Kripke-Joyal Semantics for Dependent Type Theory[1]

YaMCATS 23

Sina Hazratpour

February 2021

[1] joint work with Steve Awodey (CMU) and Nicola Gambino (Leeds)

Outline

Review of classical Beth-Kripke semantics

A **Kripke model** for an intuitionistic language $\mathcal{L}$ is a functor M of first order $\mathcal{L}$-structures over a poset K.

Beth-Kripke Semantics (I)

A **Kripke model** for an intuitionistic language $\mathcal{L}$ is a functor M of first order $\mathcal{L}$-structures over a poset K.

That is, to any $u \in K$ it associates an $\mathcal{L}$-structure M_u and to any $u \leq v$ it associates a homomorphism $f\colon M_u \to M_v$ (usually an embedding) in a functorial way.

A **Kripke model** for an intuitionistic language $\mathcal{L}$ is a functor M of first order $\mathcal{L}$-structures over a poset K.

That is, to any $u \in K$ it associates an $\mathcal{L}$-structure M_u and to any $u \leq v$ it associates a homomorphism $f\colon M_u \to M_v$ (usually an embedding) in a functorial way.

Elements of K are "worlds/stages" u,v,w, ...
$|M_u|$ is the set of possible individuals/objects (or our knowledge of them).

A **Kripke model** for an intuitionistic language $\mathcal{L}$ is a functor M of first order $\mathcal{L}$-structures over a poset K.

That is, to any $u \in K$ it associates an $\mathcal{L}$-structure M_u and to any $u \leq v$ it associates a homomorphism $f\colon M_u \to M_v$ (usually an embedding) in a functorial way.

Elements of K are "worlds/stages" u,v,w, ...
$|M_u|$ is the set of possible individuals/objects (or our knowledge of them).

Given such a model M, formulas of $\mathcal{L}$ can be interpreted at stages of K, relative to an assignment e:

Beth-Kripke Semantics (I)

- $w \Vdash R(t_1, \ldots, t_n)[e] \Leftrightarrow (t_1[e], \ldots, t_n[e]) \in M_w[R]$.
- $w \Vdash t_1 = t_2 \Leftrightarrow t_1[e] = t_2[e]$
- $w \nVdash \perp$ for every $w \in K$.
- $w \Vdash (\varphi \wedge \psi)[e] \Leftrightarrow w \Vdash \varphi[e]$ and $w \Vdash \psi[e]$.
- $w \Vdash (\varphi \vee \psi)[e] \Leftrightarrow w \Vdash \varphi[e]$ or $w \Vdash \psi[e]$.
- $w \Vdash (\varphi \to \psi)[e] \Leftrightarrow$ for all $u \geq w$, if $u \Vdash \varphi[e]$ then $u \Vdash \psi[e]$.
- $w \Vdash (\forall x)\varphi(x)[e] \Leftrightarrow$ for all $u \geq w$, for all $a \in |M_u|$, $w \Vdash (\varphi[a/x])[e]$.
- $w \Vdash (\exists x)\varphi(x)[e] \Leftrightarrow$ there exists $a \in |M_w|$ such that $w \Vdash (\varphi[a/x])[e]$.

Monotoncity/Upward persistency: For any Kripke model (K, M), and stages $w \leq u$, if $w \Vdash \varphi[e]$ then $u \Vdash \varphi[e]$.

Beth-Kripke Semantics (II)

Monotoncity/Upward persistency: For any Kripke model (K, M), and stages $w \leq u$, if $w \Vdash \varphi[e]$ then $u \Vdash \varphi[e]$.

We read this as "accumulating certainty": Once a fact is established it is never lost in the future.

Monotoncity/Upward persistency: For any Kripke model (K, M), and stages $w \leq u$, if $w \Vdash \varphi[e]$ then $u \Vdash \varphi[e]$.

We read this as "accumulating certainty": Once a fact is established it is never lost in the future.

Kripke's Soundness and Completeness Theorems establish that a sentence of $\mathcal{L}$ is provable in intuitionistic predicate logic if and only if it is forced at every stage of every Kripke model.

Monotoncity/Upward persistency: For any Kripke model (K, M), and stages $w \leq u$, if $w \Vdash \varphi[e]$ then $u \Vdash \varphi[e]$.

We read this as "accumulating certainty": Once a fact is established it is never lost in the future.

Kripke's Soundness and Completeness Theorems establish that a sentence of $\mathcal{L}$ is provable in intuitionistic predicate logic if and only if it is forced at every stage of every Kripke model.

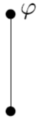
Beth-Kripke Semantics (II)

Using the soundness and completeness theorems,

- we can find simple Kripke models to show that some classically valid formulas are *not* intuitionistically valid. Two examples:

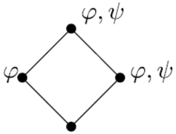
φ: an atomic sentence

$$w \nVdash \varphi \vee \neg\varphi$$



φ, ψ: atomic sentences

$$w \nVdash (\psi \to \varphi) \to (\neg\psi \vee \varphi)$$



- we can prove the disjunction and existence properties of the intuitionistic logic.

Review of classical Kripke-Joyal semantics for toposes

Classical Kripke-Joyal Semantics for toposes (review) (I)

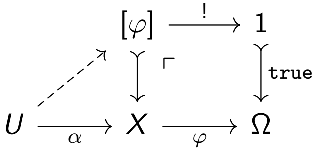
- **The Kripke–Joyal semantics** of a topos $\mathscr{E}$ gives an interpretation to formulas written in its higher order intuitionistic internal language $HoL(\Sigma_{\mathscr{E}})$.

- **The Kripke–Joyal semantics** of a topos $\mathscr{E}$ gives an interpretation to formulas written in its higher order intuitionistic internal language $HoL(\Sigma_{\mathscr{E}})$.
- The Kripke–Joyal semantics is in fact a higher order generalization of the well-known Kripke semantic for intuitionistic propositional logic.

Definition

Let $\mathscr{E}$ be an elementary topos. Given a formula $\varphi(x)$ with a free variable x of sort A in $HoL(\Sigma_{\mathscr{E}})$, and a generalized element $\alpha\colon U \to A$ in $\mathscr{E}$, we define

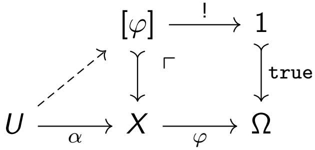
$$U \Vdash \varphi(\alpha) \quad \Leftrightarrow \quad \alpha \text{ factors through the subobject } [\varphi] \rightarrowtail A.$$



Definition

Let $\mathscr{E}$ be an elementary topos. Given a formula $\varphi(x)$ with a free variable x of sort A in $HoL(\Sigma_{\mathscr{E}})$, and a generalized element $\alpha\colon U \to A$ in $\mathscr{E}$, we define

$$U \Vdash \varphi(\alpha) \quad \Leftrightarrow \quad \alpha \text{ factors through the subobject } [\varphi] \rightarrowtail A.$$



- Call U the **stage** of forcing.
- Write $\mathscr{E} \Vdash \varphi$ if at every stage U and for every generalized element α, we have $U \Vdash \varphi(\alpha)$.

Classical Kripke-Joyal semantics for toposes (review) (III)

One can then show:

- $U \Vdash \top(\alpha)$.
- $U \Vdash \bot(\alpha)$ iff U in the initial object of $\mathscr{E}$.
- $U \Vdash (x = x')(\langle \alpha, \alpha' \rangle)$ iff $\alpha \colon U \to X$ and $\alpha' \colon U \to X$ are the same maps in $\mathscr{E}$.
- $U \Vdash (\varphi \wedge \psi)(\alpha)$ iff $U \Vdash \varphi(\alpha)$ and $U \Vdash \psi(\alpha)$.
- $U \Vdash (\varphi \vee \psi)(\alpha)$ iff there are jointly epimorphic arrows $p \colon V \to U$ and $q \colon W \to U$ such that $V \Vdash \varphi(\alpha \circ p)$ and $W \Vdash \varphi(\alpha \circ q)$.
- $U \Vdash (\varphi \Rightarrow \psi)(\alpha)$ iff for any arrow $f \colon V \to U$ such that $V \Vdash \varphi(\alpha \circ f)$ then $V \Vdash \psi(\alpha \circ f)$.
- $c \Vdash \neg\varphi(\alpha)$ iff for all maps $f \colon V \to U$ in $\mathscr{E}$, $V \not\Vdash \varphi(\alpha.f)$.
- $\vdots$

- Henceforth, $\mathscr{E} = \mathcal{P}\mathsf{Shv}(\mathcal{C}) = \mathcal{S}\mathsf{et}^{\mathcal{C}^{\mathrm{op}}}$.

Classical Kripke-Joyal semantics for presheaf toposes (review) (III)

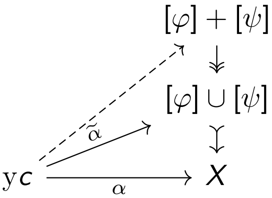
- Henceforth, $\mathscr{E} = \mathcal{P}\mathsf{Shv}(\mathcal{C}) = \mathcal{S}\mathsf{et}^{\mathcal{C}^{\mathrm{op}}}$.
- In the presheaf toposes, every presheaf is a colimit of representables.
- So it is enough to consider forcing statements $U \Vdash \varphi(\alpha)$ for representables $U = \mathrm{y}c$.

Classical Kripke-Joyal semantics for presheaf toposes (review) (III)

- Henceforth, $\mathscr{E} = \mathcal{P}\mathrm{Shv}(\mathcal{C}) = \mathcal{S}\mathrm{et}^{\mathcal{C}^{\mathrm{op}}}$.
- In the presheaf toposes, every presheaf is a colimit of representables.
- So it is enough to consider forcing statements $U \Vdash \varphi(\alpha)$ for representables $U = \mathrm{y}c$.

$$c \Vdash (\varphi \vee \psi)(\alpha) \Leftrightarrow c \Vdash \varphi(\alpha) \text{ or } c \Vdash \psi(\alpha)$$

Recall $\mathrm{y}c$ is projective & indecomposable.



Limitations of classical Kripke–Joyal semantics

- Bounded quantification. We shall overcome this by generalizing Kripke–Joyal semantics to dependent type theory with universes.
- Equality of terms is extensional and not "up to homotopy'. We will also generalize to homotopy type theory.

Kripke–Joyal semantics for dependent type theory

- Provability versus proof relevance: the classical Kripke–Joyal semantics is only concerned with the provability of a proposition; it is proof irrelevant. Kripke–Joyal semantics for dependent type theory includes the terms (proofs) in the forcing statements.

Kripke–Joyal semantics for dependent type theory

- Provability versus proof relevance: the classical Kripke–Joyal semantics is only concerned with the provability of a proposition; it is proof irrelevant. Kripke–Joyal semantics for dependent type theory includes the terms (proofs) in the forcing statements.
- We want a sound, formal and *(quasi-) mechanical* process to relate internal developments (Cohen et al., 2018), (Orton and Pitts, 2018), etc. with the diagrammatic developments (Gambino and Sattler, 2017), (Sattler, 2017), (Awodey, 2018), etc. found in the models of HoTT literature.

Kripke–Joyal semantics for dependent type theories

The setting = dependent type theory of presheaf toposes (I)

- We work in ZFC extended with an inaccessible cardinal number λ.

The setting = dependent type theory of presheaf toposes (I)

- We work in ZFC extended with an inaccessible cardinal number λ.
- We call the sets of size strictly less than λ-**small**. We write $\mathcal{S}\mathrm{et}_\lambda$ for the category of λ-small sets and $\mathcal{S}\mathrm{et}_\lambda^\bullet$ be the category of pointed λ-small sets.

The setting = dependent type theory of presheaf toposes (I)

- We work in ZFC extended with an inaccessible cardinal number λ.
- We call the sets of size strictly less than λ-**small**. We write $\mathcal{S}\mathrm{et}_\lambda$ for the category of λ-small sets and $\mathcal{S}\mathrm{et}_\lambda^\bullet$ be the category of pointed λ-small sets.
- We write $\mathcal{S}\mathrm{et}$ for the category of sets and functions.

- We work in ZFC extended with an inaccessible cardinal number λ.
- We call the sets of size strictly less than λ-**small**. We write $\mathcal{S}\mathsf{et}_\lambda$ for the category of λ-small sets and $\mathcal{S}\mathsf{et}^\bullet_\lambda$ be the category of pointed λ-small sets.
- We write $\mathcal{S}\mathsf{et}$ for the category of sets and functions.
- We fix a λ-small category $\mathcal{C}$. We define the Grothendieck topos of presheaves

$$\mathscr{E} = [\mathcal{C}^{\mathrm{op}}, \mathcal{S}\mathsf{et}]$$

We write

$$\mathrm{y} : \mathcal{C} \to \mathscr{E}$$

for the Yoneda embedding.

The setting = dependent type theory of presheaf toposes (I)

- We work in ZFC extended with an inaccessible cardinal number λ.
- We call the sets of size strictly less than λ-**small**. We write $\mathcal{S}\mathrm{et}_\lambda$ for the category of λ-small sets and $\mathcal{S}\mathrm{et}_\lambda^\bullet$ be the category of pointed λ-small sets.
- We write $\mathcal{S}\mathrm{et}$ for the category of sets and functions.
- We fix a λ-small category $\mathcal{C}$. We define the Grothendieck topos of presheaves

  $$\mathscr{E} = [\mathcal{C}^{\mathrm{op}}, \mathcal{S}\mathrm{et}]$$

  We write

  $$\mathrm{y} \colon \mathcal{C} \to \mathscr{E}$$

  for the Yoneda embedding.
- Call a family $p \colon X \to \Gamma$ in $\mathscr{E}$ λ-**small** whenever all the fibres $X(c) \to \Gamma(c)$ are small.

By the assumption of the existence of λ, the category $\mathscr{E}$ admits a **classifier for λ-small families**.

- We define $\mathsf{Type} \in \mathscr{E}$ by letting

$$\mathsf{Type}(c) = \{A\colon (\mathcal{C}/c)^{\mathrm{op}} \to \mathcal{S}\underset{\lambda}{\mathsf{et}} \mid A \text{ presheaf}\}. \qquad (1)$$

- Similarly, we define $\mathsf{Type}^{\bullet} \in \mathscr{E}$ by letting

$$\mathsf{Type}^{\bullet}(c) = \{A\colon (\mathcal{C}/c)^{\mathrm{op}} \to \mathcal{S}\mathsf{et}^{\bullet}_{\lambda} \mid A \text{ presheaf}\},$$

- Composing with the evident forgetful functor $\mathcal{S}\mathsf{et}^{\bullet}_{\lambda} \to \mathcal{S}\mathsf{et}_{\lambda}$ we get a natural transformation

$$\pi\colon \mathsf{Type}^{\bullet} \to \mathsf{Type}$$

- The map π is λ-small and it classifies **λ-small families** in $\mathscr{E}$.

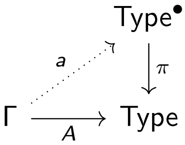
The setting = dependent type theory of presheaf toposes (III)

Following (Awodey, 2018), we get a CwF structure on $\mathscr{E}$ from the universe $\pi\colon \mathsf{Type}^\bullet \to \mathsf{Type}$:

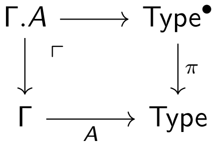
The **contexts** Γ are the objects of $\mathscr{E}$, and the substitutions $\sigma\colon \Delta \to \Gamma$ are arbitrary natural transformations.

The **types** A in context Γ are maps $A\colon \Gamma \to \mathsf{Type}$.

The **terms** $a\colon A$ in context Γ are maps $a\colon \Gamma \to \mathsf{Type}^\bullet$ with $\pi \circ a = A$.



The **context extension** of Γ by $A \in \mathrm{Ty}(\Gamma)$ is given by the pullback.



The setting = dependent type theory of presheaf toposes (IV)

Given $A \in \mathrm{Type}(\Gamma)$, for a map $\sigma\colon \Delta \to \Gamma$ we define $A(\sigma) \in \mathrm{Type}(\Delta)$ by letting

$$A(\sigma) = A \circ \sigma.$$

Given a term $a \in \mathrm{Term}(\Gamma, A)$, for a map $\sigma\colon \Delta \to \Gamma$, we define $a(\sigma) \in \mathrm{Term}(\Delta, A(\sigma))$ by letting

$$a(\sigma) = a \circ \sigma.$$

The setting = dependent type theory of presheaf toposes (IV)

Given $A \in \mathrm{Type}(\Gamma)$, for a map $\sigma\colon \Delta \to \Gamma$ we define $A(\sigma) \in \mathrm{Type}(\Delta)$ by letting

$$A(\sigma) = A \circ \sigma.$$

Given a term $a \in \mathrm{Term}(\Gamma, A)$, for a map $\sigma\colon \Delta \to \Gamma$, we define $a(\sigma) \in \mathrm{Term}(\Delta, A(\sigma))$ by letting

$$a(\sigma) = a \circ \sigma.$$

When $\Delta = \mathrm{y}c$: we write $\gamma\colon c \to \Gamma$ for $\sigma\colon \Delta \to \Gamma$. By the Yoneda lemma, $\gamma \in \Gamma(c)$.
For $A \in \mathrm{Type}(\Gamma)$ and $\gamma\colon \mathrm{y}c \to \Gamma$, the type $A(\gamma) \in \mathrm{Type}(c)$ is the composite

$$\mathrm{y}c \xrightarrow{\gamma} \Gamma \xrightarrow{A} \mathrm{Type}$$

Also, for $a \in \mathrm{Term}(c, A(\gamma))$ and $f\colon d \to c$ in $\mathcal{C}$, we have $a(\gamma f) \in \mathrm{Term}(d, A(\gamma.f))$.

Definition (Dependent Kripke–Joyal semantics– forcing terms)

For a context Γ,

Γ

Definition (Dependent Kripke–Joyal semantics– forcing terms)

For a context Γ, a type $\Gamma \vdash A\ \mathtt{Type}$,

$$\begin{array}{c} \Gamma.A \\ \big\downarrow{\scriptstyle p_A} \\ \Gamma \end{array}$$

Definition (Dependent Kripke–Joyal semantics– forcing terms)

For a context Γ, a type $\Gamma \vdash A\ \mathtt{Type}$, an object c of $\mathcal{C}$,

$$\begin{array}{ccc} & & \Gamma.A \\ & & \Big\downarrow{\scriptstyle p_A} \\ \mathrm{y}c & & \Gamma \end{array}$$

Definition (Dependent Kripke–Joyal semantics– forcing terms)

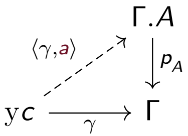
For a context Γ, a type $\Gamma \vdash A\ \mathtt{Type}$, an object c of $\mathcal{C}$, and a morphism $\gamma \colon \mathrm{y}c \to \Gamma$,

$$\begin{array}{ccc} & & \Gamma.A \\ & & \big\downarrow {\scriptstyle p_A} \\ \mathrm{y}c & \xrightarrow[\gamma]{} & \Gamma \end{array}$$

Definition (Dependent Kripke–Joyal semantics– forcing terms)

For a context Γ, a type $\Gamma \vdash A$ $\mathtt{Type}$, an object c of $\mathcal{C}$, and a morphism $\gamma \colon \mathrm{y}c \to \Gamma$,

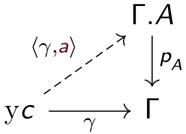
$$c \Vdash a : A(\gamma) \;\Leftrightarrow\; \text{there is a lift } \langle \gamma, a \rangle \text{ of } \gamma \text{ against } p_A \colon \Gamma.A \to \Gamma.$$



Definition (Dependent Kripke–Joyal semantics– forcing terms)

For a context Γ, a type $\Gamma \vdash A$ Type, an object c of $\mathcal{C}$, and a morphism $\gamma \colon \mathrm{y}c \to \Gamma$,

$$c \Vdash a : A(\gamma) \;\Leftrightarrow\; \text{there is a lift } \langle\gamma, a\rangle \text{ of } \gamma \text{ against } p_A : \Gamma.A \to \Gamma.$$



Proposition

$\Gamma \vdash a : A \;\Leftrightarrow\;$ There is a family $(a_\gamma \mid c : \text{an object of } \mathcal{C}, \gamma : \mathrm{y}c \to \Gamma)$ satisfying

$$c \Vdash a_\gamma : A(\gamma)$$

and for every morphism $f : d \to c$ of $\mathcal{C}$,

$$a_\gamma.f = a_{\gamma.f}$$

Proof.

By Yoneda Lemma. □

1. *(Monotonicity)* $c \Vdash a : A(\gamma)$ if and only if $d \Vdash a(f) : A(\gamma f)$ for every $f : d \to c$ in $\mathcal{C}$.

1. *(Monotonicity)* $c \Vdash a : A(\gamma)$ if and only if $d \Vdash a(f) : A(\gamma f)$ for every $f : d \to c$ in $\mathcal{C}$.
2. If A, B are types in a context Γ and $\Gamma \vdash f : A \to B$ is a function then whenever $c \Vdash a : A(\gamma)$ we get $c \Vdash f(a) : B(\gamma)$.

1. *(Monotonicity)* $c \Vdash a : A(\gamma)$ if and only if $d \Vdash a(f) : A(\gamma f)$ for every $f : d \to c$ in $\mathcal{C}$.
2. If A, B are types in a context Γ and $\Gamma \vdash f : A \to B$ is a function then whenever $c \Vdash a : A(\gamma)$ we get $c \Vdash f(a) : B(\gamma)$.
3. *(Yoneda)* Assume that b is a rule such that for for every object c of $\mathcal{C}$ and every $\gamma : \mathrm{y}c \to \Gamma$

$$c \Vdash a : A(\gamma) \text{ implies } c \Vdash b_a : B(\gamma)$$

such that $b_a.f = b_{a.f}$ for all $f : d \to c$ in $\mathcal{C}$. Then there is a function $\Gamma \vdash b : A \to B$ such that $b_a = b(a)$, for all $\Gamma \vdash a : A$.

Forcing types

Definition

For a type $\Gamma \vdash A\ \mathtt{Type}$, an object c of $\mathcal{C}$, and a morphism $\gamma \colon \mathrm{y}c \to \Gamma$, we say c **forces** A Type **at stage** γ, and we write $c \Vdash [A\,\mathrm{Type}](\gamma)$, whenever there is a presheaf $\widetilde{A}_\gamma$ and a map $p_\gamma \colon \widetilde{A}_\gamma \to \mathrm{y}c$ such that for every morphism $f \colon d \to c$ in $\mathcal{C}$ there is a presheaf $\widetilde{A}_{\gamma.f}$ and a choice of map $\widetilde{A}_{\gamma.f} \to \widetilde{A}_\gamma$, making a pullback square

$$\begin{array}{ccc} \widetilde{A}_{\gamma.f} & \longrightarrow & \widetilde{A}_\gamma \\ {\scriptstyle p_{\gamma.f}}\downarrow & \ulcorner & \downarrow{\scriptstyle p_\gamma} \\ \mathrm{y}d & \xrightarrow[\mathrm{y}f]{} & \mathrm{y}c \end{array} . \tag{2}$$

Forcing extensional equality types

Proposition

Given a context Γ*,*

Γ

Forcing extensional equality types

Proposition

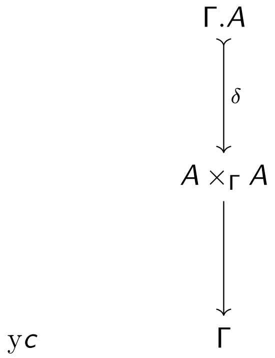
Given a context Γ, *a type* $\Gamma \vdash A \ \mathtt{Type}$,

$$\Gamma.A \overset{\delta}{\rightarrowtail} A \times_{\Gamma} A \longrightarrow \Gamma$$

Forcing extensional equality types

Proposition

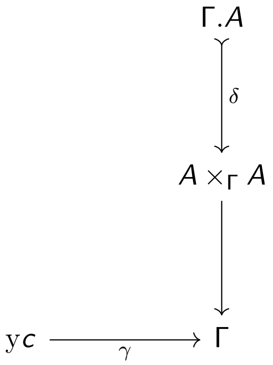
Given a context Γ, a type $\Gamma \vdash A$ $\mathtt{Type}$, an object c of $\mathcal{C}$,



Forcing extensional equality types

Proposition

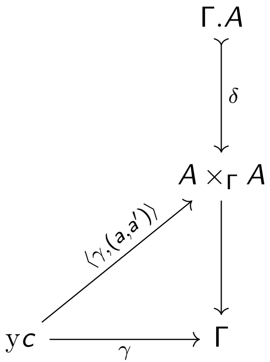
Given a context Γ, *a type* $\Gamma \vdash A$ $\mathtt{Type}$, *an object* c *of* $\mathcal{C}$, *a morphism* $\gamma \colon \mathrm{y}c \to \Gamma$,



Forcing extensional equality types

Proposition

Given a context Γ*, a type* $\Gamma \vdash A\ \mathcal{T}ype$*, an object* c *of* $\mathcal{C}$*, a morphism* $\gamma \colon \mathrm{y}c \to \Gamma$*,* $c \Vdash (a, a') : (A \times A)(\gamma)$



Forcing extensional equality types

Proposition

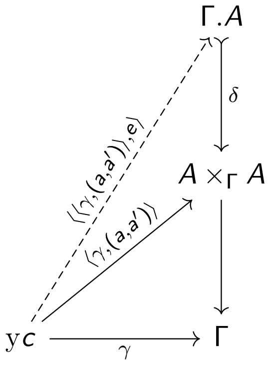
Given a context Γ, *a type* $\Gamma \vdash A \; \mathtt{Type}$, *an object* c *of* $\mathcal{C}$, *a morphism* $\gamma \colon \mathrm{y}c \to \Gamma$, $c \Vdash (a, a') : (A \times A)(\gamma)$ *we have*

$$c \Vdash e : \mathrm{Eq}_A(a, a')(\gamma) \;\Leftrightarrow$$

$$a, a' \textit{ are equal as morphisms in } \mathscr{E} \;\Leftrightarrow$$

$$a, a' \textit{ are equal elements of } A(c)\,.$$

The Type Eq_A is interpreted by the diagonal morphism $\delta \colon A \rightarrowtail A \times_\Gamma A$ over Γ.



Forcing dependent sum types

Proposition

Given a context Γ*,*

Γ

Forcing dependent sum types

Proposition

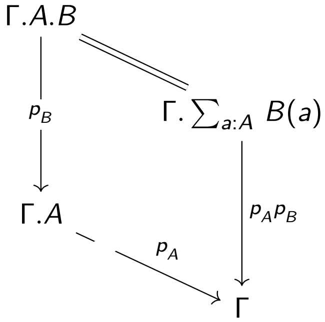
Given a context Γ, *a type* $\Gamma \vdash A\ \mathtt{Type}$,

$$\Gamma.A \xrightarrow{p_A} \Gamma$$

Forcing dependent sum types

Proposition

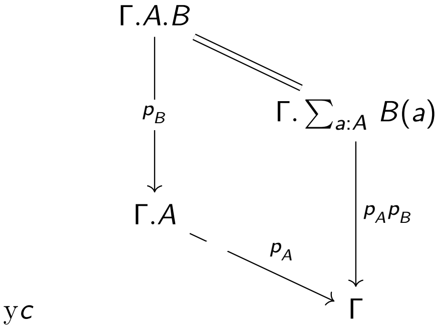
Given a context Γ, *a type* $\Gamma \vdash A\ \mathtt{Type}$, *a type* $\Gamma, x : A \vdash B\ \mathtt{Type}$,



Forcing dependent sum types

Proposition

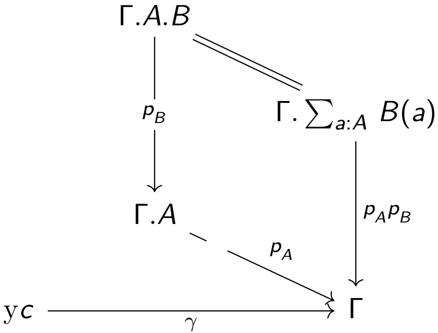
Given a context Γ*, a type* $\Gamma \vdash A$ $\mathtt{Type}$*, a type* $\Gamma, x : A \vdash B$ $\mathtt{Type}$*, an object* c *of* $\mathcal{C}$*,*



Forcing dependent sum types

Proposition

Given a context Γ*, a type* $\Gamma \vdash A$ $\mathtt{Type}$*, a type* $\Gamma, x : A \vdash B$ $\mathtt{Type}$*, an object* c *of* $\mathcal{C}$*, and a morphism* $\gamma \colon \mathrm{y}c \to \Gamma$*,*



Forcing dependent sum types

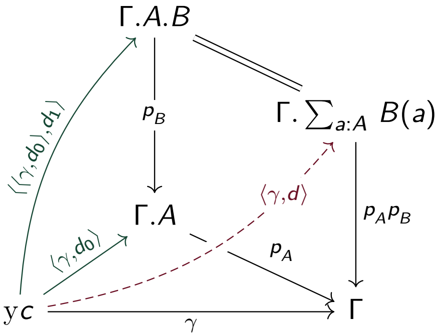
Proposition

Given a context Γ*, a type* $\Gamma \vdash A\ \mathtt{Type}$*, a type* $\Gamma, x : A \vdash B\ \mathtt{Type}$*, an object* c *of* $\mathcal{C}$*, and a morphism* $\gamma \colon \mathrm{y}c \to \Gamma$*,*

$$c \Vdash d : \left(\sum_{a:A} B(a)\right)(\gamma)$$

iff

$$d = (d_0, d_1)$$
$$c \Vdash d_0 : A(\gamma)$$
$$c \Vdash d_1 : B(\langle \gamma, d_0 \rangle)\ .$$



Forcing dependent product types

Proposition

Given a context Γ*,*

Γ

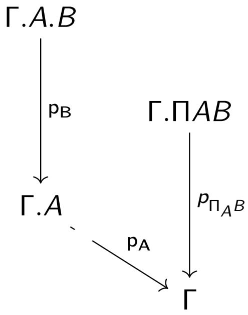
Forcing dependent product types

Proposition

Given a context Γ, *a type* $\Gamma \vdash A\ \mathtt{Type}$,

$$\Gamma.A \xrightarrow{p_A} \Gamma$$

Forcing dependent product types



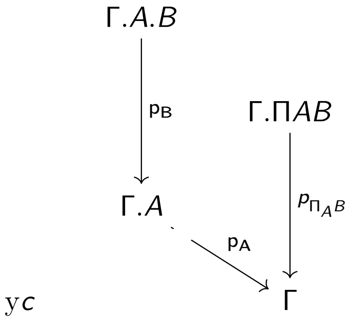
Proposition

Given a context Γ, *a type* $\Gamma \vdash A\ \mathtt{Type}$, *a type* $\Gamma, x : A \vdash B\ \mathtt{Type}$,

Forcing dependent product types

Proposition

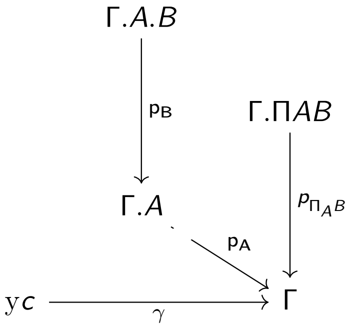
Given a context Γ, a type $\Gamma \vdash A$ $\mathtt{Type}$, a type $\Gamma, x : A \vdash B$ $\mathtt{Type}$, an object c of $\mathcal{C}$,



Forcing dependent product types

Proposition

Given a context Γ*, a type* $\Gamma \vdash A\ \mathtt{Type}$*, a type* $\Gamma, x : A \vdash B\ \mathtt{Type}$*, an object* c *of* $\mathcal{C}$*, and a morphism* $\gamma : \mathrm{y}c \to \Gamma$*,*



Forcing dependent product types

Proposition

Given a context Γ, *a type* $\Gamma \vdash A$ $\mathtt{Type}$, *a type* $\Gamma, x : A \vdash B$ $\mathtt{Type}$, *an object* c *of* $\mathcal{C}$, *and a morphism* $\gamma : \mathrm{y}c \to \Gamma$,

$$c \Vdash b : \left(\prod_{x:A} B\right)(\gamma)$$

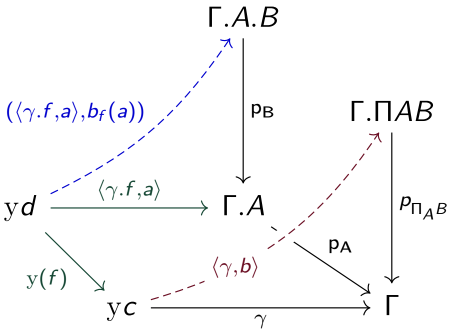
iff there is a function b *such for every morphism* $f : d \to c$ *in* $\mathcal{C}$, *if*

$$d \Vdash a : A(\gamma.f)$$

then

$$d \Vdash b_f(a) : B(\langle \gamma.f, a \rangle)$$

and for every $g : d' \to d$, $b_f(a).g = b_{f \circ g}(a.g)$.



Forcing a universe $\mathcal{V}$ of small types (I)

- We assume one more inaccessible cardinal κ with $\kappa < \lambda$.
- We call the sets of size strictly less than κ **small**.
- Write $\mathcal{S}\mathrm{et}_\kappa$ for the category of κ-small sets.
- Exactly as before, we obtain a universe $\pi_{\mathcal{V}} \colon \mathcal{V}^\bullet \to \mathcal{V}$: the set $\mathcal{V}(c)$ consists of the presheaves on $\mathcal{C}/c$ whose values are κ-small sets.
- $\mathcal{V}$ is a λ-small presheaf and hence it admits a classifier v:

$$\mathsf{v} \colon \mathsf{Type} \qquad\qquad \begin{array}{ccc} \mathcal{V} & \longrightarrow & \mathsf{Type}^\bullet \\ \big\downarrow & \ulcorner & \big\downarrow \pi \\ 1 & \underset{\mathsf{v}}{\longrightarrow} & \mathsf{Type} \end{array}$$

Forcing a universe $\mathcal{V}$ of small types (II)

We have an inclusion $\mathsf{El} \colon \mathcal{V} \hookrightarrow \mathsf{Type}$, induced by the inclusion of $\mathcal{S}\mathrm{et}_\kappa$ into $\mathcal{S}\mathrm{et}_\lambda$, and a pullback diagram

$$\begin{array}{ccc} \mathcal{V}^\bullet & \longrightarrow & \mathsf{Type}^\bullet \\ {\scriptstyle \pi_{\mathcal{V}}}\big\downarrow & \ulcorner & \big\downarrow{\scriptstyle \pi} \\ \mathcal{V} & \xrightarrow[\mathsf{El}]{} & \mathsf{Type} \end{array}$$

Forcing a universe $\mathcal{V}$ of small types (II)

We have an inclusion $\mathsf{El}\colon \mathcal{V} \hookrightarrow \mathsf{Type}$, induced by the inclusion of $\mathcal{S}\mathrm{et}_\kappa$ into $\mathcal{S}\mathrm{et}_\lambda$, and a pullback diagram

$$\begin{array}{ccc} \mathcal{V}^\bullet & \longrightarrow & \mathsf{Type}^\bullet \\ {\scriptstyle \pi_{\mathcal{V}}}\big\downarrow & \ulcorner & \big\downarrow{\scriptstyle \pi} \\ \mathcal{V} & \xrightarrow[\mathsf{El}]{} & \mathsf{Type} \end{array}$$

$$\frac{\Gamma \vdash \ a : \mathsf{v}}{\Gamma \vdash \ \mathsf{El}(a) : \mathsf{Type}}$$

$$\begin{array}{ccccc} \Gamma.\mathsf{El}(a) & \longrightarrow & \mathcal{V}^\bullet & \longrightarrow & \mathsf{Type}^\bullet \\ \big\downarrow & \ulcorner & \big\downarrow & & \big\downarrow \\ \Gamma & \xrightarrow[a]{} & \mathcal{V} & \xrightarrow[\mathsf{El}]{} & \mathsf{Type} \end{array}$$

Forcing universe $\mathcal{V}$ of small types (III)

Proposition

For an object c of $\mathcal{C}$,

$$c \Vdash a : \mathrm{v} \Leftrightarrow c \Vdash \mathrm{El}(a)\mathrm{Type},$$
$$\mathrm{El}(a.f) \equiv \mathrm{El}(a).f \text{ for every } f : d \to c \text{, and}$$
$$\mathrm{El}(a) \to \mathrm{y}c \text{ and } \mathrm{El}(a.f) \to \mathrm{y}d \ (\text{for all } f : d \to c) \text{ are small.}$$

Forcing universe $\mathcal{V}$ of small types (III)

Proposition

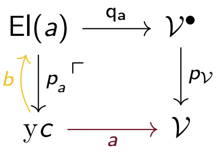
For an object c of $\mathcal{C}$,

$$c \Vdash a : \mathrm{v} \;\Leftrightarrow c \Vdash \mathsf{El}(a)\mathsf{Type},$$
$$\mathsf{El}(a.f) \equiv \mathsf{El}(a).f \;\textit{ for every } f : d \to c \textit{, and}$$
$$\mathsf{El}(a) \to \mathrm{y}c \textit{ and } \mathsf{El}(a.f) \to \mathrm{y}d \;(\textit{for all } f : d \to c) \textit{ are small.}$$

Proposition

For an object c of $\mathcal{C}$,

$$c \Vdash [a^\bullet : \mathcal{V}^\bullet] \Leftrightarrow a^\bullet = (a, b) \textit{ such that } c \Vdash a : \mathrm{v}$$
$$\textit{and } c \Vdash b : \mathsf{El}(a)(\mathrm{id}_c)$$



The impredicative universe Ω of propositions

- As usual, a impredicative universe Ω of (small) propositions in $\mathscr{E}$ is defined object-wise by
$$\Omega(c) \triangleq \mathrm{Ob}\,\mathrm{Ob}[(\mathcal{C}/c)^{\mathrm{op}}, \mathbb{2}]\ ,$$
where $\mathbb{2}$ is the class of truth values in *SET*, viewed as a partial order.
- $\Omega(c)$ is isomorphic to the class of sieves on object c, or equivalently, the class of subobjects of $\mathrm{y}c$.
- As usual, we write $\mathtt{true}\colon 1 \rightarrowtail \Omega$ for the natural transformation whose component at c is given by the maximal sieve of c. We write

$$\begin{array}{ccc} [\varphi] & \longrightarrow & 1 \\ \big\downarrow & \ulcorner & \big\downarrow \mathtt{true} \\ \mathrm{y}c & \xrightarrow[\varphi]{} & \Omega \end{array}$$

The impredicative universe Ω of propositions

- As usual, a impredicative universe Ω of (small) propositions in $\mathscr{E}$ is defined object-wise by
$$\Omega(c) \triangleq \mathrm{Ob}\,\mathrm{Ob}[(\mathcal{C}/c)^{\mathrm{op}}, \mathbb{2}]\ ,$$
where $\mathbb{2}$ is the class of truth values in *SET*, viewed as a partial order.
- $\Omega(c)$ is isomorphic to the class of sieves on object c, or equivalently, the class of subobjects of $\mathrm{y}c$.
- As usual, we write $\mathtt{true} \colon 1 \rightarrowtail \Omega$ for the natural transformation whose component at c is given by the maximal sieve of c. We write

$$\begin{array}{ccc} [\varphi] & \longrightarrow & 1 \\ \downarrow & \ulcorner & \downarrow \mathtt{true} \\ \mathrm{y}c & \xrightarrow[\varphi]{} & \Omega \end{array}$$

By the definition of the Heyting algebra structure of $\mathsf{Sub}(\Gamma)$, for $\Gamma \in \mathcal{P}\mathsf{Shv}(\mathcal{C})$, we have isomorphisms

$$
\begin{aligned}
[\top] &\cong 1 \\
[\bot] &\cong 0 \\
[\phi \wedge \psi] &\cong [\phi] \times [\psi] \\
[\phi \Rightarrow \psi] &\cong [\phi] \to [\psi] \\
[(\forall x\colon A)\phi] &\cong (\Pi x\colon A)[\phi]
\end{aligned}
$$

For disjunction, instead, $[\phi \vee \psi]$ arises via the image factorization of the coproduct

$$\begin{array}{ccc} [\phi] + [\psi] & \twoheadrightarrow & [\phi \vee \psi] \\ & \searrow & \swarrow \\ & \Gamma & \end{array}$$

For disjunction, instead, $[\phi \vee \psi]$ arises via the image factorization of the coproduct

$$\begin{array}{ccc} [\phi] + [\psi] & \longrightarrow\!\!\!\!\rightarrow & [\phi \vee \psi] \\ & \searrow \quad \swarrow & \\ & \Gamma & \end{array}$$

while the existential quantifier is the image factorization of the dependent sum

$$\begin{array}{ccc} \Sigma_A[\phi] & \longrightarrow\!\!\!\!\rightarrow & [\exists_A(\phi)] \\ & \searrow \quad \swarrow & \\ & \Gamma & \end{array}$$

Propositions as types (I)

There is a canonical map $\iota\colon \Omega \to \mathsf{Type}$ which fits into a cartesian square

$$\begin{array}{ccc} 1 & \xrightarrow{\;\widetilde{*}\;} & \mathsf{Type}^{\bullet} \\ {\scriptstyle \mathtt{true}}\big\downarrow & \lrcorner & \big\downarrow {\scriptstyle p_{\mathsf{Type}}} \\ \Omega & \underset{\iota}{\rightarrowtail} & \mathsf{Type} \end{array}$$

where $\widetilde{*} = (\iota\mathtt{true}, *)$, and $*$ is the unique term of $\mathsf{El}(\iota\mathtt{true}) \cong [\mathtt{true}] \cong 1$.

Propositions as types (I)

There is a canonical map $\iota \colon \Omega \to \mathsf{Type}$ which fits into a cartesian square

$$\begin{array}{ccc} 1 & \xrightarrow{\widetilde{*}} & \mathsf{Type}^{\bullet} \\ {\scriptstyle \mathtt{true}}\big\downarrow & \lrcorner & \big\downarrow{\scriptstyle p_{\mathsf{Type}}} \\ \Omega & \underset{\iota}{\rightarrowtail} & \mathsf{Type} \end{array}$$

where $\widetilde{*} = (\iota\mathtt{true}, *)$, and $*$ is the unique term of $\mathsf{El}(\iota\mathtt{true}) \cong [\mathtt{true}] \cong 1$.

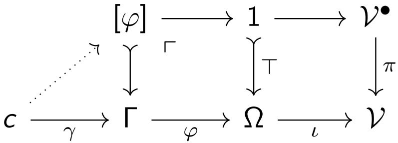
$$\begin{array}{ccccc} [\varphi] & \longrightarrow & 1 & \longrightarrow & \mathcal{V}^{\bullet} \\ \big\downarrow & \lrcorner & \big\downarrow{\scriptstyle \mathtt{true}} & & \big\downarrow{\scriptstyle \pi} \\ \Gamma & \underset{\varphi}{\longrightarrow} & \Omega & \underset{\iota}{\longrightarrow} & \mathcal{V} \end{array}$$

Forcing Ω

Theorem

Let $\varphi\colon \Gamma \to \Omega$ and $\gamma\colon c \to \Gamma$. Then the following are equivalent:

1. *$c \Vdash \varphi(\gamma)$ in the sense of the standard Kripke-Joyal semantics,*
2. *there exists a (necessarily unique) $a\colon c \to \mathcal{V}^\bullet$ such that $c \Vdash a\colon \iota\varphi(\gamma)$.*



Propositions as types (II)

Proposition

Suppose $\Gamma.A \vdash \varphi : \Omega$. *We have*

1. $\iota(\forall x : A.\varphi(x)) \equiv \Pi(a, \iota\varphi)$.
2. $||\Sigma(a, \iota\varphi)|| \cong \iota(\exists x : A.\varphi(x))$

Forcing cofibrant propositions (I)

- As in (Orton and Pitts, 2018), we consider a modality $\mathtt{cof} : \Omega \to \Omega$ satisfying:

- As in (Orton and Pitts, 2018), we consider a modality $\mathtt{cof} : \Omega \to \Omega$ satisfying:
  (i) $\mathtt{cof} \circ \mathtt{true} = \mathtt{true}$,

Forcing cofibrant propositions (I)

- As in (Orton and Pitts, 2018), we consider a modality $\mathtt{cof} : \Omega \to \Omega$ satisfying:
  (i) $\mathtt{cof} \circ \mathtt{true} = \mathtt{true}$,
  (ii) $\mathtt{cof} \circ \mathtt{false} = \mathtt{true}$,

Forcing cofibrant propositions (I)

- As in (Orton and Pitts, 2018), we consider a modality $\mathtt{cof} : \Omega \to \Omega$ satisfying:
  (i) $\mathtt{cof} \circ \mathtt{true} = \mathtt{true}$,
  (ii) $\mathtt{cof} \circ \mathtt{false} = \mathtt{true}$,
  (iii) $\forall (\varphi, \psi : \Omega).\ \mathtt{cof}\,\varphi \Rightarrow (\varphi \Rightarrow \mathtt{cof}\,\psi) \Rightarrow \mathtt{cof}(\varphi \wedge \psi)$.

Forcing cofibrant propositions

- As in (Orton and Pitts, 2018), we consider a modality $\texttt{cof} : \Omega \to \Omega$ satisfying:
  (i) $\texttt{cof} \circ \texttt{true} = \texttt{true}$,
  (ii) $\texttt{cof} \circ \texttt{false} = \texttt{true}$,
  (iii) $\forall(\varphi, \psi : \Omega).\ \texttt{cof}\, \varphi \Rightarrow (\varphi \Rightarrow \texttt{cof}\, \psi) \Rightarrow \texttt{cof}(\varphi \wedge \psi)$.
- The last axiom is called the **principle of dominance**.

- Obtain $m_{\mathtt{Cof}} : \mathtt{Cof} \rightarrowtail \Omega$ as the comprehension subtype; in the internal language

$$\mathtt{Cof} \triangleq \{\varphi \in \Omega \mid \mathtt{cof}\,\varphi\}$$

$$\begin{array}{ccc} \mathtt{Cof} & \longrightarrow & 1 \\ {\scriptstyle m_{\mathtt{cof}}} \downarrow & \ulcorner & \downarrow {\scriptstyle \mathtt{true}} \\ \Omega & \xrightarrow[\mathtt{cof}]{} & \Omega \end{array}$$

Forcing cofibrant propositions (II)

- Obtain $m_{\mathtt{Cof}} : \mathtt{Cof} \rightarrowtail \Omega$ as the comprehension subtype; in the internal language

$$\mathtt{Cof} \triangleq \{\varphi \in \Omega \mid \mathtt{cof}\,\varphi\}$$

$$\begin{array}{ccc} \mathtt{Cof} & \longrightarrow & 1 \\ {\scriptstyle m_{\mathtt{cof}}}\big\downarrow & \lrcorner & \big\downarrow{\scriptstyle \mathtt{true}} \\ \Omega & \xrightarrow[\mathtt{cof}]{} & \Omega \end{array}$$

- $\mathtt{cof}\,(\mathtt{true}) = \mathtt{true}$ implies that $\mathtt{true} = m_{\mathtt{Cof}} \circ \mathtt{t}$ for a monomorphism $\mathtt{t} : 1 \rightarrowtail \mathtt{Cof}$.

$$\begin{array}{ccccc} 1 & \overset{\mathtt{t}}{\rightarrowtail} & \mathtt{Cof} & \longrightarrow & 1 \\ \Big\| & \lrcorner & {\scriptstyle m_{\mathtt{Cof}}}\big\downarrow & \lrcorner & \big\downarrow{\scriptstyle \mathtt{true}} \\ 1 & \underset{\mathtt{true}}{\rightarrowtail} & \Omega & \xrightarrow[\mathtt{cof}]{} & \Omega \end{array}$$

Forcing cofibrant propositions (II)

- Obtain $m_{\texttt{Cof}} : \texttt{Cof} \rightarrowtail \Omega$ as the comprehension subtype; in the internal language

$$\texttt{Cof} \triangleq \{\varphi \in \Omega \mid \texttt{cof}\, \varphi\}$$

$$\begin{array}{ccc}
\texttt{Cof} & \longrightarrow & 1 \\
{\scriptstyle m_{\texttt{cof}}} \big\downarrow & \ulcorner & \big\downarrow {\scriptstyle \texttt{true}} \\
\Omega & \xrightarrow[\texttt{cof}]{} & \Omega
\end{array}$$

- $\texttt{cof}\,(\texttt{true}) = \texttt{true}$ implies that $\texttt{true} = m_{\texttt{Cof}} \circ \texttt{t}$ for a monomorphism $\texttt{t} \colon 1 \rightarrowtail \texttt{Cof}$.

$$\begin{array}{ccccc}
1 & \overset{\texttt{t}}{\rightarrowtail} & \texttt{Cof} & \longrightarrow & 1 \\
\Vert & \ulcorner & {\scriptstyle m_{\texttt{Cof}}} \big\downarrow & \ulcorner & \big\downarrow {\scriptstyle \texttt{true}} \\
1 & \underset{\texttt{true}}{\rightarrowtail} & \Omega & \xrightarrow[\texttt{cof}]{} & \Omega
\end{array}$$

- Call $\texttt{t}$ the **generic cofibrant proposition**.

Cofibrations

- A monomorphism $m \colon C \rightarrowtail Z$ is a **cofibration** if its classifying map $\chi_m \colon Z \to \Omega$ factors through $m_{\mathtt{cof}} \colon \mathtt{Cof} \rightarrowtail \Omega$.

$$\begin{tikzcd} C \arrow[r] \arrow[d, tail, "m"'] \arrow[dr, phantom, "\ulcorner", very near start] & 1 \arrow[r] \arrow[d, tail, "\mathtt{t}"] \arrow[dr, phantom, "\ulcorner", very near start] & 1 \arrow[d, tail, "\mathtt{true}"] \\ Z \arrow[r, dashed] \arrow[rr, bend right, "\chi_m"'] & \mathtt{Cof} \arrow[r, tail, "m_{\mathtt{cof}}"] & \Omega \end{tikzcd}$$

Cofibrations

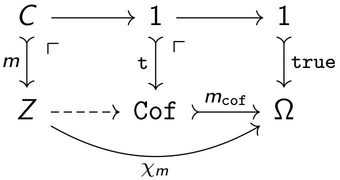
- A monomorphism $m\colon C \rightarrowtail Z$ is a **cofibration** if its classifying map $\chi_m\colon Z \to \Omega$ factors through $m_{\mathtt{cof}}\colon \mathtt{Cof} \rightarrowtail \Omega$.

$$\begin{array}{ccccc} C & \longrightarrow & 1 & \longrightarrow & 1 \\ {\scriptstyle m}\big\downarrow & \ulcorner & {\scriptstyle \mathtt{t}}\big\downarrow & \ulcorner & \big\downarrow{\scriptstyle \mathtt{true}} \\ Z & \dashrightarrow & \mathtt{Cof} & \overset{m_{\mathtt{cof}}}{\rightarrowtail} & \Omega \end{array} \qquad \chi_m\colon Z \to \Omega$$

- Therefore, a monomorphism $m\colon C \rightarrowtail Z$ is a cofibration iff it is a pullback of the generic cofibration $\mathtt{t}\colon 1 \rightarrowtail \mathtt{Cof}$.

Cofibrations

- A monomorphism $m\colon C \rightarrowtail Z$ is a **cofibration** if its classifying map $\chi_m\colon Z \to \Omega$ factors through $m_{\mathtt{cof}} : \mathtt{Cof} \rightarrowtail \Omega$.



- Therefore, a monomorphism $m\colon C \rightarrowtail Z$ is a cofibration iff it is a pullback of the generic cofibration $\mathtt{t}\colon 1 \rightarrowtail \mathtt{Cof}$.

Proposition

$m\colon C \rightarrowtail Z$ is a cofibration $\Leftrightarrow \mathscr{E} \Vdash \forall z : Z.\, \mathtt{cof}(\exists c : C. m(c) = z).$

Forcing dominance (I)

Consider the following polynomials

$$1 \overset{\mathtt{t}}{\rightarrowtail} \mathtt{Cof}$$

$$1 \overset{\mathtt{true}}{\rightarrowtail} \Omega$$

$$\mathsf{Type}^{\bullet} \xrightarrow{p_{\mathsf{Type}}} \mathsf{Type}$$

associated polynomial functor ⟹

$$\mathscr{E} \xrightarrow{P_{\mathtt{t}}} \mathscr{E}$$

$$\mathscr{E} \xrightarrow{P_{\mathtt{true}}} \mathscr{E}$$

$$\mathscr{E} \xrightarrow{P_{p_{\mathsf{Type}}}} \mathscr{E}$$

where

$$P_{\mathtt{t}}(A) = \sum_{\varphi : \mathtt{Cof}} A^{[\varphi]}$$

$$P_{\mathtt{true}}(A) = \sum_{\varphi : \Omega} A^{[\varphi]}$$

$$P_{p_{\mathsf{Type}}}(A) = \sum_{a : \mathsf{Type}} A^{\mathsf{El}(a)}$$

Forcing dominance (II)

Because the square

$$\begin{array}{ccc} 1 & \longrightarrow & 1 \\ {\scriptstyle \mathtt{t}}\downarrow & \ulcorner & \downarrow{\scriptstyle \mathtt{true}} \\ \mathtt{Cof} & \xrightarrow{m_{\mathtt{cof}}} & \Omega \end{array}$$

is cartesian, we obtain a cartesian square:

$$\begin{array}{ccc} P_{\mathtt{t}}(\mathtt{Cof}) & \rightarrowtail & P_{\mathtt{true}}(\mathtt{Cof}) \\ {\scriptstyle P_{\mathtt{t}}(m_{\mathtt{cof}})}\downarrow & \ulcorner & \downarrow{\scriptstyle P_{\mathtt{true}}(m_{\mathtt{cof}})} \\ P_{\mathtt{t}}(\Omega) & \rightarrowtail & P_{\mathtt{true}}(\Omega) \end{array} \quad = \quad \begin{array}{ccc} \sum_{\varphi : \mathtt{Cof}} \mathtt{Cof}^{[\varphi]} & \rightarrowtail & \sum_{\varphi : \Omega} \mathtt{Cof}^{[\varphi]} \\ \downarrow & \ulcorner & \downarrow \\ \sum_{\varphi : \mathtt{Cof}} \Omega^{[\varphi]} & \rightarrowtail & \sum_{\varphi : \Omega} \Omega^{[\varphi]} \end{array}$$

And because

$$\begin{array}{ccc} 1 & \xrightarrow{\;\widetilde{*}\;} & \mathsf{Type}^{\bullet} \\ {\scriptstyle \mathtt{true}}\big\downarrow & \ulcorner & \big\downarrow{\scriptstyle p_{\mathsf{Type}}} \\ \Omega & \underset{\iota}{\rightarrowtail} & \mathsf{Type} \end{array}$$

is cartesian, we obtain a cartesian square:

$$\begin{array}{ccc} P_{\mathtt{true}}(\Omega) & \rightarrowtail & P_{p_{\mathsf{Type}}}(\Omega) \\ {\scriptstyle P_{\mathtt{true}}(\iota)}\big\downarrow & \ulcorner & \big\downarrow{\scriptstyle P_{p_{\mathsf{Type}}}(\iota)} \\ P_{\mathtt{true}}(\mathsf{Type}) & \rightarrowtail & P_{p_{\mathsf{Type}}}(\mathsf{Type}) \end{array} \quad = \quad \begin{array}{ccc} \sum_{\varphi : \Omega} \Omega^{[\varphi]} & \rightarrowtail & \sum_{a : \mathsf{Type}} \Omega^{\mathsf{El}(a)} \\ \big\downarrow & \ulcorner & \big\downarrow \\ \sum_{\varphi : \Omega} \mathsf{Type}^{[\varphi]} & \rightarrowtail & \sum_{a : \mathsf{Type}} \mathsf{Type}^{\mathsf{El}(a)} \end{array}$$

Forcing dominance (II)

$$
\begin{array}{ccccc}
P_{\mathtt{t}}(\mathtt{Cof}) & \rightarrowtail & P_{\mathtt{true}}(\mathtt{Cof}) & & \\
\downarrow & \ulcorner & \downarrow & & \\
P_{\mathtt{t}}(\Omega) & \rightarrowtail & P_{\mathtt{true}}(\Omega) & \rightarrowtail & P_{p_{\mathsf{Type}}}(\Omega) \\
 & & \downarrow & \ulcorner & \downarrow \\
 & & P_{\mathtt{true}}(\mathsf{Type}) & \rightarrowtail & P_{p_{\mathsf{Type}}}(\mathsf{Type})
\end{array}
$$

Therefore, there is a composite map

$$\sum_{\varphi : \mathtt{Cof}} \mathtt{Cof}^{[\varphi]} = P_{\mathtt{t}}(\mathtt{Cof}) \rightarrowtail P_{\mathtt{true}}(\Omega) \rightarrowtail P_{p_{\mathsf{Type}}}(\mathsf{Type}) = \sum_{a : \mathsf{Type}} \mathsf{Type}^{\mathsf{El}(a)}$$

which takes (φ, ψ) to $(\iota\varphi, \iota\psi)$.

Proposition

$\mathscr{E} \Vdash [\mathsf{dom} : \forall(\varphi, \psi : \Omega).\ \mathtt{cof}\,\varphi \Rightarrow (\varphi \Rightarrow \mathtt{cof}\,\psi) \Rightarrow \mathtt{cof}(\varphi \wedge \psi)] \Leftrightarrow$ *there is a lift* dom *of* Σ *making the square commute.*

$$\begin{array}{ccc} P_{\mathtt{t}}(\mathtt{Cof}) & \overset{\mathsf{dom}}{\dashrightarrow} & \mathtt{Cof} \\ \downarrow & & \downarrow \iota \\ P_{p_{\mathsf{Type}}}(\mathsf{Type}) & \underset{\Sigma}{\longrightarrow} & \mathsf{Type} \end{array}$$

- Note that $\Sigma \colon P_{p_{\mathsf{Type}}}(\mathsf{Type}) \to (\mathsf{Type})$ in above is the Natural Model (resp. CwF) interpretation of the $\sum$ type-former following (Awodey, 2018).

Proposition

For $\varphi : \texttt{Cof}$ *and* $\psi : [\varphi] \to \texttt{Cof}$*, the following statements hold:*

(i) $\mathsf{dom}(\texttt{t}, \varphi) = \varphi = \mathsf{dom}(\varphi, \texttt{t})$.

(ii) $\mathsf{dom}(\mathsf{dom}(\varphi, \psi), \theta) = \mathsf{dom}(\varphi, \mathsf{dom}(\psi, \theta))$.

(iii) $[\mathsf{dom}(\varphi, \psi)] \equiv \sum_{x : [\varphi]} [\psi(x)]$.

Proof.

For (i), note that $\iota(\texttt{t}) = \mathsf{code}(\mathbf{1})$ where $\mathbf{1}$ is the terminal type. Since $\sum_{* : \mathbf{1}} \varphi(*) = \iota\varphi$ and ι is monic, $\mathsf{dom}(\texttt{t}, \varphi) = \varphi$.

For (ii), since $\sum_{x : \iota\varphi} \texttt{t} \cong \mathsf{code}(1)$ and the "exchange rule" of the sum types.

For (iii), observe that

$$[\mathsf{dom}(\varphi, \psi)] \equiv \mathsf{El}\iota(\mathsf{dom}(\varphi, \psi)) \equiv \mathsf{El}(\Sigma(\iota\varphi, \iota\psi)) \equiv \sum_{x : [\varphi]} [\psi(x)] .$$

□

Forcing dominance (V)

Proposition

Cofibrations are closed under composition.

Proposition

Cofibrations are closed under composition.

Proof.

It suffices to prove that if $m_\varphi \colon [\varphi] \rightarrowtail \mathrm{y}c$ and $m_\psi \colon [\psi] \rightarrowtail [\varphi]$ are cofibrations then so is their composite.

Forcing dominance (V)

Proposition

Cofibrations are closed under composition.

Proof.

It suffices to prove that if $m_\varphi : [\varphi] \rightarrowtail \mathrm{y}c$ and $m_\psi : [\psi] \rightarrowtail [\varphi]$ are cofibrations then so is their composite.

$c \Vdash \varphi : \mathtt{Cof}$ and $c \Vdash \psi : [\varphi] \to \mathtt{Cof}$ imply $c \Vdash \mathrm{dom}(\varphi, \psi) : \mathtt{Cof}$

Proposition

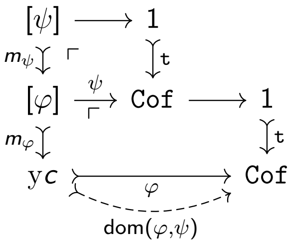
Cofibrations are closed under composition.

Proof.

It suffices to prove that if $m_\varphi : [\varphi] \rightarrowtail \mathrm{y}c$ and $m_\psi : [\psi] \rightarrowtail [\varphi]$ are cofibrations then so is their composite.

$c \Vdash \varphi : \mathtt{Cof}$ and $c \Vdash \psi : [\varphi] \to \mathtt{Cof}$ imply $c \Vdash \mathrm{dom}(\varphi, \psi) : \mathtt{Cof}$

$\mathrm{dom}(\varphi, \psi)$ classifies $m_\varphi \circ m_\psi$ since (i) $[\mathrm{dom}(\varphi, \psi)] \equiv \sum_{x : [\varphi]} [\psi(x)]$, and (ii) $m_\varphi \circ m_\psi$ is the display map of the sum type $\sum_{x : [\varphi]} [\psi(x)]$.



The **type of partial elements** of a type A is given by the polynomial functor

$$P_{\mathtt{true}}(A) = \sum_{\varphi : \Omega} [\varphi] \to A \,.$$

Forcing for partial elements (I)

The **type of partial elements** of a type A is given by the polynomial functor

$$P_{\mathtt{true}}(A) = \sum_{\varphi : \Omega} [\varphi] \to A\,.$$

The type of **cofibrant partial elements** of a type A is given by the polynomial functor

$$A^{+} = P_{\mathtt{t}}(A) = \sum_{\varphi : \mathtt{Cof}} [\varphi] \to A\,.$$

Forcing for partial elements (II)

There is a natural map

$$\begin{aligned} \eta\colon A &\longrightarrow A^{+} \\ a &\longmapsto (\texttt{true}, \lambda * . a : 1 \to A) \end{aligned}$$

Forcing for partial elements (II)

There is a natural map

$$\begin{aligned} \eta \colon A &\longrightarrow A^+ \\ a &\longmapsto (\mathtt{true}, \lambda * . a : 1 \to A) \end{aligned}$$

which fits into the pullback square

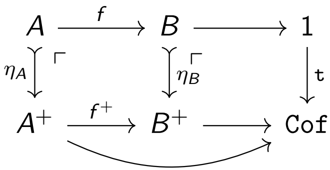
$$\begin{array}{ccc} A & \overset{\eta}{\rightarrowtail} & A^+ \\ {\scriptstyle !_A}\downarrow & \ulcorner & \downarrow{\scriptstyle \mathtt{fst}} \\ 1 & \underset{\mathtt{t}}{\rightarrowtail} & \mathtt{Cof} \end{array} .$$

Proposition ((Awodey, 2018))

The map $\eta_A \colon A \to A^+$ is a cofibration and it classifies partial maps with cofibrant domain.

Forcing for partial elements (III)

In fact, $\eta\colon \mathsf{Id} \Rightarrow +$ is cartesian:



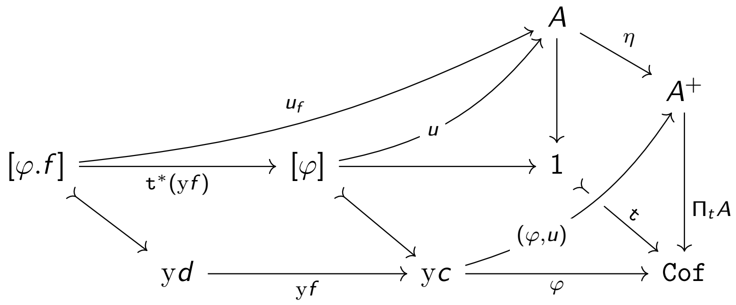
The right square & the outer rectangle are cartesian $\Rightarrow$ The left square is cartesian.

Proposition

$c \Vdash [(\varphi, u) : A^+](\gamma) \iff$
$c \Vdash [\varphi : \texttt{Cof}](\gamma)$ *and for all* $f : d \to c$, *if* $d \Vdash [x : \varphi.f](\gamma.f)$ *then* $d \Vdash [u_f(x) : A](\gamma.f)$, *where* $u_f(x).g = u_{fg}(x)$, *for all* $g : d' \to d$.

Proposition

$c \Vdash [(\varphi, u) : A^{+}](\gamma) \iff$
$c \Vdash [\varphi : \mathtt{Cof}](\gamma)$ *and for all* $f : d \to c$, *if* $d \Vdash [x : \varphi.f](\gamma.f)$ *then* $d \Vdash [u_f(x) : A](\gamma.f)$, *where* $u_f(x).g = u_{fg}(x)$, *for all* $g : d' \to d$.



Forcing for partial elements (IV)

The above gets simplified when $\Gamma = 1$.

$$c \Vdash [(\varphi, u) : A^+] \quad \Leftrightarrow \quad \begin{array}{ccccc} 1 & \longleftarrow & [\varphi] & \xrightarrow{u} & A \\ {\scriptstyle \mathtt{t}}\downarrow & \lrcorner & \downarrow{\scriptstyle m} & \ulcorner & \downarrow{\scriptstyle \eta_A} \\ \mathtt{Cof} & \xleftarrow[\varphi]{} & \mathrm{y}c & \xrightarrow[(\varphi, u)]{} & A^+ \end{array}$$

Monad structure from dominance

Proposition ((Awodey, 2018))

$+ : \mathscr{E} \to \mathscr{E}$ *is a (fibred) monad.*

Monad structure from dominance

Proposition ((Awodey, 2018))

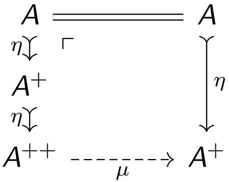
$+ \colon \mathscr{E} \to \mathscr{E}$ *is a (fibred) monad.*

First, we give a category-theoretic proof.

1st Proof.

η_A, η_{A^+}: cofibrations $\Rightarrow$ $\eta_{A^+} \circ \eta_A$: cofibration by dominance.

η_A: cofibrant partial map classifier $\Rightarrow$ there is a unique morphism μ_A classifying the partial map $(\eta_{A^+} \circ \eta_A, \mathrm{id}_A)$.

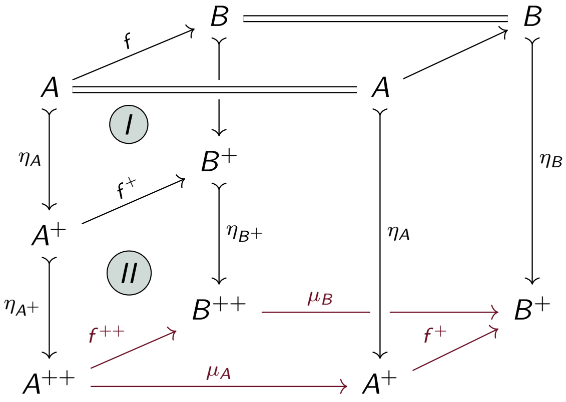


Monad structure from dominance

(1st Proof cont'd.)

μ_A thus obtained is natural in A:

By classifying property of η_B the bottom square commutes since
(i) all vertical squares are pullbacks (I and II because η is cartesian),
(ii) the top square commutes,
(iii) $\eta_{A^+} \circ \eta_A$: cofibration by dominance.



Monad structure from dominance

(1st Proof cont'd.)

To see that $\mu \circ \eta_{A^+} = \mathrm{id}_{A^+}$, observe that the following is a pullback by an easy diagram chase using the previous diagram and the fact that η is always monic.

$$\begin{array}{ccccc}
A & = \!= \!= \!= \!= \!= \!= \!= \!= \!= & & & A \\
{\scriptstyle \eta_A}\big\downarrow & \ulcorner & & & \big\downarrow{\scriptstyle \eta_A} \\
A^+ & \xrightarrow[\eta_{A^+}]{} & A^{++} & \xrightarrow[\mu]{} & A^+
\end{array}$$

By the uniqueness of the classifying map of (η_A, id_A), we have $\mu_A \circ \eta_{A^+} = \mathrm{id}_{A^+}$. By naturality of η,

$$\eta_{A^+} \circ \eta_A = (\eta_A)^+ \circ \eta_A$$

The same argument above shows

$$\mu_A \circ \eta_{A^+} = \mathrm{id}_{A^+} \quad .$$

Monad structure from dominance (II)

Proposition ((Awodey, 2018))

$+ \colon \mathscr{E} \to \mathscr{E}$ *is a (fibred) monad.*

Now, we give a proof using Kripke–Joyal semantics.

2nd Proof.

Monad structure from dominance (II)

Proposition ((Awodey, 2018))

$+ \colon \mathscr{E} \to \mathscr{E}$ is a (fibred) monad.

Now, we give a proof using Kripke–Joyal semantics.

2nd Proof. Write $A^{++} = (A^{+})^{+}$.

$c \Vdash (\varphi, u) : A^{++}$

$\Leftrightarrow$ $u = (\psi, u')$, $c \Vdash [\varphi : \mathtt{Cof}]$, and for every $f \colon c' \to c$, if $c' \Vdash [x : \varphi.f]$ then $c' \Vdash [\psi_f(x) : \mathtt{Cof}]$, and for every $g \colon d \to c'$, if $d \Vdash [y : \psi.g]$ then $d \Vdash [u'_g(y) : A]$ and u' is uniform.

Monad structure from dominance (II)

Proposition ((Awodey, 2018))

$+ \colon \mathscr{E} \to \mathscr{E}$ is a (fibred) monad.

Now, we give a proof using Kripke–Joyal semantics.

2nd Proof. Write $A^{++} = (A^{+})^{+}$.

$c \Vdash (\varphi, u) : A^{++}$

$\Leftrightarrow$ $u = (\psi, u')$, $c \Vdash [\varphi : \mathtt{Cof}]$, and for every $f \colon c' \to c$, if $c' \Vdash [x : \varphi.f]$ then $c' \Vdash [\psi_f(x) : \mathtt{Cof}]$, and for every $g \colon d \to c'$, if $d \Vdash [y : \psi.g]$ then $d \Vdash [u'_g(y) : A]$ and u' is uniform.

Now, set $f = \mathrm{id}_c$.

The statement above (after $\Leftrightarrow$) becomes $u = (\psi, u')$ and $c \Vdash \varphi : \mathtt{Cof}$, $c \Vdash \psi : [\varphi] \to \mathtt{Cof}$, $c \Vdash u' : \sum_{x : [\varphi]} [\psi(x)] \to A$

Monad structure from dominance (II)

The latter implies

$$c \Vdash \mathsf{dom}(\varphi, \psi) : \mathtt{Cof} \text{ and } c \Vdash u' : \mathsf{dom}(\varphi, \psi) \to A.$$

Monad structure from dominance (II)

The latter implies

$$c \Vdash \mathsf{dom}(\varphi, \psi) : \mathtt{Cof} \text{ and } c \Vdash u' : \mathsf{dom}(\varphi, \psi) \to A.$$

Hence

$$c \Vdash (\mathsf{dom}(\varphi, \psi), u') : A^+.$$

The latter implies

$$c \Vdash \mathrm{dom}(\varphi, \psi) : \mathtt{Cof} \text{ and } c \Vdash u' : \mathrm{dom}(\varphi, \psi) \to A.$$

Hence

$$c \Vdash (\mathrm{dom}(\varphi, \psi), u') : A^{+}.$$

Uniformity of u' implies $\mathscr{E} \Vdash \mu : A^{++} \to A^{+}$.

By Yoneda, we get $\mu \colon A^{++} \to A^{+}$.

Monad structure from dominance (II)

The latter implies

$$c \Vdash \mathsf{dom}(\varphi,\psi) : \mathtt{Cof} \text{ and } c \Vdash u' : \mathsf{dom}(\varphi,\psi) \to A.$$

Hence

$$c \Vdash (\mathsf{dom}(\varphi,\psi), u') : A^+.$$

Uniformity of u' implies $\mathscr{E} \Vdash \mu : A^{++} \to A^+$.
By Yoneda, we get $\mu\colon A^{++} \to A^+$.

Also, $\mu \circ \eta_{A^+} = \mathsf{id} = \mu \circ +(\eta_A)$ because $\mathsf{dom}(\varphi, \mathtt{t}) = \varphi$ and $\mathsf{dom}(\mathtt{t}, \psi) = \psi$.
$\mu \circ \mu_{A^+} = \mu \circ +(\mu_A)$ because $\mathsf{dom}(\mathsf{dom}(\varphi,\psi),\theta) = \mathsf{dom}(\varphi, \mathsf{dom}(\psi,\theta))$.

For any type A define

$$\mathrm{TFib}(A) := \prod_{\varphi:\mathtt{Cof}} \prod_{u:[\varphi]\to A} \sum_{a:A} u =_{\varphi} a,$$

where the type $u =_{\varphi} a$ (written $(\varphi, u) \nearrow a$ in Orton and Pitts 2018) is defined

$$(u =_{\varphi} a) := \prod_{p:[\varphi]} \mathrm{Eq}_A(up, a)\,.$$

For any type A define

$$\mathrm{TFib}(A) := \prod_{\varphi:\mathtt{Cof}} \prod_{u:[\varphi]\to A} \sum_{a:A} u =_\varphi a,$$

where the type $u =_\varphi a$ (written $(\varphi, u) \nearrow a$ in Orton and Pitts 2018) is defined

$$(u =_\varphi a) := \prod_{p:[\varphi]} \mathrm{Eq}_A(up, a)\,.$$

Proposition

The map $p_A : \Gamma.A \to \Gamma$ is a uniform trivial fibration $\Leftrightarrow$ there is a term $\Gamma \vdash \alpha : \mathrm{TFib}(A)$.

An application of Kripke–Joyal semantics: Uniform trivial fibration (I)

Proposition

The map $p_A : \Gamma.A \to \Gamma$ *is a uniform trivial fibration* $\Leftrightarrow$ *there is a term* $\Gamma \vdash \alpha : \mathrm{TFib}(A)$.

Recall that p_A being a **a uniform trivial fibration** means that

$$\begin{array}{c} \Gamma.A \\ \downarrow{\scriptstyle p_A} \\ \Gamma \end{array}$$

An application of Kripke–Joyal semantics: Uniform trivial fibration (I)

Proposition

The map $p_A : \Gamma.A \to \Gamma$ *is a uniform trivial fibration* $\Leftrightarrow$ *there is a term* $\Gamma \vdash \alpha : \mathsf{TFib}(A)$.

Recall that p_A being a **a uniform trivial fibration** means that for every cofibration $C \rightarrowtail Z$

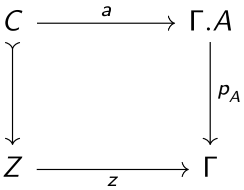
$$\begin{array}{ccc} C & & \Gamma.A \\ \big\downarrow & & \big\downarrow{\scriptstyle p_A} \\ Z & & \Gamma \end{array}$$

An application of Kripke–Joyal semantics: Uniform trivial fibration (I)

Proposition

The map $p_A : \Gamma.A \to \Gamma$ *is a uniform trivial fibration* $\Leftrightarrow$ *there is a term* $\Gamma \vdash \alpha : \mathrm{TFib}(A)$.

Recall that p_A being a **a uniform trivial fibration** means that for every cofibration $C \rightarrowtail Z$ and commutative square

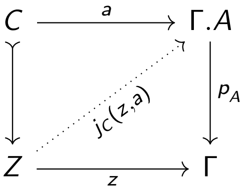


An application of Kripke–Joyal semantics: Uniform trivial fibration (I)

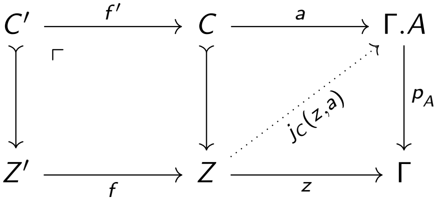
Proposition

The map $p_A \colon \Gamma.A \to \Gamma$ *is a uniform trivial fibration* $\Leftrightarrow$ *there is a term* $\Gamma \vdash \alpha : \mathrm{TFib}(A)$.

Recall that p_A being a **a uniform trivial fibration** means that for every cofibration $C \rightarrowtail Z$ and commutative square there is a diagonal filler $j_C(z, a) \colon Z \to \Gamma.A$ making both triangles commute,



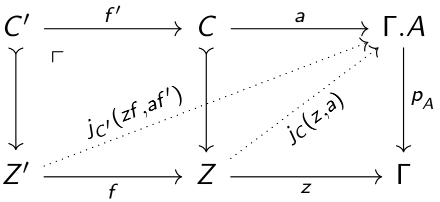
Recall that p_A being a **a uniform trivial fibration** means that for every cofibration $C \rightarrowtail Z$ and commutative square there is a diagonal filler $j_C(z,a)\colon Z \to \Gamma.A$ making both triangles commute, and for any map $f\colon Z' \to Z$,



An application of Kripke–Joyal semantics: Uniform trivial fibration (I)

Recall that p_A being a **a uniform trivial fibration** means that for every cofibration $C \rightarrowtail Z$ and commutative square there is a diagonal filler $j_C(z,a)\colon Z \to \Gamma.A$ making both triangles commute, and for any map $f\colon Z' \to Z$, these maps j satisfy the indicated condition,

$$j_{C'}(zf, af') = j_C(z,a) \circ f\ .$$



An application of Kripke–Joyal semantics: Uniform trivial fibration (II)

Lemma

For $\Gamma \vdash A\,\mathrm{Type}$, $\gamma : \mathrm{y}c \to \Gamma$ *such that*

$$c \Vdash a : A(\gamma)$$

$$c \Vdash \varphi : \mathtt{Cof}(\gamma)$$

$$c \Vdash u : \big([\varphi] \to A\big)(\gamma)\,.$$

then we also have

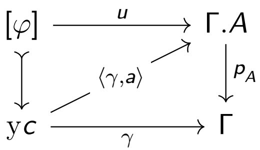
$$c \Vdash e : (u =_\varphi a)(\gamma) \qquad \Leftrightarrow \qquad \begin{array}{ccc} [\varphi] & \xrightarrow{\;u\;} & \Gamma.A \\ \big\downarrow & \nearrow^{\langle\gamma, a\rangle} & \big\downarrow{\scriptstyle p_A} \\ \mathrm{y}c & \xrightarrow[\;\gamma\;]{} & \Gamma \end{array} \quad \textit{commutes,}$$

where

$$(u =_\varphi a) := \prod_{x:[\varphi]} \mathrm{Eq}_A(ux, a)\,.$$

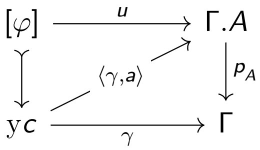
An application of Kripke–Joyal semantics: Uniform trivial fibration (II)

Proof of Lemma.



$c \Vdash a : A(\gamma) \;\Leftrightarrow\;$ the lower triangle commutes.

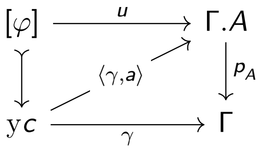
Proof of Lemma.



$c \Vdash a : A(\gamma) \;\Leftrightarrow\;$ the lower triangle commutes.

$c \Vdash \varphi : \mathtt{Cof}(\gamma)$ and $c \Vdash \big(u : [\varphi] \to A\big)(\gamma) \;\Leftrightarrow\;$ the outer square commutes.

Proof of Lemma.



$c \Vdash a : A(\gamma) \;\Leftrightarrow\;$ the lower triangle commutes.

$c \Vdash \varphi : \mathtt{Cof}(\gamma)$ and $c \Vdash \big(u : [\varphi] \to A\big)(\gamma) \;\Leftrightarrow\;$ the outer square commutes.

$c \Vdash e : u =_{\varphi} a(\gamma)$

$\Leftrightarrow\; c \Vdash e : \prod_{x:[\varphi]} \mathsf{Eq}_A(ux, a)(\gamma)$

$\Leftrightarrow\;$ for all $f : d \to c$ in $\mathcal{C}$, $d \Vdash x : [\varphi](\gamma.f)$ returns $d \Vdash e_f(x) : \mathsf{Eq}_A(ux, a)(\gamma.f)$

$\Leftrightarrow\;$ the top triangle commutes. QED.

Proof of Theorem.

Suppose $\Gamma \vdash \alpha : \mathsf{TFib}(A)$.

Thus for all $\gamma \colon \mathrm{y}c \to \Gamma$, we have $c \Vdash \alpha_\gamma : \mathsf{TFib}(A)(\gamma)$, coherently in γ.

Proof of Theorem.

Suppose $\Gamma \vdash \alpha : \mathsf{TFib}(A)$.

Thus for all $\gamma : \mathrm{y}c \to \Gamma$, we have $c \Vdash \alpha_\gamma : \mathsf{TFib}(A)(\gamma)$, coherently in γ.

Note that

$$\begin{aligned}\mathsf{TFib}(A) &= \prod_{\varphi:\mathtt{Cof}} \prod_{u:[\varphi]\to A} \sum_{a:A} \prod_{x:[\varphi]} \mathsf{Eq}_A(ux, a) \\ &= \prod_{(\varphi,u)\,:\,A^+} \sum_{a:A} u =_\varphi a\end{aligned}$$

We thus obtain

$$c \Vdash \alpha_\gamma : \prod_{(\varphi,u)\,:\,A^+} \sum_{a:A} (u =_\varphi a)(\gamma)\ .$$

Proof of Theorem (cont'd).

By Kripke–Joyal semantics of $\prod$ and $\sum$, we have for every $f: d \to c$ in $\mathcal{C}$, if

$$d \Vdash (\varphi, u) : A^+(\gamma.f) \tag{3}$$

then

$$d \Vdash \alpha_{\gamma.f}(\varphi, u)^0 : A(\gamma.f) \tag{4}$$

and

$$d \Vdash \alpha_{\gamma.f}(\varphi, u)^1 : \left(u =_\varphi \alpha_{\gamma.f}(\varphi, u)^0\right)(\gamma.f) \tag{5}$$

and, for any $g: d' \to d$,

$$\alpha_{\gamma.f}(\varphi, u).g = \alpha_{(\gamma.fg)}(\varphi[g], u[g])\,. \tag{6}$$

An application of Kripke–Joyal semantics: Uniform trivial fibration (III)

Unfolding the condition (3) yields the following commutative diagram.

$$
\begin{array}{ccc}
[\varphi.f] & \xrightarrow{\langle \gamma.f, u_f \rangle} & \Gamma.A \\
\big\downarrow & & \big\downarrow p_A \\
\mathrm{y}d & \xrightarrow[\gamma.f]{} & \Gamma
\end{array}
$$

Unfolding the condition (3) yields the following commutative diagram.

$$
\begin{array}{ccc}
[\varphi.f] & \xrightarrow{\langle \gamma.f, u_f \rangle} & \Gamma.A \\
\downarrow & & \downarrow p_A \\
\mathrm{y}d & \xrightarrow[\gamma.f]{} & \Gamma
\end{array}
$$

Lemma applied to (4) and (5) yields the following commuting diagram.

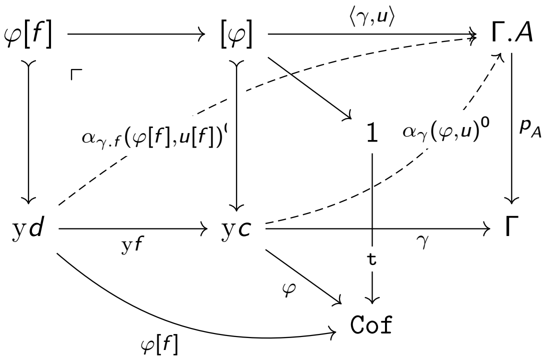
$$
\begin{array}{ccc}
\varphi.f & \xrightarrow{\langle \gamma.f, u_f \rangle} & \Gamma.A \\
\downarrow & \nearrow^{\alpha_{\gamma.f}(\varphi, u)^0} & \downarrow p_A \\
\mathrm{y}d & \xrightarrow[\gamma.f]{} & \Gamma
\end{array}
$$

An application of Kripke–Joyal semantics: Uniform trivial fibration (III)

Thus forcing TFib(A) produces diagonal fillers

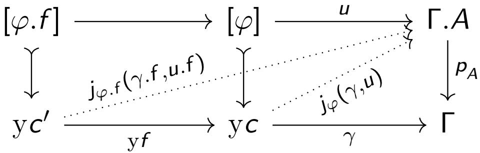
$$j_\varphi(\gamma, u) \triangleq \alpha_{\gamma.f}(\varphi, u)^0$$

for each lifting problem as in the right hand square below:



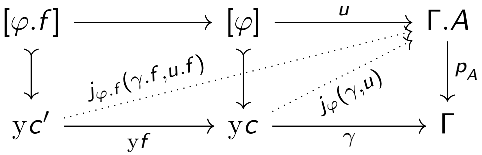
Proof of Theorem (cont'd) – converse argument

If $p_A \colon \Gamma.A \to \Gamma$ is a uniform trivial fibration then in particular for every *basic* cofibration $[\varphi] \rightarrowtail \mathrm{y}c$ and square as on the right below, there is a diagonal filler $j_\varphi(\gamma, u)$ as indicated.

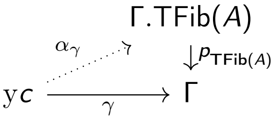


Proof of Theorem (cont'd) – converse argument

If $p_A : \Gamma.A \to \Gamma$ is a uniform trivial fibration then in particular for every *basic* cofibration $[\varphi] \rightarrowtail \mathrm{y}c$ and square as on the right below, there is a diagonal filler $j_\varphi(\gamma, u)$ as indicated.



By the lemma, this corresponds to an element $\alpha_\gamma : \mathrm{y}c \to \mathsf{TFib}(A)$ over $\gamma : \mathrm{y}c \to \Gamma$,



Proof of Theorem (cont'd) – converse argument

The uniformity condition says exactly that for all $f : c' \to c$, the elements α_γ cohere,

$\alpha_{(\gamma . \mathrm{y} f)} = \alpha_\gamma \circ f$.

Proof of Theorem (cont'd) – converse argument

The uniformity condition says exactly that for all $f : c' \to c$, the elements α_γ cohere,

$\alpha_{(\gamma.\mathrm{y}f)} = \alpha_\gamma \circ f$.

By Yoneda for the slice category $\mathscr{E}/\Gamma$ that there is a term $\Gamma \vdash \alpha : \mathsf{TFib}(A)$. QED.

Next ...

Further use of Kripke–Joyal semantics for dependent type theory in

- Extending to uniform **fibrations** using an interval I.
- Showing the fibrancy of path types.
- Showing the universe of fibrations is itself fibrant.
- Showing Frobenius property of fibrations.

References I

Steve Awodey. "A cubical model of homotopy type theory". In: *Ann. Pure Appl. Logic* 169.12 (2018).

Steve Awodey. "Natural models of homotopy type theory". In: *Math. Structures Comput. Sci.* 28.2 (2018).

Cyril Cohen et al. "Cubical type theory: a constructive interpretation of the univalence axiom". In: *21st International Conference on Types for Proofs and Programs*. Vol. 69. 2018.

Nicola Gambino and Christian Sattler. "The Frobenius condition, right properness, and uniform fibrations". In: *Journal of Pure and Applied Algebra* 221.12 (2017).

Ian Orton and Andrew M. Pitts. "Axioms for Modelling Cubical Type Theory in a Topos". In: *Logical Methods in Computer Science* 14 (4 2018).

References II

Christian Sattler. "The Equivalence Extension Property and Model Structures". In: (2017). url: http://arxiv.org/abs/1704.06911.

The End

Thanks for your attention!