Realizability Models of $\lambda 2$

Niels van der Weide

Overview of the next 45 minutes

- Last week, we saw that $\lambda 2$ does not have a nontrivial set-theoretic model
- We can encode the booleans

$$\mathbb{B} = \prod X.X \to X \to X$$

- The power set (i.e., set of all subsets) is then given by $X \to \mathbb{B}$
- Because the function type $B_1 \to B_2$ is interpreted as the set of all functions, we get a contradiction

Overview of the next 45 minutes

- Last week, we saw that $\lambda 2$ does not have a nontrivial set-theoretic model
- We can encode the booleans

$$\mathbb{B} = \prod X.X \to X \to X$$

- The power set (i.e., set of all subsets) is then given by $X \to \mathbb{B}$
- Because the function type $B_1 \to B_2$ is interpreted as the set of all functions, we get a contradiction

Goal of this talk: we give a model of $\lambda 2$ based on PERs, which we saw in the previous talk

Table of Contents

Outline

Recap of $\lambda 2$

Brief Recap on D-Sets

Realizability Semantics for $\lambda 2$

Components of $\lambda 2$

$\lambda 2$ has

- Sort contexts: list of type variables

$$X_1, \ldots, X_m$$

- Types depend on a sort context:

$$X_1, \ldots, X_m \vdash B$$

Components of $\lambda 2$

$\lambda 2$ has

- Sort contexts: list of type variables

$$X_1, \ldots, X_m$$

- Types depend on a sort context:

$$X_1, \ldots, X_m \vdash B$$

- Type contexts: list of of term variables

$$x_1 : A_1, \ldots, x_n : A_n$$

- Terms depend on a sort context and a type context:

$$X_1, \ldots, X_m \,|\, x_1 : A_1, \ldots, x_n : A_n \vdash t : B$$

Components of $\lambda 2$

$\lambda 2$ has

- Sort contexts: list of type variables

$$X_1, \ldots, X_m$$

- Types depend on a sort context:

$$X_1, \ldots, X_m \vdash B$$

- Type contexts: list of of term variables

$$x_1 : A_1, \ldots, x_n : A_n$$

- Terms depend on a sort context and a type context:

$$X_1, \ldots, X_m \,|\, x_1 : A_1, \ldots, x_n : A_n \vdash t : B$$

We abbreviate $X_1, \ldots, X_m$ by Θ and $x_1 : A_1, \ldots, x_n : A_n$ by Γ

Types in $\lambda 2$

We have three ways to make types:

- Type variables

$$X_1, \ldots, X_m \vdash X_i$$

- Function types

$$\frac{\Theta \vdash B_1 \qquad \Theta \vdash B_2}{\Theta \vdash B_1 \to B_2}$$

- Polymorphism

$$\frac{\Theta, Y \vdash B}{\Theta \vdash \prod Y.B}$$

Rules of $\lambda 2$ (part 1)

We have five ways to make terms

- Variables

$$\Theta \mid x_1 : A_1, \ldots, x_n : A_n \vdash x_i : A_i$$

- λ-abstraction

$$\frac{\Theta \mid \Gamma, y : B_1 \vdash t : B_2}{\Theta \mid \Gamma \vdash \lambda y : B_1 . t : B_1 \to B_2}$$

- Application

$$\frac{\Theta \mid \Gamma \vdash f : B_1 \to B_2 \qquad \Theta \mid \Gamma \vdash t : B_1}{\Theta \mid \Gamma \vdash f\ t : B_2}$$

Rules of $\lambda 2$ (part 2)

We have five ways to make terms

- Polymorphic λ-abstraction

$$\frac{\Theta, Y \mid \Gamma \vdash t : B}{\Theta \mid \Gamma \vdash \Lambda Y.t : \prod Y.B}$$

- Application

$$\frac{\Theta \mid \Gamma \vdash f : \prod Y.B \qquad \Theta \vdash A}{\Theta \mid \Gamma \vdash f\, A : B[Y \mapsto A]}$$

Outline

Important Concepts

To give realizability semantics for $\lambda 2$, we use the following concepts that we discussed throughout the reading group

- Partial combinatory algebras
- Combinatory completeness
- D-sets (or assemblies)
- Modest sets

We will recall D-sets and modest sets

Brief Recap: D-sets

Definition

A D-**set** consists of a set X together with a a relation $\Vdash_X$ between D and X such that for each $x \in X$ there is $d \in D$ with $d \Vdash x$.

When we write $d \Vdash_X x$, we say d **realizes** x.

Brief Recap: *D*-sets

Code		Math
$\Lambda xy.x$	$\Vdash$	true
$\Lambda xy.y$	$\Vdash$	false

Brief Recap: Morphisms of$D$-sets

Definition

Suppose we have D-sets $(X, \Vdash_X)$ and $(Y, \Vdash_Y)$. Let $f : X \to Y$ be a function. We say that an element $e \in D$ **tracks** f if for all $x \in X$ and $d \in D$ with $d \Vdash_X x$ we have $e \cdot d \Vdash_Y f(x)$ and $e \cdot d \downarrow$.

Idea: e is an implementation of f in a the PCA D.

Brief Recap: Morphisms of D-sets

Definition

Suppose we have D-sets $(X, \Vdash_X)$ and $(Y, \Vdash_Y)$. Let $f : X \to Y$ be a function. We say that an element $e \in D$ **tracks** f if for all $x \in X$ and $d \in D$ with $d \Vdash_X x$ we have $e \cdot d \Vdash_Y f(x)$ and $e \cdot d \downarrow$.

Idea: e is an implementation of f in a the PCA D.

Definition

Suppose we have D-sets $(X, \Vdash_X)$ and $(Y, \Vdash_Y)$. A **morphism** from $(X, \Vdash_X)$ to $(Y, \Vdash_Y)$ is a function $f : X \to Y$ for which there exists $e : D$ that tracks f.

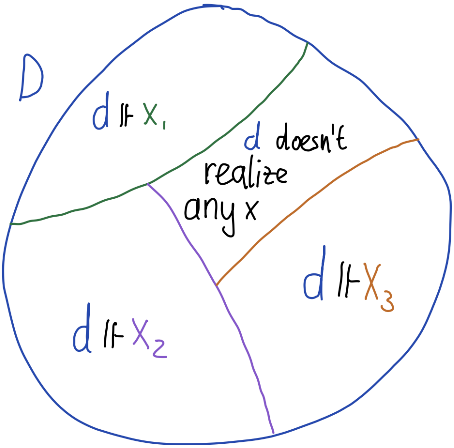
Brief Recap: Modest Sets

Definition

A D-set $(X, \Vdash_X)$ is called **modest** if for all $d \in D$ and $x, x' \in X$ we have $x = x'$ whenever $d \Vdash_X x$ and $d \Vdash_X x'$.

Concretely: a D-set $(X, \Vdash_X)$ is modest if each element in D realizes at most 1 element

Brief Recap: Modest Sets



$$X = \{X_1, X_2, X_3\}$$

Brief Recap: PERs

Definition

A D-set $(X, \Vdash_X)$ is called a $\mathbf{PER}_D$ **object** if $X \subseteq \mathcal{P}(D)/\{\emptyset\}$ such that for all $A, B \in X$ we have

- $A \cap B = \emptyset$ if $A \neq B$
- $d \Vdash A$ iff $d \in A$

Brief Recap: PERs

Definition

A D-set $(X, \Vdash_X)$ is called a $\mathbf{PER}_D$ **object** if $X \subseteq \mathcal{P}(D)/\{\emptyset\}$ such that for all $A, B \in X$ we have

- $A \cap B = \emptyset$ if $A \neq B$
- $d \Vdash A$ iff $d \in A$

Equivalently, one can represent PER_D objects as **partial equivalence relations on** D

A partial equivalence relation on D is a relation on D that is symmetric and transitive.

Brief Recap: PERs and Modest Sets

Theorem

We have maps

- f *sending modest sets to* PER_D *objects*
- g *sending* PER_D *objects to modest sets*

such that $f(g(R)) = R$ *and* $g(f(X, \Vdash_X))$ *is isomorphic to* $(X, \Vdash_X)$.

Brief Recap: PERs and Modest Sets

Theorem

We have maps

- f *sending modest sets to* PER_D *objects*
- g *sending* PER_D *objects to modest sets*

such that $f(g(R)) = R$ *and* $g(f(X, \Vdash_X))$ *is isomorphic to* $(X, \Vdash_X)$.

Why is this important?

- The collection of all modest sets is very "large": i.e., we don't have a set of modest sets
- This theorem tells us modest sets are always of a certain shape: they are PER_D objects
- And we have a set of PER_D objects

This is key in interpreting polymorphism

Constructions on PERs

In the last talk, we saw various constructions of D-sets

- Given D-sets $(X, \Vdash_X)$ and $(Y, \Vdash_Y)$, then their product (carrier: $X \times Y$) is a D-set
- Given D-sets $(X, \Vdash_X)$ and $(Y, \Vdash_Y)$, then $(X, \Vdash_X) \to (Y, \Vdash_Y)$ (carrier: all morphisms from $(X, \Vdash_X)$ to $(Y, \Vdash_Y)$) is a D-set

Constructions on PERs

In the last talk, we saw various constructions of D-sets

- Given D-sets $(X, \Vdash_X)$ and $(Y, \Vdash_Y)$, then their product (carrier: $X \times Y$) is a D-set
- Given D-sets $(X, \Vdash_X)$ and $(Y, \Vdash_Y)$, then $(X, \Vdash_X) \to (Y, \Vdash_Y)$ (carrier: all morphisms from $(X, \Vdash_X)$ to $(Y, \Vdash_Y)$) is a D-set

One can show that these also are constructions on modest sets!

Constructions on PERs

- Given modest sets $(X, \Vdash_X)$ and $(Y, \Vdash_Y)$, then their product (carrier: $X \times Y$) is a modest set
- Given a D-set $(X, \Vdash_X)$ and a modest set $(Y, \Vdash_Y)$, then $(X, \Vdash_X) \to (Y, \Vdash_Y)$ (carrier: all morphisms from $(X, \Vdash_X)$ to $(Y, \Vdash_Y)$) is a modest set

Note that we only need that $(Y, \Vdash_Y)$ is modest

Functions to a modest set

- Let $(X, \Vdash_X)$ be a D-set and let $(Y, \Vdash_Y)$ be a modest set

Functions to a modest set

- Let $(X, \Vdash_X)$ be a D-set and let $(Y, \Vdash_Y)$ be a modest set
- Recall: since Y is modest, we have $y = y'$ whenever $d \Vdash_Y y$ and $d \Vdash_Y y'$ for all $d \in D$ and $y, y' \in Y$

Functions to a modest set

- Let $(X, \Vdash_X)$ be a D-set and let $(Y, \Vdash_Y)$ be a modest set
- Recall: since Y is modest, we have $y = y'$ whenever $d \Vdash_Y y$ and $d \Vdash_Y y'$ for all $d \in D$ and $y, y' \in Y$
- Goal: $(X, \Vdash_X) \to (Y, \Vdash_Y)$ is modest
- Suppose, we have morphisms f, g from $(X, \Vdash_X)$ to $(Y, \Vdash_Y)$, and suppose e tracks both f and g
- To show $f = g$, we need to show $f(x) = g(x)$ for all $x \in X$

Functions to a modest set

- Let $(X, \Vdash_X)$ be a D-set and let $(Y, \Vdash_Y)$ be a modest set
- Recall: since Y is modest, we have $y = y'$ whenever $d \Vdash_Y y$ and $d \Vdash_Y y'$ for all $d \in D$ and $y, y' \in Y$
- Goal: $(X, \Vdash_X) \to (Y, \Vdash_Y)$ is modest
- Suppose, we have morphisms f, g from $(X, \Vdash_X)$ to $(Y, \Vdash_Y)$, and suppose e tracks both f and g
- To show $f = g$, we need to show $f(x) = g(x)$ for all $x \in X$
- Since Y is modest, it suffices to find a d such that $d \Vdash_Y f(x)$ and $d \Vdash_Y g(x)$

Functions to a modest set

- Let $(X, \Vdash_X)$ be a D-set and let $(Y, \Vdash_Y)$ be a modest set
- Recall: since Y is modest, we have $y = y'$ whenever $d \Vdash_Y y$ and $d \Vdash_Y y'$ for all $d \in D$ and $y, y' \in Y$
- Goal: $(X, \Vdash_X) \to (Y, \Vdash_Y)$ is modest
- Suppose, we have morphisms f, g from $(X, \Vdash_X)$ to $(Y, \Vdash_Y)$, and suppose e tracks both f and g
- To show $f = g$, we need to show $f(x) = g(x)$ for all $x \in X$
- Since Y is modest, it suffices to find a d such that $d \Vdash_Y f(x)$ and $d \Vdash_Y g(x)$
- Since X is a D-set, we can find c such that $c \Vdash_X x$
- Now $e \cdot c$ realizes both $f(x)$ and $g(x)$

Functions to a modest set

- Let $(X, \Vdash_X)$ be a D-set and let $(Y, \Vdash_Y)$ be a modest set
- Recall: since Y is modest, we have $y = y'$ whenever $d \Vdash_Y y$ and $d \Vdash_Y y'$ for all $d \in D$ and $y, y' \in Y$
- Goal: $(X, \Vdash_X) \to (Y, \Vdash_Y)$ is modest
- Suppose, we have morphisms f, g from $(X, \Vdash_X)$ to $(Y, \Vdash_Y)$, and suppose e tracks both f and g
- To show $f = g$, we need to show $f(x) = g(x)$ for all $x \in X$
- Since Y is modest, it suffices to find a d such that $d \Vdash_Y f(x)$ and $d \Vdash_Y g(x)$
- Since X is a D-set, we can find c such that $c \Vdash_X x$
- Now $e \cdot c$ realizes both $f(x)$ and $g(x)$
- Hence, $(X, \Vdash_X) \to (Y, \Vdash_Y)$ is modest

Constructions on PERs

Since modest sets and PERs are equivalent, we get constructions on PERs!

- Given PER_D objects $(X, \Vdash_X)$ and $(Y, \Vdash_Y)$, then their product (carrier: $X \times Y$) is a PER_D object
- Given a D-set $(X, \Vdash_X)$ and a PER_D object $(Y, \Vdash_Y)$, then $(X, \Vdash_X) \to (Y, \Vdash_Y)$ (carrier: all morphisms from $(X, \Vdash_X)$ to $(Y, \Vdash_Y)$) is a PER_D object

Outline

Overview

- Overall goal: model of $\lambda 2$
- Our focus shall be on how to interpret polymorphism
- The main idea is to interpret types as PERs and terms as morphisms
- However, we must also take type variables into account

Dependence on Variables

Let us recall terms and types

$$X_1, \ldots, X_m \vdash B$$

$$X_1, \ldots, X_m \,|\, x_1 : A_1, \ldots, x_n : A_n \vdash t : B$$

Note:

- Both types and terms can contain variables
- There are two kinds of variables: type variables and term variables
- Types only contain type variables, and terms can contain both term and type variables

Interpretation of Types

When we give the semantics of types, we must take the interpretation of the type variables into account

- Let $X_1, \ldots, X_m \vdash B$ be a type
- Its interpretation is a function that assigns to all PERs $\theta_1, \ldots, \theta_m$ a PER

$$[\![X_1, \ldots, X_m \vdash B]\!](\theta_1, \ldots, \theta_m)$$

Realizability Semantics for $\lambda 2$: Types

To interpret

$$X_1, \ldots, X_m \vdash B$$

Given PERs $\theta_1, \ldots, \theta_m$, we have a PER

$$[\![X_1, \ldots, X_m \vdash B]\!](\theta_1, \ldots, \theta_m)$$

Realizability Semantics for $\lambda 2$: Types

To interpret

$$X_1, \ldots, X_m \vdash B$$

Given PERs $\theta_1, \ldots, \theta_m$, we have a PER

$$[\![X_1, \ldots, X_m \vdash B]\!](\theta_1, \ldots, \theta_m)$$

For variables, we use the θ_i:

$$[\![X_1, \ldots, X_m \vdash X_i]\!](\theta_1, \ldots, \theta_m) = \theta_i$$

Realizability Semantics for $\lambda 2$: Types

To interpret

$$X_1, \ldots, X_m \vdash B$$

Given PERs $\theta_1, \ldots, \theta_m$, we have a PER

$$[\![X_1, \ldots, X_m \vdash B]\!](\theta_1, \ldots, \theta_m)$$

For variables, we use the θ_i:

$$[\![X_1, \ldots, X_m \vdash X_i]\!](\theta_1, \ldots, \theta_m) = \theta_i$$

We interpret function types via the collection of all morphisms

$$\begin{aligned} &[\![\Theta \vdash B_1 \to B_2]\!](\theta_1, \ldots, \theta_m) \\ &= [\![\Theta \vdash B_1]\!](\theta_1, \ldots, \theta_m) \to [\![\Theta \vdash B_2]\!](\theta_1, \ldots, \theta_m) \end{aligned}$$

Realizability Semantics for $\lambda 2$: Types

To interpret

$$X_1, \ldots, X_m \vdash B$$

Given PERs $\theta_1, \ldots, \theta_m$, we have a PER

$$[\![X_1, \ldots, X_n \vdash B]\!](\theta_1, \ldots, \theta_m)$$

To interpret polymorphism, we use **an intersection**.

$$[\![\Theta \vdash \prod Y.B]\!](\theta_1, \ldots, \theta_m) = \bigcap_{X \in \mathrm{PER}_D} [\![\Theta \vdash B]\!](\theta_1, \ldots, \theta_m, X)$$

Realizability Semantics for $\lambda 2$: Types

To interpret

$$X_1, \ldots, X_m \vdash B$$

Given PERs $\theta_1, \ldots, \theta_m$, we have a PER

$$[\![X_1, \ldots, X_n \vdash B]\!](\theta_1, \ldots, \theta_m)$$

To interpret polymorphism, we use **an intersection**.

$$[\![\Theta \vdash \prod Y.B]\!](\theta_1, \ldots, \theta_m) = \bigcap_{X \in \mathrm{PER}_D} [\![\Theta \vdash B]\!](\theta_1, \ldots, \theta_m, X)$$

B has an additional type variable, which we interpret by X

Realizability Semantics for $\lambda 2$: Types

To interpret

$$X_1, \ldots, X_m \vdash B$$

Given PERs $\theta_1, \ldots, \theta_m$, we have a PER

$$[\![X_1, \ldots, X_n \vdash B]\!](\theta_1, \ldots, \theta_m)$$

To interpret polymorphism, we use **an intersection**.

$$[\![\Theta \vdash \prod Y.B]\!](\theta_1, \ldots, \theta_m) = \bigcap_{X \in \mathsf{PER}_D} [\![\Theta \vdash B]\!](\theta_1, \ldots, \theta_m, X)$$

B has an additional type variable, which we interpret by X
Note: we use that the intersection of PER_D objects is a PER_D object.

Realizability Semantics for $\lambda 2$: Contexts

Suppose we have types $\Theta \vdash A_i$ where $\Theta = X_1, \ldots, X_m$.
This means that we have $[\![\Theta \vdash A_i]\!](\theta_1, \ldots, \theta_m)$ for all PERs $\theta_1, \ldots, \theta_m$.
We interpret contexts as follows:

$$\begin{aligned}&[\![x : A_1, \ldots, x_n : A_n]\!](\theta_1, \ldots, \theta_m)\\ &= [\![\Theta \vdash A_1]\!](\theta_1, \ldots, \theta_m) \times \ldots \times [\![\Theta \vdash A_n]\!](\theta_1, \ldots, \theta_m)\end{aligned}$$

Realizability Semantics for $\lambda 2$: Contexts

Suppose we have types $\Theta \vdash A_i$ where $\Theta = X_1, \ldots, X_m$.
This means that we have $[\![\Theta \vdash A_i]\!](\theta_1, \ldots, \theta_m)$ for all PERs $\theta_1, \ldots, \theta_m$.
We interpret contexts as follows:

$$\begin{aligned} &[\![x : A_1, \ldots, x_n : A_n]\!](\theta_1, \ldots, \theta_m) \\ &= [\![\Theta \vdash A_1]\!](\theta_1, \ldots, \theta_m) \times \ldots \times [\![\Theta \vdash A_n]\!](\theta_1, \ldots, \theta_m) \end{aligned}$$

Note: the empty context is interpreted as the PER with a single element and every element of D realizes that element

Realizability Semantics for $\lambda 2$: Terms

Next we look at terms

$$\Theta \mid \Gamma \vdash t : B$$

The interpretation of term assigns to each well-typed term $\Theta \mid \Gamma \vdash t : B$ an element

$$[\![\Theta \mid \Gamma \vdash t : B]\!] : \bigcap_{\theta_1,\ldots,\theta_m} [\![\Gamma]\!](\theta_1,\ldots,\theta_m) \to [\![\Theta \vdash B]\!](\theta_1,\ldots,\theta_m)$$

Realizability Semantics for $\lambda 2$: Terms

Next we look at terms

$$\Theta \mid \Gamma \vdash t : B$$

The interpretation of term assigns to each well-typed term $\Theta \mid \Gamma \vdash t : B$ an element

$$[\![\Theta \mid \Gamma \vdash t : B]\!] : \bigcap_{\theta_1,\ldots,\theta_m} [\![\Gamma]\!](\theta_1, \ldots, \theta_m) \to [\![\Theta \vdash B]\!](\theta_1, \ldots, \theta_m)$$

Key idea: terms are interpreted **uniformly** over all type variables

Realizability Semantics for $\lambda 2$: Terms

$$\frac{\Theta, X \mid \Gamma \vdash t : B}{\Theta \mid \Gamma \vdash \Lambda Y.t : \prod X.B}$$

Induction hypothesis: we have

$$[\![\Theta, Y \mid \Gamma \vdash t : B]\!] \in \bigcap_{\theta, X \in \mathsf{PER}_D} [\![\Gamma]\!](\theta) \to [\![\Theta, Y \vdash B]\!](\theta, X)$$

Realizability Semantics for $\lambda 2$: Terms

$$\frac{\Theta, X \mid \Gamma \vdash t : B}{\Theta \mid \Gamma \vdash \Lambda Y.t : \prod X.B}$$

Induction hypothesis: we have

$$[\![\Theta, Y \mid \Gamma \vdash t : B]\!] \in \bigcap_{\theta, X \in \mathrm{PER}_D} [\![\Gamma]\!](\theta) \to [\![\Theta, Y \vdash B]\!](\theta, X)$$

Our goal: we need to make

$$[\![\Theta \mid \Gamma \vdash \Lambda Y.t : \prod X.B]\!] \in \bigcap_{\theta \in \mathrm{PER}_D} [\![\Gamma]\!](\theta) \to \bigcap_{X \in \mathrm{PER}_D} [\![\Theta, Y \vdash B]\!](\theta, X)$$

Realizability Semantics for $\lambda 2$: Terms

$$\frac{\Theta, X \mid \Gamma \vdash t : B}{\Theta \mid \Gamma \vdash \Lambda Y.t : \prod X.B}$$

Induction hypothesis: we have

$$[\![\Theta, Y \mid \Gamma \vdash t : B]\!] \in \bigcap_{\theta, X \in \mathsf{PER}_D} [\![\Gamma]\!](\theta) \to [\![\Theta, Y \vdash B]\!](\theta, X)$$

Our goal: we need to make

$$[\![\Theta \mid \Gamma \vdash \Lambda Y.t : \prod X.B]\!] \in \bigcap_{\theta \in \mathsf{PER}_D} [\![\Gamma]\!](\theta) \to \bigcap_{X \in \mathsf{PER}_D} [\![\Theta, Y \vdash B]\!](\theta, X)$$

If $x \in [\![\Gamma]\!](\theta)$ for all θ, then for all θ and X

$$[\![\Theta, Y \mid \Gamma \vdash t : B]\!](x) \in [\![\Theta, Y \vdash B]\!](\theta, X)$$

Realizability Semantics for $\lambda 2$: Terms

$$\frac{\Theta \mid \Gamma \vdash f : \prod Y.B \qquad \Theta \vdash A}{\Theta \mid \Gamma \vdash f\ A : B[Y \mapsto A]}$$

Induction hypothesis: we have

$$[\![\Theta \mid \Gamma \vdash f : \prod Y.B]\!] \in \bigcap_{\theta \in \mathrm{PER}_D} [\![\Gamma]\!](\theta) \to \bigcap_{X \in \mathrm{PER}_D} [\![\Theta \vdash B]\!](\theta, X)$$

and a map assigning to PERs θ a PER $[\![\Theta \vdash A]\!](\theta)$

Realizability Semantics for $\lambda 2$: Terms

$$\frac{\Theta \mid \Gamma \vdash f : \prod Y.B \qquad \Theta \vdash A}{\Theta \mid \Gamma \vdash f\ A : B[Y \mapsto A]}$$

Induction hypothesis: we have

$$[\![\Theta \mid \Gamma \vdash f : \prod Y.B]\!] \in \bigcap_{\theta \in \mathrm{PER}_D} [\![\Gamma]\!](\theta) \to \bigcap_{X \in \mathrm{PER}_D} [\![\Theta \vdash B]\!](\theta, X)$$

and a map assigning to PERs θ a PER $[\![\Theta \vdash A]\!](\theta)$
Our goal: we need to make

$$[\![\Theta \mid \Gamma \vdash f\ A : B[Y \mapsto A]]\!] \in \bigcap_{\theta \in \mathrm{PER}_D} [\![\Gamma]\!](\theta) \to [\![\Theta, Y \vdash B]\!](\theta, [\![\Theta \vdash A]\!](\theta))$$

Realizability Semantics for $\lambda 2$: Terms

$$\frac{\Theta \mid \Gamma \vdash f : \prod Y.B \qquad \Theta \vdash A}{\Theta \mid \Gamma \vdash f\ A : B[Y \mapsto A]}$$

Induction hypothesis: we have

$$[\![\Theta \mid \Gamma \vdash f : \prod Y.B]\!] \in \bigcap_{\theta \in \mathrm{PER}_D} [\![\Gamma]\!](\theta) \to \bigcap_{X \in \mathrm{PER}_D} [\![\Theta \vdash B]\!](\theta, X)$$

and a map assigning to PERs θ a PER $[\![\Theta \vdash A]\!](\theta)$
Our goal: we need to make

$$[\![\Theta \mid \Gamma \vdash f\ A : B[Y \mapsto A]]\!] \in \bigcap_{\theta \in \mathrm{PER}_D} [\![\Gamma]\!](\theta) \to [\![\Theta, Y \vdash B]\!](\theta, [\![\Theta \vdash A]\!](\theta))$$

If $x \in [\![\Gamma]\!](\theta)$ for all θ, then, for all θ,

$$[\![\Theta \mid \Gamma \vdash f : \prod Y.B]\!](x) \in [\![\Theta \vdash B]\!](\theta, [\![\Theta \vdash A]\!](\theta))$$

Conclusion

We constructed a realizability model of $\lambda 2$. **Key ideas**:

- Types are interpreted via PERs
- The model is not set-theoretic: the function type is not interpreted as the set of all functions, but as all morphisms of D-sets
- Polymorphic function types are interpreted via intersections
- Terms are interpreted as a uniform family of morphisms, which is the basis for interpreting polymorphism