Theories, interpretations, and pretoposes

Jesse Han

May 21, 2018

Abstract

We explicate the relationships between interpretations of first-order theories, interpretations of first-order structures, and elementary functors between categories of definable sets and their pretopos completions.

Contents

Notation and conventions

- Unless explicitly stated otherwise, we are always working in multisorted classical first-order logic; every sort S has its own equality symbol $=_S$.
- Unless explicitly stated otherwise, definable means definable without parameters.
- Unadorned variables in formulas will generally stand for finite tuples of appropriately-sorted variables.
- Similarly, when we say "sort" we mean a finite tuple of the basic sorts of the language. We allow the empty finite tuple of sorts, and denote it by 1.
- If we have already mentioned a tuple of variables x, then we will write S_x for the sort corresponding to x.
- Greek letters (except for σ and ρ, which will usually denote automorphisms) will usually mean first-order formulas.

- If $\mathcal{L}$ is a first-order language, we write $\mathsf{Sorts}(\mathcal{L})$, $\mathsf{Functions}(\mathcal{L})$, $\mathsf{Relations}(\mathcal{L})$, $\mathsf{Constants}(\mathcal{L})$, and $\mathsf{Formulas}(\mathcal{L})$ to mean the collections of sorts, function symbols, relation symbols, constant symbols, and first-order $\mathcal{L}$-formulas, respectively.

1 Theories and abstract interpretations

Definition 1.1. Fix a first-order language $\mathcal{L}$. An **$\mathcal{L}$-theory** (when we wish to avoid emphasizing the ambient language, we will just say **theory**) is a set of $\mathcal{L}$-sentences.

Definition 1.2. If φ is an $\mathcal{L}$-formula, we say that the **sort**, or **ambient sort**, of φ is the sort S_x corresponding to the tuple x of free variables in φ. If φ is a sentence, i.e. an $\mathcal{L}$-formula with no free variables, we understand that in this case the ambient sort of φ is the empty sort.

Definition 1.3. Let T be an $\mathcal{L}$-theory and let T' be an $\mathcal{L}'$-theory. An **abstract interpretation** I of T in T', written $I : T \to T'$, consists of the following data:

1. For each sort $S \in \mathsf{Sorts}(\mathcal{L})$, we assign an $\mathcal{L}'$-formula $I(S)$, with the convention that the empty sort 1 of T is sent to the empty sort 1 of T'.

   We also require this assignment to preserve the operation of forming finite tuples of sorts: if the basic sorts B_1 and B_2 are sent to the sorts S_1 and S_2, then the sort B_1B_2 is sent to the sort S_1S_2 (where we write sorts right next to each other to indicate their concatenation as tuples of basic sorts).

2. For each basic non-logical symbol $c \in \mathsf{Constants}(\mathcal{L})$, $R \in \mathsf{Relations}(\mathcal{L})$, or $f \in \mathsf{Functions}(\mathcal{L})$, we assign an $\mathcal{L}'$-formula $I(c)$ (resp. $I(R)$, $I(f)($, such that the assignment is compatible with sorts: if a nonlogical symbol X belongs to the sort S_X, then $I(X)$ is a subset of the sort $I(S_X)$.

3. By an induction on complexity of formulas, the above assignments determine an assignment of $\mathcal{L}$-formulas to $\mathcal{L}'$-formulas, in particular of $\mathcal{L}$-sentences to $\mathcal{L}'$-sentences. We finally require that if ψ is an $\mathcal{L}$-sentence such that $T \models \psi$, then $T' \models I(\psi)$.

Definition 1.4. If an abstract interpretation $I : T \to T'$ interprets all equalities $x = y$ as equivalence relations with singleton equivalence classes, we say that I is a **strict** abstract interpretation. This is sometimes called a **definition** of T in T'.

2 Structures and concrete interpretations

Definition 2.1. Let T be an $\mathcal{L}$-theory. A **model** M of T consists of the following data:

1. For each sort $S \in \mathsf{Sorts}(\mathcal{L})$, we assign a set $M(S)$, with the convention that the empty sort 1 of T is sent to the empty product 1 of **Set**.

   We also require that this assignment preserves the operation of forming finite tuples of sorts: if the basic sorts B_1 and B_2 are sent to the set $M(B_1)$ and $M(B_2)$, then the sort B_1B_2 is sent to the set $M(B_1B_2) = M(B_1) \times M(B_2)$.

2. For each basic non-logical symbol $c \in \mathsf{Constants}(\mathcal{L})$, $R \in \mathsf{Relations}(\mathcal{L})$, or $f \in \mathsf{Functions}(\mathcal{L})$, we assign a set $M(c)$ (resp. $M(R)$, $M(f)$), such that the assignment is compatible with sorts: if a nonlogical symbol X belongs to the sort S_X, then $M(X)$ belongs to $M(S_X)$.

3. By an induction on complexity of formulas, the above assignments determine an assignment of $\mathcal{L}$-formulas to sets. Since the assignments above respect sorts, any $\mathcal{L}$-sentence, which lives in the empty sort 1, will be sent to a subset of the terminal set 1, of which there are only two possibilities, the entire terminal set 1, or $\varnothing$. We finally require that if ψ is an $\mathcal{L}$-sentence such that $T \models \psi$, then $M(\psi) = 1$.

Definition 2.2. Let $\mathcal{L}$ be a language. An **$\mathcal{L}$-structure** is a model M of the empty $\mathcal{L}$-theory.

Definition 2.3. Let M be an $\mathcal{L}$-structure. By Definition 2.1, M includes the data of an function $\mathsf{Sentences}(\mathcal{L}) \to \{\varnothing, 1\}$. The **$\mathcal{L}$-theory** of M, written $\mathbf{Th}(M)$, is the preimage of 1 along this function:

$$\mathbf{Th}(M) \stackrel{\text{df}}{=} \{\psi \in \mathsf{Sentences}(\mathcal{L}) \,\big|\, M(\psi) = 1\}.$$

Definition 2.4. Let M be an $\mathcal{L}$-structure. A **definable set** U of M is some set U such that $U = M(\varphi(x))$ for some $\mathcal{L}$-formula $\varphi(x)$.

Definition 2.5. Let M be an $\mathcal{L}$-theory, and let M' be an $\mathcal{L}'$-theory. A **concrete interpretation** (f, f^*) of M in M', written $(f, f^*) : M \to M'$, consists of the following data:

1. For each sort S of $\mathcal{L}$, we assign a definable set U_S of M' and a surjective function $f_S : U_S \twoheadrightarrow M(S)$.

   We also require that this assignment preserves the operation of forming finite tuples of sorts: if the basic sorts B_1 and B_2 are assigned the functions $f_{B_1} : U_{B_1} \twoheadrightarrow B_1$ and $f_{B_2} : U_{B_2} \twoheadrightarrow B_2$, then the sort B_1B_2 is assigned $U_{B_1B_2} = U_{B_1} \times U_{B_2}$ and the function $f_{B_1B_2} = f_{B_1} \times f_{B_2}$.

2. These surjective functions must satisfy the following property: for every sort S and for definable subset $V \subseteq M(S)$, the preimage f^*V of V along f is definable in M'.

Definition 2.6. If a concrete interpretation (f, f^*) additionally satisfies that f is injective, we say that (f, f^*) is a **strict concrete interpretation**.

Remark 2.7. Since we have defined models so that models always interpret tuples of basic sorts as products of basic sorts, this gives the correct definition of a model in the 1-sorted case, when there is a unique basic sort.

An important consequence of the fact that models interpret tuples of basic sorts as products of basic sorts is that the definable projection functions from tuples of sorts to their subtuples are interpreted as literal projections in any model.

Example 2.8. Let M be an $\mathcal{L}$-structure, and let $\sigma : M \to M$ be an automorphism. Then $\sigma_S : M(S) \to M(S)$ is a strict concrete interpretation $(\sigma, \sigma^*) : M \to M$. Since definable sets are invariant under automorphisms, σ^* is the identity.

3 Categories of definable sets and elementary functors

The starting point for first-order categorical logic is the identification of a theory with its category of definable sets.

Definition 3.1. Let T be a first-order $\mathcal{L}$-theory. The **category of definable sets** $\mathbf{Def}(T)$ comprises:

$$\mathbf{Def}(T) \stackrel{\text{df}}{=} \begin{cases} \text{Objects: } \mathsf{Formulas}(\mathcal{L})/_{\sim}, \text{ where } \phi(x) \sim \psi(x) \iff T \models \phi(x) \leftrightarrow \psi(x), \\ \text{Morphisms: } \{\phi \in \mathsf{Formulas}(\mathcal{L}) \,|\, T \models \phi \text{ is a function } \varphi(x) \to \psi(y)\}/_{\sim}. \end{cases}$$

Remark 3.2. Above, we are defining morphisms to be equivalence classes of graphs of definable functions, where we are using the same equivalence relation as we did for objects. By the completeness theorem for first-order logic, the notion of equivalence of formulas proved in defining the objects of $\mathbf{Def}(T)$ is the same as *T-provable equivalence*, i.e. $\varphi(x) \sim \psi(y) \iff T \vdash \varphi(x) \leftrightarrow \psi(x)$. By the downward Löwenheim-Skolem theorem, it also suffices to check $\sim$-equivalence by checking if two formulas have the same points on just those models whose sizes are less than or equal to the size of the theory.

Remark 3.3. Every sort S has its own equality symbol $=_S$, and the formula $x =_S x$ represents S in $\mathbf{Def}(T)$. Sorts are "maximal" objects in $\mathbf{Def}(T)$; just as, syntactically, they provide the contexts in which we reason about formulas, every definable set $A \in \mathbf{Def}(T)$ belongs to, and thus embeds into, a sort.

Notation 3.4. From now on, unless if we are explicitly working in a model, "definable set" will mean an equivalence class of formulas in the above sense.

We now note some important features of the category $\mathbf{Def}(T)$:

Proposition 3.5. $\mathbf{Def}(T)$ *has all finite limits.*

Proof. By the canonical product-equalizer decomposition for limits (see e.g. [2]), it suffices to see that $\mathbf{Def}(T)$ has all equalizers and finite products.

Given a pair of morphisms in $\mathbf{Def}(T)$, as in $f, g : \varphi(x) \rightrightarrows \psi(y)$, we can write a first-order formula $\mathrm{eq}(f, g)$ with a canonical inclusion $\mathrm{eq}(f, g) \hookrightarrow \varphi(x)$ whose points in any model will be the equalizer of the functions f and g: $\mathrm{eq}(f, g) \stackrel{\text{df}}{\equiv} f(x) =_{S_x} g(x)$. We will show that $\mathrm{eq}(f, g)$ has the expected universal property. So suppose the universal property fails, and we have in $\mathbf{Def}(T)$ a commutative diagram like this:

$$\begin{array}{ccccc} \mathrm{eq}(f,g) & \longrightarrow & X & \overset{g}{\underset{f}{\rightrightarrows}} & Y \\ {\scriptstyle h_1}\uparrow\uparrow{\scriptstyle h_2} & \nearrow & & & \\ E & & & & \end{array}$$

such that $h_1 \neq h_2$. Then by definition of the equivalence relation defining objects and morphisms in $\mathbf{Def}(T)$, this is witnessed by a model M such that we can take two formulas $\phi_1(x_1, x_2)$ and $\phi_2(x_1, x_2)$ representing the graphs of h_1 and h_2 and $M(\phi_1) \neq M(\phi_2)$. This contradicts that $M(\mathrm{eq}(f, g))$ is the equalizer of $M(X \rightrightarrows Y)$. We conclude that $\mathbf{Def}(T)$ has all equalizers.

Now we will show that $\mathbf{Def}(T)$ has all finite products. We note that the empty product (i.e. a terminal object) is the equivalence class of the empty sort, which we think of as being the ambient sort for the empty tuple of variables; we can view *any* finite tuple of variables as being padded by an empty variable of the empty sort, and so *any* formula is vacuously a definable function from itself to the empty sort, so the empty sort is a terminal object. Now, for $n \geqslant 1$, let $\varphi_1(x_1), \ldots, \varphi_n(x_n)$ be a finite collection of $\mathcal{L}$-formulas. Then, just as for the case of equalizers, we can write a first-order formula $(\varphi_1 \times \cdots \times \varphi_n)(x_1, \ldots, x_n)$ whose points in any model will be the product of the

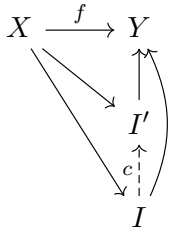
sets $\varphi_1(x_1), \ldots, \varphi_n(x_n)$: replacing x_i with distinct variables of the same sort as necessary, so that $x_1, \ldots, x_n$ are all distinct (this respects T-provable equivalence), put

$$\varphi_1 \times \cdots \times \varphi_n(x_1, \ldots, x_n) \stackrel{\text{df}}{\equiv} \varphi_1(x_1) \wedge \cdots \wedge \varphi_n(x_n).$$

(Note that in any model, this is a subset of $M(S_{x_1 \ldots x_n}) = M(S_{x_1}) \times \cdots \times M(S_{x_n})$.)

Now we can repeat the argument we used for equalizers word-for-word, except replacing equalizer diagrams with product diagrams throughout. We conclude that $\mathbf{Def}(T)$ has all finite products. □

Definition 3.6. Let $f : X \to Y$ be a morphism in some category. The **image** of f, if it exists, is a subobject $I \hookrightarrow Y$ of Y such that there is a factorization of f through I, and if f factors through any other subobject I' of Y, then there is a unique map of subobjects $I \stackrel{c}{\to} I'$ making the diagram



commute.

Proposition 3.7. *Every morphism $f : \varphi(x) \to \psi(y)$ has an image.*

Proof. We claim that the image of a definable function f is described by the formula $\mathrm{im}(f) \stackrel{\text{df}}{\equiv} \exists x \Gamma(f)(x, y)$. This is equipped with the canonical projection to $\psi(y)$. By how we defined models' interpretations of formulas built with existential quantifiers, $M(\exists x \Gamma(f)(x, y))$ is the literal projection of $M(\Gamma(f)(x, y)$ to $M(\psi(y))$, and is therefore the image of $M(f)$. The same argument we used for equalizers and limits then shows that $\mathrm{im}(f)$ is the image of f in $\mathbf{Def}(T)$. □

Definition 3.8. Let $S_1, \ldots, S_n$ be finitely many subobjects of X in some category. The **finite sup** of $S_1, \ldots, S_n$, if it exists, is a subobject $S_1 \vee \cdots \vee S_n$ of X such that any other subobject S' containing $S_1, \ldots, S_n$ contains $S_1 \vee \cdots \vee S_n$.

Proposition 3.9. $\mathbf{Def}(T)$ *has all finite sups.*

Proof. Note that since the empty set is a definable subset of every definable set, $\mathbf{Def}(T)$ always has empty sups. It then suffices to obtain binary sups. Let $\varphi(x)$ and $\psi(x)$ be subobjects of $\theta(x)$ in $\mathbf{Def}(T)$. Then $T \models \varphi(x) \to \theta(x) \& \& \psi(x) \to \theta(x)$; therefore, $T \models \varphi(x) \vee \psi(x) \to \theta(x)$. In every model M of T, $M(\varphi(x) \vee \psi(x))$ is the sup of the subsets $M(\varphi(x)), M(\psi(x))$ of $M(\theta(x)$. Now we argue as in the previous two propositions: if this were not true in $\mathbf{Def}(T)$, this must be witnessed in a model, an impossibility, so $\varphi(x) \vee \psi(x)$ is the sup of $\varphi(x)$ and $\psi(x)$ in $\mathbf{Def}(T)$. □

Definition 3.10. Let $\mathbf{C}$ be a category with pullbacks.

1. Let $f : X \to Y$ be a morphism in $\mathbf{C}$ which has an image. We say that the **image** $\mathrm{im}(f)$ **is stable** if for every morphism $g : Z \to Y$, the two horizontal maps at the bottom of the

diagram of pullback squares

$$\begin{array}{ccccc} X & \xrightarrow{y} & Y & \longleftarrow & \mathrm{im}(f) \\ \uparrow & & \uparrow{\scriptstyle g} & & \uparrow \\ X \times_Y Z & \longrightarrow & Z & \longleftarrow & \mathrm{im}(f) \times_Y Z \end{array}$$

have the same image (colloquially, "the pullback of the image is the image of the pullback").

2. Let $S_1, \ldots, S_n$ be subobjects of B in $\mathbf{C}$ which have a finite sup. We say that the **finite sup $\bigvee_{i \leqslant n} S_i$ is stable** if for every morphism $g : Z \to B$, the pullback of $\bigvee_{i \leqslant n} S_i$ along g, as in the pullback diagram

$$\begin{array}{ccc} \bigvee_{i \leqslant n} S_i & \longrightarrow & B \\ \uparrow & & \uparrow \\ \left(\bigvee_{i \leqslant n} S_i\right) \times_B Z & \longrightarrow & Z \end{array}$$

is the finite sup of the pullbacks of the S_i (this makes sense because the pullback of a monomorphism is always a monomorphism).

We say that **C has stable images and stable finite sups** if $\mathbf{C}$ has all images and finite sups, and they are all stable.

Proposition 3.11. $\mathbf{Def}(T)$ *has stable images and stable finite sups.*

Proof. Let $f : X \to Y$ and $g : Z \to Y$ be morphisms in $\mathbf{Def}(T)$. The pullback of X and Z with respect to f and g is the equalizer of the following pair of maps:

$$X \times Z \overset{f \circ \pi_X}{\underset{g \circ \pi_Z}{\rightrightarrows}} Y$$

and can therefore be represented by the formula $(f(x) = g(z))$, and the image of its canonical projection to Z can be represented by $\exists x\, (f(x) = g(z))$.

Similarly, the image of $\mathrm{im}(f) \times_Y Z$ in Z is represented by the formula $\exists y\, (y = g(z))$. For any model M, it is true that $M\big(\exists x\, (f(x) = g(z))\big) = M\big(\exists y\, (y = g(z))\big)$. By the completeness theorem for first-order logic, $T \models \exists x\, (f(x) = g(z)) \leftrightarrow \exists y\, (y = g(z))$. Since f and g were arbitrary, we conclude $\mathbf{Def}(T)$ has stable images.

Now, let $S_1, \ldots, S_n$ be subobjects of B in $\mathbf{Def}(T)$. It is easy to check, using the description of pullbacks we used in the previous paragraph, that the pullback of any subobject of B along a map $g : Z \to B$ is just the (formula describing the) preimage of that subobject along g. Since models interpret finite sups as unions, and in the category of sets, taking preimages commute with unions, then we may argue as before using the completeness theorem that finite sups are stable. This applies equally well to the empty sup; we conclude $\mathbf{Def}(T)$ has stable finite sups. □

Definition 3.12. Let $\mathbf{C}$ be a category with pullbacks and finite sups.

We say that $\mathbf{C}$ is **Boolean** if for every subobject S of B in $\mathbf{C}$, there exists another subobject $\neg S$ such that the pullback of S and $\neg S$ over B is the empty sup in B, and the finite sup $S \vee \neg S$ is all of B.

Proposition 3.13. $\mathbf{Def}(T)$ *is Boolean.*

Proof. If $T \models \varphi(x) \to \psi(x)$, then

$$T \models \neg\big(\varphi(x) \wedge (\neg\varphi(x)) \wedge B\big)$$

and

$$T \models \varphi(x) \vee (\neg\varphi(x) \wedge B) \leftrightarrow B.$$

□

In [4], Makkai and Reyes showed that the properties we studied above actually *characterize* those categories of the form $\mathbf{Def}(T)$ for some theory T. They call such categories (Boolean) *logical categories*.

Definition 3.14. A category $\mathbf{C}$ is a **logical category** if it has finite limits, has stable images, and has stable sups.

Proposition 3.15. *Let T be a theory. Then $\mathbf{Def}(T)$ is a Boolean logical category.*

Proof. We saw in Propositions 3.5, 3.7, 3.9, 3.11, and 3.13 that for every theory T, $\mathbf{Def}(T)$ has finite limits, stable images, stable sups, and is Boolean. □

By requiring the preservation of those categorical properties which define logical categories, we can define what it means for a functor to be a morphism of logical categories. [1]

Definition 3.16. Let $\mathbf{C}$ and $\mathbf{C}'$ be logical categories. An **elementary functor $\mathbf{C} \to \mathbf{C}'$** is a functor which preserves finite limits, finite sups of subobjects, and images.

Elementary functors preserve complements whenever they exist:

Lemma 3.17. *Let A be a subobject of C in a logical category $\mathbf{C}$. Suppose that A has a complement B inside C. Let $I : \mathbf{C} \to \mathbf{C}'$ be an elementary functor. Then $I(A)$ and $I(B)$ are complements inside $I(C)$.*

Proof. A and B satisfy that their pullback $A \cap B$ over $\mathbf{C}$ is the empty sup $\varnothing$ of $\mathbf{C}$. Since I preserves pullbacks and finite sups, $I(A) \cap I(B) = I(A \cap B) = I(\varnothing) = \varnothing$. Similarly, $A \vee B = C$ and since I preserves finite sups, $I(C) = I(A \vee B) = I(A) \vee I(B)$. □

Up until now, we have seen how the categorical features of a Boolean logical category correspond to the logical operations of a theory: finite limits correspond to meets and forming products of sorts, images (of projections) correspond to existential quantification, finite sups correspond to finite joins, and complementation corresponds to taking negations. So modulo an induction on complexity of formulas, elementary functors should correspond to interpretations.

We will devote much of our energy in the rest of this section, and the following section, into making this intuition precise.

Proposition 3.18. *Let T and T' be $\mathcal{L}$ and $\mathcal{L}'$-theories. Let $F : \mathbf{Def}(T) \to \mathbf{Def}(T')$ be an elementary functor. Then F induces a strict interpretation $I_F : T \to T'$.*

[1] In [4] these are called, aptly, *logical functors*, but we follow the terminology of [3], wherein logical functors between pretoposes are called *elementary*.

Proof. Let S be a basic sort in $\mathbf{Def}(T)$. We choose a representative formula of the equivalence class $F(S)$ and make that $I_F(S)$. After specifying the basic sorts, I_F is determined on all sorts. Since 1 is the limit of the empty diagram in $\mathbf{Def}(T)$ and elementary functors preserve finite limits, the empty sort 1 of T gets sent to the empty sort 1 of T'. This satisfies part 1 of the definition of an abstract interpretation.

Let c be a constant symbol of sort S. Then in $\mathbf{Def}(T)$, c is interpreted as a nullary function $1 \xrightarrow{c} S$. Applying F, we get a nullary function $1 \to F(S)$. We now abuse notation and identify the formula $I_F(S)$ with the definable set in $\mathbf{Def}(T')$ it represents. Since F preserves finite products, $I_F(S) \simeq F(S)$ and we define $I_F(c)$ to be the definable nullary function $1 \to F(S) \xrightarrow{\sim} I_F(S)$ of T'.

Let R be a relation symbol of sort S. Then $F(R) \hookrightarrow F(S)$; composing by the isomorphism $F(S) \simeq I_F(S)$, we have that $F(R) \hookrightarrow I_F(S)$. We define $I_F(R)$ to be the image of this embedding.

Let f be a function symbol whose graph relation $\Gamma(f)$ is of sort S_1S_2. Then we define $I_F(f)$ by defining $I_F(\Gamma(f))$ just as we did for a relation symbol.

We have now defined I_F up to part 2 of the definition of an abstract interpretation. By an induction on complexity of formulas, I_F determines a map of $\mathcal{L}$-sentences to $\mathcal{L}'$-sentences. We must now check that whenever $T \models \psi$, $T' \models I_F(\psi)$. We will show that at each step of the inductive definition of $I_F : \mathsf{Formulas}(\mathcal{L}) \to \mathsf{Formulas}\mathcal{L}'$, the truth of sentences is preserved.

I_F preserves negations of formulas: given $\psi(x)$ and $\neg\psi(x)$ of sort S, we have that $I(\psi(x))$ and $I(\neg\psi(x))$ have empty intersection (since elementary functors preserves pullbacks and finite sups) and that $I(\psi(x)) \vee I(\neg\psi(x)) = I(S)$ (since elementary functors preserve finite sups), so $I(\neg\psi(x)) \equiv \neg I(\psi(x))$. Since F preserves finite sups and pullbacks, I_F preserves disjunctions and conjunctions of formulas.

By viewing sentences as subobjects of the empty sort 1, we see that I_F preserves the truth of negations, conjunctions, and disjunctions of sentences.

I_F preserves existential quantification of formulas: generally, $\exists x\varphi(x, y)$ is the image of the projection $\varphi(x, y) \xrightarrow{\pi_{S_y}} S_y$, and since F preserves finite limits, it preserves images and projection maps from products, so

$$I_F(\exists x\varphi(x, y)) \simeq F(\exists x\varphi(x, y)) \simeq \mathrm{im}\left(I(\varphi(x, y)) \xrightarrow{\pi_{F(S_Y))}} I(S_y)\right) \simeq \exists x F(\varphi(x, y)) \simeq \exists x I_F(\varphi(x, y)).$$

Note that this applies equally well when y is an empty variable belonging to the empty sort 1, and we are considering a sentence $\exists x\varphi(x)$. If $T \models \exists x\varphi(x)$, then in $\mathbf{Def}(T)$, $\exists x\varphi(x) = 1$. Since F is elementary, it preserves terminal objects, so $F(\exists x\varphi(x)) = I_F(\exists x\varphi(x)) = 1$.

Now, let ψ be an atomic $\mathcal{L}$-sentence. Then ψ is of the form $R(t)$ or $t_1 =_S t_2$ for terms t, t_1, t_2. Since ψ is atomic, these terms are definable constants of T, and so can be thought of as nullary definable functions $1 \xrightarrow{t} R(t)$ or $1 \underset{t_2}{\overset{t_1}{\rightrightarrows}} S$. Applying F, we see that $F(t)$ goes into $F(R)$ and $F(t_1) = F(t_2)$ (since F is at least a function), so $I_F(t) \in I_F(R)$ and $I_F(t_1) = I_F(t_2)$. This provides the base of the induction and completes the proof. □

This has an obvious converse:

Proposition 3.19. *Let $I : T \to T'$ be a strict abstract interpretation. Let $[\psi(x)]$ denote the $\sim$-equivalence class of ψ (as in the definition of $\mathbf{Def}(T)$).*

Let $X \xrightarrow{f} Y$ be a definable function in T. Then the assignment

$$X \xrightarrow{f} Y \mapsto [X] \xrightarrow{[f]} [Y]$$

determines an elementary functor $F_I : \mathbf{Def}(T) \to \mathbf{Def}(T')$.

Proof. F_I is well-defined on objects since I is an interpretation, and so preserves the sentences which assert that any two given formulas are provably equivalent. F_I is well-defined on morphisms since I is a strict interpretation, so that for every definable function f of T, T' proves that $I(f)$ is a function.

Now we check that F_I preserves finite products. We saw in Proposition 3.5 that any finite collection of formulas $\varphi_1(x_1), \ldots, \varphi_n(x_n)$ has a canonical product $\varphi_1 \times \cdots \times \varphi_n(x_1, \ldots, x_n)$. Given any tuple $(a_i \in \psi_i(x))$, T proves that there is a unique $\overline{a} \in \varphi_1 \times \cdots \times \varphi_n(x_1, \ldots, x_n)$ which projects to each a_i. Since I was an interpretation, T' proves the same thing about $I(\varphi_1 \times \varphi_n(x_1, \ldots, x_n))$, so $I(\varphi_1 \times \varphi_n(x_1, \ldots, x_n))$ satisfies the universal property of the product $I(\varphi_1(x_1)) \times \cdots \times I(\varphi_n(x_n))$.

The same argument works for equalizers, so F_I preserves finite limits.

We saw in Proposition 3.9 that finite sups in $\mathbf{Def}(T)$ are given by disjunctions of formulas. I_T was defined by induction to send $\varphi(x) \vee \psi(x)$ to $I(\varphi(x)) \vee I(\psi(x))$, so F_I preserves finite sups.

Similarly, I_T was defined by induction to send an existentially quantified formula $\exists x \varphi(x, y)$ to $\exists x I(\varphi(x, y))$. So F_I preserves images of the canonical projections between tuples of sorts. We saw in Proposition 3.7 that the image of a definable function f in $\mathbf{Def}(T)$ is given by binding the domain variable of the graph relation $\Gamma(f)(x, y)$, so that $\mathrm{im}(f) = \exists x \Gamma(x, y)$. Therefore, F_I preserves the images of definable functions. □

We have seen how strict interpretations between theories induce elementary functors between the categories of definable sets, and vice-versa. It is natural to ask if this implements a one-one correspondence, and if such a correspondence is functorial. Indeed, we will finish this section by showing that the constructions of Proposition 3.18 and Proposition 3.19 essentially implement a equivalence of categories between theories and Boolean logical categories.

Definition 3.20. Let $I, I' : T_1 \to T_2$ be strict abstract interpretations. A **transformation** $\eta : I \to T'$ comprises the following data:

1. For every sort S of T_1, a definable function $\eta_S : I(S) \to I'(S)$ in T_2, such that
2. for every formula $\varphi(x)$ of sort S in T_1, $\eta_S : I(S) \to I'(S)$ restricts to a definable function $\eta_{\varphi(x)} : I(\varphi(x)) \to I'(\varphi(x))$.

Definition 3.21. Let us say that two strict interpretations $I, I' : T_1 \to T_2$ are **provably equivalent** if for every formula φ of T_1,

$$T_1 \vdash I(\varphi) \leftrightarrow I'(\varphi).$$

Similarly, we say that two transformations $I \underset{\eta'}{\overset{\eta}{\rightrightarrows}} I'$ are **provably equivalent** if for every sort S, the graph of η_S is T_2-provably equivalent to the graph of η'_S.

Let $[I]_\sim$ and $[I']_\sim$ be two equivalence classes of interpretations mod provable equivalence. We define a transformation $[I]_\sim \to [I']_\sim$ to be an equivalence class of transformations $I \to I'$ mod

provable equivalence. It is easy to see that this does not depend on our choice of representatives I, I' of the equivalence classes of interpretations.

Proposition 3.22. *Let $I, I' : T_1 \to T_2$ be strict abstract interpretations, and let $\eta : I \to I'$ be a transformation. Then the construction $I \mapsto F_I$ from Proposition 3.19 determines from η a natural transformation of elementary functors $\widetilde{\eta} : F_I \to F_{I'}$.*

Proof. A natural transformation $F_I \to F_{I'}$ assigns to each object A of $\mathbf{Def}(T_1)$ a definable function $F_I(A) \to F_{I'}(A)$ which is natural with respect to definable functions $A \to B$.

Our definition of natural transformation already gives us a collection of definable functions $\widetilde{\eta}_A : F_I(A) \to F_{I'}(A)$ (by taking the mod-T_2-provable equivalence classes of the definable functions η_A) and so it remains to verify that for every definable function $f : A \to B$, the following diagram commutes:

$$\begin{array}{ccc} F_I(A) & \xrightarrow{\widetilde{\eta}_A} & F_{I'}(A) \\ {\scriptstyle F_I(f)}\downarrow & & \downarrow{\scriptstyle F_{I'}(f)} \\ F_I(B) & \xrightarrow[\widetilde{\eta}_B]{} & F_{I'}(B) \end{array}$$

□

Definition 3.23. We define the 2**-category of first-order theories**, written **Th**, by

$$\mathbf{Th} \stackrel{\text{df}}{=} \begin{cases} \text{Objects: first-order theories} \\ \text{1-morphisms: strict interpretations mod provable equivalence} \\ \text{2-morphisms: transformations.} \end{cases}$$

Definition 3.24. We define the 2**-category of Boolean logical categories**, written **BoolLogCat**, by

$$\mathbf{BoolLogCat} \stackrel{\text{df}}{=} \begin{cases} \text{Objects: Boolean logical categories} \\ \text{1-morphisms: elementary functors} \\ \text{2-morphisms: natural transformations.} \end{cases}$$

Our goal for the remainder of this section is to prove the following theorem.

Theorem 3.25. *The assignment*

$$\begin{array}{ccc} T_1 & & \mathbf{Def}(T_1) \\ I \left(\Downarrow \eta\right) I' & \mapsto & F_I \left(\Downarrow \widetilde{\eta}\right) F_{I'} \\ T_2 & & \mathbf{Def}(T_2) \end{array}$$

determines an equivalence of 2*-categories* $\mathbf{Def}(-) : \mathbf{Th} \xrightarrow{\sim} \mathbf{BoolLogCat}$.

Lemma 3.26. *If $I, I' : T \to T'$ are different morphisms in* **Th***, then $F_I \neq F'_I$ as elementary functors.*

Similarly, if $\eta_1, \eta_2 : I \to I'$ are different transformations, then $\widetilde{\eta}_1 \neq \widetilde{\eta}_2$ as natural transformations of elementary functors.

Proof. By definition I and I' are different morphisms in **Th** if and only if they are not provably equivalent. This means that there exists some formula φ of T such that $I(\varphi)$ are $I'(\varphi)$ do not represent the same equivalence class in $\mathbf{Def}(T')$, so $F_I(\varphi) \neq F_{I'}(\varphi)$.

Similarly, $\eta_1 \neq \eta_2$ in **Th** if for some sort S, the definable functions η_1 and η_2 are not provably equivalent. Then $(\widetilde{\eta}_1)_S \neq (\widetilde{\eta}_2)_S$, so $\widetilde{\eta}_1 \neq \widetilde{\eta}_2$. □

Lemma 3.27. *Let $F : \mathbf{Def}(T_1) \to \mathbf{Def}(T_2)$ be an elementary functor. Let $\widetilde{F}$ be the elementary functor induced by the interpretation I_F, as in Proposition 3.19.*

Then $F = \widetilde{F}$.

Proof. We use $[-]_\sim$ to denote an equivalence class modulo provable equivalence. Unwinding definitions, we see that for any formula φ of T_1, we have

$$F\left([\varphi]_\sim\right) = [I_F(\varphi)]_\sim = \widetilde{F}\left([\varphi]_\sim\right)$$

□

Lemma 3.28. *A natural transformation of elementary functors $\epsilon : F \to F'$, where $F, F' : \mathbf{Def}(T_1) \to \mathbf{Def}(T_2)$ determines a transformation (mod provable equivalence) of interpretations $\eta : I_F \to I_{F'}$, where $I_F, I_{F'} : T_1 \to T_2$.*

Then $\widetilde{\eta} = \epsilon$.

Proof. For each sort $S \in \mathbf{Def}(T)$, put $\eta_S \stackrel{\text{df}}{=} [\epsilon_S]_\sim$. Since I_F and $I_{F'}$ are obtained by choosing representatives from $\sim$-equivalence classes and ϵ_S was natural, η_S is a definable function (mod provable equivalence) $I_F(S) \to I_{F'}(S)$ and for any formula $\varphi(x)$ of sort S restricts to a definable function (mod provable equivalence) $I_F(\varphi(x)) \to I_{F'}(\varphi(x))$.

Then, as we did for Lemma 3.27, we see that for an arbitrary formula φ,

$$\widetilde{\eta}_{[\varphi]_\sim} = [\eta_\varphi]_\sim = \epsilon_{[\varphi]_\sim}.$$

□

The following definition is essentially due to Makkai and Reyes (see "internal theories" in [4]).

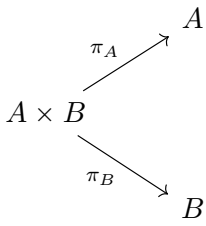
Definition 3.29. Let $\mathbf{C}$ be a Boolean logical category. The **internal logic** of $\mathbf{C}$ is a theory $T_{\mathbf{C}}$ defined as follows:

1. We form a language $\mathcal{L}_{\mathbf{C}}$ by associating to every object A of $\mathbf{C}$ a sort $\mathsf{L}(A)$, to every morphism $f : A \to B$ of $\mathbf{C}$ a function symbol $\mathsf{L}(f)$ of sort $\mathsf{L}(A) \to \mathsf{L}(B)$, to every morphism from the terminal object $1 \xrightarrow{c} A$ a constant symbol $\mathsf{L}(c)$ of sort $\mathsf{L}(A)$, and to every subobject $B \hookrightarrow A$ a relation symbol $\mathsf{L}_A(B)$ of sort $\mathsf{L}(A)$.
2. We form a theory $T_{\mathbf{C}}$ whose sentences specify the Boolean logical category structure on $\mathbf{C}$. That is,
   (a) For every identity morphism $\mathrm{id}_A : A \to A$,

$$T_{\mathbf{C}} \vdash (\forall x)\left(\mathsf{L}(\mathrm{id}_A)(x) = x\right).$$

(b) If $g_1 \circ g_2 = h$ in $\mathbf{C}$, then

$$T_{\mathbf{C}} \vdash (\forall x)\,(\mathsf{L}(g_1)(\mathsf{L}(g_2)(x)) = \mathsf{L}(h)(x))\,.$$

(c) If 

is a product diagram in $\mathbf{C}$, then

$$T_{\mathbf{C}} \vdash (\forall a \in \mathsf{L}(A))\,(\forall b \in \mathsf{L}(B))\,(\exists! c \in \mathsf{L}(A \times B))\,\big[\mathsf{L}(\pi_A)(c) = a \;\;\wedge\;\; \mathsf{L}(\pi_B)(c) = b\big]$$

(c.f. Proposition 3.5).

(d) If $C \xrightarrow{u} A \underset{g}{\overset{f}{\rightrightarrows}} B$ is an equalizer diagram in $\mathbf{C}$, then

$$T_{\mathbf{C}} \vdash (\forall a \in \mathsf{L}(A))\,(\exists! c \in \mathsf{L}(C))\,\big[\mathsf{L}(f)(a) = \mathsf{L}(g)(a) \leftrightarrow \mathsf{L}(u)(c) = a\big].$$

(e) If $B_1 \vee B_2$ is the finite sup of B_1 and B_2 in the subobject lattice of A in $\mathbf{C}$, then

$$T_{\mathbf{C}} \vdash (\forall a \in \mathsf{L}(A))\,\big[\mathsf{L}_A(B_1 \vee B_2)(a) \leftrightarrow \mathsf{L}_A(B_1)(a) \vee \mathsf{L}_A(B_2)(a)\big].$$

(f) If $C \xrightarrow{f} A$ is a morphism in $\mathbf{C}$ with image $B = \mathrm{im}(f) \hookrightarrow A$, then

$$T_{\mathbf{C}} \vdash (\forall a \in \mathsf{L}(A))\,\big[(\exists c \in \mathsf{L}(C))\,(\mathsf{L}(f)(c) = a) \leftrightarrow \mathsf{L}_A(B)(a)\big].$$

(g) If $1_{\mathbf{C}}$ is the terminal object of $\mathbf{C}$, then

$$T_{\mathbf{C}} \vdash (\exists! a \in \mathsf{L}(1_{\mathbf{C}}))\big[a = a\big].$$

(h) If $\varnothing_{\mathbf{C}}$ is the empty sup in the subobject lattice of an object A, then

$$T_{\mathbf{C}} \vdash (\forall a \in \mathsf{L}(A))\big[\neg \mathsf{L}(\varnothing_{\mathbf{C}})(a)\big].$$

The above data clearly equip $\mathbf{C}$ with a canonical functor $\mathsf{L} : \mathbf{C} \to \mathbf{Def}(T_{\mathbf{C}})$, by identifying each object of $\mathbf{C}$ with its corresponding sort in $\mathcal{L}_{\mathbf{C}}$, and by identifying each morphism of $\mathbf{C}$ with its corresponding function symbol in $\mathcal{L}_{\mathbf{C}}$.

Theorem 3.30. *The canonical functor $\mathsf{L} : \mathbf{C} \to \mathbf{Def}(T_{\mathbf{C}})$ is an equivalence of categories.*

Proof. We will show that the functor is essentially surjective by inducting on the complexity of formulas in $T_{\mathbf{C}}$ and showing that at every step of the induction, we can find a corresponding object of $\mathbf{C}$ gotten by carrying out the construction of the formula "internally" in $\mathbf{C}$.

The base of the induction is almost completely provided by the definition of $T_{\mathbf{C}}$. We remark that, given a comparison of two terms $t_1 = t_2$, we can treat constants as nullary functions and reduce to the case where t_1 is some composition of function symbols $\mathsf{L}(\overline{f})$ applied to some tuple of variables

$\overline{x}$ and where t_2 is some composition of function symbols $\mathsf{L}(\overline{g})$ applied to some tuple of variables $\overline{y}$. Letting S denote the sort of the equality relation we're using to compare t_1 and t_2, we have that

$$\mathsf{L}(\overline{f})(\overline{x}) = \mathsf{L}(\overline{g})(\overline{y}) \simeq \mathsf{L}(S_x \times_S S_y)$$

(where $S_x \times_S S_y$ is the pullback computed in $\mathbf{C}$ of S_y and S_y over S with respect to the morphisms $\overline{f}$ and $\overline{g}$.)

This is because by viewing a pullback as a composition of products and equalizers, our axiomatization of $T_{\mathbf{C}}$ gives that $\mathsf{L}(S_x \times_S S_y)$ injects into $\mathsf{L}(S_x \times S_y)$ with image $\mathsf{L}_A(S_x \times_S S_y)$, and

$$T_{\mathbf{C}} \vdash (\forall a \in \mathsf{L}(S_x))(\forall b \in \mathsf{L}(S_y))(\exists! c \in \mathsf{L}_A(S_x \times_S S_y))\big[\mathsf{L}(\overline{f})(a) = \mathsf{L}(\overline{g})(b) \leftrightarrow \mathsf{L}(\pi_{S_x})(c) = a \wedge \mathsf{L}(\pi_{S_y})(c) = b\big],$$

which defines a bijection ϵ^{-1} from $\mathsf{L}(\overline{f})(\overline{x}) = \mathsf{L}(\overline{g})(\overline{y})$ to $\mathsf{L}_A(S_x \times_S S_y)$, so that

$$T_{\mathbf{C}} \vdash \big(\mathsf{L}(\overline{f})(\overline{x}) = \mathsf{L}(\overline{g})(\overline{y})\big) \leftrightarrow \big(\mathsf{L}_A(S_x \times_S S_y)(\epsilon^{-1}(x, y))\big).$$

Note that above, we are in the following situation. There is a canonical bijection $\widehat{\epsilon} : \mathsf{L}(S_x \times S_y) \to \mathsf{L}(S_x) \times \mathsf{L}(S_y)$, given by sending a c to the pair of projections $\big(\mathsf{L}(\pi_{S_x})(c), \mathsf{L}(\pi_{S_y})(c)\big)$, and it restricts along the inclusion $\mathsf{L}(S_x \times_S S_y) \hookrightarrow \mathsf{L}(S_x \times S_y)$ to the bijection $\epsilon : L(S_x \times_S S_y) \xrightarrow{\sim} \mathsf{L}(\overline{f})(\overline{x}) = \mathsf{L}(\overline{g})(\overline{y})$. That is, the following diagram, with the horizontal arrows bijections, commutes:

$$\begin{array}{ccc} \mathsf{L}(S_x) \times \mathsf{L}(S_y) & \xrightarrow{\widehat{\epsilon^{-1}}} & \mathsf{L}(S_x \times S_y) \\ \uparrow & & \uparrow \\ \mathsf{L}(\overline{f})(\overline{x}) = \mathsf{L}(\overline{g})(\overline{y}) & \xleftarrow[\epsilon]{} & \mathsf{L}(S_x \times_S S_y). \end{array}$$

So the definable bijection ϵ is not just any definable bijection. It has the additional property that a diagram like the one above commutes, where in particular the right vertical arrow is L of an embedding in $\mathbf{C}$. It is clear that for atomic formulas, the definable bijections witnessing the base of the induction all satisfy this additional property.

We proceed with the induction. Let us say that an $\mathcal{L}_{\mathbf{C}}$-formula $\varphi(x)$, viewed as an object of $\mathbf{Def}(T_{\mathbf{C}})$, has a *counterpart* if there exists an object $A_{\varphi(x)}$ of $\mathbf{C}$ such that there is a definable bijection $\mathsf{L}(A_{\varphi(x)}) \overset{\epsilon_{\varphi(x)}}{\cong} \varphi(x)$ in $\mathbf{Def}(T_{\mathbf{C}})$, which satisfies the additional property we discussed above. If every $\mathcal{L}_{\mathbf{C}}$-formula $\varphi(x)$ has a counterpart, then L is essentially surjective.

If we know that the $\mathcal{L}_{\mathbf{C}}$-formulas $\varphi(x)$ and $\psi(x)$ of sort $S \simeq \mathsf{L}(C)$ have counterparts $A_{\varphi(x)}$ and $A_{\psi(x)}$, then we can form their join

$$T_{\mathbf{C}} \vdash \varphi(x) \vee \psi(x) \leftrightarrow \big(\mathsf{L}(A_{\varphi(x)})(\epsilon_{\varphi(x)}(x)) \vee \mathsf{L}(A_{\psi(x)})(\epsilon_{\psi(x)})(x)\big),$$

and we put

$$\epsilon_{\varphi(x) \vee \psi(x)} \overset{\text{df}}{=} \begin{cases} \epsilon_{\varphi(x)} \text{ if } x \in \varphi(x) \\ \epsilon_{\psi(x)} \text{ if } x \in \psi(x) \backslash \varphi(x) \end{cases},$$

so that $\varphi(x) \vee \psi(x)$ has counterpart $A_{\varphi(x)} \vee A_{\psi(x)}$ (finite sup computed in the subobject lattice of C), witnessed by the definable bijection $\epsilon_{\varphi(x) \vee \psi(x)}$. We can argue analogously that $\varphi(x) \wedge \psi(x)$ has a counterpart.

Now suppose that $\varphi(x)$ of sort $S \xleftarrow{\widehat{\epsilon}} \mathcal{L}(C)$ has a counterpart $A_{\varphi(x)}$. Then $\neg\varphi(x)$ has counterpart $\neg A_{\varphi(x)}$, with definable bijection $\epsilon_{\neg\varphi(x)}$ defined to be the restriction of the canonical bijection $\widehat{\epsilon}$ to the complement of $\mathsf{L}(A_{\varphi(x)}) \hookrightarrow \mathsf{L}(C)$.

Suppose that $\varphi(x,y)$ of sort $S_x \times S_y \xleftarrow{\widehat{\epsilon}} \mathsf{L}(C_x \times C_y)$ has a counterpart. It is easy to check that the canonical bijections (where we have abused notation by so far calling all of them $\widehat{\epsilon}$) are compatible with projections, so that the diagram

$$\begin{array}{ccccc} \mathsf{L}(C_x) & \longleftarrow & \mathsf{L}(C_x \times C_y) & \longrightarrow & \mathsf{L}(C_y) \\ \downarrow & & \downarrow & & \downarrow \\ S_x & \longleftarrow & S_x \times S_y & \longrightarrow & S_y, \end{array}$$

commutes, where the top horizontal maps are L of the projections $C_x \times C_y \rightrightarrows C_x, C_y$, the bottom horizontal maps are the projections $S_x \times S_y \rightrightarrows S_x, S_y$, and the vertical maps are the $\widehat{\epsilon}$.

Let ι be the inclusion of $A_{\varphi(x,y)}$ into $C_x \times C_y$. Then $\exists x\varphi(x,y)$ will have counterpart $A_{\exists x\varphi(x,y)} \overset{\text{df}}{=} \mathrm{im}\big(\pi_{C_y} \circ \iota\big) \hookrightarrow C_y$, with $\epsilon_{\varphi(x,y)}$ the unique map which makes the following diagram commute:

$$\begin{array}{ccc} \varphi(x,y) & \twoheadrightarrow & \exists x\varphi(x,y) \\ {\scriptstyle\simeq}\uparrow & & \uparrow \\ \mathsf{L}(A_{\varphi(x,y)}) & \twoheadrightarrow & \mathsf{L}(\mathrm{im}(\pi \circ \iota)). \end{array}$$

This completes the induction on complexity of formulas, and we conclude that L is essentially surjective.

Fullness does not immediately follow from essential surjectivity. Instead, we must look closer at the proof and use the additional property which allowed us to conclude that every $\mathcal{L}_{\mathbf{C}}$-formula had a counterpart.

Indeed, let $f : \mathsf{L}(A) \to \mathsf{L}(B)$ be a definable function in $\mathbf{Def}(T_{\mathbf{C}})$. Then there is a subobject G of $A \times B$ such that the following diagram commutes:

$$\begin{array}{ccc} \mathsf{L}(G) & \longrightarrow & \mathsf{L}(A \times B) \\ {\scriptstyle\simeq}\downarrow & & \downarrow{\scriptstyle\simeq} \\ \Gamma(f) & \longrightarrow & \mathsf{L}(A) \times \mathsf{L}(B). \end{array}$$

Since $\Gamma(f)$ was a definable function, the canonical projection $\mathsf{L}(A) \times \mathsf{L}(B) \to \mathsf{L}(A)$ restricts to a bijection $\Gamma(f) \simeq \mathsf{L}(A)$. Then L of the canonical projection $A \times B \to A$ restricted to G must be a definable bijection.

It now remains to show that this implies that the canonical map $G \to A$ is an isomorphism, since then G will be the graph of the morphism $A \xrightarrow{\sim} G \to B$.

From how we defined $T_{\mathbf{C}}$, L preserves diagonal embeddings $A \to A \times A$. If K is the kernel pair $K \rightrightarrows A \xrightarrow{f} B$ of a morphism f in $\mathbf{C}$, then the diagram

$$\begin{array}{ccccccc} \mathsf{L}(A \times A) & \longrightarrow & \mathsf{L}(A) \times \mathsf{L}(A) & \rightrightarrows & \mathsf{L}(A) & \xrightarrow{\mathsf{L}(f)} & \mathsf{L}(B) \\ \uparrow & & \uparrow & & & & \\ \mathsf{L}(K) & \longrightarrow & \ker(\mathsf{L}(f)) & & & & \end{array}$$

commutes. If $\mathsf{L}(f)$ is injective, then $\ker(\mathsf{L}(f))$ is the diagonal relation on $\mathsf{L}(A)$ and hence $\mathsf{L}(K)$ is the diagonal relation on $\mathsf{L}(A)$.

Similarly, if $\mathsf{L}(f)$ is surjective, then its image is all of $\mathsf{L}(B)$.

It therefore suffices to show that, for subobjects $A, B \hookrightarrow C$ of C, whenever $T_{\mathbf{C}} \vdash \mathsf{L}_C(A) \to \mathsf{L}_C(B)$ as predicates on $\mathsf{L}(C)$, $A \subseteq B$ as subobjects of C in $\mathbf{C}$.

By the categorical completeness theorem and the discussion of internal theories in [4], we know that:

1. Whenever the inclusion of A into C does not factor through the inclusion of B into C in $\mathbf{C}$[2], there exists an elementary functor $F : \mathbf{C} \to \mathbf{Set}$ such that $F(B) \backslash F(A) \neq \varnothing$, and
2. every elementary functor $F : \mathbf{C} \to \mathbf{Set}$ can be expanded to a model of M_F of $T_{\mathbf{C}}$.

Therefore, if $\mathbf{C}$ does not know that $A \subseteq B$ as subobjects of C in $\mathbf{C}$, there is a model M_F of $T_{\mathbf{C}}$ such that $M_F(B) \backslash M_F(A) \neq \varnothing$. Therefore, $T \not\vdash \mathsf{L}(A) \to \mathsf{L}(B)$.

This shows that L is full. This also shows that L is faithful: if $f_1 \neq f_2$ in $\mathbf{C}$, then $T \not\vdash \mathsf{L}(\Gamma(f_1)) \leftrightarrow \mathsf{L}(\Gamma(f_2))$. □

Proof of Theorem 3.25. Lemma 3.26 shows that $\mathbf{Def}(-)$ is faithful on 1-morphisms and 2-morphisms, Lemma 3.27 and Lemma 3.28 shows that $\mathbf{Def}(-)$ is full on 1-morphisms and 2-morphisms, and Theorem 3.30 shows that $\mathbf{Def}(-)$ is surjective up to equivalence. □

4 Pretoposes and the $(-)^{\mathrm{eq}}$-construction

Previously, we established an equivalence

$$\begin{array}{c} \{\text{first-order theories and strict interpretations}\} \\ \updownarrow \\ \{\text{Boolean logical categories and elementary functors}\}. \end{array}$$

We would like to expand this picture to incorporate arbitrary interpretations of theories. As we saw during the proof of Proposition 3.19, we needed to assume strictness of the interpretation $I : T \to T'$ to ensure that the graph $\Gamma(f)$ of a definable function f in T is interpreted as a relation $I(\Gamma(f))$ which T' proves to be the graph of a function. Without the strictness assumption, equality is interpreted as a proper equivalence relation E, and then the definition of an abstract interpretation only ensures that $I(f)$ is a function only *on E-equivalence classes*, i.e. only after quotienting out by E.

Therefore, if we can always form quotients of definable sets by definable equivalence relations, then we can canonically associate to any non-strict interpretation a "homotopic" strict interpretation, by quotienting the non-equality equivalence relations back into equality relations.

[2]Warning: this is *not* the same as saying that the terminal map from the complement $B \backslash A$ to $1_{\mathbf{C}}$ has image $= 1_{\mathbf{C}}$. This would mean that the internal theory of $\mathbf{C}$ *proves* that $B \backslash A$ is nonempty; however, if $\mathbf{C}$ merely does not contain a factorization of $A \hookrightarrow C$ through $B \hookrightarrow C$, then the internal theory leaves the proposition $\mathsf{L}(A) \to \mathsf{L}(B)$ undecided.

First-order theories don't always have definable quotients of definable sets by definable equivalence relations, and Boolean logical categories don't always have quotients of objects by equivalence-relation-objects. However, we can canonically enlarge any theory T (and so, by Theorem 3.25, any Boolean logical category $\mathbf{C}$) to a theory T^{eq} (resp., to a Boolean logical category $\widetilde{\mathbf{C}}$) which *does* have definable quotients of definable sets by definable equivalence relations (resp. *does* have quotients of objects by equivalence-relation-objects). Then arbitrary abstract interpretations $T_1 \to T_2$ will correspond to strict abstract interpretations $T_1 \to T_2^{\mathrm{eq}}$.

Our goal in this section will be to make everything we have just said precise, and to characterize those logical categories of the form $\widetilde{C} = \mathbf{Def}(T^{\mathrm{eq}})$.

Definition 4.1. An **equivalence relation** (or **internal congruence**) in a category $\mathbf{C}$ with finite limits is the following data:

1. An object X and a subobject $E \hookrightarrow X \times X$,
2. A *reflexivity map* $r : X \to E$ such that r is a section to both projections $\pi_1, \pi_2 : X \times X \to X$,
3. A *symmetry map* $s : E \to E$ such that $\pi_1 \circ s = \pi_2$ and $\pi_2 \circ s = \pi_1$,
4. A *transitivity map* $r : E \times_X E \to E$, where $E \times_X E$ is the pullback of π_1 and π_2, as in the following pullback square (where $i : R \hookrightarrow X \times X$ is the inclusion map):

$$\begin{array}{ccc} E \times_X E & \xrightarrow{p_2} & R \\ {\scriptstyle p_1}\downarrow & & \downarrow{\scriptstyle \pi_1 \circ i} \\ R & \xrightarrow{\pi_2 \circ i} & X \end{array}$$

   such that $\pi_1 \circ i \circ p_2 = \pi_1 \circ i \circ t$, and $\pi_2 \circ i \circ p_2 = \pi_2 \circ i \circ t$.

Example 4.2. Let $f : X \to Y$ be a morphism in $\mathbf{Def}(T)$. The pullback $X \times_Y X$ with respect to f and f, as in

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ {\scriptstyle \pi_1}\uparrow & & \uparrow{\scriptstyle f} \\ X \times_Y X & \xrightarrow[\pi_2]{} & X, \end{array}$$

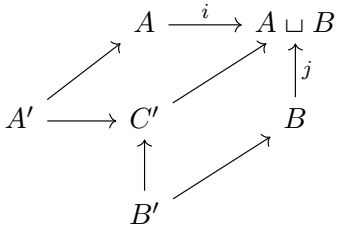
can be canonically viewed as an internal congruence $X \times_Y X \overset{\pi_1}{\underset{\pi_2}{\rightrightarrows}} X$, called the **kernel** or **kernel pair** of f. We write $\ker(f) \overset{\mathrm{df}}{=} X \times_Y X$.

Definition 4.3. Let $E \rightrightarrows X$ be an equivalence relation in $\mathbf{C}$. A **quotient** of E, written X/E, is the coequalizer of $E \rightrightarrows X$.

Definition 4.4. A **pretopos** is a Boolean logical category $\mathbf{C}$ that additionally satisfies:

1. $\mathbf{C}$ has a stable disjoint sum of any pair of objects. A disjoint sum $A \sqcup B$ of objects A, B is a coproduct of A and B such that, for the canonical maps $i : A \hookrightarrow A \sqcup B$ and $j : B \hookrightarrow A \sqcup B$, i and j are monomorphisms and the pullback $A \times_{A \sqcup B} B$ is isomorphic to 0.

Stability for disjoint sums means that whenever we have a diagram of the form



with A' and B' pullbacks, then C' is the disjoint sum of A' and B'.

2. **C** has quotients of equivalence relations. Equivalently, every equivalence relation in **C** is the kernel pair of some map.

The condition for theories analogous to condition 2 above is *elimination of imaginaries*.

Definition 4.5. A theory T **eliminates imaginaries** if for every definable equivalence relation $E \rightrightarrows X$, there exists a definable set X/E and a definable function π_E such that $E \rightrightarrows X$ is the kernel of $X \stackrel{\pi_E}{\to} X/E$.

Definition 4.6. The following construction, due to Shelah, associates to any theory T a larger theory T^{eq} and a strict interpretation $T \to T^{\mathrm{eq}}$, such that T^{eq} eliminates imaginaries.

1. First, we expand the language $\mathcal{L}$ of T to a language $\mathcal{L}^{\mathrm{eq}}$ by adding, for every definable equivalence relation $E \rightrightarrows X$, a new sort S_E and a new partial function symbol $f_E : X \to S_E$.
2. T^{eq} is axiomatized by the theory of T on the original sorts, plus sentences which assert that for every definable equivalence relation $E \rightrightarrows X$ in T, $f_E : X \to S_E$ is a surjection and $E = \ker(f_E)$.

The new sorts S_E are called **imaginary sorts**, and their elements are called **imaginaries**. The original sorts are called **real sorts**. There is a canonical strict interpretation $T \to T^{\mathrm{eq}}$ which is determined by the inclusion of languages $\mathcal{L} \subseteq \mathcal{L}^{\mathrm{eq}}$.

The empty sort 1 has itself as a unique quotient, so we view it as both a real and imaginary sort.

The construction suggests that T determines T^{eq}. We can make this precise.

Proposition 4.7. *Any product $S_E \times S_{E'}$ of imaginary sorts is canonically isomorphic to a single imaginary sort $S_{E''}$, and any definable set in T^{eq} is canonically isomorphic to the quotient of a definable set of T.*

Proof. Let $E \rightrightarrows X$ and $E' \rightrightarrows X'$ be two definable equivalence relations on definable sets X and X'. There is a canonical surjective map

$$X \times X' \to X/E \times X'/E', \quad \text{by} \quad (x, x') \mapsto (x/E, x'/E)$$

with kernel relation $E'' \rightrightarrows X \times X' \twoheadrightarrow X/E \times X'/E'$. Therefore, there is a canonical definable bijection $S_E \times S_{E'} \simeq S_{E''}$. Note that this implies that in T^{eq}, definable subsets of $S_E \times S_{E'}$ correspond precisely to definable subsets of $S_{E''}$.

Now, let $S_E = S_{E_1}$ be an imaginary sort. We write $\overline{x} = x_1, \ldots, x_n, y_1, \ldots, y_m$, where $x_1, \ldots, x_n$ are the real-sorted variables and $y_1, \ldots, y_m$ are the imaginary-sorted variables belonging to the imaginary sorts $S_{E_1}, \ldots, S_{E_m}$, where each E_i is an equivalence relation on a definable set X_i of sort

B_i. Let $\psi(\overline{x})$ be a quantifier-free definable set in T^{eq}. Putting $\psi(\overline{x})$ into disjunctive normal form, we reduce to the case where $\psi(\overline{x})$ is a conjunction of atomic and negated-atomic $\mathcal{L}^{\text{eq}}$-formulas. We will proceed in two steps. First, we will show that if we put quantifiers over all the real-sorted variables of $\psi(\overline{x})$, the imaginary-sorted result is canonically isomorphic to a product of images of T-definable sets. Second, we will show that given an imaginary-sorted product of images of T-definable sets, putting quantifiers over any of the imaginary-sorted variables still results in an imaginary-sorted product of images of T-definable sets. Quantifying over all the variables except x_1, we will get that each disjunctand of the disjunctive normal form of $\exists \overline{x} \backslash x_1 \psi(\overline{x})$ is the image of a T-definable set, and so their union will be the image of a T-definable set.

So, assuming that $\psi(\overline{x})$ is a conjunction of atomic and negated-atomic $\mathcal{L}^{\text{eq}}$-formulas, we rearrange the conjunctands according to which kinds of variables appear: purely real, mixed, and pure imaginary, so that

$$\psi(\overline{x}) \equiv \left(\bigwedge_i \theta_i^R(\overline{x}) \right) \wedge \left(\bigwedge_j \theta_j^M(\overline{x}) \right) \wedge \left(\bigwedge_k \theta_k^I(\overline{x}) \right).$$

The only purely imaginary-sorted atomic $\mathcal{L}^{\text{eq}}$-formulas are just the equality relations on each sort S_{E_i}. Some of the E_i might coincide. Fix an E_i; then, the pure-imaginary part of $\psi(x)$ asserts some equalities and inequalities among the n' distinct variables belonging to some S_{E_i}. Replacing "$=_{S_{E_i}}$" and "$\neq_{S_{E_i}}$" by E_i and $\neg E_I$ throughout, obtain a definable subset of $X_i^{n'}$ whose image under f_{E_i} is the conjunction of those S_{E_i}-sorted conjunctands of $\bigwedge_k \theta_k^I(\overline{x})$. Repeating this for each distinct equivalence relation among the $E_1, \ldots, E_n$, we write $\bigwedge_k \theta_k^I(\overline{x})$ as a product of images of T-definable sets.

To finish the first step, it remains to see what happens to the purely-real and mixed parts of $\psi(\overline{x})$ when we quantify over all the real variables. The purely-real part just becomes an $\mathcal{L}$-sentence. The mixed part consists of the graph relations $\Gamma_i(f)(x, y)$ of the projections $f_{E_i} : X_i \to S_{E_i}$ (and their negations). So there are four cases, depending on whether the real variable x is existentially or universally quantified over, and whether or not Γ_i is negated. It is easy to see that for all the cases except for the case where x is existentially quantified over and Γ_i is positive (so that we have an image), the resulting formula $Qx(\neg)\Gamma(f)(x, y)$ is equivalent mod T^{eq} to a purely-real sentence about E_i and X_i. If the real variable x does not appear in the pure real part, then $\exists x \Gamma_i(x, y)$ is equivalent mod T^{eq} to S_{E_i}.

Summarizing, we have that

$$\exists x_1 \ldots \exists x_n \left(\bigwedge_i \theta_i^R(\overline{x}) \right) \wedge \left(\bigwedge_j \theta_j^M(\overline{x}) \right)$$

is canonically isomorphic to a product of images of T-definable sets. Since the purely-imaginary part contains no real variables to quantify over and each S_{E_i} is the image of X_i under f_{E_i}, $\exists x_1 \ldots \exists x_n \psi(\overline{x})$ is canonically isomorphic to a product of images of T-definable sets. This completes the first step.

If $\psi(\overline{y}) \equiv (y_1 \in f_{E_1}(Y_1)) \wedge \cdots \wedge (y_n \in f_{E_n}(Y_n))$ is an imaginary-sorted product of images of T-definable sets, then existentially quantifying over any y_k has the same effect as deleting $(y_k \in f_{E_k}(Y_k))$, so the result is still a product of images of T-definable sets. Similarly, universally quantifying over any y_k has no effect on the conjunctands $(y_i \in f_{E_i}(Y_i))$ when $i \neq k$, and $\forall y_k \exists x \left[(x \in Y_k) \wedge f_{E_k}(x) = y_k \right]$ is equivalent mod T^{eq} to the assertion in T that Y_k is equal to X_k, the domain of the equivalence relation E_k. Therefore, when we universally quantify over y_k,

the conjunctand $(y_k \in f_{E_k}(Y_k))$ can be replaced by this sentence in T, which is a subterminal object of 1 whose image is itself. So both $\exists y_k \psi(\overline{y})$ and $\forall y_k \psi(\overline{y})$ are canonically isomorphic to products of images of T-definable sets. □

Proposition 4.8. *For any theory T,* $\mathbf{Def}(T^{\mathrm{eq}})$ *is a pretopos.*

Proof. $\mathbf{Def}(T^{\mathrm{eq}})$ is already a Boolean logical category, so it remains to check the existence of quotients and stable disjoint sums.

Quotients: If $E \rightrightarrows X$ is a definable equivalence relation of T^{eq}, then by Proposition 4.7, the diagram $E \rightrightarrows X$ is the image of an T-definable equivalence relation $E' \rightrightarrows X'$, which therefore has quotient $S_{E'}$.

Disjoint sums: In any sort S of T, $S \times S$ has at least two disjoint definable sets, given by the diagonal relation $x =_S y$ and its complement $x \neq_S y$. We can define an equivalence relation E on $S \times S$ whose classes are precisely these two definable sets, and S_E will have two distinct constants 0 and 1. Let X and Y be two definable sets in T^{eq}. We put

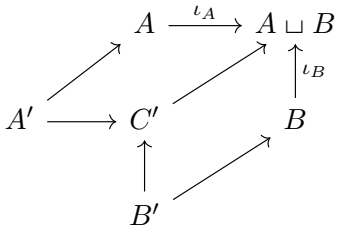
$$X \sqcup Y \stackrel{\mathrm{df}}{=} X \times Y \times \{0,1\} \,/_{E_{X \sqcup Y}},$$

where

$$(x,y,\epsilon) \sim_{E_{X\sqcup Y}} (x',y',\epsilon') \iff \epsilon = \epsilon' \text{ and } \begin{cases} \text{if } \epsilon = \epsilon' = 0, \text{ then } x = x' \\ \text{if } \epsilon = \epsilon' = 1, \text{ then } y = y'. \end{cases}$$

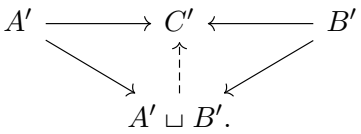
Whenever we take points in a model, $X \sqcup Y$ as we have defined it is canonically isomorphic to $\{(x,0) \,\big|\, x \in X\} \cup \{(y,1) \,\big|\, y \in Y\}$ (unions of sets are not immediately available to us via $\vee$ if X and Y are differently-sorted, which is why we had to use the equivalence relation.) This determines the canonical inclusions $A \xrightarrow{\iota_A} A \sqcup B \xleftarrow{\iota_B} B$, and it is easy to see that their pullback is 0.

Stability of disjoint sums: Consider a diagram of T^{eq}-definable sets of the form



where C' is a subobject of $A \sqcup B$ and A' and B' are pullbacks.

There is a canonical definable map $A' \sqcup B' \to C'$ such that the following diagram commutes:



Since ι_A and ι_B are mono, their pullbacks $A' \to C'$ and $B' \to C'$ are also mono. Therefore, the map $A' \sqcup B' \to C'$ is also mono.

On the other hand, in any model M of T^{eq}, every point $c \in M(C')$ is the image of something in $M(A')$ or $M(B')$; since the above diagram commutes, the function $M(A' \sqcup B') \to M(C')$ is surjective. Since M was arbitrary, by the completeness theorem T^{eq} proves that $A' \sqcup B' \to C'$ is a bijection.

□

We will spend the rest of this section proving that one may as well replace abstract interpretations between theories with elementary functors between $\mathbf{Def}(-)$ of $(-)^{\mathrm{eq}}$ of those theories.

Here is what we are going to do. Given an abstract interpretation $I : T_1 \to T_2$, we can lift I to a *strict* abstract interpretation $I_t : T_1 \to T_2^{\mathrm{eq}}$ by quotienting out the equivalence relations interpreting the equality symbols in T_1, which then corresponds by Theorem 3.25 to an elementary functor $\mathbf{Def}(T_1) \to \mathbf{Def}(T_2^{\mathrm{eq}})$. This determines a "functor"

$$(-)_t : \mathbf{Int}(T_1, T_2) \to \mathbf{BoolLogCat}(\mathbf{Def}(T_1), \mathbf{Def}(T_2^{\mathrm{eq}}))$$

which will be an equivalence of categories.

(We put scare quotes around "functor" because we have not defined a notion of morphism for abstract interpretations. In fact, we will tautologically *define* morphisms of abstract interpretation $T_1 \to T_2$ to be transformations of the associated strict abstract interpretations $T_1 \to T_2^{\mathrm{eq}}$.)

Dually, instead of lifting I to T_2^{eq}, we can extend I to an abstract interpretation $\widehat{I} : T_1^{\mathrm{eq}} \to T_2$. This determines a functor

$$\widehat{(-)} : \mathbf{Int}(T_1, T_2) \to \mathbf{Int}(T_1^{\mathrm{eq}}, T_2)$$

which will be an equivalence of categories.

Let $\widetilde{(-)}$ denote the composition $(-)_t \circ \widehat{(-)}$. Then the "functor"

$$\widetilde{(-)} : \mathbf{Int}(T_1, T_2) \to \mathbf{Pretop}(\mathbf{Def}(T_1^{\mathrm{eq}}), \mathbf{Def}(T_2^{\mathrm{eq}}))$$

will be an equivalence of categories.

Definition 4.9. Let $I : T_1 \to T_2$ be an abstract interpretation. We define a strict abstract interpretation $I_t : T_1 \to T_2^{\mathrm{eq}}$ as follows. Let X be a definable set of T_1 of sort S with equality relation $=_S$. Let E be the equality relation on X. Then we put $I_t(X)$ to be the following imaginary sort of T^{eq}:

$$I_t(X) \stackrel{\mathrm{df}}{=} I(X) \Big/ I(=_S) \simeq S_{I(E)} \in T_2^{\mathrm{eq}}.$$

I_t is a lift of I along the canonical interpretation $T_2 \to T_2^{\mathrm{eq}}$, i.e. the following diagram commutes:

$$\begin{array}{ccc} & & T_2^{\mathrm{eq}} \\ & \overset{I_t}{\nearrow} & \uparrow \\ T_1 & \xrightarrow[I]{} & T_2. \end{array}$$

By Theorem 3.25, I_t determines an elementary functor $F_{I_t} : \mathbf{Def}(T_1) \to \mathbf{Def}(T_2^{\mathrm{eq}})$.

We *define* a morphism of two abstract interpretations $I, I' : T_1 \to T_2$ to be a transformation of the strict abstract interpretations I_t, I'_t. This obviously defines a category structure $\mathbf{Int}(T_1, T_2)$ on the collection of abstract interpretations $T_1 \to T_2$, such that $I \mapsto I_t \mapsto F_{I_t}$ determines an equivalence of categories $\mathbf{Int}(T_1, T_2) \simeq \mathbf{BoolLogCat}(\mathbf{Def}(T_1), \mathbf{Def}(T_2^{\mathrm{eq}}))$.

Definition 4.10. Let $I : T_1 \to T_2$ be an abstract interpretation. We define an abstract interpretation $\widehat{I} : T_1^{\mathrm{eq}} \to T_2$ as follows. For every imaginary sort $S_E = X/E$ in T_1^{eq},

$$\widehat{I}(S_E) \stackrel{\mathrm{df}}{=} I(X), \widehat{I}(=_{S_E}) \stackrel{\mathrm{df}}{=} I(E), \quad \text{and} \quad \widehat{I}(\Gamma(f_E)) \stackrel{\mathrm{df}}{=} I(E).$$

Recall from Definition 3.23 that $\mathbf{Th}(T_1, T_2)$ denotes the category of strict interpretation $T_1 \to T_2$ with transformations as morphisms. The $\widehat{(-)}$ construction determines a functor $\widehat{(-)} : \mathbf{Th}(T_1, T_2^{\mathrm{eq}}) \to \mathbf{Th}(T_1^{\mathrm{eq}}, T_2^{\mathrm{eq}})$ as follows: for each transformation $\eta : I \to I'$, and for each imaginary sort $S_E \simeq X/E$ in T_1^{eq}, it is easy to see that by virtue of being a transformation, $\eta_X : I(X) \to I'(X)$ descends to a map $\eta_{X/E} : I(X)/I(E) \to I'(X)/I'(E)$, and so we put

$$\widehat{\eta}(S_E) \stackrel{\mathrm{df}}{=} \eta_{X/E}.$$

We claim that $\widehat{(-)} : \mathbf{Th}(T_1, T_2^{\mathrm{eq}}) \to \mathbf{Th}(T_1^{\mathrm{eq}}, T_2^{\mathrm{eq}}$ is an equivalence.

There is a functor going the other way given by precomposing by the canonical interpretation $T_1 \to T_1^{\mathrm{eq}}$. This is a retract of $\widehat{(-)}$, so $\widehat{(-)}$ is faithful.

$\widehat{(-)}$ is full: for the definition of a transformation to be satisfied, the components of any transformation $\widehat{I} \to \widehat{I'}$ at imaginary sorts S_E for $E \rightrightarrows X$ is determined by the component at X.

$\widehat{(-)}$ is essentially surjective: let $J : T_1^{\mathrm{eq}} \to T_2^{\mathrm{eq}}$ be some strict interpretation. Let J' be the interpretation obtained by precomposing by the interpretation $T_1 \to T_1^{\mathrm{eq}}$ and then applying the $\widehat{(-)}$ construction again. Since J was an interpretation, $J(S_E)$ is canonically isomorphic to the quotient of $J(X)$ by $J(E)$, which is $J'(S_E)$. These canonical isomorphisms determine a natural isomorphism $J \simeq J'$.

The functor $\mathbf{Th}(T_1, T_2^{\mathrm{eq}}) \to \mathbf{Th}(T_1^{\mathrm{eq}}, T_2^{\mathrm{eq}})$ determines a functor

$$\mathbf{Int}(T_1, T_2) \to \mathbf{Int}(T_1^{\mathrm{eq}}, T_2),$$

and since $\mathbf{Th}(T_1, T_2^{\mathrm{eq}}) \to \mathbf{Th}(T_1^{\mathrm{eq}}, T_2^{\mathrm{eq}})$ was an equivalence, so is $\mathbf{Int}(T_1, T_2) \to \mathbf{Int}(T_1^{\mathrm{eq}}, T_2)$.

Definition 4.11. Let $\mathbf{C}$ be a Boolean logical category. The **pretopos completion** of $\mathbf{C}$ is

$$\widetilde{\mathbf{C}} \stackrel{\mathrm{df}}{=} \mathbf{Def}\left((T_{\mathbf{C}})^{\mathrm{eq}}\right).$$

Definition 4.12. Let $F : \mathbf{C} \to \mathbf{C}'$ be an elementary functor between Boolean logical categories. We define the **pretopos completion** of F to be

$$\widetilde{F} : \widetilde{\mathbf{C}} \to \widetilde{\mathbf{C}'}$$

in the sense of $\widetilde{(-)} \stackrel{\mathrm{df}}{=} \widehat{(-)} \circ (-)_t$ in the above discussion.

Identifying theories with Boolean logical categories, we also define the pretopos completion $\widetilde{I}$ of an abstract interpretation $I : T_1 \to T_2$ to be the strict abstract interpretation $\widetilde{I} : T_1^{\mathrm{eq}} \to T_2^{\mathrm{eq}}$.

5 Categories of models

Definition 5.1. Let M and N be models of T. An **elementary embedding** $f : M \to N$ comprises the following data:

1. For every sort S of T, a function $f_S : M(S) \to N(S)$, such that
2. the collection $\{f_S\}$ is compatible with forming tuples of sorts: if S is a tuple of basic sorts $S = (B_1, \ldots, B_n)$, $f_S = f_{B_1} \times \cdots \times f_{B_n}$, and furthermore
3. for every tuple $\overline{a}$ of sort S and every formula $\varphi(\overline{x})$ such that $M \models \varphi(\overline{a})$, $N \models \varphi(f_S(\overline{a}))$.

Definition 5.2. The **category of models** of a theory T is defined to be:

$$\mathbf{Mod}(T) \stackrel{\text{df}}{=} \begin{cases} \text{Objects: models of } T, \\ \text{Morphisms: elementary embeddings.} \end{cases}$$

The category **Set** of all sets is a Boolean logical category, although unlike those Boolean loogical categories of the form $\mathbf{Def}(T)$ for theories T, **Set** is not small.

However, for every regular cardinal κ, the category $\mathbf{Set}_\kappa$ of all hereditarily κ-small sets is a small Boolean logical category. By the downward Löwenheim-Skolem theorem, for every theory there exists some κ such that one only needs to test points in κ-small models to invoke the completeness theorem, and $\mathbf{Set} = \bigcup_\kappa \mathbf{Set}_\kappa$.

Proposition 5.3. *Every model $M \models T$ determines an elementary functor $\mathbf{Def}(T) \to \mathbf{Set}$.*

Proof. By item 3 of Definition 2.1, a model is an assignment of sets to $\mathcal{L}$-formulas. Since T proves that every definable function is a function, this must be true after taking points in a model, so this assignment is a functor $M : \mathbf{Def}(T) \to \mathbf{Set}$. Now we must show that the functor M is elementary.

To see that M preserves finite limits, it suffices to check the preservation and reflection of limits on just products and equalizers.

The usual construction of an equalizer of two maps $f, g : X \to Y$ in **Set** is always definable: it is the subset of X consisting of those elements x such that $f(x) = g(x)$.

Similarly, if X and Y are definable, then $X \times Y$ is definable, and the projections $X \times Y \underset{\pi_Y}{\overset{\pi_X}{\rightrightarrows}} X, Y$ are definable.

If J is a finite diagram in $\mathbf{Def}_M(T)$ and $\varprojlim J$ its limit, and $Z \in \mathbf{Def}_M(T)$ is a definable set in M equipped with a cone of definable maps to J, then Z has (in **Set**) a unique mediating map to $\varprojlim J$, which is definable because it is definable in the cases when J is a product or equalizer diagram, the limit is finite, and by the canonical product-equalizer decomposition the mediating map for a general finite J is a composition of finitely many mediating maps for products and equalizers.

To check preservation of finite sups, let $\{\varphi_1(x), \ldots, \varphi_n(x)\}$ be a finite collection of formulas of the same sort. Then their sup is given by $\bigvee_n \varphi_i(x)$, and the sup of $\{\varphi_1(M), \ldots, \varphi_n(M)\}$ is precisely $\bigcup_n \varphi_i(M)$. The empty sup is the empty formula, represented in $\mathbf{Def}(T)$ by the T-provable equivalence class of "$x \neq x$", and this is interpreted by M as the empty set, which is the empty sup for any set in **Set**.

To check preservation of images, let f be a definable function. The image of f in $\mathbf{Def}(T)$ is just the formula which describes the image of f, and M interprets this formula as the image of $f(M)$. $\square$

Proposition 5.3 determines an inclusion $\mathbf{Mod}(T) \hookrightarrow \mathbf{BoolLogCat}(\mathbf{Def}(T), \mathbf{Set})$.

Proposition 5.4. *The inclusion* $\mathbf{Mod}(T) \hookrightarrow \mathbf{BoolLogCat}(\mathbf{Def}(T), \mathbf{Set})$ *is an equivalence of categories.*

Proof. Fix an elementary functor $F : \mathbf{Def}(T) \to \mathbf{Set}$. We must find a model M such that $M \simeq F$ as elementary functors. Equivalently, we will show that we can "perturb" F to a model (which is just an elementary functor with some additional strictness conditions) without changing its isomorphism type as a functor.

For every basic sort B, there are canonical isomorphisms $F(B^k) \simeq F(B)^k$. Up to isomorphism of functors (where the isomorphism of functors is given by conjugating by these canonical isomorphisms), we can assume therefore that $F(B^k) = F(B)^k$.

Furthermore, for every sort $\vec{B} = B_1 \times \cdots \times B_n$, there are canonical isomorphisms $F(B_1 \times \cdots \times B_n) \simeq F(B_1) \times \cdots \times F(B_n)$. Again, up to isomorphism of functors, we can assume that $F(\vec{B}) = F(\vec{B})$. Furthermore, if $\varphi(x)$ is a formula of sort B, then there is a canonical definable injection $\varphi(x) \hookrightarrow B$ such that the image of $F(\varphi(x) \hookrightarrow B)$ is a subset of $F(B)$; arguing as before, we can assume up to an isomorphism of functors that $F(\varphi(x)) \subseteq F(B)$. Similarly, we can assume up to an isomorphism of functors that if $T \models \forall x(\varphi(x) \to \psi(x))$, then $F(\varphi(x)) \subseteq F(\psi(x))$.

The canonical isomorphisms described so far induce isomorphisms of Boolean algebras $2^{\vec{B}} \simeq 2^{F(\vec{B})}$. Therefore, up to isomorphism of functors, we can assume that $F(\varphi(x) \vee \psi(x)) = F(\varphi(x)) \cup F(\psi(x))$ (resp. $\wedge$ and negations).

Since F preserves images, then for every definable function f, $F(\mathrm{im}(f)) \simeq \mathrm{im}(F(f))$. Then up to isomorphism of functors, $F(\mathrm{im}(f)) = \mathrm{im}(F(f))$.

Now we have, up to isomorphism, completely "strictified" F. It remains to show that an elementary functor which strictly preserves products, finite sups, and images is a model.

Indeed, let $\vec{c}$ be a tuple of terms such that $R(\vec{c})$ is an atomic sentence. Then by our previous reductions, $F(x = \vec{c}) \subseteq F(R(x))$, so $F \models R(\vec{c})$.

It is obvious that if φ and ψ satisfy that $(T \models \varphi \implies F \models \varphi)$ and $(T \models \psi \implies F \models \psi)$, then $(T \models \varphi \wedge \psi \implies F \models \varphi \wedge \psi)$.

If $\varphi(x)$ is a formula, then $T \models \exists x \varphi(x)$ if and only if the image of the projection of $\varphi(x)$ to the empty sort (which is the empty product, so is the terminal object 1) is all of 1. Since F is a logical functor, it preserves the terminal object and all maps into the terminal object, so F of the image of the projection of $\varphi(x)$ to the empty sort is still 1. Then $F(\varphi(x))$ cannot be empty, since if it were, the image of its canonical map to 1 would be the empty set. So $F \models \exists x \varphi(x)$.

Similarly, if $T \models \neg\psi$, then if ψ is quantifier-free it is easy to see that $F \models \neg\psi$. If ψ is of the form $\exists\varphi(x)$, then as a subobject of the terminal object 1, $\exists x \varphi(x) = \varnothing$ the empty sup. Since F is logical, it preserves empty sups, so again $\exists x \varphi(x) = \varnothing$ as a subobject of the terminal set 1, and therefore, $F \models \neg\exists x \varphi(x)$.

This concludes the induction on complexity of formulas, and finishes the proof. $\square$

For any regular cardinal κ, $\mathbf{Set}_\kappa$ is a pretopos in addition to being a Boolean logical category. So, by the discussion above, we have equivalences

$$\mathbf{Mod}(T) \simeq \mathbf{BoolLogCat}(\mathbf{Def}(T), \mathbf{Set}_\kappa), \quad \text{and} \quad \mathbf{Mod}(T^{\mathrm{eq}}) \simeq \mathbf{Pretop}(\mathbf{Def}(T^{\mathrm{eq}}), \mathbf{Set}_\kappa).$$

By the discussion following Definition 4.10, $\mathbf{BoolLogCat}(\mathbf{Def}(T), \mathbf{Set}_\kappa)$ and $\mathbf{Pretop}(\mathbf{Def}(T^{\mathrm{eq}}), \mathbf{Set})$ are equivalent. We conclude:

Proposition 5.5. *For any theory T, the categories $\mathbf{Mod}(T)$ and $\mathbf{Mod}(T^{\mathrm{eq}})$ are equivalent.*

Remark 5.6. In the discussion following Definition 4.10, one can show that the canonical functor $\mathbf{Pretop}(T^{\mathrm{eq}}, \mathbf{Set}) \to \mathbf{BoolLogCat}(T, \mathbf{Set})$ induced by the canonical interpretation $T \to T^{\mathrm{eq}}$ is pseudo-inverse to the functor $\widehat{(-)}$, so that the equivalence $\mathbf{Mod}(T) \simeq \mathbf{Mod}(T^{\mathrm{eq}})$ is given by the canonical functor $\mathbf{Mod}(T^{\mathrm{eq}}) \to \mathbf{Mod}(T)$ induced by the canonical interpretation $T \to T^{\mathrm{eq}}$.

Definition 5.7. In general, an interpretation $T \to T'$ induces a strict interpretation of pretopos completions $T^{\mathrm{eq}} \to T'^{\mathrm{eq}}$ and thus an elementary functor $\mathbf{Def}(T^{\mathrm{eq}}) \to \mathbf{Def}(T'^{\mathrm{eq}})$. Since models are essentially elementary functors into $\mathbf{Set}$, the elementary functor $\mathbf{Def}(T^{\mathrm{eq}}) \to \mathbf{Def}(T'^{\mathrm{eq}})$ pulls back models of T' to models of T, inducing a functor $\mathbf{Mod}(T') \to \mathbf{Mod}(T)$. We call such functors between categories of models **reduct functors**. If $I : T \to T'$ is an abstract interpretation, we write $I^* : \mathbf{Mod}(T') \to \mathbf{Mod}(T)$ for the induced reduct functor.

6 Notions of equivalence between the notions of interpretations

In this section, we examine various notions of equivalence between abstract interpretations, concrete interpretations, and elementary functors.

The first notion is due to [1].

Definition 6.1. Let $M \models T$ and $M' \models T'$, and let $(f, f^*), (g, g^*) : M \to M'$ be concrete interpretations. We use the letter U for the preimages of sorts along f, and we use the letter V for the preimages of sorts along g.

We say that (f, f^*) and (g, g^*) are **homotopic** if for every sort S of T, the pullbacks

$$\begin{array}{ccc} U_S \times_{M(S)} V_S & \longrightarrow & V_S \\ \downarrow & & \downarrow g \\ U_S & \xrightarrow[f]{} & M(S) \end{array}$$

are definable in M'.

Remark 6.2. Note that if f_S and g_S are injective, then the above pullback describes the graph of a bijection $U_S \simeq V_S$.

We define the analogous notion for abstract interpretations.

Definition 6.3. Let $I, I' : T_1 \to T_2$ be abstract interpretations. For every sort S, we denote by E_S the definable equivalence relation in T_2 given by $I(x =_S y)$ (resp. E'_S, I'). A **homotopy** between I and I' comprises the following data:

1. For every sort S of T_1, a definable relation $R_S \hookrightarrow I(S) \times I'(S)$, such that the following conditions are satisfied:

2. (Naturality mod E and E') For every formula $\varphi(x)$ of sort S,

$$T_2 \vdash (\forall x \in I(\varphi(x)))\big[R_S(x,y) \to (\exists y' \in I'(\varphi(x)))[E'_S(y,y')]\big].$$

3. (Univalence mod E and E')

$$T_2 \vdash (\forall x_1, x_2 \in I(S))(\forall y_1, y_2 \in I'(S))\big[E_S(x_1, X_2) \to \big(R_S(x_1,y_1) \wedge R_S(x_2,y_2) \to E'_S(y_1,y_2)\big)\big].$$

4. (Injectivity mod E and E')

$$T_2 \vdash (\forall x_1, x_2 \in I(S))(\forall y_1, y_2 \in I'(S))\big[E'_S(y_1,y_2) \wedge R_S(x_1,y_1) \wedge R_S(x_2,y_2) \to E_S(x_1,x_2)\big].$$

5. (Surjectivity mod E and E')

$$T_2 \vdash (\forall y \in I'(S))(\exists x \in I(S))\big[R_S(x,y)\big].$$

Finally, given two elementary functors $F, F' : \mathbf{C}_1 \to \mathbf{C}_2$, a natural notion of equivalence is just natural isomorphism of functors.

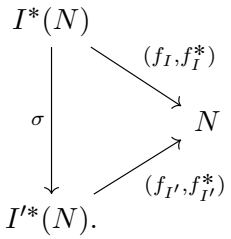
An immediate consequence of Definition 6.3 is:

Proposition 6.4. *Let I and I' be abstract interpretations $T_1 \to T_2$. I and I' are abstractly homotopic if and only if the elementary functors associated to their pretopos completions*

$$F_{\widetilde{I}}, F_{\widetilde{I'}} : \mathbf{Def}(T_1^{\mathrm{eq}}) \to \mathbf{Def}(T_2^{\mathrm{eq}})$$

are naturally isomorphic.

Corollary 6.5. *Let $I, I' : T_1 \to T_2$ be homotopic abstract interpretations. For any model $N \models T_2$, there exists an isomorphism $\{\sigma_S\}_{S \in \mathsf{Sorts}(T_1)}$ of models of T_1 such that the following diagram commutes:*



From Remark 6.2, we get:

Proposition 6.6. *Let M and N be models of T and T'. If two concrete interpretations*

$$(f, f^*), (g, g^*) : M \to N$$

are homotopic, then the underlying abstract interpretations $f^*, g^* : T \to T'$ are homotopic.

References

[1] G. Ahlbrandt and M. Ziegler, *Quasi-finitely axiomatizable totally categorical theories*, Annals of Pure and Applied Logic, 30(1) (1986), pp. 63–82.

[2] S. M. Lane, *Categories for the working mathematician, 2nd ed.*, Springer-Verlag, 1998.

[3] M. Makkai, *Stone duality for first-order logic*, Annals of Pure and Applied Logic, 40 (1988), pp. 167–215.

[4] M. Makkai and G. Reyes, *First-order categorical logic*, Springer-Verlag, 1977.