A Constructive Model of Directed Univalence in Bicubical Sets

Matthew Weaver
Princeton University

HoTTEST. April 16th, 2020

joint work with Dan Licata

Directed Type Theory

- Riehl-Shulman defines a type theory for ∞-categories with a model in bisimplicial sets
  1. Begin with HoTT
  2. Add Hom-types
  3. ∞-categories (Segal types) and univalent ∞-category (Rezk types) given internally as predicates on types
  4. Predicate isCov(B : A → U) for covariant discrete fibrations
  5. Cavallo, Riehl and Sattler have also (externally) defined the universe of covariant fibrations (the ∞-category of spaces and continuous functions) and shown *Directed Univalence:* $\mathrm{Hom}_{U_{Cov}}\, A\, B \simeq A \to B$

Constructive(?) Directed Type Theory

- Can we make this constructive?
  1. Begin with Cubical Type Theory
  2. Use a second cubical interval to define Hom-types
  3. Use LOPS to define universe of covariant fibrations and construct directed univalence internally...
  - ...unfortunately, directed univalence is a bit trickier than expected
  - ...fortunately, we can still make it work!

Let's see how far the techniques from cubical type theory get us!

Defining Bicubical Directed Type Theory

Cubical Type Theory
(in the style of Orton-Pitts)

1. Begin with a topos

2. Add an interval: $\mathbb{I}$

3. Specify gen. cofibrations for $\mathbb{I}$

4. Define filling problem for Kan fibrations

5. Define universe of Kan fibrations

6. Construct univalence

Directed Type Theory

1. Begin with Cubical Type Theory

2. Add an interval: $\mathbb{2}$

3. Specify gen. cofibrations for $\mathbb{2}$

4. Define filling problem for covariant fibrations

5. Define universe of covariant fibrations

6. Construct directed univalence

Defining Bicubical Directed Type Theory

Cubical Type Theory
(in the style of Orton-Pitts)

1. Begin with a topos

2. Add an interval: $\mathbb{I}$

3. Specify gen. cofibrations for $\mathbb{I}$

4. Define filling problem for Kan fibrations

5. Define universe of Kan fibrations

6. Construct univalence

Directed Type Theory

1. Begin with Cubical Type Theory

2. Add an interval: $\mathbb{2}$

3. Specify gen. cofibrations for $\mathbb{2}$

4. Define filling problem for covariant fibrations

5. Define universe of covariant fibrations

6. Construct directed univalence

Defining Bicubical Directed Type Theory

Cubical Type Theory
(in the style of Orton-Pitts)

2. Add an interval: $\mathbb{I}$

$$\frac{}{\mathbb{I} : \mathrm{Type}}$$

$$\frac{}{0_{\mathbb{I}} : \mathbb{I}} \qquad \frac{}{1_{\mathbb{I}} : \mathbb{I}}$$

e.g. generators for the Cartesian cubes, although any cubical type theory works

Directed Type Theory

2. Add an interval: $\mathbb{2}$

$$\frac{}{\mathbb{2} : \mathrm{Type}}$$

$$\frac{}{0_{\mathbb{2}} : \mathbb{2}} \qquad \frac{}{1_{\mathbb{2}} : \mathbb{2}}$$

$$\frac{x : \mathbb{2} \quad y : \mathbb{2}}{x \wedge y : \mathbb{2}} \qquad \frac{x : \mathbb{2} \quad y : \mathbb{2}}{x \vee y : \mathbb{2}}$$

and equations...

i.e. generators for the Dedekind cubes

The Directed Interval

- Why Dedekind cubes instead of Cartesian?
  $x \leq y := x = x \wedge y$

- We also add the following axioms:

  - $p : \mathbb{I} \to \mathbb{2}$ is constant $(\Pi\ x\ y : \mathbb{I},\ p\ x = p\ y)$

  - $p : \mathbb{2} \to \mathbb{2}$ is monotone $(\Pi\ x\ y : \mathbb{2}$, if $x \leq y$ then $p\ x \leq p\ y)$

Defining Bicubical Directed Type Theory

Cubical Type Theory
(in the style of Orton-Pitts)

1. Begin with a topos
2. Add an interval: $\mathbb{I}$
3. **Specify gen. cofibrations for $\mathbb{I}$**
4. Define filling problem for Kan fibrations
5. Define universe of Kan fibrations
6. Construct univalence

Directed Type Theory

1. Begin with Cubical Type Theory
2. Add an interval: $\mathbb{2}$
3. **Specify gen. cofibrations for $\mathbb{2}$**
4. Define filling problem for covariant fibrations
5. Define universe of covariant fibrations
6. Construct directed univalence

Defining Bicubical Directed Type Theory

Cubical Type Theory
(in the style of Orton-Pitts)

3. Specify gen. cofibrations for $\mathbb{I}$

$$\frac{}{\text{isCof} : \Omega \to \Omega}$$

Cof := $\Sigma\ \phi : \Omega\ .\ \text{isCof}\ \phi$

Cof closed under $\_\wedge\_$, $\_\vee\_$, $\bot$, $\top$

$$\frac{x : \mathbb{I} \quad y : \mathbb{I}}{\_ : \text{isCof}\ (x = y)}$$

$$\frac{\phi : \mathbb{I} \to \text{Cof}}{\_ : \text{isCof}\ (\Pi\ x : \mathbb{I}\ .\ \phi\ x)}$$

Directed Type Theory

3. Specify gen. cofibrations for $\mathbb{2}$

$$\frac{x : \mathbb{2} \quad y : \mathbb{2}}{\_ : \text{isCof}\ (x = y)}$$

$$\frac{\phi : \mathbb{2} \to \text{Cof}}{\_ : \text{isCof}\ (\Pi\ x : \mathbb{2}\ .\ \phi\ x)}$$

Defining Bicubical Directed Type Theory

Cubical Type Theory
(in the style of Orton-Pitts)

1. Begin with a topos

2. Add an interval: $\mathbb{I}$

3. Specify gen. cofibrations for $\mathbb{I}$

4. Define filling problem for Kan fibrations

5. Define universe of Kan fibrations

6. Construct univalence

Directed Type Theory

1. Begin with Cubical Type Theory

2. Add an interval: $\mathbb{2}$

3. Specify gen. cofibrations for $\mathbb{2}$

4. Define filling problem for covariant fibrations

5. Define universe of covariant fibrations

6. Construct directed univalence

Defining Bicubical Directed Type Theory

Cubical Type Theory

(in the style of Orton-Pitts)

4. Define filling problem for Kan fibrations

$\mathrm{hasCom} : (\mathbb{I} \to U) \to U$

$\mathrm{hasCom}\ A = \Pi\ i\ j : \mathbb{I}\ .$

$\quad \Pi\ \alpha : \mathrm{Cof}\ .$

$\quad \Pi\ t : (\Pi\ x : \mathbb{I}\ .\ \alpha \to A\ x)$

$\quad \Pi\ b : (A\ i)[\alpha \mapsto t\ i]\ .$

$\quad (A\ j)[\alpha \mapsto t\ j;\ i = j \mapsto b]$

$\mathrm{relCom} : (A : U) \to (A \to U) \to U$

$\mathrm{relCom}\ A\ B = \Pi\ p : \mathbb{I} \to A\ .$

$\quad \mathrm{hasCom}\ (B \circ p)$

Directed Type Theory

4. Define filling problem for covariant fibrations

$\mathrm{hasCov} : (\mathbb{2} \to U) \to U$

$\mathrm{hasCov}\ A = \Pi\ \alpha : \mathrm{Cof}\ .$

$\quad \Pi\ t : (\Pi\ x : \mathbb{2}\ .\ \alpha \to A\ x)$

$\quad \Pi\ b : (A\ \mathbb{0}_{\mathbb{2}})[\alpha \mapsto t\ \mathbb{0}_{\mathbb{2}}]\ .$

$\quad (A\ \mathbb{1}_{\mathbb{2}})[\alpha \mapsto t\ \mathbb{1}_{\mathbb{2}}]$

$\mathrm{relCov} : (A : U) \to (A \to U) \to U$

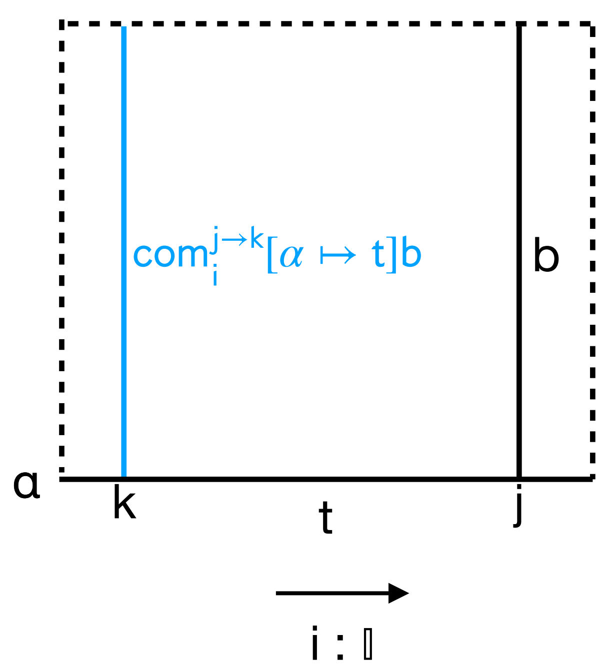
$\mathrm{relCov}\ A\ B = \Pi\ p : \mathbb{2} \to A\ .$

$\quad \mathrm{hasCov}\ (B \circ p)$

Defining Bicubical Directed Type Theory

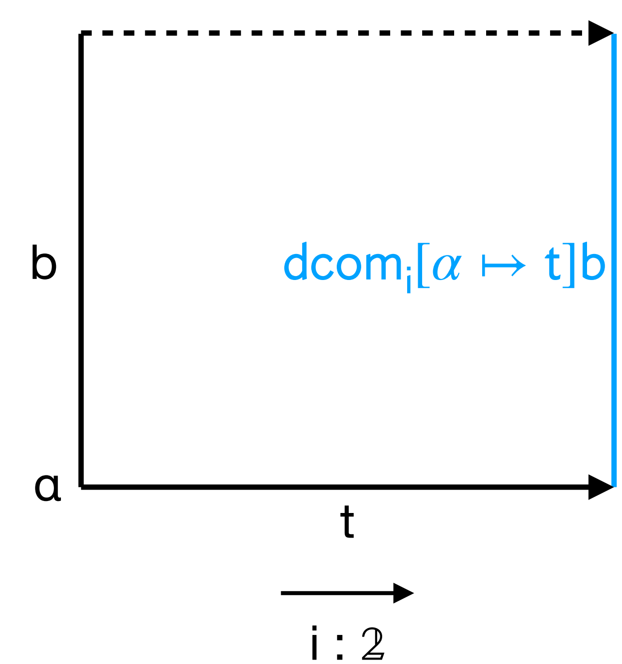
Cubical Type Theory
(in the style of Orton-Pitts)

4. Define filling problem for Kan fibrations



Directed Type Theory

4. Define filling problem for covariant fibrations



Defining Bicubical Directed Type Theory

Cubical Type Theory
(in the style of Orton-Pitts)

1. Begin with a topos

2. Add an interval: $\mathbb{I}$

3. Specify gen. cofibrations for $\mathbb{I}$

4. Define filling problem for Kan fibrations

5. Define universe of Kan fibrations

6. Construct univalence

Directed Type Theory

1. Begin with Cubical Type Theory

2. Add an interval: $\mathbb{2}$

3. Specify gen. cofibrations for $\mathbb{2}$

4. Define filling problem for covariant fibrations

5. Define universe of covariant fibrations

6. Construct directed univalence

Defining Bicubical Directed Type Theory

Cubical Type Theory
(in the style of Orton-Pitts)

5. Define universe of Kan fibrations

- U_{Kan} given by LOPS construction for relCom

Directed Type Theory

5. Define universe of covariant fibrations

- U_{Cov} given by LOPS construction for relCov
- **Lemma**: relCov is in U_{Kan}, so $El_{Cov} : U_{Cov} \to U_{Kan}$

Defining Bicubical Directed Type Theory

Cubical Type Theory
(in the style of Orton-Pitts)

1. Begin with a topos
2. Add an interval: $\mathbb{I}$
3. Specify gen. cofibrations for $\mathbb{I}$
4. Define filling problem for Kan fibrations
5. Define universe of Kan fibrations
6. **Construct univalence**

Directed Type Theory

1. Begin with Cubical Type Theory
2. Add an interval: $\mathbb{2}$
3. Specify gen. cofibrations for $\mathbb{2}$
4. Define filling problem for covariant fibrations
5. Define universe of covariant fibrations
6. **Construct directed univalence**

Defining Bicubical Directed Type Theory

Cubical Type Theory
(in the style of Orton-Pitts)

6. Construct univalence

- Key Idea: Glue type to attach equivalences to path structure

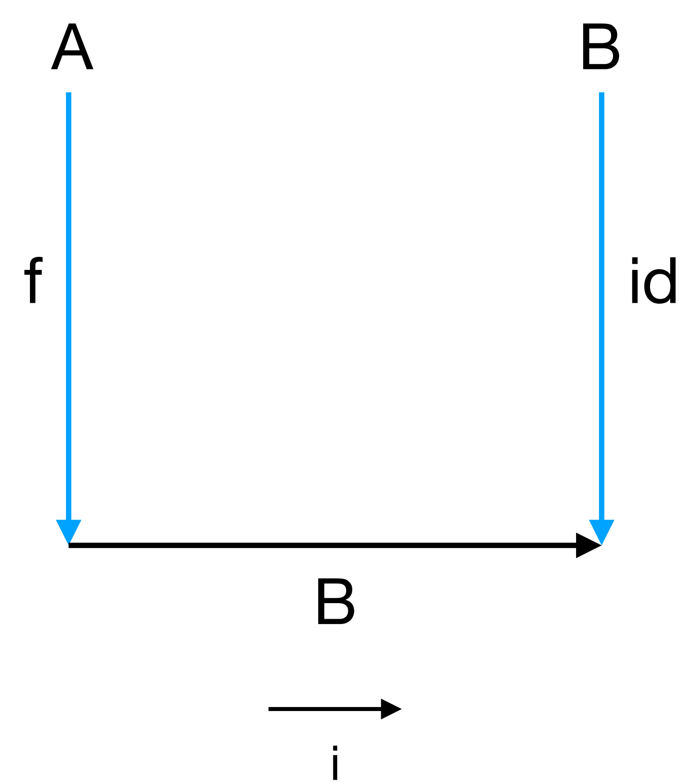
Directed Type Theory

6. Construct directed univalence

- Key Idea: Glue type to attach ***functions*** to morphism structure

Defining Directed Univalence

$$\text{dua } i\ A\ B\ f := \lambda\, i\, .\ \text{Glue}\ [\, i = \mathbb{0}_{\mathbb{2}} \mapsto (A\ ,\ f : A \to B)\ ,\ i = \mathbb{1}_{\mathbb{2}} \mapsto (B\ ,\ id)\,]\ B \quad : \text{Hom}_{U}\ A\ B$$

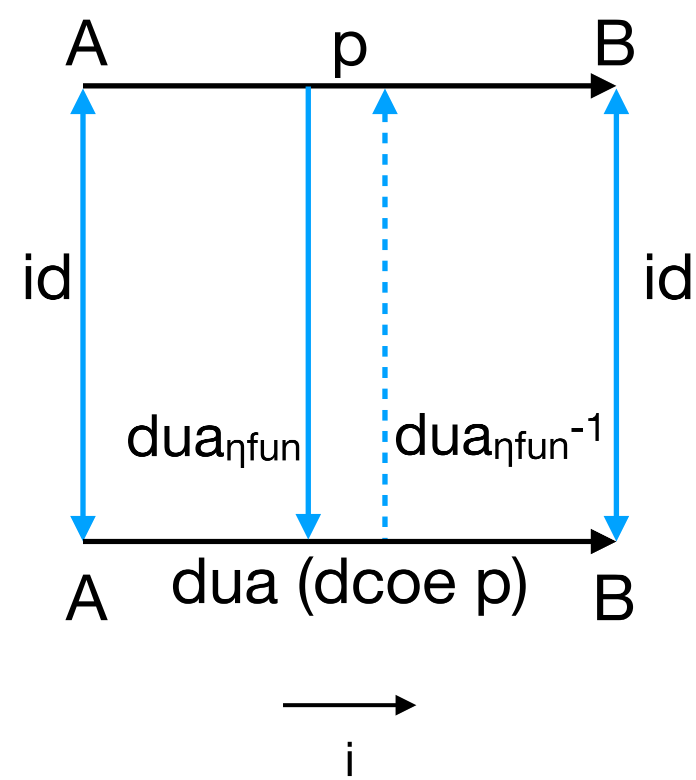


Naive Directed Univalence

- dua is Kan + covariant, and thus lands in U_{Cov}
- U_{Cov} itself is Kan
- Path univalence holds in U_{Cov}
- These allow us to define the following for U_{Cov}:
    - dcoe : (Hom A B) → (A → B)
    - dua : (A → B) → Hom A B
    - dua_{β} : Π f : A → B . Path f (dcoe (dua f))
    - $dua_{\eta fun}$: Π p : Hom A B . Π i : $\mathbb{2}$. p i → (dua (dcoe p)) i

Naive Directed Univalence

- We're thus left with the following picture:



- To complete directed univalence, we need $dua_{\eta fun}^{-1}$

What next?

- The proof in the bisimplicial model relies on the fact weak equivalences in the model are level-wise weak equivalences of simplicial sets

- Three potential model structures with level-wise weak equivalences...with three separate challenges.

Reedy	Dedekind cubes aren't Reedy
Projective	not all types we need are fibrant
Injective	not easily defined as cofibrantly generated

What next?

- The proof in the bisimplicial model relies on the fact weak equivalences in the model are level-wise weak equivalences of simplicial sets

- Three potential model structures with level-wise weak equivalences...with three separate challenges.

~~Reedy~~	Run into constructivity issue: degenerate cells not always decidable
Projective	not all types we need are fibrant
Injective	**not easily defined as cofibrantly generated**

Cobar and the Injective Model Structure

- Shulman classifies an injective fibrant object A : C → M as:
  - For every c in C, A c is fibrant in the underlying model structure M (i.e. A is object-wise fibrant)
  - A is equivalent to cobar(A)
- Coquand and Ruch internalize the cobar construction in a syntactic setting,
  - ...and constructively show weak equivalences are object-wise!
- Idea: use the internal cobar and prove all types A we care about are equivalent to cobar(A)
  - Spoiler: This works to finish the construction of directed univalence!
  - Fine Print: The formal connection between the internal version and Shulman's work has not yet been worked out.

Lex Operators and Stack Models of Type Theory

- Coquand and Ruch define a general framework for internalizing lex operators and defining models of type theory localized at them:
  - D : Type → Type is a strict lex endofunctor on types
  - η is a strict natural transformation Id → D
  - We can restrict the model to types A that are stacks, i.e. η_A is an equivalence from A to D A
- Note: Cobar is a lex operator

Lex Axioms

D is an endofunctor on U_{Kan}

$$\frac{}{D : U_{Kan} \to U_{Kan}}$$

$$\frac{f : A \to B}{D\ f : D\ A \to D\ B}$$

$$\frac{f : A \to B \quad g : B \to C}{D\ (g \circ f) = D\ g \circ D\ f}$$

$$\frac{}{D\ (\lambda\ x : A\ .\ x) = \lambda\ x : D\ A\ .\ x}$$

η is a natural transformation $Id_{U_{Kan}} \to D$

$$\frac{A : U_{Kan}}{\eta_A : A \to D\ A}$$

$$\frac{f : A \to B}{D\ f \circ \eta_A = \eta_B \circ f}$$

Lex Axioms

Additionally...

$$\frac{}{\eta\text{-Path}_A : \text{Path}_{D\,A \to D^2 A}\ (D\ \eta_A)\ (\eta_{D\,A})}$$

$$\frac{}{L : D\ U_{Kan} \to U_{Kan}}$$

$dD : (A \to U_{Kan}) \to (D\ A \to U_{Kan})$
$dD\ B := L \circ D\ B$

D is Lex

$$\frac{A : U_{Kan} \quad B : A \to U_{Kan}}{D\Sigma\text{-snd}_B : (x : D\ \Sigma A.B) \to dD\ B\ (D\ \text{fst}\ x)}$$

$$\frac{A : U_{Kan} \quad B : A \to U_{Kan}}{D\Sigma\text{-iso}_B : \text{isIso}\ (\lambda\ x \to D\ \text{fst}\ x\ ,\ D\Sigma\text{-snd}_B\ x)}$$

(i.e. $D\ \Sigma A.B \cong \Sigma D\ A.\ dD\ B$)

Closure Properties

- For an arbitrary lex endofunctor D...
  - ...if $B : A \to U$ is a family of stacks, then $\Pi\ A\ .\ B$ is a stack.
  - ...if A is a stack and $B : A \to U$ is a family of stacks, then $\Sigma\ A\ .\ B$ is a stack.
- For the other type formers we care about (i.e. Path, Hom, and Glue), we need specific information D and η.

Internalizing Cobar

- Main Idea: A natural transformation $A \to \mathrm{cobar}(B)$ corresponds to a homotopy coherent transformation $A \rightsquigarrow B$

- We define our internal cobar operator D by first defining a helper operator E.

- Intuition for E: For a type A and every X in $\square_{\mathrm{Ded}}$, an element of E A(X) is an element in a in A(X) along with a choice for the action of every substitution $Y \to X$ as an element in A(Y).

- Intuition for D: For every type A and X in $\square_{\mathrm{Ded}}$, an element of D A(X) is a choice of n elements of A for every chain of n composable morphisms into X that are weakly coherent with respect to the substitution action given by A.

Definition of E

- Given a bicubical set A, we define the bicubical set E A:
  - For X in $\square_{\mathrm{Ded}}$,

$$E\ A(X) := \Pi\ f : \mathrm{Hom}(Y, X))\ .\ A(Y)$$

  - For f : Hom(Y , X),

$$E\ A(f) := u : E\ A(X) \mapsto \lambda\ g : \mathrm{Hom}(Z\ ,\ Y).\ u(f \circ g)$$

- We also define a natural transformation α : Id → E:

$$\alpha_A\ (X) := a :\ A(X) \mapsto \lambda\ f : \mathrm{Hom}(Y\ ,\ X).\ af$$

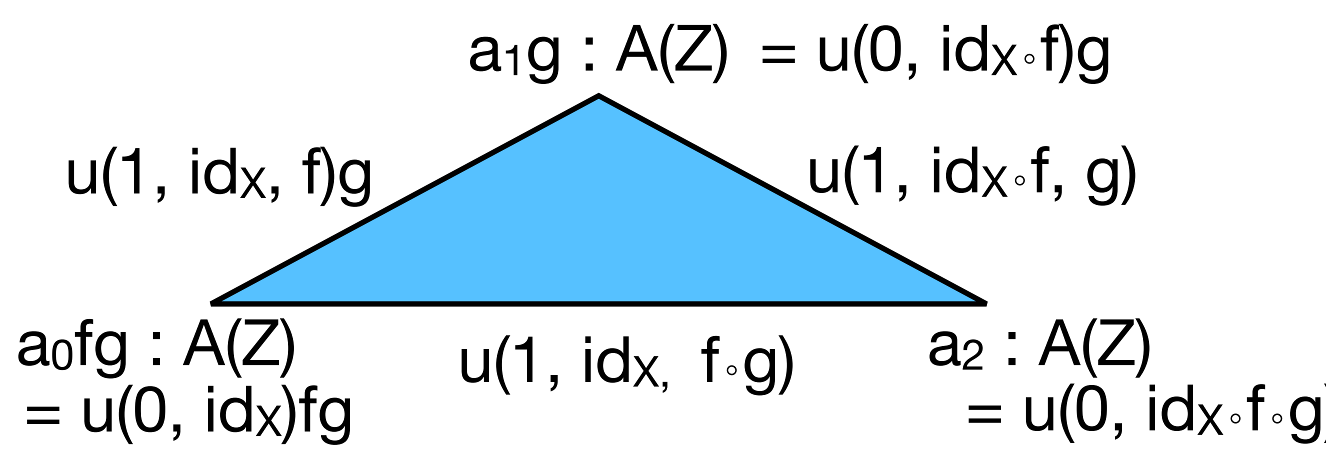
Definition of D

- Given a bicubical set A, we define the bicubical set D A:
  - For X in $\square_{\text{Ded}}$,

$$D\ A(X) := \Pi\ n : \mathbb{N}\ .\ \mathbb{I}^n \to E^{n+1}\ A(X)$$

    - The family u must additionally satisfy some conditions

So...what is D?

n	u : D A(X)
n = 0	$id_X \mapsto$ $a_0 : A(X)$
n = 1	$id_X, f : Hom(Y, X) \mapsto$ $a_0f : A(Y) = u(0, id_X)f$ —— $a_1 : A(Y) = u(0, id_X \circ f)$
n = 2	$id_X, f : Hom(Y, X), g : Hom(Z, Y) \mapsto$ 
⋮	

Definition of D

- diag : $\mathbb{I}^n[\mathbb{0} = i_1 \vee i_1 = i_2 \vee \ldots \vee i_{n-1} = i_n \vee i_n = \mathbb{1}] \to \mathbb{I}^{n-1}$ by forgetting i_1 on $\mathbb{0} = i_1$, i_k on $i_k = i_{k+1}$ and i_n on $i_n = \mathbb{1}$.
- Given a bicubical set A, we define the bicubical set D A:
  - For X in $\square_{Ded}$,

$$D\,A(X) := \Pi\ n : \mathbb{N}\ .\ \mathbb{I}^n \to E^{n+1}\ A(X)$$

    - The family u must additionally satisfy the following:
      - $u = \alpha \circ u \circ \text{diag}$ when $\mathbb{0} = i_1$
      - $u = E^k(\alpha) \circ u \circ \text{diag}$ when $i_k = i_{k+1}$
      - $u = E^n(\alpha) \circ u \circ \text{diag}$ when $i_n = \mathbb{1}$
  - For f : Hom(Y , X),

$$D\,A(f) := u : D\,A(X) \mapsto \lambda\ n, i_1, \ldots, i_n.\ E^{n+1}\ A(f)(u(n\ , i_1, \ldots, i_n))$$

- We also define a natural transformation η : Id $\to$ D:

$$\eta_A\ (X) := a :\ A(X) \mapsto \lambda\ n, i_1, \ldots, i_n.\ \alpha^{n+1}(a)$$

Additional Internal Axioms

$$\frac{}{\text{DPath-iso : Iso (D (Path}_A(a_0, a_1)) \text{ Path}_{D\ A}(\eta_A\ a_0, \eta_A\ a_1)}$$

$$\frac{}{\text{DHom-iso : Iso (D (Hom}_A(a_0, a_1)) \text{ Hom}_{D\ A}(\eta_A\ a_0, \eta_A\ a_1)}$$

$$\frac{}{\text{Ddua-iso : Iso (D (dua i A B f)) (dua i (D A) (D B) (D f))}}$$

(actual axioms specify how these isomorphisms compute in relation to η)

More Closure Properties

- For this specific lex endofunctor D...
  - If a type A is a stack, then for any terms a_0, a_1 in A, both $Path_A(a_0, a_1)$ and $Hom_A(a_0, a_1)$ are stacks.
  - If types A and B are stacks, then for any i : $\mathbb{2}$ and function f : A → B, dua i A B f is a stack.

Completing Directed Univalence

- The construction of directed univalence follows in two steps:

  1. Given a function f : A → B between stacks, if f is an object-wise equivalence of cubical sets then it is an equivalence of bicubical sets (Coquand and Ruch).

  2. The function $dua_{\eta fun}$ is an object-wise equivalence of cubical sets (modified from bisimplicial proof of Cavallo, Riehl and Sattler).

The Universe of Covariant Stacks

- Lastly, we define the universe that supports directed univalence:

$$U_{CovStack} := \Sigma\ A : U_{Cov}\ .\ \text{isStack}\ A$$

Our Results

- **Main Theorem:** There exists a constructive model of type theory in bicubical sets with a universe of fibrant types (U_{Kan}) and a universe of covariant fibrations ($U_{CovStack}$) such that:
  - $U_{CovStack}$ has a decode function into U_{Kan};
  - U_{Kan} is closed under Π, Σ, DPath, DHom and contains codes for smaller $U_{CovStack}$ and U_{Kan};
  - $U_{CovStack}$ is closed under Π (with fixed closed domain), Σ, DPath and DHom;
  - U_{Kan} and $U_{CovStack}$ are both path univalent;
  - $U_{CovStack}$ is morphism (directed path) univalent.

- Formalized in Agda!