibrant $Z$, by lifting in an arbitrary

\[
\begin{array}{ccc}
X & \xrightarrow{r} & Z^{\Delta^n} \\
\downarrow & & \downarrow \\
L_f X & \xrightarrow{s} & Z^{\partial\Delta^n}
\end{array}
\]

2. Since $L_f X$, $Z^{\partial\Delta^n}$, and $Z^{\Delta^n}$ are $f$-local, so is $s^*Z^{\Delta^n}$.

3. The map $s^*Z^{\Delta^n} \to L_f X$ is a fibration, so it has a section by the previous theorem.
Outline

1. Cell complexes
2. Presentations of 1-monads
3. Algebraic model category theory
4. Presentations of homotopical monads
5. More abstract cell complexes
6. Higher Inductive Types
Well-behaved monads

What is special about localization, and what isn’t? We need:

1. $T$-algebra structures can be rectified along fibrations.
2. $T$-algebra structures lift to path objects, etc.
Well-behaved monads

What is special about localization, and what isn’t? We need:

1. $T$-algebra structures can be rectified along fibrations.
2. $T$-algebra structures lift to path objects, etc.

These work for localization because the pushout

$$\begin{array}{ccc}
2 \otimes S_1 & \longrightarrow & S_A \sqcup S_A \\
\downarrow & & \downarrow \\
\Delta^1 \otimes S_A & \longrightarrow & P_f
\end{array}$$

is homotopically well-behaved.
**Definition**

- A **generating cofibration of monads** is a map

\[ S \times A \rightarrow S \times B \]

where $S$ is a well-behaved endofunctor and $A \rightarrow B$ is a generating cofibration.

- A **cell complex of monads** is a composite of pushouts of generating cofibrations.

- A **cell monad** $T$ is such that $\text{Id} \rightarrow T$ is a cell complex.
  (Recall $\text{Id}$ is the initial object in the category of monads.)
Homotopy theory of cell monads, I

“$T$-algebra structures lift to path objects, etc.”

**Theorem**

If $T$ is a cell monad, then the category $T$-$\text{Alg}_f$ of fibrant $T$-algebras is a *fibration category* à la Brown.

**Sketch of proof.**

Fibrations, weak equivalences, and limits are inherited from the base category. For factorizations, we argue inductively up the cell complex $\text{Id} \to T$. In the case of a generating cofibration, we factor downstairs and lift the $(S \times B)$-algebra structure, using the pushout product.
Theorem (in progress)

1. If $T_1 \rightarrow T_2$ is a cell complex, then $T_2$-$\text{Alg}_f \rightarrow T_1$-$\text{Alg}_f$ is almost a “fibration of fibration categories” à la Szumiło.

2. The pushouts in a cell complex yield homotopy pullbacks of fibration categories, hence (Szumiło) pullbacks of $\infty$-cats.

3. For a cell monad $T$,
   - the homotopy $\infty$-category of $T$-$\text{Alg}_f$ coincides with the algebras for the analogous $\infty$-monad;
   - any map of $\infty$-algebras can be presented by a $T$-algebra fibration; and hence
   - free $(T \sqcup R)$-algebras present free objects of this $\infty$-category.
Theorem (in progress)

1. If $T_1 \to T_2$ is a cell complex, then $T_2\text{-Alg}_f \to T_1\text{-Alg}_f$ is almost a “fibration of fibration categories” à la Szumiło.

2. The pushouts in a cell complex yield homotopy pullbacks of fibration categories, hence (Szumiło) pullbacks of ∞-cats.

3. For a cell monad $T$,
   - the homotopy ∞-category of $T\text{-Alg}_f$ coincides with the algebras for the analogous ∞-monad;
   - any map of ∞-algebras can be presented by a $T$-algebra fibration; and hence
   - free $(T \sqcup R)$-algebras present free objects of this ∞-category.

tl;dr: Cell monads are homotopically meaningful.
Replacing big messy cell complexes with smaller and simpler but more abstract ones

**Before**

The $f$-localization of a space is a cell complex with *transfinitely* many cells.

**After**

The $f$-localization *monad* is a cell monad with *four* cells.
Question

Why not work directly with presentations of ∞-monads?

In theory, we could. But reasons to use cell monads include:

1. We can leverage Kelly’s existing “package”.
2. Fits in a model-categorical framework, if we prefer that for other reasons.
3. Free \((T \Box R)\)-algebras have a stronger universal property than just ∞-freeness.
Question

Why not work directly with presentations of $\infty$-monads?

In theory, we could. But reasons to use cell monads include:

1. We can leverage Kelly’s existing “package”.
2. Fits in a model-categorical framework, if we prefer that for other reasons.
3. Free $(T \square R)$-algebras have a stronger universal property than just $\infty$-freeness. ... and it corresponds exactly to “induction principles” in type theory.
Outline

1. Cell complexes
2. Presentations of 1-monads
3. Algebraic model category theory
4. Presentations of homotopical monads
5. More abstract cell complexes
6. Higher Inductive Types
Higher Inductive Types

Slogan

Higher inductive types are a notation for describing families of cell monads that exist “uniformly” in all nice model categories.

Example

For \( f : A \to B \), the \( f \)-localization \( L_f X \) is generated by four “constructors” (monad cells):

1. \( \text{ext} : \forall (g : A \to L_f X), B \to L_f X \)
2. \( \text{rinv} : \forall (g : A \to L_f X), \forall (a : A), \text{ext}_g(f(a)) = g(a) \)
3. \( \text{ext}' : \forall (g : A \to L_f X), B \to L_f X \)
4. \( \text{linv} : \forall (h : B \to L_f X), \forall (b : B), \text{ext}'_{h \circ f}(b) = h(b) \)
Type theory is a system of notations for describing constructions that exist “uniformly” in all categories of a certain sort. Each “type constructor” represents a category-theoretic operation, e.g. $x : A \vdash \prod_{y : B(x)} C$ represents the right adjoint to pullback along a fibration $B \to A$.

Homotopy type theory is a system of notations for describing constructions that exist in all “categories with homotopy theory”.

Fine print: there are unresolved coherence questions in making this completely precise in generality.
Some history

- Traditional type theory has “ordinary” inductive types, which describe free monads and coproducts of monads.
- These include coproducts of spaces, but not other colimits.
- HITs were invented to describe other colimits in type theory.
- Only afterwards did we discover cell monads, when trying to model them categorically.
Recursion and induction for \( \mathbb{N} \)

**Recursion principle**

Given any \( X \) together with \( x_0 : 1 \to X \) and \( x_s : \mathbb{N} \times X \to X \), there is an \( f : \mathbb{N} \to X \) such that

\[
\begin{align*}
    f(0) &= x_0 \\
    f(n + 1) &= x_s(n, f(n))
\end{align*}
\]
Recursion and induction for $\mathbb{N}$

**Recursion principle**

Given any $X$ together with $x_0 : 1 \to X$ and $x_s : \mathbb{N} \times X \to X$, there is an $f : \mathbb{N} \to X$ such that

\[
\begin{align*}
f(0) &= x_0 \\
f(n + 1) &= x_s(n, f(n))
\end{align*}
\]

**Induction principle**

Given any property $P(n)$ such that $P(0)$ and $P(n) \to P(n + 1)$, we have $\forall n, P(n)$.
Type-theoretic induction for $\mathbb{N}$

Generalized induction principle

Given any map $p : C \to \mathbb{N}$ together with $x_0 : p^{-1}(0)$ and functions $x_{s,n} : p^{-1}(n) \to p^{-1}(n + 1)$ for all $n$, we have a section $f : \mathbb{N} \to C$ of $p$ such that

\[
\begin{align*}
  f(0) &= x_0 \\
  f(n + 1) &= x_{s,n}(f(n))
\end{align*}
\]
Type-theoretic induction for $\mathbb{N}$

**Generalized induction principle**

Given any map $p : C \to \mathbb{N}$ together with $x_0 : p^{-1}(0)$ and functions $x_{s,n} : p^{-1}(n) \to p^{-1}(n + 1)$ for all $n$, we have a section $f : \mathbb{N} \to C$ of $p$ such that

$$f(0) = x_0$$
$$f(n + 1) = x_{s,n}(f(n))$$

- $C = \mathbb{N} \times X$ gives recursion.
- $C = \{ n \in \mathbb{N} \mid P(n) \}$ gives induction.
Given any map \( p : C \to \mathbb{N} \) together with \( x_0 : 1 \to C \) and \( x_s : C \to C \) such that

\[
\begin{align*}
1 & \xrightarrow{x_0} C \\
\downarrow & \downarrow \psi \\
1 & \xrightarrow{0} \mathbb{N}
\end{align*}
\quad \text{and} \quad
\begin{align*}
C & \xrightarrow{x_s} C \\
\downarrow & \downarrow p \\
\mathbb{N} & \xrightarrow{1} \mathbb{N}
\end{align*}
\]

we have a section \( f : \mathbb{N} \to C \) of \( p \) such that

\[
\begin{align*}
f(0) &= x_0 \\
f(n + 1) &= x_s(f(n))
\end{align*}
\]
Generalized induction principle

Let $S(X) = 1 + X$. Then any $S$-algebra map $C \to \mathbb{N}$ has a section.

In type theory, these maps are restricted to be fibrations, corresponding to “dependent types”.
Endofunctor induction

**Generalized induction principle**

Let $S(X) = 1 + X$. Then any $S$-algebra map $C \to \mathbb{N}$ has a section.

In type theory, these maps are restricted to be fibrations, corresponding to “dependent types”.

**Ordinary inductive types**

For any well-behaved endofunctor $S$, there is an $S$-algebra $W_S$ such that any $S$-algebra fibration $C \to W_S$ has a section.

(It’s $(\overline{S} \sqcup R)(\emptyset)$.)
For any cell monad $T$, there is a $T$-algebra $W_T$ such that any $T$-algebra fibration $C \to W_T$ has a section.

Gives type-theoretic notations for:

- Small concrete cell complexes (spheres, tori, etc.)
- Homotopy colimits
- Localization
- Postnikov towers
- Spectrification
- And much more!