SJÄLVSTÄNDIGA ARBETEN I MATEMATIK

MATEMATISKA INSTITUTIONEN, STOCKHOLMS UNIVERSITET

Functorial Semantics for Fragments of First-Order Logic

av

Anton Christenson

2025 – No M6

MATEMATISKA INSTITUTIONEN, STOCKHOLMS UNIVERSITET, 106 91 STOCKHOLM

Functorial Semantics for Fragments of First-Order Logic

Anton Christenson

Självständigt arbete i matematik 30 högskolepoäng, avancerad nivå

Handledare: Peter LeFanu Lumsdaine

2025

ABSTRACT

Functorial semantics provides a close connection between logic and category theory, where models of a theory can be seen as functors out of a classifying category. This perspective also shows how set-theoretic models can be generalized in a natural way to models in any other category with the appropriate structure.

In this thesis we introduce functorial semantics first for algebraic theories and then for intuitionistic first-order logic, and show how the categorical framework can be used to prove soundness and completeness of the corresponding deductive systems.

Different treatments of this topic make different equivalent choices both on the logical side (in terms of which deductive rules to include) and on the categorical side (in terms of which extra structure to require). We aim here to make the connection between the two sides as clear as possible, by stating the structural requirements in a way that directly mirrors the deductive rules. This approach also simplifies the process of comparing and contrasting different logical fragments, such as regular and coherent logic.

SAMMANFATTNING

Funktoriell semantik ger en nära koppling mellan logik och kategoriteori, där modeller av en teori kan ses som funktorer från en klassificerande kategori. Detta perspektiv visar också hur mängdteoretiska modeller på ett naturligt sätt kan generaliseras till modeller i andra kategorier med lämplig struktur.

I denna uppsats introducerar vi funktoriell semantik först för algebraiska teorier och sedan för intuitionistisk första ordningens logik, och visar hur det kategoriteoretiska ramverket kan användas för att bevisa sundhet och fullständighet för de motsvarande deduktiva systemen.

Olika framställningar av detta ämne utgör från olika ekvivalenta val, både på den logiska sidan (vad gäller vilka deduktiva regler som inkluderas) och på den kategoriteoretiska sidan (vad gäller vilken extra struktur som krävs). Vi strävar här efter att göra kopplingen mellan de två sidorna så tydlig som möjligt, genom att formulera de strukturella kraven på ett sätt som direkt speglar de deduktiva reglerna. Detta tillvägagångssätt förenklar också processen att jämföra och kontrastera olika logiska fragment, såsom reguljär och koherent logik.

Contents

1 Introduction

In logic, a primary object of study is logical theories and *models* of those theories. The theories can range from simple algebraic structures such as groups, to powerful first-order theories such as Zermelo–Fraenkel set theory that are suitable as foundations for mathematics. Classically, a model of a theory consists of a *set* together with functions and subsets that interpret the meaning of function and relation symbols. In *categorical semantics*, this definition is generalized to allow for models in a much broader class of categories. We can for instance consider models of the theory of a group in the category of topological spaces (giving the notion of a *topological group*) or in the category of smooth manifolds (giving the notion of a *Lie group*).

A key construction is that of a *classifying category* $\mathcal{C}_{\mathbb{T}}$ of a theory $\mathbb{T}$. In this category there is a *universal model* of the theory $\mathbb{T}$, that enables an efficient way of proving completeness of a given deductive system. Further, any model of $\mathbb{T}$ in another category $\mathcal{C}$ corresponds to a functor $\mathcal{C}_{\mathbb{T}} \to \mathcal{C}$. These observations together can be summarized as "theories are categories – models are functors" and provides the basis for the field of *functorial semantics.*

In this thesis our goal is to provide an accessible introduction to this topic, and in particular highlight the fruitful connection between logic and category theory provided therein. We follow the framework of [Joh02, Sections D1.1–D1.4], but with a more narrow focus on developing just the core parts necessary to show completeness and give the functorial semantics equivalence. Our exposition also draws heavily on the lecture notes [AB24, Sections 1.1 and 3.2–3.3].

The main contributions of this work are:

1. To fill in some of the routine details that are omitted from the above sources.
2. To reorganize the structural requirements on categories in a way that more directly mirrors the deductive rules.

The overall structure of the thesis is as follows.

- In Chapter 2, we cover some necessary prerequisites from category theory.
- In Chapter 3, we introduce functorial semantics for the class of algebraic theories.
- In Chapters 4-7, we develop functorial semantics for first-order logic and fragments.

Acknowledgements

I am deeply grateful to my advisor Peter Lumsdaine for his encouragement and patience, and to my friend Giacomo Cozzi for his support.

2 Categorical prerequisites

We assume familiarity with some basic notions from category theory, including products, pullbacks, functors and natural transformations. A good introduction can be found in [Lei14, Chapter 1 and 5]. In the rest of this chapter we briefly cover a few other topics that will be of particular importance to us.

2.1 FP-categories and FP-functors

Definition 2.1.1 An FP-category is a category that has all finite products. Equivalently, it has a terminal object and binary products.

If $f_1 : X \to Y_1$ and $f_2 : X \to Y_2$, we write (f_1, f_2) for the corresponding morphism $X \to Y_1 \times Y_2$. More generally, if $f_i : X \to Y_i$, then $(f_1, \dots, f_n) : X \to Y_1 \times \dots \times Y_n$.

Definition 2.1.2 A functor $F : \mathcal{C} \to \mathcal{D}$ between two FP-categories is said to *preserve finite products* if for every finite product diagram in $\mathcal{C}$, we get a product diagram in $\mathcal{D}$ by applying F to all the objects and morphisms. A functor with this property is also called an FP-functor.

To prove that a functor preserves finites products, it suffices to check whether terminal objects and binary products are preserved. Thus if $T \in \mathcal{C}$ is terminal then $F(T) \in \mathcal{D}$ should be terminal, and if

$$\begin{tikzcd} & A \times B \arrow[dl, "p"'] \arrow[dr, "q"] & \\ A & & B \end{tikzcd}$$

is any binary product diagram in $\mathcal{C}$, then

$$\begin{tikzcd} & F(A \times B) \arrow[dl, "F(p)"'] \arrow[dr, "F(q)"] & \\ F(A) & & F(B) \end{tikzcd}$$

should be a product diagram in $\mathcal{D}$. Note that it is *not* enough to check that

$$F(A \times B) \cong F(A) \times F(B),$$

we need the stronger property that the specific map

$$F(A \times B) \overset{(Fp,Fq)}{\longrightarrow} F(A) \times F(B)$$

is an isomorphism.

Definition 2.1.3 If $\mathcal{C}$ and $\mathcal{D}$ are FP-categories, we denote by $\mathsf{FP}(\mathcal{C}, \mathcal{D})$ the category of FP-functors $\mathcal{C} \to \mathcal{D}$ and natural transformations.

2.2 Equivalences between categories

An *equivalence* between two categories consists of functors

$$\mathcal{C} \underset{G}{\overset{F}{\rightleftarrows}} \mathcal{D}$$

together with natural transformations $\varepsilon : FG \Rightarrow 1_{\mathcal{D}}$ and $\eta : 1_{\mathcal{C}} \Rightarrow GF$.

Commonly, an equivalence is shown in practice via the following lemma:

Lemma 2.2.1 A functor gives rise to an equivalence if and only if it is fully faithful and essentially surjective on objects.

If one has a functor satisfying the above properties, an inverse functor can be constructed using the axiom of choice. However, for our purposes we will always prove "essentially surjective" by constructing an inverse mapping, and then we do not need to appeal to choice.

Lemma 2.2.2 Let $F : \mathcal{C} \to \mathcal{D}$ be a fully faithful functor, and suppose that we can construct a map $G : \mathsf{ob}(\mathcal{D}) \to \mathsf{ob}(\mathcal{C})$ and isomorphisms $f_X : F(G(X)) \to X$ for each X. Then $\mathcal{C} \simeq \mathcal{D}$.

Proof. The first step is to extend the definition of G to morphisms. Let $h : X \to Y$ be a morphism in $\mathcal{D}$. Then because $F(G(X)) \cong X$ and $F(G(Y)) \cong Y$ we can compose with isomorphisms to get a map $h' : F(G(X) \to F(G(Y))$. Since F is fully faithful, there is a unique map $h'' : G(X) \to G(Y)$ such that $F(h'') = h'$. We can then define $G(h) = h''$, and it is easy to check that this is functorial.

Next we need a natural transformation $\varepsilon : FG \Rightarrow 1_{\mathcal{D}}$. We already know that $F(G(X)) \cong X$ so we can define the component ε_X to be that isomorphism f_X. Now for any $h : X \to Y$ in $\mathcal{D}$, $G(h)$ by definition has the property that $F(G(H)$ is the composition $f_Y^{-1} \circ h \circ f_X$, so naturality is satisfied.

For the other natural transformation $\eta : 1_{\mathcal{C}} \Rightarrow GF$, we need to define the component $\eta_X : X \to G(F(X))$. For this, we start with the identity map $F(X) \to F(X)$ and compose with the isomorphism $F(X) \to F(G(F(X)))$. Then we use the fact that F is fully faithful to pull this back to a map $X \to G(F(X))$. The naturality condition is again easily checked. □

2.3 Posets as categories

A *poset* or *partially ordered set* is a set equipped with an order relation that is reflexive, antisymmetric and transitive (see Example 4.2.5). Any poset can also be viewed as a category, with one object for each element and an arrow $x \to y$ whenever $x \leq y$. Under this construction, a meet is exactly a product, and a join is exactly a coproduct:

Order	**Category**
Maximal element	Terminal object
Meet (greatest lower bound)	Product
Minimal element	Initial object
Join (lowest upper bound)	Coproduct

Functors between posets correspond to order-preserving maps:

$$x \leq y \Rightarrow F(x) \leq F(y)$$

2.4 Subobjects

A *monic* or *monomorphism* is a morphism f such that $f \circ g_1 = f \circ g_2 \Rightarrow g_1 = g_2$. We define a preorder $\lesssim$ on the collection of all monics into a fixed object X by

$$f \lesssim f' :\Leftrightarrow \exists g(f = f' \circ g).$$

Forming equivalence classes by $f \sim f' :\Leftrightarrow f \lesssim f' \wedge f' \lesssim f$ we obtain the partially ordered class of subobjects of X. We denote this class by $\mathsf{Sub}(X)$ and the partial order by $\leq$.

For simplicity we will assume that all our categories are *well-powered*, so that every $\mathsf{Sub}(X)$ is in fact a partially ordered *set*. None of our results depend in a significant way on this assumption, but it avoids us being distracted by size issues.

A subobject is really specified by a monic, not by an object, but we will often refer to a subobject by the domain of its monic. This is harmless as long as we keep in mind that the domain does not uniquely specify a monic.

Remark 2.4.1. In Set, an arbitrary injective function

$$m : S \rightarrowtail X$$

represents the same subobject as the inclusion

$$i : \mathrm{im}(m) \hookrightarrow X.$$

Let $\mathcal{C}$ be a category with finite limits. Then for each morphism $f : A \to B$ in $\mathcal{C}$, there is an associated functor between posets

$$f^* : \mathsf{Sub}(B) \to \mathsf{Sub}(A)$$

which turns a subobject of B into a subobject of A by this pullback diagram:

$$\begin{tikzcd}
f^*(S) \arrow[r, dashed] \arrow[d, dashed, tail] & S \arrow[d, tail] \\
A \arrow[r, "f"] & B
\end{tikzcd}$$

There are two implicit claims in saying that f^* is a functor:

1. The pullback of the monic $S \rightarrowtail B$ along f gives a monic $f^*(S) \rightarrowtail A$.
2. This operation preserves $\leq$: if $S_1 \leq S_2$ in $\mathsf{Sub}(B)$, then $f^*(S_1) \leq f^*(S_2)$ in $\mathsf{Sub}(A)$.

If the morphism f is a monic m, then the pullback functor additionally reflects $\leq$:

3. If $m^*(S_1) \leq m^*(S_2)$ in $\mathsf{Sub}(A)$, then $S_1 \leq S_2$ in $\mathsf{Sub}(B)$.

In practice this means that if we have a diagram

$$\begin{tikzcd}
S_1 \arrow[dr, tail] & & \\
& A \arrow[r, tail] & B \\
S_2 \arrow[ur, tail] & &
\end{tikzcd}$$

then S_1 and S_2 can be considered as subobjects of either A or B, and when writing $S_1 \leq S_2$ we need not specify which of the domains we are talking about.

Finally we observe that

$$(\mathrm{id}_X)^* = \mathrm{id}_{\mathsf{Sub}(X)} \qquad (g \circ f)^* = f^* \circ g^*$$

which means that we have a contravariant functor

$$\begin{aligned} \mathsf{Sub} : \mathcal{C}^{\mathsf{op}} &\to \mathsf{Pos} \\ X &\mapsto \mathsf{Sub}(X) \\ f &\mapsto f^* \end{aligned}$$

from $\mathcal{C}$ to the category of posets, which we will refer to as the subobject functor.

2.5 Adjunctions

An adjunction $F \dashv G$ of functors

$$\begin{tikzcd} \mathcal{C} \arrow[r, shift left, "F"] & \mathcal{D} \arrow[l, shift left, "G"] \end{tikzcd}$$

is a family of isomorphisms

$$\hom_{\mathcal{D}}(F(C), D) \cong \hom_{\mathcal{C}}(C, G(D))$$

which is natural in C and D. In the case where $\mathcal{C}$ and $\mathcal{D}$ are posets, the condition simplifies to

$$F(C) \leq D \quad \iff \quad C \leq G(D)$$

and naturality is trivially satisfied.

An adjunction between posets is also known as a *Galois connection.*

Remark 2.5.1. If F and G are arbitrary maps between posets with the above property, then it can be shown that they are order-preserving as follows:

$$\dfrac{\dfrac{C_1 \leq C_2 \qquad \dfrac{\dfrac{}{F(C_2) \leq F(C_2)}}{C_2 \leq G(F(C_2))}}{C_1 \leq G(F(C_2))}}{F(C_1) \leq F(C_2)} \qquad\qquad \dfrac{\dfrac{\dfrac{\dfrac{}{G(D_1) \leq G(D_1)}}{F(G(D_1)) \leq D_1} \qquad D_1 \leq D_2}{F(G(D_1)) \leq D_2}}{G(D_1) \leq G(D_2)}$$

Thus if we want to prove an adjunction $F \dashv G$, we do not need to separately prove that F and G are functors: it follows for free once we have shown the adjunction.

3 Algebraic theories

In this chapter we will consider a particularly simple class of logical theories called *algebraic theories.* These theories describe structures with some number of operations defined on them, that satisfy certain equations (the *axioms* of the theory). Prototypical examples are algebraic structures such as groups, rings and modules. We will also see that lattices and directed multigraphs can be described by algebraic theories.

The functorial semantics of algebraic theories is interesting in its own right, but it also serves as a stepping stone to understanding the functorial semantics of first-order theories. The work we do here to interpret contexts and terms carries directly over to the more general first-order case.

In the introduction, we summarized functorial semantics with the slogan "theories are categories – models are functors". Exactly which kinds of categories and functors this refers to depends on the logical fragment under consideration. In this chapter we will see that there is a one-to-one (up to equivalence) correspondence between algebraic theories and finite product categories, and models of algebraic theories correspond to the FP-functors that we defined in Section 2.1.

3.1 Signatures, contexts and terms

Definition 3.1.1 An algebraic signature Σ consists of a set of sorts and a set of function symbols, each with an associated type $A_1, \dots, A_n \to B$. The type of a function symbol consists of

1. An *arity* $A_1, \dots, A_n$ which is a tuple of sorts that describes how many inputs the function takes, and the sort of each input.
2. A single sort B that describes the output sort of the function.

The arity of a function may be empty, in which case we will denote the type as $1 \to B$. Nullary function symbols take no input, so we can think of them as constants.

We assume that we have a fixed countably infinite set $\{v_i\}_{i \in \mathbb{N}}$ of variables which can be used to construct the terms inside any signature. Note that any given term/equation/formula/proof will only use finitely many variables, which means that we always have the option to introduce a new fresh variable that has not been used yet. We even have a canonical choice: the first v_i that is not used.

In our meta-language, we want to use any variable symbols that suit us in the moment, to maximize readability. We will have expressions such as "$x_1, \dots, x_n$" for a collection of variables. It is then understood that these meta-variables stand in for any concrete variables. For instance x_1, x_2, x_3 may stand in for the concrete variables v_2, v_1, v_{34}.

Similarly, the sorts of the signature form a concrete set. The symbols $A_1, \dots, A_n$ are not the concrete sorts, but meta-variables representing sorts. We do not for instance preclude the possibility that A_1 and A_2 represent the same sort.

Example 3.1.2. The signature of a *group* has a single sort G and function symbols

$$m : G \times G \to G, \quad i : G \to G, \quad e : 1 \to G$$

representing the group operation, inverse and identity. Note that the constant e is represented by a nullary function symbol.

Example 3.1.3. The signature of a *ring* has a single sort R and function symbols

$$+, \times : R \times R \to R, \quad - : R \to R, \quad 0, 1 : 1 \to R$$

representing the ring operations $\{+, \times, -\}$ and units $\{0, 1\}$.

Example 3.1.4. The signature of a *multidigraph* has two sorts V, A (representing vertices and arrows) and two function symbols s, t of type $A \to V$, that associate to each arrow a source and target vertex.

Definition 3.1.5 A *context* consists of a (possibly empty) list of variables, each with an associated sort. The *type* of a context is the list of sorts. The same variable symbol never appears twice in a context, but multiple variables may have the same sort.

Example 3.1.6. The following are examples of contexts:

$$[-], \quad [x{:}A], \quad [x{:}B], \quad [x{:}A, y{:}A], \quad [y{:}A, x{:}A]$$

Note that all of these are *different* contexts: both the order and the sort of variables is significant. A variable in isolation does not have an associated sort, it is assigned a sort by the context.

A generic context $\Gamma = [\, x_1{:}A_1, \dots, x_n{:}A_n \,]$ can be written as $\overline{x} = [\, x_1, \dots, x_n \,]$ if we do not need to explicitly refer to the sorts, and $\overline{A} = [\, A_1, \dots, A_n \,]$ if we do not need to explicitly refer to the variable names. For any of these notations we do not exclude the possibility that $n = 0$ such that $\Gamma = \overline{x} = \overline{A} = [-]$ is the empty context.

We notate concatenation of contexts by Γ, Δ or $\overline{x}, \overline{y}$ or Γ, A etc. It is then implicitly claimed that the same variable does not appear in both contexts.

Definition 3.1.7 The set of terms in a context Γ is defined recursively by:

1. For every variable $x{:}A$ of the context, x is a term of sort A.
2. If f is a function symbol of type $A_1, \dots, A_n \to B$, and for each $1 \leq i \leq n$ we have a term t_i (also in the context Γ) of sort A_i, then $f(t_1, \dots, t_n)$ is a term of sort B.

We use the notation $t : \Gamma \to A$ for a term in context Γ of sort A, and refer to $\Gamma \to A$ as the "type" of the term t.

Example 3.1.8. The following are examples of terms in the signature of a group:

$$[\, x{:}G \mid m(x, i(x)) \,], \quad [\, x{:}G, y{:}G \mid m(x, i(x)) \,]$$

Note that these are *different* terms, since they have different contexts. We will sometimes omit the sorts and/or names of variables in the context if we do not need to explicitly refer to them, but formally they should always be specified.

3.2 Substitution

Definition 3.2.1 Let $\overline{A}$ and $\overline{B}$ be contexts. A *substitution* $\overline{s} : \overline{A} \to \overline{B}$ consists of, for each sort B_i of $\overline{B}$, a term $s_i : \overline{A} \to B_i$. Thus a substitution is described by a tuple of terms, and we can notate it by $\left[\overline{A} \mid s_1{:}B_1, \dots, s_n{:}B_n\right]$ or simply $[\, s_1, \dots, s_n \,]$.

Remark 3.2.2. Note that there is no requirement that $\overline{A}$ and $\overline{B}$ have the same length or type.

Definition 3.2.3 Let $t : \Gamma \to A$ be a term and $\overline{s} : \Delta \to \Gamma$ a substitution. The term

$$t(\overline{s}) : \Delta \to A$$

is defined by recursion on the structure of t:

1. If $t = x_i$, then $t(\overline{s}) = s_i$.
2. If $t = f(t_1, \dots, t_k)$, then $t(\overline{s}) = f(t_1(\overline{s}), \dots, t_k(\overline{s}))$.

Example 3.2.4. Consider a term $[\,x, y \mid t\,]$. We can think of this term as analagous to a function $t(x, y)$, and the substitution $t(s_1, s_2)$ is simply the term where we have "plugged in" the values s_1 and s_2 for the variables x and y. The notation $t(x, y)$ here corresponds to applying the trivial substition (x, y) which replaces x with x and y with y.

Remark 3.2.5. Another common notation for substitution is $t\big[s/x_j\big]$, where s is a single term substituted for the single variable x_j. In our setup, there is no need to specify which variable is substituted: we always substitute *all* variables in the context *simultaneously*. The notation $t\big[s/x_j\big]$ can then be seen as shorthand for applying the substitution

$$\big(x_1, \dots, x_{j-1}, s, x_{j+1}, \dots, x_n\big)$$

which leaves all variables except x_j unchanged.

Example 3.2.6. Let f be a function symbol of type $A_1, \dots, A_n \to B$. Then we have a term

$$s_1 = [\,x_1 : A_1, \dots, x_n : A_n \mid f(x_1, \dots, x_n) : B\,]$$

which we can also view as a substitution

$$\overline{s} : [\,A_1, \dots, A_n\,] \to [\,B\,].$$

The substitution takes a term $t(y)$ in the context $[\,y : B\,]$ and transforms it into a term in the context $\overline{A}$ by "plugging in" the value $f(\overline{x})$ in place of y:

$$[\,x_1 : A_1, \dots, x_n : A_n \mid t(f(x_1, \dots, x_n))\,]$$

As our notation suggests, there is a category $\mathsf{Syn}(\Sigma)$ (the *syntactic category* of the signature Σ) where objects are contexts and the morphisms are substitutions. The composition of a substitution $\overline{s}$ with a single term t is defined as:

$$\begin{tikzcd}
\Delta \arrow[d, "\overline{s}"'] \arrow[dr, "t \circ \overline{s} = t(\overline{s})"] & \\
\Gamma \arrow[r, "t"'] & A
\end{tikzcd}$$

More generally, if $\overline{s}$ and $\overline{t}$ are both substitutions, then $\overline{t} \circ \overline{s}$ is defined as

$$\overline{t}(\overline{s}) = (t_1(\overline{s}), \dots, t_n(\overline{s})).$$

It is immediate to see that for each context there is an identity substitution which substitutes each variable for itself. Associativity is saying that if we have three substitutions

$$\big[\overline{a} : \overline{A}\big] \xrightarrow{\overline{r}} \big[\overline{b} : \overline{B}\big] \xrightarrow{\overline{s}} \big[\overline{c} : \overline{C}\big] \xrightarrow{\overline{t}} \big[\overline{d} : \overline{D}\big]$$

then

$$(\overline{r}(\overline{s}))(\overline{t}) = \overline{r}(\overline{s}(\overline{t}))$$

which can be proved by a straightforward induction on the structure of $\overline{r}$.

Remark 3.2.7. Contexts of the same type are isomorphic as objects in $\mathsf{Syn}(\Sigma)$, with isomorphism given by the trivial substitution

$$[\,x_1, ..., x_n\,] : [\,x_1 \colon A_1, \dots , x_n \colon A_n\,] \to [\,y_1 \colon A_1, \dots , y_n \colon A_n\,].$$

Similarly, contexts which only differ in the order of variables are isomorphic, with isomorphism given by the permutation substitution.

Remark 3.2.8. The category $\mathsf{Syn}(\Sigma)$ has finite products. The empty context is the terminal object, and the product of two contexts is

$$[\,x_1, \dots , x_n\,] \times [\,y_1, \dots , y_m\,] = [\,x_1, \dots , x_n, y_1, \dots , y_m\,],$$

with projections given by the trivial substitutions

$$\pi_1 = [\,x_1, \dots , x_n\,] : [\,\overline{x}, \overline{y}\,] \to [\,\overline{x}\,],$$

$$\pi_2 = [\,y_1, \dots , y_m\,] : [\,\overline{x}, \overline{y}\,] \to [\,\overline{y}\,].$$

The substitution π_1 acts on a term in context $[\,\overline{x}\,]$ and turns it into a term in context $[\,\overline{x}, \overline{y}\,]$, by adding new variables to the context and leaving the term itself unchanged.

Example 3.2.9. Let $\Gamma = [x_1 \colon A]$. What is $\Gamma \times \Gamma$? We cannot directly concatenate, because

$$[\,x_1 \colon A, x_1 \colon A\,]$$

is not a valid context. However, Γ is isomorphic to $\Gamma' = [x_2 \colon A]$, and we can compute $\Gamma \times \Gamma'$ by concatenation:

$$\begin{tikzcd} & {[\,x_1 \colon A, x_2 \colon A\,]} \arrow[dl, "{[x_1]}"'] \arrow[dr, "{[x_2]}"] & \\ {[\,x_1 \colon A\,]} & & {[\,x_2 \colon A\,]} \end{tikzcd}$$

By composing the projection $\pi_2 : \Gamma \times \Gamma' \to \Gamma'$ with the isomorphism $\Gamma' \cong \Gamma$ we obtain a projection ${\pi_2}' : \Gamma \times \Gamma' \to \Gamma$, and then $(\Gamma \times \Gamma', \pi_1, {\pi_2}')$ is a product of Γ with Γ.

In general, computing the product of two contexts may require renaming some variables to avoid collisions, but the types (lists of sorts) of the contexts are always simply concatenated. For example, if A, B and C are sorts then $[\,A, B\,] \times [\,B, C\,] = [\,A, B, B, C\,]$.

3.3 Interpretations

A Σ-interpretation I in an FP-category $\mathcal{C}$ consists of:

1. For each sort A, an object $[\![\,A\,]\!]^I \in \mathcal{C}$.
2. For each function symbol $f : A_1, \dots , A_n \to B$, a morphism

$$[\![\,f\,]\!]^I : [\![\,A_1\,]\!]^I \times ... \times [\![\,A_n\,]\!]^I \to [\![\,B\,]\!]^I.$$

Often we will omit the I superscript if we do not need to explicitly refer to the name of the interpretation.

Definition 3.3.1 The interpretation $[\![\, t \,]\!]$ of a term t is defined recursively by:

$$[\![\, x_1 \!:\! A_1, \dots, x_n \!:\! A_n \mid x_j \,]\!] = \pi_j : [\![\, A_1 \,]\!] \times \dots \times [\![\, A_n \,]\!] \to [\![\, A_j \,]\!]$$

$$[\![\, f(r_1, \dots, r_n) \,]\!] = [\![\, f \,]\!] \circ ([\![\, r_1 \,]\!], \dots, [\![\, r_n \,]\!])$$

We extend the interpretation to contexts by $[\![\, A_1, \dots, A_n \,]\!] = [\![\, A_1 \,]\!] \times \dots \times [\![\, A_n \,]\!]$. In particular the empty context is interpreted by the terminal object.

A term $t : \Gamma \to A$ is then interpreted by a morphism $[\![\, t \,]\!] : [\![\, \Gamma \,]\!] \to [\![\, A \,]\!]$.

Example 3.3.2. Let Σ be the signature (V, A, s, t) of a multidigraph (Example 3.1.4), and take $\mathcal{C}$ to be the category of sets. An interpretation of Σ inside $\mathcal{C}$ then consists of:

1. A *set* $[\![\, V \,]\!]$ of vertices
2. A *set* $[\![\, A \,]\!]$ of arrows
3. A *function* $[\![\, s \,]\!] : [\![\, A \,]\!] \to [\![\, V \,]\!]$ assigning to each arrow $a \in [\![\, A \,]\!]$ a source vertex $[\![\, s \,]\!](a) \in [\![\, V \,]\!]$
4. A *function* $[\![\, t \,]\!] : [\![\, A \,]\!] \to [\![\, V \,]\!]$ assigning to each arrow $a \in [\![\, A \,]\!]$ a target vertex $[\![\, t \,]\!](a) \in [\![\, V \,]\!]$

This matches our intuition for what a multidigraph should be.

Example 3.3.3. An interpretation of the signature (G, m, i, e) of a group (Example 3.1.2) in the category of sets consists of:

1. A *set* $[\![\, G \,]\!]$ of group elements.
2. A *function* $[\![\, m \,]\!] : [\![\, G \,]\!] \times [\![\, G \,]\!] \to [\![\, G \,]\!]$ sending each pair of elements $g, h \in [\![\, G \,]\!]$ to their product $[\![\, m \,]\!](g, h) = g \cdot h \in [\![\, G \,]\!]$.
3. A *function* $[\![\, i \,]\!] : [\![\, G \,]\!] \to [\![\, G \,]\!]$ sending each $g \in [\![\, G \,]\!]$ to its inverse $[\![\, i \,]\!](g) = g^{-1} \in [\![\, G \,]\!]$.
4. A *function* $[\![\, e \,]\!] : 1 \to [\![\, G \,]\!]$, where 1 is the terminal object in Set, which is the singleton set $\{\star\}$. The function $[\![\, e \,]\!]$ is uniquely determined by the value $[\![\, e \,]\!](\star) \in [\![\, G \,]\!]$, so we can identify the function with this value and write $[\![\, e \,]\!] \in [\![\, G \,]\!]$ to suggest that $[\![\, e \,]\!]$ behaves as a constant.

Note that we do not yet have any requirement that the axioms of a group are satisfied: we have a set with certain operations defined on it, but we have not asserted that $[\![\, e \,]\!]$ should behave as a unit or that $[\![\, i \,]\!]$ should behave as an inverse. We will do this in the next section.

Example 3.3.4. Let Σ be an arbitrary signature and take $\mathcal{C}$ to be the syntactic category $\mathsf{Syn}(\Sigma)$. There is a *canonical interpretation* C of Σ inside $\mathsf{Syn}(\Sigma)$ where each sort is interpreted as the context of length one with a variable of that sort, and each function symbol is interpreted as the corresponding substitution:

$$[\![\, A \,]\!]^{\mathsf{C}} = [\, x_1 \!:\! A \,]$$

$$[\![\, f : A_1, \dots, A_n \to B \,]\!]^{\mathsf{C}} = [f] : [\, A_1, \dots, A_n \,] \to [\, B \,]$$

We extend the interpretation to substitutions by $[\![\, \overline{s} \,]\!] = ([\![\, s_1 \,]\!], \dots, [\![\, s_n \,]\!])$.

Lemma 3.3.5 Substitution is interpreted by composition:

$$\begin{tikzcd}
\llbracket \Delta \rrbracket \arrow[d, "\llbracket \overline{s} \rrbracket"'] \arrow[dr, dashed, "\llbracket t(\overline{s}) \rrbracket = \llbracket t \rrbracket \circ \llbracket \overline{s} \rrbracket"] & \\
\llbracket \Gamma \rrbracket \arrow[r, "\llbracket t \rrbracket"'] & \llbracket A \rrbracket
\end{tikzcd}$$

Proof. By induction on the structure of t. If $t = x_j$ then

$$\llbracket t \rrbracket \circ \llbracket \overline{s} \rrbracket = \pi_j \circ (\llbracket s_1 \rrbracket, \dots, \llbracket s_n \rrbracket) = \llbracket s_j \rrbracket = \llbracket t(\overline{s}) \rrbracket$$

and if $t = f(r_1, \dots, r_n) = f(\overline{r})$ then

$$\llbracket f(\overline{r}) \rrbracket \circ \llbracket \overline{s} \rrbracket = (\llbracket f \rrbracket \circ \llbracket \overline{r} \rrbracket) \circ \llbracket \overline{s} \rrbracket = \llbracket f \rrbracket \circ (\llbracket \overline{r} \rrbracket \circ \llbracket \overline{s} \rrbracket) = \llbracket f \rrbracket \circ \llbracket \overline{r}(\overline{s}) \rrbracket = \llbracket f(\overline{r}(\overline{s})) \rrbracket = \llbracket f(\overline{r})(\overline{s}) \rrbracket.$$

Note that for the step $\llbracket \overline{r} \rrbracket \circ \llbracket \overline{s} \rrbracket = \llbracket \overline{r}(\overline{s}) \rrbracket$ we are really applying the inductive hypothesis to each of the n terms r_i individually:

$$\llbracket r_i \rrbracket \circ \llbracket \overline{s} \rrbracket = \llbracket r_i(\overline{s}) \rrbracket$$

The final step $\llbracket f(\overline{r}(\overline{s})) \rrbracket = \llbracket f(\overline{r})(\overline{s}) \rrbracket$ is using associativity of composition in $\mathsf{Syn}(\Sigma)$. □

Definition 3.3.6 A Σ-*interpretation homomorphism* $h : I \to J$ consists of maps

$$h_A : \llbracket A \rrbracket^I \to \llbracket A \rrbracket^J$$

for each sort A, such that the following diagram commutes for every function symbol f:

$$\begin{tikzcd}
\llbracket A_1 \rrbracket^I \times \dots \times \llbracket A_n \rrbracket^I \arrow[r, "\llbracket f \rrbracket^I"] \arrow[d, "h_{A_1} \times \dots \times h_{A_n}"'] & \llbracket B \rrbracket^I \arrow[d, "h_B"] \\
\llbracket A_1 \rrbracket^J \times \dots \times \llbracket A_n \rrbracket^J \arrow[r, "\llbracket f \rrbracket^J"'] & \llbracket B \rrbracket^J
\end{tikzcd}$$

Example 3.3.7. Let G_1 and G_2 be multidigraphs in Set. A homomorphism $h : G_1 \to G_2$ consists of:

1. A function $h_V : \llbracket V \rrbracket^{G_1} \to \llbracket V \rrbracket^{G_2}$ sending each vertex of G_1 to a vertex of G_2.
2. A function $h_A : \llbracket A \rrbracket^{G_1} \to \llbracket A \rrbracket^{G_2}$ sending each arrow of G_1 to an arrow of G_2.

The homomorphism criterion requires that the following diagrams commute:

$$\begin{tikzcd}
\llbracket A \rrbracket^{G_1} \arrow[r, "\llbracket s \rrbracket^{G_1}"] \arrow[d, "h_A"'] & \llbracket V \rrbracket^{G_1} \arrow[d, "h_V"] \\
\llbracket A \rrbracket^{G_2} \arrow[r, "\llbracket s \rrbracket^{G_2}"'] & \llbracket V \rrbracket^{G_2}
\end{tikzcd}
\qquad
\begin{tikzcd}
\llbracket A \rrbracket^{G_1} \arrow[r, "\llbracket t \rrbracket^{G_1}"] \arrow[d, "h_A"'] & \llbracket V \rrbracket^{G_1} \arrow[d, "h_V"] \\
\llbracket A \rrbracket^{G_2} \arrow[r, "\llbracket t \rrbracket^{G_2}"'] & \llbracket V \rrbracket^{G_2}
\end{tikzcd}$$

This is saying that if a is an arrow from v_1 to v_2, then $h(a)$ must be an arrow from $h(v_1)$ to $h(v_2)$.

Example 3.3.8. Let G and H be sets with group structure (i.e. sets equipped with operations m, i, e, which do not necessarily satisfy any axioms). A homomorphism is a function $\varphi : G \to H$ such that

$$\varphi(a \cdot b) = \varphi(a) \cdot \varphi(b) \qquad \varphi(a^{-1}) = \varphi(a)^{-1} \qquad \varphi(e_G) = e_H.$$

Note that if G and H also satisfy the group axioms, then the latter two equations can be proved from the first one, so the common definition of group homomorphism only needs to include the first equation. But all three equations are part of "respecting the structure", and in a general structure without any axioms we should include each of these requirements for a homomorphism.

Example 3.3.9. If $F : \mathcal{C} \to \mathcal{D}$ is an FP-functor and I a Σ-interpretation in $\mathcal{C}$, then we can define a Σ-interpretation in $\mathcal{D}$. On sorts we simply let $[\![A]\!]^{F(I)} = F([\![A]\!]^I)$. For a function symbol f of type $A_1, \dots, A_n \to B$, it seems clear that we should define $[\![f]\!]^{F(I)}$ as:

$$F([\![f]\!]^I) : F\Big([\![A_1]\!]^I \times \dots \times [\![A_n]\!]^I\Big) \to F([\![B]\!]^I)$$

However, the domain here is not quite equal to $F\Big([\![A_1]\!]^I\Big) \times \dots \times F\Big([\![A_n]\!]^I\Big)$. Since F is FP-preserving, we do have a canonical isomorphism

$$F\Big([\![A_1]\!]^I\Big) \times \dots \times F\Big([\![A_n]\!]^I\Big) \cong F\Big([\![A_1]\!]^I \times \dots \times [\![A_n]\!]^I\Big)$$

so we can define $[\![f]\!]^{F(I)}$ to be the composition

$$F\Big([\![A_1]\!]^I\Big) \times \dots \times F\Big([\![A_n]\!]^I\Big) \overset{\cong}{\to} F\Big([\![A_1]\!]^I \times \dots \times [\![A_n]\!]^I\Big) \overset{F([\![f]\!]^I)}{\to} F([\![B]\!]^I).$$

The Σ-interpretations and homomorphisms in an FP-category $\mathcal{C}$ form a category $\Sigma_{\mathsf{int}}(\mathcal{C})$. Composition of homomorphisms is defined componentwise: $(g \circ h)_A = g_A \circ h_A$. The morphisms $(g \circ h)_A$ commute with function symbols since g_A and h_A commute with function symbols. The identity homomorphism is the identity on each component.

If we have an interpretation I of the signature Σ in the category $\mathcal{C}$, then we can define a functor $I^\sharp : \mathsf{Syn}(\Sigma) \to \mathcal{C}$. This functor maps a context Γ to $[\![\Gamma]\!]^I$ and a substitution $\overline{s}$ to $[\![\overline{s}]\!]^I$. Because of how we extended the interpretation to contexts, this functor preserves finite products.

Conversely, if $F : \mathsf{Syn}(\Sigma) \to \mathcal{C}$ is any FP-functor, then we can define an interpretation $\mathsf{ev}_{\mathsf{C}}(F) = F(\mathsf{C})$, where C is the canonical interpretation from Example 3.3.4, and $F(\mathsf{C})$ is defined as in Example 3.3.9.

We thus have mappings

$$\begin{tikzcd}
\Sigma_{\mathsf{int}}(\mathcal{C}) \arrow[r, bend left, "(-)^\sharp"] & \mathsf{FP}(\mathsf{Syn}(\Sigma), \mathcal{C}) \arrow[l, bend left, "\mathsf{ev}_{\mathsf{C}}"]
\end{tikzcd}$$

which we now want to extend to an equivalence of categories. We start by extending ev_{C} to a functor. We have so far only defined it on the objects of $\mathsf{FP}(\mathsf{Syn}(\Sigma), \mathcal{C})$, but we also need to define it on the morphisms, which are natural transformations between functors.

If $\eta : F \Rightarrow G$ is a natural transformation, define the model homomorphism

$$\mathsf{ev_C}(\eta) : \mathsf{ev_C}(F) \to \mathsf{ev_C}(G)$$

by the components

$$\mathsf{ev_C}(\eta)_A = \eta_{[A]} : F([A]) \to G([A]).$$

From the naturality of η it follows that this commutes with all the function symbols, as required.

Lemma 3.3.10 The functor $\mathsf{ev_C}$ is essentially surjective on objects.

Proof. Let $I \in \Sigma_{\mathsf{int}}(\mathcal{C})$ be an arbitrary. We can turn I into a functor $I^\sharp$ and back into an interpretation $\mathsf{ev_C}\big(I^\sharp\big) = I^\sharp(\mathsf{C})$. For every sort A we have

$$[\![\, A \,]\!]^{I^\sharp(\mathsf{C})} = I^\sharp\big([\![\, A \,]\!]^{\mathsf{C}}\big) = I^\sharp([\, A \,]) = [\![\, A \,]\!]^I$$

and similarly for every function symbol f

$$[\![\, f \,]\!]^{I^\sharp(\mathsf{C})} = I^\sharp\big([\![\, f \,]\!]^{\mathsf{C}}\big) \circ i = I^\sharp([\, f \,]) = [\![\, f \,]\!]^I$$

where i is the canonical isomorphism from Example 3.3.9.

This shows that $\mathsf{ev_C}\big(I^\sharp\big) = I$, so we are done. □

Lemma 3.3.11 The functor $\mathsf{ev_C}$ is full and faithful.

We omit the details of this proof, but the key observation is that by making a naturality diagram with the projections $[x_1, \dots, x_n] \to [x_i]$, it can be seen that the component $\eta_{[x_1,\dots,x_n]}$ is uniquely determined by the components $\eta_{[x_1]}, \dots, \eta_{[x_n]}$.

Putting the lemmas together we arrive at the desired equivalence:

Theorem 3.3.12 For any signature Σ and for any FP-category $\mathcal{C}$, there is an equivalence

$$\Sigma_{\mathsf{int}}(\mathcal{C}) \simeq \mathsf{FP}(\mathsf{Syn}(\Sigma), \mathcal{C}).$$

Remark 3.3.13. It can also be shown that this equivalence is "natural in $\mathcal{C}$", but we will not use this fact.

Remark 3.3.14. Under this equivalence, the canonical interpretation C corresponds to the identity functor $\mathsf{Syn}(\Sigma) \to \mathsf{Syn}(\Sigma)$.

3.4 Equations, theories and models

An *equation* is an expression $t_1 = t_2$, where t_1 and t_2 are two terms of the same type $\Gamma \to A$.

To specify the context and sort of an equation we may use either of the following notations:

$$\Gamma \mid t_1 =_A t_2 \qquad\qquad t_1 = t_2 : \Gamma \to A$$

If two terms are interpreted as the same morphism by a given interpretation I, i.e. we have

$$[\![\, t_1 \,]\!]^I = [\![\, t_2 \,]\!]^I,$$

then we say that the interpretation *believes* the equation $t_1 = t_2$, and we write $I \vDash t_1 = t_2$.

An *algebraic theory* $\mathbb{T}$ consists of an algebraic signature Σ together with a set of equations over that signature, the *axioms* of the theory.

A $\mathbb{T}$-*model* is a Σ-interpretation that believes all the axioms of $\mathbb{T}$.

A $\mathbb{T}$-model homomorphism is the same thing as a Σ-interpretation homomorphism – we do not need to add any extra requirements about respecting the axioms. In other words, the $\mathbb{T}$-models and homomorphisms in an FP-category $\mathcal{C}$ form a *full* subcategory $\mathbb{T}_{\mathsf{mod}}(\mathcal{C})$ of $\Sigma_{\mathsf{int}}(\mathcal{C})$.

Example 3.4.1. Take Σ to be the signature of a multidigraph (Example 3.1.4) and make a theory $\mathbb{T}$ without any axioms. Then a $\mathbb{T}$-model is the same thing as a Σ-interpretation: an arbitrary multidigraph, without any special properties.

In fact, there are not many meaningful things that we are able to add as axioms at this point. We could add the axiom $s(a) =_V t(a)$ which says that every arrow is a loop, or we could add the axiom $a_1 =_A a_2$ which says that all arrows are equal (i.e. the graph has either no arrows or exactly one arrow). However, in our current fragment we have no way of asserting that two things are *not* equal, so for instance adding an axiom forbidding loops is not possible.

Example 3.4.2. Take Σ to be the signature of a group (Example 3.1.2). The axiom of associativity, spelled out in full, is:

$$[\, a{:}G, b{:}G, c{:}G \mid m(m(a,b),c) = m(a, m(b,c)) \,]$$

By using infix notation $a \cdot b := m(a,b)$ and omitting the context we get the more readable equation

$$(a \cdot b) \cdot c = a \cdot (b \cdot c).$$

Note that there is little harm in just supplying the more readable version when giving an axiom of a theory. The context is not uniquely determined, but we can use the convention that an equation without context should be taken in its *canonical context*, which includes only the variables that appear inside the equation, in the order that they first occur. To complete the theory of a group, we add each of the familiar axioms:

$$\begin{aligned} a \cdot e &= a \qquad & a \cdot a^{-1} &= e \\ e \cdot a &= a \qquad & a^{-1} \cdot a &= e \end{aligned}$$

Now a model of this theory inside Set coincides exactly with the usual definition of a group (and a model homomorphism is exactly a group homomorphism).

Remark 3.4.3. The existence of inverses could be encoded in a first-order formula as follows:

$$(\forall a{:}G).(\exists b{:}G).(a \cdot b = e \wedge b \cdot a = e)$$

We do not have access to $\forall, \exists$ or even $\wedge$ in the axioms of an algebraic theory. Nonetheless, by including the function symbol i in our signature, we automatically get existence of an element $i(a)$ for each a. Further, all our equations are implicitly universally quantified – an equation

$$m(m(a,b),c) = m(a, m(b,c))$$

can also be thought of as an equality of functions $G^3 \to G$. To further emphasize this, we can replace the variables with the projection maps $\pi_i : G^3 \to G$:

$$m(m(\pi_1, \pi_2), \pi_3) = m(\pi_1, m(\pi_2, \pi_3))$$

Claiming that these two function compositions are equal is equivalent to saying that the previous equality holds for all values of a, b, c.

Example 3.4.4. The theory of a *bounded lattice* has a single sort L, nullary function symbols $\top$ and $\bot$ and binary function symbols $\wedge$ and $\vee$, satisfying the following axioms:

$$\begin{array}{cc} x \wedge \top = x & x \vee \bot = x \\ x \wedge y = y \wedge x & x \vee y = y \vee x \\ x \wedge (y \wedge z) = (x \wedge y) \wedge z & x \vee (y \vee z) = (x \vee y) \vee z \\ x \wedge (x \vee y) = x & x \vee (x \wedge y) = x \end{array}$$

Remark 3.4.5. There is no requirement in general that a sort is non-empty. For instance, to interpret the multidigraph signature (V, A, s, t) we can let $[\![V]\!]$ and $[\![A]\!]$ be the empty set, and $[\![s]\!]$ and $[\![t]\!]$ the empty function.

However, when interpreting the group signature (G, m, i, e), if we try to make $[\![G]\!]$ empty then there is no valid choice for $[\![e]\!]$ since there are no functions from the singleton set to the empty set. The nullary function symbol $e : 1 \to G$ thus forces the sort G to be non-empty.

Example 3.4.6. Let $\mathbb{T}$ be the theory of a group and take $\mathcal{C}$ to be the category Top of topological spaces. A $\mathbb{T}$-model in $\mathcal{C}$ then consists of:

1. A topological space G.
2. Continuous maps $m : G \times G \to G$, $i : G \to G$ and a constant $e \in G$, satisfying the group axioms.

This is exactly the definition of a *topological group.*

Lemma 3.4.7 If $F : \mathcal{C} \to \mathcal{D}$ is an FP-functor and M a $\mathbb{T}$-model in $\mathcal{C}$, then $F(M)$ (as defined in Example 3.3.9) is a $\mathbb{T}$-model in $\mathcal{D}$.

Proof. If $[\![t_1]\!]^M = [\![t_2]\!]^M$ then

$$[\![t_1]\!]^{F(M)} = F\Big([\![t_1]\!]^M\Big) \circ i = F\Big([\![t_2]\!]^M\Big) \circ i = [\![t_2]\!]^{F(M)}$$

where i is the canonical isomorphism $F(A_1) \times \ldots \times F(A_n) \to F(A_1 \times \ldots \times A_n)$. This shows that $F(M)$ believes every equation that M believes – in particular all the axioms of $\mathbb{T}$. $\square$

Example 3.4.8. The image of a topological group under the forgetful functor $\mathsf{Top} \to \mathsf{Set}$ is an ordinary group in Set.

3.5 The equational calculus

Having established what a model is, and what it means for an equation to be true in a given model, we are now interested in the question of which deductive system is appropriate for this fragment of logic. Our rules will be of the form

$$\frac{\sigma_1 \; \ldots \; \sigma_n}{\tau}$$

where the equations $\sigma_1, \ldots, \sigma_n$ are *assumptions* and the equation τ is the *conclusion.*

Equational calculus for algebraic theories

$$\frac{}{\Gamma \mid t_1 = t_1}\,\mathsf{refl} \qquad \frac{\Gamma \mid t_1 = t_2}{\Gamma \mid t_2 = t_1}\,\mathsf{sym} \qquad \frac{\Gamma \mid t_1 = t_2 \quad \Gamma \mid t_2 = t_3}{\Gamma \mid t_1 = t_3}\,\mathsf{trans}$$

$$\frac{\Gamma \mid t_1 = t_2 \quad \overline{r} = \overline{s} : \Delta \to \Gamma}{\Delta \mid t_1(\overline{r}) = t_2(\overline{s})}\,\mathsf{sub}$$

Here the assumption $\overline{r} = \overline{s}$ is shorthand for a list of assumptions $r_1 = s_1, \dots, r_n = s_n$. To apply the substitution rule, all of these assumptions must be satisfied.

If the equation τ can be proved from the axioms of $\mathbb{T}$ using the above rules, we say that $\mathbb{T}$ proves τ and write $\mathbb{T} \vdash \tau$.

Example 3.5.1. Let $\mathbb{T}$ be the theory of an abelian group, with axioms:

$$a(bc) = (ab)c \qquad aa^{-1} = 1 \qquad a^{-1}a = 1 \qquad 1a = a \qquad a1 = a \qquad ab = ba$$

Then $\mathbb{T} \vdash (ab)a^{-1} = b$. Here is one possible derivation:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{xa^{-1} = xa^{-1}}\,\mathsf{refl} \quad \dfrac{}{ab = ba}\,\mathsf{ax}}{(ab)a^{-1} = (ba)a^{-1}}\,\mathsf{sub} \quad \dfrac{\dfrac{\dfrac{}{(ab)c = a(bc)}\,\mathsf{ax}}{(ba)x = b(ax)}\,\mathsf{sub} \quad \dfrac{}{a^{-1} = a^{-1}}\,\mathsf{refl}}{(ba)a^{-1} = b(aa^{-1})}\,\mathsf{sub}}{(ab)a^{-1} = b(aa^{-1})}\,\mathsf{trans} \quad \dfrac{\dfrac{\dfrac{}{bx = bx}\,\mathsf{refl} \quad \dfrac{}{aa^{-1} = 1}\,\mathsf{ax}}{b(aa^{-1}) = b1}\,\mathsf{sub} \quad \dfrac{\dfrac{}{a1 = a}\,\mathsf{ax}}{b1 = b}\,\mathsf{sub}}{b(aa^{-1}) = b}\,\mathsf{trans}}{(ab)a^{-1} = b}\,\mathsf{trans}$$

Theorem 3.5.2 (Soundness) If $\mathbb{T} \vdash \sigma$, then $M \vDash \sigma$ holds for all $\mathbb{T}$-models M.

Proof. We want to show that if $\mathbb{T} \vdash (t_1 = t_2)$, then for any model M in any FP-category $\mathcal{C}$ we have $M \vDash (t_1 = t_2)$, which is to say that $[\![t_1]\!]^M = [\![t_2]\!]^M$. It suffices to show that each of the rules "preserves truth": if all the assumptions are true in an interpretation, then the conclusion is also true in that interpretation. Formally this corresponds to a proof by induction on the derivation tree.

Introducing an axiom is sound since if $t_1 = t_2$ is an axiom then any model satisfies $[\![t_1]\!] = [\![t_2]\!]$ by assumption.

The soundness of the three equality rules follows directly from the properties of equality in the meta-theory:

1. For any term t_1 it holds that $[\![t_1]\!] = [\![t_1]\!]$.
2. If $[\![t_1]\!] = [\![t_2]\!]$, then $[\![t_2]\!] = [\![t_1]\!]$.
3. If $[\![t_1]\!] = [\![t_2]\!]$ and $[\![t_2]\!] = [\![t_3]\!]$, then $[\![t_1]\!] = [\![t_3]\!]$.

Finally if $[\![\Gamma \mid t_1]\!] = [\![\Gamma \mid t_2]\!]$ and $[\![\Delta \mid r_i]\!] = [\![\Delta \mid s_i]\!]$ for each i, then by Lemma 3.3.5:

$$\begin{aligned} &[\![\Delta \mid t_1(\overline{r})]\!] = [\![\Gamma \mid t_1]\!] \circ [\![\Delta \mid r_1, \dots, r_k]\!] = [\![\Gamma \mid t_1]\!] \circ ([\![\Delta \mid r_1]\!], \dots, [\![\Delta \mid r_k]\!]) = \\ &= [\![\Gamma \mid t_2]\!] \circ ([\![\Delta \mid s_1]\!], \dots, [\![\Delta \mid s_k]\!]) = [\![\Gamma \mid t_2]\!] \circ [\![\Delta \mid s_1, \dots, s_k]\!] = [\![\Delta \mid t_2(\overline{s})]\!] \end{aligned}$$

which shows that substitution is sound.

□

Suppose that we have an equation $\overline{x}, y \mid t_1 = t_2$ where the variable y does not appear in either term. It would be tempting then to deduce the equation $\overline{x} \mid t_1 = t_2$. If y has the same sort as some variable x_j we could achieve this by performing the substitution $\left[x_j/y\right]$. Similarly, if y has the same sort as t_1, then we could perform the substitution $[t_1/y]$. However, if we do not have access to any terms of the same sort as y, then removing y from the context would be unsound, as the following example shows.

Example 3.5.3. Consider the algebraic theory with two sorts (A, B), no function symbols and the single axiom

$$x{:}A, y_1{:}B, y_2{:}B \mid y_1 = y_2.$$

Take M to be the set-model where $A = \varnothing$ and $B = \{0,1\}$. Then both $[\![\, x{:}A, y_1{:}B, y_2{:}B \mid y_1 \,]\!]^M$ and $[\![\, x{:}A, y_1{:}B, y_2{:}B \mid y_2 \,]\!]^M$ are equal to the empty function $\varnothing \times B \times B \to B$, so M does believe the above equation. However, M does *not* believe the equation

$$y_1{:}B, y_2{:}B \mid y_1 = y_2$$

because the functions

$$[\![\, y_1{:}B, y_2{:}B \mid y_1 \,]\!]^M = \pi_1 : \{0,1\} \times \{0,1\} \to \{0,1\}$$
$$[\![\, y_1{:}B, y_2{:}B \mid y_2 \,]\!]^M = \pi_2 : \{0,1\} \times \{0,1\} \to \{0,1\}$$

are distinct. What is happening here is that having a variable of sort A in the context amounts to an assumption that the sort is inhabited. The axiom of this theory could be read as "if the sort A is inhabited, then all elements of sort B are equal".

3.6 The syntactic category of a theory

Recall that in the syntactic category $\mathsf{Syn}(\Sigma)$ of a signature Σ, objects are contexts and morphisms are substitutions. We will construct the syntactic category $\mathsf{Syn}(\mathbb{T})$ of a *theory* $\mathbb{T}$ as a quotient category of $\mathsf{Syn}(\Sigma)$: the objects will still be contexts, but the morphisms will now be equivalence classes of substitutions.

Let $\overline{A}$ and $\overline{B}$ be two contexts, and consider the set of all substitutions $\overline{A} \to \overline{B}$. Define an equivalence relation $=_{\mathbb{T}}$ on this set by

$$\overline{r} =_{\mathbb{T}} \overline{s} \quad \Longleftrightarrow \quad \mathbb{T} \vdash \overline{r} = \overline{s}$$

(recall that $\mathbb{T} \vdash \overline{r} = \overline{s}$ means that $\mathbb{T} \vdash r_i = s_i$ for each component of the substitutions).

Lemma 3.6.1 The equivalence relation $=_{\mathbb{T}}$ respects composition in $\mathsf{Syn}(\mathbb{T})$: If $\overline{r} =_{\mathbb{T}} \overline{s} : \overline{A} \to \overline{B}$ and $\overline{t} =_{\mathbb{T}} \overline{u} : \overline{B} \to \overline{C}$, then $\overline{t} \circ \overline{r} = \overline{u} \circ \overline{s} : \overline{A} \to \overline{C}$.

Proof. For each component of $\overline{t}$ and $\overline{u}$, an application of the substitution rule gives:

$$\frac{\overline{B} \mid t_i = u_i \qquad \overline{A} \mid \overline{r} = \overline{s}}{\overline{A} \mid t_i(\overline{r}) = u_i(\overline{s})}\,\mathsf{sub}$$

Putting the components back together we get $\mathbb{T} \vdash \overline{t}(\overline{r}) = \overline{u}(\overline{s})$ as desired. ☐

Definition 3.6.2 We define the syntactic category $\mathsf{Syn}(\mathbb{T})$ of a theory $\mathbb{T}$ to be $\mathsf{Syn}(\Sigma)/ =_{\mathbb{T}}$, where morphisms are $=_{\mathbb{T}}$-equivalence classes of substitutions.

Remark 3.6.3. If $\mathbb{T}$ is the empty theory with no axioms, then $\mathsf{Syn}(\mathbb{T}) = \mathsf{Syn}(\Sigma)$ since the only provable equations are $t = t$.

We can define a Σ-interpretation $\mathcal{U}$ in the category $\mathsf{Syn}(\mathbb{T})$ as follows:

1. $[\![A]\!]^{\mathcal{U}} = [A]$
2. $[\![f]\!]^{\mathcal{U}} = [f]$

Note the similarity to our definition of the canonical interpretation C inside $\mathsf{Syn}(\Sigma)$ (Example 3.3.4). While C corresponded to the identity functor $\mathsf{Syn}(\Sigma) \to \mathsf{Syn}(\Sigma)$, the interpretation $\mathcal{U}$ corresponds to the quotient functor $Q : \mathsf{Syn}(\Sigma) \to \mathsf{Syn}(\mathbb{T})$ that sends each substitution to its equivalence class.

Lemma 3.6.4 $[\![t]\!]^{\mathcal{U}} = [t]$ for every term t.

Proof. Straightforward induction on the structure of t. ☐

Lemma 3.6.5 The interpretation $\mathcal{U}$ is a $\mathbb{T}$-model.

Proof. For any axiom $t_1 = t_2$ we have $t_1 =_{\mathbb{T}} t_2$, so $[t_1]$ and $[t_2]$ are the same equivalence class and

$$[\![t_1]\!]^{\mathcal{U}} = [t_1] = [t_2] = [\![t_2]\!]^{\mathcal{U}}.$$

☐

Lemma 3.6.6 The model $\mathcal{U}$ is *universal*: it believes the equation σ if and only if $\mathbb{T} \vdash \sigma$.

Proof. We have

$$\mathcal{U} \vDash t_1 = t_2 \iff [\![t_1]\!]^{\mathcal{U}} = [\![t_2]\!]^{\mathcal{U}} \iff [t_1] = [t_2] \iff \mathbb{T} \vdash t_1 = t_2$$

where the second step is applying Lemma 3.6.4 and the last step is by definition of the equivalence classes in $\mathsf{Syn}(\mathbb{T})$. ☐

Let's take a step back and consider what this universal model would look like for a concrete algebraic theory. Let $\mathbb{T}$ be the the theory of groups, and let G be some particular group. Certainly all equations provable from the axioms of $\mathbb{T}$ also hold as equations in G – this is soundness. But for most groups G we could think of, it is also the case that there are equations true in G that are *not* provable from $\mathbb{T}$. For instance, if the group G happens to be abelian, then

$$ab = ba$$

is such an equation. But the special "universal group" $\mathcal{U}$ (which we must recall is not a group in Set, but instead a group in the category $\mathsf{Syn}(\mathbb{T})$) has no such "extra" equations, it *only* believes the equations that it is *required* by the axioms to believe.

Theorem 3.6.7 (Completeness) If $M \vDash \sigma$ holds for all models in all FP-categories, then $\mathbb{T} \vdash \sigma$.

Proof. If $M \vDash (t_1 = t_2)$ holds for all models, then in particular $\mathcal{U} \vDash (t_1 = t_2)$, which implies that $\mathbb{T} \vdash (t_1 = t_2)$. ☐

Note that this completeness theorem is weaker than completeness for set-models. It is in fact possible to derive completeness for set-models as a corollary, via an embedding lemma: see Appendix A.1.

3.7 Models are functors

Suppose we are given a model M. The interpretation functor $[\![\cdot]\!]^M : \mathsf{Syn}(\Sigma) \to \mathcal{C}$ then factors uniquely through the quotient functor $Q : \mathsf{Syn}(\Sigma) \to \mathsf{Syn}(\mathbb{T})$. Conversely, any functor $F : \mathsf{Syn}(\mathbb{T}) \to \mathcal{C}$ can be composed with Q to give an interpretation:

$$\begin{tikzcd}
\mathsf{Syn}(\Sigma) \arrow[r, "Q"] \arrow[rr, dashed, bend right, "{[\![\cdot]\!]^F = F \circ Q}"'] & \mathsf{Syn}(T) \arrow[r, "F"] & \mathcal{C}
\end{tikzcd}$$

Recall that $\mathbb{T}_{\mathsf{mod}}(\mathcal{C})$ is a full subcategory of $\Sigma_{\mathsf{int}}(\mathcal{C})$. Similarly, in $\mathsf{FP}(\mathsf{Syn}(\Sigma), \mathcal{C})$, there is a full subcategory consisting of those functors that factor through Q, and by the bijection above (given by composition with Q) this subcategory is equivalent to $\mathsf{FP}(\mathsf{Syn}(\mathbb{T}), \mathcal{C})$. Putting these facts together we conclude that the equivalence from Theorem 3.3.12,

$$\Sigma_{\mathsf{int}}(\mathcal{C}) \simeq \mathsf{FP}(\mathsf{Syn}(\Sigma), \mathcal{C}),$$

restricts to an equivalence of subcategories:

Theorem 3.7.1 For any algebraic theory $\mathbb{T}$ and for any FP-category $\mathcal{C}$, there is an equivalence of categories

$$\mathbb{T}_{\mathsf{mod}}(\mathcal{C}) \simeq \mathsf{FP}(\mathsf{Syn}(\mathbb{T}), \mathcal{C}).$$

3.8 The internal language of an FP-category

We have seen how to take any algebraic theory $\mathbb{T}$ and turn it into an FP-category $\mathsf{Syn}(\mathbb{T})$. This partly justifies our claim that "theories are categories" – but does the correspondence go both ways? A natural next question to ask is whether given an arbitrary FP-category $\mathcal{C}$, we can find an algebraic theory $\mathbb{T}_{\mathcal{C}}$ such that $\mathsf{Syn}(\mathbb{T}_{\mathcal{C}}) \simeq \mathcal{C}$. This is indeed possible, via the construction of the *internal language* (also known as the *internal logic*) of a category.

Let $\mathcal{C}$ be a small[1] FP-category. We define a signature $\Sigma_{\mathcal{C}}$ as follows:

1. For each $A \in \mathsf{ob}(\mathcal{C})$, include a sort A in the signature.
2. For each morphism $A_1 \times \ldots \times A_n \to B$, include a function symbol $f : A_1, \ldots, A_n \to B$ in the signature. Note that this means that a single morphism $A \to B$ gives rise to many different function symbols, one for each way of writing A as a product $A_1 \times \ldots \times A_n$.

We have a canonical interpretation of $\Sigma_{\mathcal{C}}$ inside $\mathcal{C}$: each sort comes directly from an object, and each function symbols arises from a morphism of the right type, so the canonical interpretation can simply invert the above procedure. We now define a theory $\mathbb{T}_{\mathcal{C}}$ over the signature $\Sigma_{\mathcal{C}}$ by including the axiom $t_1 = t_2$ for any terms satisfying $[\![t_1]\!] = [\![t_2]\!]$ in the canonical interpretation.

By construction the canonical interpretation is then a model of the theory $\mathbb{T}_{\mathcal{C}}$, so it corresponds to an FP-functor $[\![\cdot]\!] : \mathsf{Syn}(\mathbb{T}_{\mathcal{C}}) \to \mathcal{C}$.

Theorem 3.8.1 For any small FP-category $\mathcal{C}$, the functor $[\![\cdot]\!]$ is an equivalence $\mathsf{Syn}(\mathbb{T}_{\mathcal{C}}) \simeq \mathcal{C}$.

---

[1] We can also speak of the internal language of large categories, but then we need to expand our definition of signature to allow proper classes of sorts and function symbols, and our definition of theory to allow for a proper class of axioms. This comes with the usual subtleties of dealing with proper classes, but the construction remains unchanged.

Proof. We show that $[\![\cdot]\!]$ is essentially surjective and fully faithful:

Essentially surjective: Let A be any object of $\mathcal{C}$. There is then a sort A in $\Sigma_{\mathcal{C}}$, and a context object $[\,A\,]$ in $\mathsf{Syn}(\mathbb{T}_{\mathcal{C}})$. The interpretation of this context is the canonical interpretation of the sort A, which is the object A.

Full: Let $\overline{A}$ and $\overline{B}$ be any two objects of $\mathsf{Syn}(\mathbb{T}_{\mathcal{C}})$. Their canonical interpretations are

$$[\![\overline{A}]\!] = [\![A_1]\!] \times \ldots \times [\![A_m]\!] = A_1 \times \ldots \times A_m$$
$$[\![\overline{B}]\!] = [\![B_1]\!] \times \ldots \times [\![B_n]\!] = B_1 \times \ldots \times B_n.$$

Now an arbitrary morphism $f : A_1 \times \ldots \times A_m \to B_1 \times \ldots \times B_n$ is by the universal property of the product equal to $(f_1, \ldots, f_n)$, where $f_i : A_1 \times \ldots \times A_m \to B_i$. Each f_i has a corresponding function symbol $f_i : A_1, \ldots, A_m \to B_i$ in the signature $\Sigma_{\mathcal{C}}$. Putting these together gives a substitution $[f_1, \ldots, f_n] : \overline{A} \to \overline{B}$, the interpretation of which is by definition equal to $(f_1, \ldots, f_n) = f$.

Faithful: Let $[\,\overline{s}\,]$ and $[\,\overline{t}\,]$ be any two (equivalence classes of) substitutions in $\mathsf{Syn}(\mathbb{T}_{\mathcal{C}})$ such that $[\![\overline{s}]\!] = [\![\overline{t}]\!]$. Then for each i we have $[\![s_i]\!] = [\![t_i]\!]$, so by construction $s_i = t_i$ is an axiom of $\mathbb{T}_{\mathcal{C}}$. But then $\mathbb{T} \vdash s_i = t_i$, so $\mathbb{T} \vdash \overline{s} = \overline{t}$, so $[\,\overline{s}\,] = [\,\overline{t}\,]$ represent the same equivalence class of substitutions. $\square$

We now have a one-to-one (up to equivalence) correspondence between (small) FP-categories and (small) algebraic theories. This can be extended to an equivalence of 2-categories:

$$\mathsf{FP} \simeq \mathsf{Alg}$$

where Alg is the 2-category of algebraic theories and translations, although we will not make this idea formal here.

4 First-order logic

We now turn our attention to first-order logic. The end goal is to obtain the same results for first-order theories as we did for algebraic theories: we want to make a connection between theories and categories with appropriate structure, as well as between models and structure-preserving functors. There is considerably more structure to consider than in the algebraic case, so we have organized the pieces into four chapters. We begin in this chapter on the purely syntactic side, defining first-order formulas, sequents, theories and proofs.

4.1 Signatures and formulas

Definition 4.1.1 A first-order signature Σ is defined by the following data.

1. A set of sorts.
2. A set of function symbols, each with an associated arity $A_1, \dots, A_n$ and sort B.
3. A set of relation symbols, each with an associated arity $A_1, \dots, A_n$.
4. For each sort A, a distinguished relation symbol $=_A$ of arity A, A.

The equality symbols are special in that their interpretation will be fixed, but from the perspective of the syntax they behave just like any other relation symbol.

The sets of contexts, terms and substitutions are defined the same as in the algebraic case. These can then be put together into *formulas* using the relation symbols, logical *connectives* and *quantifiers*.

Full first-order logic includes the nullary connectives $\{\top, \bot\}$, the binary connectives $\{\wedge, \vee, \to\}$ and the quantifiers $\{\exists, \forall\}$. We do not include the unary connective $\neg$, and instead consider $\neg\varphi$ as notational shorthand for $\varphi \to \bot$.

Rather than fixing a specific set of connectives in our definition of formulas, we make the definition for an arbitrary "logical signature". This allows us to re-use the same definition for different logical fragments.

The logical signature could in principle include any number of connectives and quantifiers, but for our purposes it will always be a subset of $\{\top, \wedge, \bot, \vee, \to, \exists, \forall\}$.

Definition 4.1.2 The set of first-order formulas-in-context over signature Σ is recursively defined by the following clauses:

1. If R is a relation symbol with arity $A_1, \dots, A_k$ and $t_i : \Gamma \to A_i$ are terms, then $\Gamma \mid R(t_1, \dots, t_k)$ is a formula.
2. If C is an n-ary logical connective and $\varphi_1, \dots, \varphi_n$ are formulas in context Γ, then $\Gamma \mid C(\varphi_1, \dots, \varphi_n)$ is a formula.
3. If Q is a quantifier and $\Gamma, x{:}A \mid \varphi$ is a formula, then $\Gamma \mid (Qx{:}A).\varphi$ is a formula.

Definition 4.1.3 If we restrict the logical signature to $\{\top, \wedge, \exists\}$ we obtain the set of *regular* formulas, and from $\{\top, \wedge, \bot, \vee, \exists\}$ we get the *coherent* formulas.

Remark 4.1.4. The sort of a variable bound by a quantifier may be omitted if we do not need to directly refer to it.

Remark 4.1.5. We may abbreviate $R(t_1, \dots, t_k)$ to $R(\bar{t})$ and $C(\varphi_1, \dots, \varphi_n)$ to $C(\overline{\varphi})$. For binary relation symbols and binary logical connectives we use infix notation when convenient.

Remark 4.1.6. We previously defined the set of terms $t : \Gamma \to A$ for a *fixed* context Γ. Now we are simultaneously defining the set of formulas in all contexts. Specifically, the quantifiers require us to allow the context to vary: to construct the formula $\exists x.\varphi$ in context Γ we must first construct the formula φ in the extended context Γ, x.

Remark 4.1.7. By the above definition, bound variables are always disjoint from the free variables. This is just one possible choice of how to deal with bound variables, and what choice we make here does not affect any of the upcoming results in a significant way. For some more discussion on exactly which formulas are valid in our convention, see Appendix A.2.

We wish to identify formulas that only differ in the names of variables. To do this we first make precise the notion of renaming variables. For a given formula $[\Gamma \mid \varphi]$, consider the set V of variables that appear either in Γ or in some quantifier inside φ. Equivalently, these are the variables that appeared in some context at some stage of the construction of φ. A renaming is then an injective function from V to the set of variables.

The renaming r is applied to a formula by replacing every occurence of a variable x (in the context, in quantifiers, free occurences as well as bound occurences) with the variable $r(x)$. Note that a renaming of variables is applied simultaneously to the context and the formula, and must not collapse distinct variables. If the formula φ can be transformed into the formula ψ by a renaming of variables, we say that φ and ψ are *α-equivalent.*

Example 4.1.8. The formulas

$$x, y \mid x = y \qquad y, x \mid y = x \qquad a, b \mid a = b$$

are all α-equivalent. However, neither of the formulas

$$x \mid x = x \qquad x, y \mid y = x$$

are α-equivalent to $x, y \mid x = y$.

Intuitively, two α-equivalent formulas convey the same information: the variables are just placeholders, referring back to a specific position in the context or a specific quantifier.

Our definition of simultaneous substitution can now be extended to formulas: If $\Gamma \mid \varphi$ is a formula and $\overline{s} : \Delta \to \Gamma$ is a substitution (that is, a term $s_i : A_i$ for each variable $x_i : A_i$ of Γ) then we obtain a new formula $\Delta \mid \varphi(\overline{s})$ by substituting s_i for each free occurence of x_i (some bound variables may need to be renamed to avoid capturing new variables appearing in s_i).

Definition 4.1.9 Let $\Gamma \mid \varphi$ be a formula and $\overline{s} : \Delta \to \Gamma$ a substitution. If necessary, rename the bound variables of φ so that they are disjoint from the variables of Δ. Then define the formula $\Delta \mid \varphi(\overline{s})$ by recursion as follows:

1. $R(t_1, \dots, t_n)(\overline{s}) = R(t_1(\overline{s}), \dots, t_n(\overline{s}))$
2. $C(\psi_1, \dots, \psi_n)(\overline{s}) = C(\psi_1(\overline{s}), \dots, \psi_n(\overline{s}))$
3. $(Qx.\psi)(\overline{s}) = Qx.(\psi(\overline{s}, x))$

Remark 4.1.10. Because of the way we defined our syntax, the expression $Qx.(\psi(\overline{s}, x))$ isn't even a valid formula in a context which includes the variable x. If we try to apply a substitution without first doing the necessary renaming of bound variables, we can imagine the substitution "failing" with an "error message" that says "the attempted substitution is not allowed since it could lead to accidental capturing of free variables".

Remark 4.1.11. We recall from the algebraic case that a term $t(x_1, \dots, x_n)$ behaves much the same as a function $f(x_1, \dots, x_n)$. The substitution $t(\overline{s})$ can be thought of as "plugging in" the values s_i into the expression t, just as $f(\overline{s})$ can be thought of as evaluating f at $\overline{s}$. Similarly, a formula $\varphi(x_1, \dots, x_n)$ behaves much the same as a relation $R(x_1, \dots, x_n)$, and a substitution $\varphi(s_1, \dots, s_n)$ can be thought of analagously to evaluating R at $\overline{s}$.

Remark 4.1.12. Sometimes it is more natural to think of the substitution as acting on the formula, in which case we may write $\overline{s}^*(\varphi)$ in place of $\varphi(\overline{s})$.

4.2 Sequents and theories

In place of the equations that played a central role for algebraic theories, we will now consider *sequents* of the form

$$[\,\Gamma \mid \varphi \vdash \psi\,]$$

where φ and ψ are both formulas in context Γ[2]. We can read a sequent as "φ entails ψ (in the context Γ)". We will sometimes omit the context Γ for readability, but formally it should always be included. Note that a sequent without specified context can be assigned a canonical context consisting of all the free variables that occur in either formula, in the order of first appearance.

Remark 4.2.1. In logical fragments which include implication ($\to$) the sequent $\varphi \vdash \psi$ carries the same information as the formula $\varphi \to \psi$ (we will make this precise in Remark 4.3.2). An important difference between them is that while implication can be used multiple times inside a formula, sequents are never nested. Thus being able to interpret sequents is a weaker assumption than being able to interpret implication.

Remark 4.2.2. It is also possible to consider sequents of the form $\varphi_1, \dots, \varphi_n \vdash \psi$. We will always include $\wedge$ in our fragments, so we can represent such a sequent by $\varphi_1 \wedge \dots \wedge \varphi_n \vdash \psi$. Restricting to one formula on either side of the sequent simplifies our notation somewhat.

Definition 4.2.3 A *first-order theory* consists of a signature together with a set of sequents, the axioms of the theory. A theory is regular/coherent if all of its axioms are regular/coherent.

Example 4.2.4. Any algebraic theory can also be considered as a first-order theory, by writing the axioms as $\top \vdash t_1 = t_2$.

Example 4.2.5. The (regular) theory of a partially ordered set has a single sort, a binary relation symbol $\leq$ and three axioms:

$$\begin{aligned} \top &\vdash x \leq x && \text{(reflexivity)} \\ x \leq y \wedge y \leq x &\vdash x = y && \text{(antisymmetry)} \\ x \leq y \wedge y \leq z &\vdash x \leq z && \text{(transitivity)} \end{aligned}$$

If we additionally include the axiom

$$\top \vdash x \leq y \vee y \leq x$$

we obtain the (coherent) theory of a totally ordered set.

Example 4.2.6. We can formulate the theory of simple graphs as a coherent theory, using a sort V for vertices and a relation $E \rightarrowtail V \times V$ for edges, with the two axioms

---

[2] More precisely, $[\,\Gamma \mid \varphi\,]$ and $[\,\Gamma \mid \psi\,]$ are α-equivalence classes of formulas. While other representatives from these equivalence classes may have different variable names in the context, the *type* of the context (consisting of the list of sorts) is always the same as the type of Γ.

$$E(x,x) \vdash \bot \qquad\qquad E(x,y) \vdash E(y,x)$$

Example 4.2.7. The theory of categories can be described as a regular theory.

The signature consists of two sorts, three function symbols and one relation symbol:

$$O \text{ (objects)} \qquad M \text{ (morphisms)}$$

$$\mathrm{id} : O \to M \qquad \mathrm{dom} : M \to O \qquad \mathrm{cod} : M \to O \qquad C \rightarrowtail M, M, M$$

The intended meaning of $C(f,g,h)$ is that $f \circ g = h$.

The first three axioms specify under which conditions morphisms can be composed, and assert that when the composition exists it is unique:

1. $C(f,g,h) \vdash \mathrm{dom}(f) = \mathrm{cod}(g) \wedge \mathrm{cod}(f) = \mathrm{cod}(h) \wedge \mathrm{dom}(g) = \mathrm{dom}(h)$
2. $\mathrm{dom}(f) = \mathrm{cod}(g) \vdash (\exists h{:}M).C(f,g,h)$
3. $C(f,g,h) \wedge C(f,g,h') \vdash h = h'$

The next two axioms assert that the identity morphism works as expected and that composition is associative:

4. $\top \vdash C(f, \mathrm{id}(\mathrm{dom}(f)), f) \wedge C(\mathrm{id}(\mathrm{cod}(f)), f, f)$
5. $C(f,g,a) \wedge C(a,h,b) \wedge C(g,h,c) \vdash C(f,c,b)$

The last axiom uses auxiliary definitions

$$\begin{aligned} a &= f \circ g \\ b &= a \circ h = (f \circ g) = h \\ c &= g \circ h \end{aligned}$$

in order to rewrite the associativity rule as a single composition:

$$\underbrace{\overbrace{(f \circ g)}^{a} \circ h}_{b} = f \circ \overbrace{(g \circ h)}^{c} \quad \Longleftrightarrow \quad b = f \circ c$$

Example 4.2.8. The theory of an *inhabited object* has a single sort A and an axiom:

$$\top \vdash (\exists x{:}A).\top$$

This is subtly different from the algebraic theory of a *pointed object*, which instead includes a function symbol $p : 1 \to A$. A homomorphism between pointed objects must preserve the base point, while a homomorphism between inhabited objects will have no such requirements. (We will define homomorphisms between models of first-order theories in Section 7.2).

4.3 The sequent calculus

For algebraic theories, we had an equational calculus with rules that allowed us to conclude that a certain equation is true given that some other equations are true. For the more general first-order theories we consider now, the rules have the same structure but with sequents in place of equations. A typical rule thus looks like

$$\frac{\sigma_1 \quad \sigma_2}{\tau}$$

where σ_1, σ_2 and τ are all sequents. The intended interpretation of the rule is that if σ_1 and σ_2 are true, then τ is also true. If both of the rules

$$\begin{array}{c} \sigma \\ \hline \tau \end{array} \quad \text{and} \quad \begin{array}{c} \tau \\ \hline \sigma \end{array}$$

are valid, we summarize them with a single "double rule":

$$\begin{array}{c} \sigma \\ \hline\hline \tau \end{array}$$

Similarly, if all three of the rules

$$\begin{array}{c} \sigma_1 \quad \sigma_2 \\ \hline \tau \end{array} \qquad \begin{array}{c} \tau \\ \hline \sigma_1 \end{array} \qquad \begin{array}{c} \tau \\ \hline \sigma_2 \end{array}$$

are valid, we summarize them with the double rule:

$$\begin{array}{c} \sigma_1 \quad \sigma_2 \\ \hline\hline \tau \end{array}$$

In general, a double rule says the following: If *all* of the sequents on one side of the double line are true, then you are allowed to conclude *any one* of the sequents on the other side.

We will use the following set of deductive rules:

Sequent calculus for first-order theories

$$\begin{array}{c} \\ \hline \varphi \vdash \varphi \end{array}\ \mathsf{refl} \qquad \begin{array}{c} \varphi \vdash \psi \quad \psi \vdash \chi \\ \hline \varphi \vdash \chi \end{array}\ \mathsf{cut} \qquad \begin{array}{c} \varphi \vdash \psi \\ \hline \varphi(\overline{s}) \vdash \psi(\overline{s}) \end{array}\ \mathsf{sub}$$

$$\begin{array}{c} \top \vdash \varphi(x, x) \\ \hline\hline x = y \vdash \varphi(x, y) \end{array}\ =$$

$$\begin{array}{c} \\ \hline \varphi \vdash \top \end{array}\ \top \qquad \begin{array}{c} \varphi \vdash \psi \quad \varphi \vdash \chi \\ \hline\hline \varphi \vdash \psi \wedge \chi \end{array}\ \wedge$$

$$\begin{array}{c} \\ \hline \bot \vdash \varphi \end{array}\ \bot \qquad \begin{array}{c} \varphi \vdash \chi \quad \psi \vdash \chi \\ \hline\hline \varphi \vee \psi \vdash \chi \end{array}\ \vee$$

$$\begin{array}{c} \varphi \wedge \psi \vdash \chi \\ \hline\hline \varphi \vdash \psi \to \chi \end{array}\ \to$$

$$\begin{array}{c} \varphi(\overline{x}, y) \vdash \psi(\overline{x}) \\ \hline\hline \exists y . \varphi(\overline{x}, y) \vdash \psi(\overline{x}) \end{array}\ \exists \qquad \begin{array}{c} \varphi(\overline{x}) \vdash \psi(\overline{x}, y) \\ \hline\hline \varphi(\overline{x}) \vdash \forall y . \psi(\overline{x}, y) \end{array}\ \forall$$

If the sequent σ can be derived from the axioms of $\mathbb{T}$ we write $\mathbb{T} \vdash \sigma$ and say that $\mathbb{T}$ proves σ.

Remark 4.3.1. In the equational calculus we had three rules asserting that equality is reflexive, symmetric and transitive. We have now replaced those with a more general double rule for equality, from which all the other properties can be derived. For instance, reflexivity of equality can be proved by

$$\begin{array}{cl} \hline x = y \vdash x = y & \mathsf{refl} \\ \hline\hline \top \vdash x = x & = \\ \hline \top \vdash t_1 = t_1 & \mathsf{sub} \end{array}$$

where we applied the =-rule with $x = y$ filling the role of the formula $\varphi(x, y)$,

Remark 4.3.2. Let $[\overline{x} \mid \psi \vdash \chi]$ be an arbitrary sequent. Then

$$\begin{array}{c} \overline{x} \mid \psi \vdash \chi \\ \hline\hline \overline{x} \mid \top \land \psi \vdash \chi \\ \hline\hline \overline{x} \mid \top \vdash \psi \to \chi \\ \hline\hline - \mid \top \vdash \forall \overline{x}.(\psi \to \chi) \end{array}$$

so that an arbitrary sequent-in-context is logically equivalent to a sequent of the form $\top \vdash \varphi$ in the empty context. Indeed, many first-order theories are formulated in this way, without mention of sequents or contexts.

4.4 Rules with extra assumptions

We want to be able to apply each deductive rule in a situation where we have some side assumptions $H = h_1 \land \ldots \land h_n$. For instance, for the rules $=$, $\exists$ and $\lor$, we want to be able to use the following stronger rules:

$$\begin{array}{c} H(x) \vdash \varphi(x,x) \\ \hline\hline H(x) \land x = y \vdash \varphi(x,y) \end{array} =_H \qquad \begin{array}{c} H(\overline{x}) \land \varphi(\overline{x},y) \vdash \psi(\overline{x}) \\ \hline\hline H(\overline{x}) \land \exists y.\varphi(\overline{x},y) \vdash \psi(\overline{x}) \end{array} \exists_H \qquad \begin{array}{c} H \land \varphi \vdash \chi \quad H \land \psi \vdash \chi \\ \hline\hline H \land (\varphi \lor \psi) \vdash \chi \end{array} \lor_H$$

Each of these are easily proved by temporarily shifting the assumption to the right of the sequent, as indicated below.

$$\begin{array}{c} H \vdash \varphi(x,x) \\ \hline\hline \top \vdash H \to \varphi(x,x) \\ \hline\hline (x = y) \vdash H \to \varphi(x,y) \\ \hline\hline H \land (x = y) \vdash \varphi(x,y) \end{array} \qquad \begin{array}{c} H \land \varphi \vdash \psi \\ \hline\hline \varphi \vdash H \to \psi \\ \hline\hline \exists y.\varphi \vdash H \to \psi \\ \hline\hline H \land \exists y.\varphi \vdash \psi \end{array} \qquad \begin{array}{c} H \land \varphi \vdash \chi \quad H \land \psi \vdash \chi \\ \hline\hline \varphi \vdash H \to \chi \quad \psi \vdash H \to \chi \\ \hline\hline \varphi \lor \psi \vdash H \to \chi \\ \hline\hline H \land (\varphi \lor \psi) \vdash \chi \end{array}$$

In fragments without implication, these proofs are not possible, so instead we replace the rules $\{=, \exists, \lor\}$ with their stronger versions $\{=_H, \exists_H, \lor_H\}$. This is not necessary for any of the other rules: for instance the rule

$$\begin{array}{c} H \land \varphi \vdash \psi \quad H \land \varphi \vdash \chi \\ \hline\hline H \land \varphi \vdash \psi \land \chi \end{array} \land_H$$

is already an instance of the $\land$-rule, and the rule $\bot_H$ can be derived as follows:

$$\dfrac{\dfrac{\dfrac{}{H \land \bot \vdash H \land \bot}\,\text{refl}}{H \land \bot \vdash \bot}\,\land \qquad \dfrac{}{\bot \vdash \varphi}\,\bot}{H \land \bot \vdash \varphi}\,\text{cut}$$

4.5 Stability of subformulas

Suppose we have a formula φ where ψ appears as a subformula (ψ is a subformula of φ if it was one of the intermediate stages of constructing φ). It would then be reasonable to assume that if $\psi \dashv\vdash \psi'$ then $\varphi \dashv\vdash \varphi[\psi'/\psi]$, where we have made some informal use of the substitution notation.

To make this precise, we want for each connective and quantifier to prove a stability rule that says that replacing any input by an equivalent formula gives an equivalent output:

$$\dfrac{\varphi \dashv\vdash \varphi' \qquad \psi \dashv\vdash \psi'}{C(\varphi,\psi) \dashv\vdash C(\varphi',\psi')} \qquad \dfrac{\varphi \dashv\vdash \varphi'}{Qx.\varphi \dashv\vdash Qx.\varphi'}$$

For the nullary connectives there is nothing to prove – the corresponding rules

$$\dfrac{}{\top \dashv\vdash \top} \qquad \dfrac{}{\bot \dashv\vdash \bot}$$

follow directly from reflexivity.

Lemma 4.5.1 The connectives $\wedge$ and $\vee$ are monotone increasing in both arguments:

$$\frac{\varphi \vdash \varphi' \qquad \psi \vdash \psi'}{\varphi \wedge \psi \vdash \varphi' \wedge \psi'} \qquad \frac{\varphi \vdash \varphi' \qquad \psi \vdash \psi'}{\varphi \vee \psi \vdash \varphi' \vee \psi'}$$

Proof. The soundness of the first rule can be shown as follows:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\varphi \wedge \psi \vdash \varphi \wedge \psi}\,\text{refl}}{\varphi \wedge \psi \vdash \varphi}\,\wedge \qquad \varphi \vdash \varphi'}{\varphi \wedge \psi \vdash \varphi'}\,\text{cut} \qquad \dfrac{\dfrac{\dfrac{}{\varphi \wedge \psi \vdash \varphi \wedge \psi}\,\text{refl}}{\varphi \wedge \psi \vdash \psi}\,\wedge \qquad \psi \vdash \psi'}{\varphi \wedge \psi \vdash \psi'}\,\text{cut}}{\varphi \wedge \psi \vdash \varphi' \wedge \psi'}\,\wedge$$

The derivation of the second rule is dual. □

Lemma 4.5.2 The connective $\to$ is monotone decreasing in the first argument and monotone increasing in the second argument:

$$\frac{\varphi \vdash \varphi' \qquad \psi \vdash \psi'}{\varphi' \to \psi \vdash \varphi \to \psi'}$$

Proof.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{(\varphi' \to \psi) \wedge \varphi \vdash (\varphi' \to \psi) \wedge \varphi}\,\text{refl} \qquad \varphi \vdash \varphi'}{(\varphi' \to \psi) \wedge \varphi \vdash (\varphi' \to \psi) \wedge \varphi'}\,\star \qquad \dfrac{\dfrac{}{\varphi' \to \psi \vdash \varphi' \to \psi}\,\text{refl}}{(\varphi' \to \psi) \wedge \varphi' \vdash \psi}\,\to}{(\varphi' \to \psi) \wedge \varphi \vdash \psi}\,\text{cut} \qquad \psi \vdash \psi'}{(\varphi' \to \psi) \wedge \varphi \vdash \psi'}\,\text{cut}}{\varphi' \to \psi \vdash \varphi \to \psi'}\,\to$$

We have made use of the previous lemma with the deduction marked $\star$. □

Lemma 4.5.3 The quantifiers $\exists$ and $\forall$ are monotone increasing:

$$\frac{\varphi \vdash \psi}{\exists y.\varphi \vdash \exists y.\psi} \qquad \frac{\varphi \vdash \psi}{\forall y.\varphi \vdash \forall y.\psi}$$

Proof. The derivations for $\exists$ and $\forall$ are dual. We show them here side by side in two different extremes of notational explicity:

$$\dfrac{\dfrac{\dfrac{\overline{x}, y \mid \varphi(\overline{x}, y) \vdash \psi(\overline{x}, y) \qquad \dfrac{\dfrac{\overline{x} \mid \exists y.\psi(\overline{x}, y) \vdash \exists y.\psi(\overline{x}, y)}{\overline{x} \mid \exists y.\psi(\overline{x}, y) \vdash \exists y'.\psi(\overline{x}, y')}}{\overline{x}, y \mid \psi(\overline{x}, y) \vdash \exists y'.\psi(\overline{x}, y')}\,\exists}{\overline{x}, y \mid \varphi(\overline{x}, y) \vdash \exists y'.\psi(\overline{x}, y')}\,\text{cut}}{\overline{x} \mid \exists y.\varphi(\overline{x}, y) \vdash \exists y'.\psi(\overline{x}, y')}\,\exists}{\overline{x} \mid \exists y.\varphi(\overline{x}, y) \vdash \exists y.\psi(\overline{x}, y)} \qquad \dfrac{\dfrac{\dfrac{\forall y.\varphi \vdash \forall y.\varphi}{\forall y.\varphi \vdash \varphi}\,\forall \qquad \varphi \vdash \psi}{\forall y.\varphi \vdash \psi}\,\text{cut}}{\forall y.\varphi \vdash \forall y.\psi}\,\forall$$

The left one is explicit about the context of each formula, and avoids any overlap between free and bound variables (which is not strictly speaking allowed by the way we constructed our syntax) by renaming the bound y to y'[3]. Once we've convinced ourselves that this sort of renaming trick would work in any similar situation, we can feel free to use the more readable form on the right as a shorthand. □

Lemma 4.5.4 If ψ is a subformula of φ and $\psi \dashv\vdash \psi'$, then $\varphi \dashv\vdash \varphi'$, where φ' has the subformula ψ replaced with ψ'.

Proof. Induction on the structure of φ, applying the previous lemmas. □

---

[3] This is allowed since we have identified α-equivalent formulas. By renaming y to y', we are switching to a different representative of the same equivalence class.

5 Categorical semantics

We now move on from syntax to semantics, and investigate what extra structure a category $\mathcal{C}$ must have in order to be able to contain models of a given first-order theory $\mathbb{T}$.

We will then show that the sequent calculus is sound for this class of models.

5.1 Interpretation

An interpretation of a first-order theory inside a category $\mathcal{C}$ consists of:

1. For each sort A, an object $\llbracket A \rrbracket$
2. For each function symbol $f : A_1, \ldots, A_n \to B$, a morphism $\llbracket f \rrbracket : \llbracket A_1 \rrbracket \times \ldots \times \llbracket A_n \rrbracket \to \llbracket B \rrbracket$
3. For each relation symbol $R \rightarrowtail A_1, \ldots, A_n$, a subobject $\llbracket R \rrbracket \rightarrowtail \llbracket A_1 \rrbracket \times \ldots \times \llbracket A_n \rrbracket$

The equality relation $=_A$ is always interpreted as the diagonal subobject

$$\Delta : \llbracket A \rrbracket \rightarrowtail \llbracket A \rrbracket \times \llbracket A \rrbracket.$$

For this definition to make sense, we must require the category $\mathcal{C}$ to have finite products. The interpretation can then be extended to contexts, terms and substitutions just like in the algebraic case, and it still holds true that substition into a term is interpreted as composition, as we proved in Lemma 3.3.5.

The next step is to extend the interpretation to formulas. Before going into the details, we illustrate the intuitive idea with a concrete example in the category Set.

Suppose that our signature is that of a ring: we have a sort R with addition and multiplication operations and constants 0 and 1. An interpretation of this signature inside the category Set could for instance be $\llbracket R \rrbracket = \mathbb{R}$, equipped with the usual ring structure $(0, 1, +, \times)$.

Now consider the following formula-in-context:

$$[\, x{:}R, y{:}R \mid x^2 + y^2 = 1 \,]$$

Our notation suggests that it might be natural to interpret this formula as the set

$$\{(x, y) \in \mathbb{R} \times \mathbb{R} \mid x^2 + y^2 = 1\}$$

which we can think of as a *subset* of the plane $\mathbb{R} \times \mathbb{R}$. We generalize this idea to arbitrary theories and categories by interpeting each formula as a *subobject* of the interpretation of the context:

$$\llbracket \Gamma \mid \varphi \rrbracket \rightarrowtail \llbracket \Gamma \rrbracket$$

We break this definition down into the subproblems of interpreting first atomic formulas (relation symbols), then each of the connectives, then both of the quantifiers. Each of these steps will require some new extra structure to exist in the category $\mathcal{C}$. Rather than stating what structure $\mathcal{C}$ needs up front, we add new requirements as we go along. Afterwards we will summarize the whole list of requirements and show how it can be simplified.

Interpreting relations

We define the interpretation of a relation symbol by $\llbracket R(\bar{t}) \rrbracket := \llbracket \bar{t} \rrbracket^*(\llbracket R \rrbracket)$:

$$\begin{tikzcd}
\llbracket R(t_1, \dots , t_n) \rrbracket \arrow[d, tail] & \llbracket R \rrbracket \arrow[d, tail] \\
\llbracket \Gamma \rrbracket \arrow[r, "\llbracket t_1, \dots , t_n \rrbracket"] & \llbracket A_1, \dots , A_n \rrbracket
\end{tikzcd}$$

For this definition to make sense we need $\mathcal{C}$ to have *pullbacks of monics.*

Remark 5.1.1. In particular, since we defined $\llbracket =_A \rrbracket = \Delta_{\llbracket A \rrbracket} : \llbracket A \rrbracket \rightarrowtail \llbracket A \rrbracket \times \llbracket A \rrbracket$, equality is always interpreted as the pullback

$$\begin{tikzcd}
\llbracket t_1 =_A t_2 \rrbracket \arrow[r, tail, dashed] \arrow[d, dashed] & \llbracket \Gamma \rrbracket \arrow[d, "\llbracket t_1, t_2 \rrbracket"] \\
\llbracket A \rrbracket \arrow[r, tail, "\Delta_{\llbracket A \rrbracket}"'] & \llbracket A \rrbracket \times \llbracket A \rrbracket
\end{tikzcd}$$

which is also an equalizer:

$$\begin{tikzcd}
\llbracket t_1 =_A t_2 \rrbracket \arrow[r, tail, dashed] & \llbracket \Gamma \rrbracket \arrow[r, shift left, "\llbracket t_1 \rrbracket"] \arrow[r, shift right, "\llbracket t_2 \rrbracket"'] & \llbracket A \rrbracket
\end{tikzcd}$$

Interpreting connectives

To interpret an n-ary logical connective C, we need for each context Γ an n-ary operation $\llbracket C \rrbracket_\Gamma$ on $\mathsf{Sub}(\llbracket \Gamma \rrbracket)$. We will then define the interpretation as

$$\llbracket \Gamma \mid C(\varphi_1, \dots , \varphi_n) \rrbracket := \llbracket C \rrbracket_\Gamma(\llbracket \Gamma \mid \varphi_1 \rrbracket, \dots , \llbracket \Gamma \mid \varphi_n \rrbracket).$$

For each of the connectives $\top, \wedge, \bot, \vee, \rightarrow$ we thus need corresponding operations

$$\llbracket \top \rrbracket_\Gamma, \llbracket \wedge \rrbracket_\Gamma, \llbracket \bot \rrbracket_\Gamma, \llbracket \vee \rrbracket_\Gamma, \llbracket \rightarrow \rrbracket_\Gamma$$

on each $\mathsf{Sub}(\llbracket \Gamma \rrbracket)$. We will usually omit the Γ subscript, and sometimes also the interpretation brackets, for readability.

To ensure the correct properties, we require each connective to satisfy a specific condition. We choose to formulate them in a way that directly mirrors the deductive rules associated to each connective.

Connective	**Condition**	**Categorical interpretation**	**Adjunction**
$\top$	$A \leq \top$	$\top$ is a terminal object in $\mathsf{Sub}(\Gamma)$	$* \dashv \top$
$\wedge$	$\begin{array}{c} A \leq B \quad A \leq C \\ \hline\hline A \leq B \wedge C \end{array}$	$\wedge$ is a product on $\mathsf{Sub}(\Gamma)$	$\Delta \dashv \wedge$
$\bot$	$\bot \leq A$	$\bot$ is an initial object in $\mathsf{Sub}(\Gamma)$	$\bot \dashv *$
$\vee$	$\begin{array}{c} A \leq C \quad B \leq C \\ \hline\hline A \vee B \leq C \end{array}$	$\wedge$ is a coproduct on $\mathsf{Sub}(\Gamma)$	$\vee \dashv \Delta$
$\rightarrow$	$\begin{array}{c} A \wedge B \leq C \\ \hline\hline A \leq B \rightarrow C \end{array}$	$\rightarrow$ is an exponential on $\mathsf{Sub}(\Gamma)$	$(- \wedge B) \dashv (B \rightarrow -)$

From the above conditions we can derive the properties

$$\frac{A \leq A' \quad B \leq B'}{A \wedge B \leq A' \wedge B'}, \qquad \frac{A \leq A' \quad B \leq B'}{A \vee B \leq A' \vee B'}, \qquad \frac{A \leq A' \quad B \leq B'}{A' \rightarrow B \leq A \rightarrow B'}$$

in exactly the same way we did in Section 4.5. Thus the operations are not just functions between sets but in fact *functors* between posets:

$$\begin{aligned} \top, \bot &: 1 \rightarrow \mathsf{Sub}(\Gamma) \\ \wedge, \vee &: \mathsf{Sub}(\Gamma) \times \mathsf{Sub}(\Gamma) \rightarrow \mathsf{Sub}(\Gamma) \\ \rightarrow &: \mathsf{Sub}(\Gamma)^{\mathsf{op}} \times \mathsf{Sub}(\Gamma) \rightarrow \mathsf{Sub}(\Gamma) \end{aligned}$$

The rightmost column in the above table shows that each “condition” is in fact an adjunction between functors. Here $*$ denotes the unique functor $\mathsf{Sub}(\Gamma) \rightarrow 1$ and Δ denotes the diagonal $(\mathrm{id}, \mathrm{id}) : \mathsf{Sub}(\Gamma) \rightarrow \mathsf{Sub}(\Gamma) \times \mathsf{Sub}(\Gamma)$.

Remark 5.1.2. Limits, colimits and adjoints are unique up to isomorphism, and in a poset isomorphism is equality. Thus the structure $\{\top, \wedge, \bot, \vee, \rightarrow\}$ is unique if it exists.

For each connective, we impose the extra condition that the interpretation must be *stable under pullbacks.* This means that for any $f : A \rightarrow B$ and $S_1, \dots, S_n \in \mathsf{Sub}(B)$ we have

$$f^*(\llbracket C \rrbracket_B(S_1, \dots, S_n)) = \llbracket C \rrbracket_A(f^*(S_1), \dots, f^*(S_n)).$$

This condition is needed to ensure that the connectives commute with substitutions.

Example 5.1.3. If $X \in \mathsf{Set}$, then $\mathsf{Sub}(X)$ can be identified with the poset of subsets of X, ordered by inclusion. In this case all the structure $\{\top, \wedge, \bot, \vee, \rightarrow\}$ exists and is given by:

$$\top = X, \qquad A \wedge B = A \cap B,$$

$$\bot = \varnothing, \qquad A \vee B = A \cup B,$$

$$A \rightarrow B = A^{\mathsf{C}} \cup B$$

Interpreting quantifiers

To interpret a quantifier Qx in the context Γ, we need an operation

$$\llbracket Q \rrbracket : \mathsf{Sub}(\llbracket \Gamma, x{:}A \rrbracket) \rightarrow \mathsf{Sub}(\llbracket \Gamma \rrbracket)$$

and due to the results of Section 4.5, both $[\![\,\exists\,]\!]$ and $[\![\,\forall\,]\!]$ will in fact be covariant functors.

Rather than indexing these functors by their domain, we index them by the projection $\pi : [\![\,\Gamma\,]\!] \times [\![\,\Gamma, x\!:\!A\,]\!] \to [\![\,\Gamma\,]\!]$, since the condition on each quantifier will be stated in terms of the associated pullback functor

$$\pi^* : \mathsf{Sub}([\![\,\Gamma\,]\!]) \to \mathsf{Sub}([\![\,\Gamma, x\!:\!A\,]\!]).$$

In more general terms, we require for each projection $\pi : X \times Y \to X$ that there exists two associated functors $\exists_\pi$ and $\forall_\pi$, satisfying the following adjunctions.

Quantifier	**Condition**	**Adjunction**
$\exists$	$\dfrac{A \leq \pi^* B}{\exists_\pi A \leq B}$	$\exists_\pi \dashv \pi^*$
$\forall$	$\dfrac{\pi^* A \leq B}{A \leq \forall_\pi B}$	$\pi^* \dashv \forall_\pi$

Once again, if this structure exists it is unique.

Example 5.1.4. In Set, both the adjoints exist. If $\pi : X \times Y \to X$ and $S \subseteq X \times Y$ then:

$$\exists_\pi(S) = \{x \in X \mid \exists y.(y \in Y \ \wedge \ (x, y) \in S)\}$$

$$\forall_\pi(S) = \{x \in X \mid \forall y.(y \in Y \to (x, y) \in S)\}$$

In order for a quantifier to commute with substitutions, we want the following diagram to commute for every substition $\overline{s} : \Delta \to \Gamma$:

$$\begin{tikzcd}
\mathsf{Sub}([\![\,\Delta, x\!:\!A\,]\!]) \arrow[d, "Q_\pi"] & \mathsf{Sub}([\![\,\Gamma, x\!:\!A\,]\!]) \arrow[l, "{[\![\,\overline{s}, x\,]\!]^*}"'] \arrow[d, "Q_\pi"'] \\
\mathsf{Sub}([\![\,\Delta\,]\!]) & \mathsf{Sub}([\![\,\Gamma\,]\!]) \arrow[l, "{[\![\,\overline{s}\,]\!]^*}"']
\end{tikzcd}$$

This is saying that given a subobject of $[\![\,\Gamma, x\!:\!A\,]\!]$, the following two compositions give the same result:

1. First apply the quantifier to get a subobject of $[\![\,\Gamma\,]\!]$, then apply the substitution $\overline{s}$ to get a subobject of $[\![\,\Delta\,]\!]$.
2. First apply the substitution $(\overline{s}, x)$ which leaves x unchanged and gives a subobject of $[\![\,\Delta, x\!:\!A\,]\!]$, then apply the quantifier to get a subobject of $[\![\,\Delta\,]\!]$.

Note that this requirement

$$[\![\,\overline{s}\,]\!]^* \circ Q_\pi = Q_\pi \circ [\![\,\overline{s}, x\,]\!]^*$$

corresponds exactly to how we *defined* substitution in the syntax:

$$(\overline{s})^*(Qx.\psi) = Qx.((\overline{s}, x)^*(\psi)).$$

Definition 5.1.5 The family Q_π of functors is *stable under pullback* if for any $f : X \to Y$, the following diagram commutes:

$$\begin{tikzcd}
\mathsf{Sub}(X \times A) \arrow[d, "Q_\pi"] & \mathsf{Sub}(Y \times A) \arrow[l, "(f, \mathrm{id})^*"'] \arrow[d, "Q_\pi"] \\
\mathsf{Sub}(X) & \mathsf{Sub}(Y) \arrow[l, "f^*"']
\end{tikzcd}$$

This stability condition is also known as the *Beck–Chevalley condition.*

Interpreting formulas

We now know how to interpret relation symbols, connectives and quantifiers, and we can collect these together in a recursive definition to interpret arbitrary formulas.

Definition 5.1.6 The interpretation of an arbitrary formula is recursively defined by the following three clauses:

1. $[\![R(t_1, \dots, t_n)]\!] := [\![t_1, \dots, t_n]\!]^*([\![R]\!])$
2. $[\![C(\varphi_1, \dots, \varphi_n)]\!] := [\![C]\!]([\![\varphi_1]\!], \dots, [\![\varphi_n]\!])$
3. $[\![\Gamma \mid Qx.\varphi]\!] := [\![Q]\!]([\![\Gamma, x \mid \varphi]\!])$

Lemma 5.1.7 If φ and φ' are α-equivalent, then $[\![\varphi]\!] = [\![\varphi']\!]$.

Proof. There are only two places in our recursive definition where we make any reference at all to variables:

1. In the base case for interpretation of a term (Definition 3.3.1), where a variable that appears at position j in the context is interpreted as the projection π_j.
2. In the quantifier clause above, where a bound variable is moved into the free variables.

In both of these cases, the actual name of the variable has no effect on the interpretation, only the position inside the context matters. Since α-equivalence preserves this relationship by renaming the context and the formula simultaneously, it follows that α-equivalent formulas share the same interpretation. □

Lemma 5.1.8 If $[\, \Gamma \mid \varphi \,]$ is a formula and $\overline{s} : \Delta \to \Gamma$ is a substitution, then the interpretation of $[\, \Delta \mid \varphi(\overline{s}) \,]$ is given by pullback:

$$\begin{tikzcd}
{[\![\varphi(\overline{s})]\!]} \arrow[r] \arrow[d, tail] & {[\![\varphi]\!]} \arrow[d, tail] \\
{[\![\Delta]\!]} \arrow[r, "{[\![\overline{s}]\!]}"] & {[\![\Gamma]\!]}
\end{tikzcd}$$

Proof. We show that $[\![\varphi(\overline{s})]\!] = [\![\overline{s}]\!]^*([\![\varphi]\!])$ by induction over the structure of φ. The proof is straightforward, just applying definitions and using the properties we have ensured. In particular, we use that the interpretations of connectives and quantifers are stable under pullback.

$$\begin{aligned}
[\![R(\overline{t})(\overline{s})]\!] &= [\![R(\overline{t}(\overline{s}))]\!] = [\![\overline{t}(\overline{s})]\!]^*([\![R]\!]) = [\![\overline{t} \circ \overline{s}]\!]^*([\![R]\!]) = ([\![\overline{t}]\!] \circ [\![\overline{s}]\!])^*([\![R]\!]) \\
&= ([\![\overline{s}]\!]^* \circ [\![\overline{t}]\!]^*)([\![R]\!]) = [\![\overline{s}]\!]^*([\![\overline{t}]\!]^*([\![R]\!])) = [\![\overline{s}]\!]^*([\![R(\overline{t})]\!])
\end{aligned}$$

$$\begin{aligned}[\![\, C(\varphi_1, \dots, \varphi_n)(\overline{s}) \,]\!] &= [\![\, C(\varphi_1(\overline{s}), \dots, \varphi_n(\overline{s})) \,]\!] \\ &= [\![\, C \,]\!]([\![\, \varphi_1(\overline{s}) \,]\!], \dots, [\![\, \varphi_n(\overline{s}) \,]\!]) \\ &= [\![\, C \,]\!]([\![\, \overline{s} \,]\!]^*([\![\, \varphi_1 \,]\!]), \dots, [\![\, \overline{s} \,]\!]^*([\![\, \varphi_n \,]\!])) \\ &= [\![\, \overline{s} \,]\!]^*([\![\, C \,]\!]([\![\, \varphi_1 \,]\!], \dots, [\![\, \varphi_n \,]\!])) \\ &= [\![\, \overline{s} \,]\!]^*([\![\, C(\varphi_1, \dots, \varphi_n) \,]\!])\end{aligned} \qquad \begin{aligned}[\![\, (Qx.\varphi)(\overline{s}) \,]\!] &= [\![\, Qx.(\varphi(\overline{s}, x)) \,]\!] \\ &= [\![\, Q \,]\!]([\![\, \varphi(\overline{s}, x) \,]\!]) \\ &= [\![\, Q \,]\!]([\![\, \overline{s}, x \,]\!]^*([\![\, \varphi \,]\!])) \\ &= [\![\, \overline{s} \,]\!]^*([\![\, Q \,]\!]([\![\, \varphi \,]\!])) \\ &= [\![\, \overline{s} \,]\!]^*([\![\, Qx.\varphi \,]\!])\end{aligned}$$

□

5.2 Regular, coherent and Heyting categories

The following table summarizes all the structure we needed in order to define the interpretation of a formula:

To interpret...	**We need...**
Contexts	Finite products in $\mathcal{C}$
Relation symbols	Pullbacks of monics in $\mathcal{C}$
$\top$	Terminal object $\top$ in each $\mathsf{Sub}(X)$
$\wedge$	Binary product $\wedge$ on each $\mathsf{Sub}(X)$
$\bot$	Initial object $\bot$ in each $\mathsf{Sub}(X)$
$\vee$	Binary coproduct $\vee$ on each $\mathsf{Sub}(X)$
$\to$	Exponential $\to$ on each $\mathsf{Sub}(X)$
$\exists$	Left adjoint $\exists_\pi \dashv \pi^*$ for each $\pi : X \times Y \to X$
$\forall$	Right adjoint $\pi^* \dashv \forall_\pi$ for each $\pi : X \times Y \to X$

We always require all the structure to be stable under pullback, so if we say that a category "has $\exists$" we mean that it has all the adjoints $\exists_\pi$ and that they satisfy the Beck–Chevalley condition.

Definition 5.2.1 A category is…

- …*regular* if it has $\{\mathsf{FP}, \mathsf{PM}, \top, \wedge, \exists_\pi\}$,
- …*coherent* it it has $\{\mathsf{FP}, \mathsf{PM}, \top, \wedge, \bot, \vee, \exists_\pi\}$,
- …*Heyting* if it has $\{\mathsf{FP}, \mathsf{PM}, \top, \wedge, \bot, \vee, \to, \exists_\pi, \forall_\pi\}$.

Here FP stands for finite products and PM stands for pullbacks of monics.

Note that this definition matches exactly the logical signatures of Definition 4.1.3, so that a category is regular/coherent/Heyting if and only if it has the necessary structure to interpret any regular/coherent/first-order theory.

We will now analyze which parts of the structure can be constructed in terms of others, to simplify our list of requirements.

Lemma 5.2.2 A category with finite products and pullbacks of monics has all finite limits.

Proof. An equalizer can be constructed as a pullback of monics (see Remark 5.1.1), and any finite limit can be constructed from equalizers and finite products. □

Lemma 5.2.3 Every category has $\top$.

Proof. The top element of $\mathsf{Sub}(X)$ is represented by $\mathrm{id}_X : X \rightarrowtail X$. □

Lemma 5.2.4 Having pullbacks of monics implies having $\wedge$.

Proof. For $A, B \in \mathsf{Sub}(X)$ the meet $A \wedge B$ can be constructed by the pullback:

$$\begin{tikzcd}
A \wedge B \arrow[r, tail] \arrow[d, tail] & B \arrow[d, tail] \\
A \arrow[r, tail] & X
\end{tikzcd}$$

By the results in Section 2.4 it follows that $A \wedge B$ is indeed a subobject of X, and that we can also view it as a subobject of A or B if we prefer. To claim that this operation lives up to the name of a meet, we must show that

$$C \leq (A \wedge B) \quad \Longleftrightarrow \quad (C \leq A) \wedge (C \leq B)$$

for all $C \in \mathsf{Sub}(X)$. This is exactly saying that $A \wedge B$ is a greatest lower bound of A and B. It is clear from the above diagram that $(A \wedge B) \leq A$ and $(A \wedge B) \leq B$. Now if $C \leq (A \wedge B)$, then $C \leq A$ and $C \leq B$ follow by transitivity.

For the other direction, if $(C \leq A) \wedge (C \leq B)$ then we have a diagram:

$$\begin{tikzcd}
C \arrow[r, tail] \arrow[d, tail] \arrow[dr, tail] & B \arrow[d, tail] \\
A \arrow[r, tail] & X
\end{tikzcd}$$

Here the monic $C \rightarrowtail A$ is chosen such that the left triangle commutes (that such a monic exists is the definition of $C \leq A$), and similarly $C \rightarrowtail B$ is chosen such that the right triangle commutes. It follows that the square commutes, and we can apply the universal property of $A \wedge B$ as a product: C must factor through $A \wedge B$, which gives $C \leq (A \wedge B)$. □

Lemma 5.2.5 Having $\rightarrow$ is equivalent to having right adjoints $m^* \dashv \forall_m$ for each monic m.

Proof. We show first how to use $\forall_m$ to construct the operation $\rightarrow$. Let $A, B \in \mathsf{Sub}(X)$ be arbitrary, and let m denote the monic $A \rightarrowtail X$. Consider the following pullback diagram:

$$\begin{tikzcd}
A \wedge B \arrow[r, tail] \arrow[d, tail] & B \arrow[d, tail] \\
A \arrow[r, tail, "m"] & X
\end{tikzcd}$$

Since the functor $\forall_m$ goes from $\mathsf{Sub}(A)$ to $\mathsf{Sub}(X)$, we can define

$$A \rightarrow B := \forall_m (A \wedge B)$$

provided that we consider $A \wedge B$ as a subobject of A. To prove the adjunction

$$\begin{array}{c} C \wedge A \leq B \\ \hline\hline C \leq A \to B \end{array}$$

we argue as follows:

$$\dfrac{C \wedge A \leq B \qquad \dfrac{\dfrac{}{C \wedge A \leq C \wedge A}\,\mathsf{refl}}{C \wedge A \leq A}\,\wedge}{\dfrac{C \wedge A \leq A \wedge B}{\dfrac{m^*(C) \leq A \wedge B}{\dfrac{C \leq \forall_m(A \wedge B)}{C \leq A \to B}\ \text{definition of } \to}\ \text{adjunction } m^* \dashv \forall_m}\ \text{definition of } m^*}\,\wedge$$

There is some slight subtlety swept under the rug here, as we are temporarily switching codomain for our subobjects in the middle of the proof:

$$\begin{array}{rl} C \wedge A \leq A \wedge B & \in \mathsf{Sub}(X) \\ m^*(C) \leq A \wedge B & \in \mathsf{Sub}(A) \\ C \leq \forall_m(A \wedge B) & \in \mathsf{Sub}(X) \end{array}$$

As we observed in Section 2.4, the relation $\leq$ is stable under such a change of domain.

To prove the other direction of the adjunction, read the proof tree bottom-up and take the left branch.

Next, if we have $\to$ then we can define $\forall_m$ by

$$\forall_m(S) = X \to S$$

where $S \rightarrowtail X \overset{m}{\rightarrowtail} Y$. The adjunction $m^* \dashv \forall_m$ follows from the adjunction $(- \wedge X) \dashv (X \to -)$ as follows:

$$\begin{array}{c} m^*(A) \leq B \\ \hline\hline A \wedge X \leq B \\ \hline\hline A \leq X \to B \\ \hline\hline A \leq \forall_m(B) \end{array}$$

□

Lemma 5.2.6 Every finitely complete category has left adjoints $\exists_m \dashv m^*$ for each monic m.

Proof. Looking at the diagram

$$\begin{tikzcd} A \arrow[d, tail, "a"'] & B \arrow[d, tail, "b"] \\ X \arrow[r, tail, "m"'] & Y \end{tikzcd}$$

we see that to get a map $\mathsf{Sub}(X) \to \mathsf{Sub}(Y)$ we can simply compose monics: we define $\exists_m(a) = m \circ a$. We want to show that there is an adjunction

$$\begin{array}{c} A \leq m^* B \\ \hline\hline \exists_m A \leq B \end{array}$$

By our construction of $\wedge$ as a pullback, m^*B is simply $X \wedge B$. And $\exists_m(A)$ is just A considered as a subobject of Y. Thus we are trying to prove

$$\begin{array}{c} A \leq X \wedge B \\ \hline\hline A \leq B \end{array}$$

which is clearly true since $A \leq X$ is true.

$\square$

Lemma 5.2.7 If $\mathcal{C}$ has left adjoints $\exists_m \dashv m^*$ for monics m and $\exists_\pi \dashv \pi^*$ for projections π, then $\mathcal{C}$ has left adjoints $\exists_f \dashv f^*$ for all morphisms f.

Proof. An arbitrary morphism f can be factored through its graph as follows:

$$\begin{tikzcd} X \arrow[r, "{(f, \mathrm{id})}"] \arrow[dr, "f"'] & Y \times X \arrow[d, "\pi"] \\ & Y \end{tikzcd}$$

We observe that (f, id) is a monic which we will denote m. By assumption we have adjoints $\exists_\pi \dashv \pi^*$ and $\exists_m \dashv m^*$, and we now claim that $\exists_f := \exists_\pi \circ \exists_m$ is the adjoint we are looking for.

Indeed, since $(\mathsf{Sub}, -^*)$ is a contravariant functor we have

$$f^* = (\pi \circ m)^* = m^* \circ \pi^*,$$

from which the proof of $\exists_f \dashv f^*$ follows:

$$\begin{array}{c} \exists_f(A) \leq B \\ \hline\hline \exists_\pi(\exists_m(A)) \leq B \\ \hline\hline \exists_m(A) \leq \pi^*(B) \\ \hline\hline A \leq m^*(\pi^*(B)) \\ \hline\hline A \leq f^*(B) \end{array}$$

$\square$

Lemma 5.2.8 If $\mathcal{C}$ has right adjoints $m^* \dashv \forall_m$ for monics m and $\pi^* \dashv \forall_\pi$ for projections π, then $\mathcal{C}$ has right adjoints $f^* \dashv \forall_f$ for all morphisms f.

Proof. By exactly the dual argument as for $\exists$:

$$f^* = m^* \circ \pi^* \dashv \forall_\pi \circ \forall_m = \forall_f$$

$\square$

It follows from the preceding lemmas that every regular category has not only $\exists_\pi$ but also $\exists_f$, and that every Heyting category has not only $\forall_\pi$ but also $\forall_f$. Putting together all the results so far we have proven that the following simplified definition is equivalent to Definition 5.2.1:

Definition 5.2.9 A finitely complete category is…

- …regular if and only if it has $\{\exists_f\}$,
- …coherent if and only if it has $\{\bot, \vee, \exists_f\}$,
- …Heyting if and only if it has $\{\bot, \vee, \exists_f, \forall_f\}$.

Remark 5.2.10. We could equivalently replace $\exists_f$ with $\exists_\pi$ throughout: we chose $\exists_f$ here for the aesthetic reason that the requirement "$\exists_f \dashv f^* \dashv \forall_f$ for *all* morphisms f" is slightly neater than having to restrict to a particular class of morphisms.

Remark 5.2.11. We note that our definition of regular category is unusual. The standard definition requires the existence of coequalizers of *kernel pairs* (see [AB24, Section 3.2.1]), and this can then be used to construct the existential quantifer. Conversely, given the existential quantifier it is possible to construct coequalizers of kernel pairs (although we will not do this here), so the two definitions are in fact equivalent.

5.3 Models and soundness

We say that an interpretation M *believes* the sequent $[\Gamma \mid \varphi \vdash \psi]$ if

$$[\![\Gamma \mid \varphi]\!]^M \leq [\![\Gamma \mid \psi]\!]^M$$

in the poset $\mathsf{Sub}([\![\Gamma]\!]^M)$. We notate this as $M \vDash [\Gamma \mid \varphi \vdash \psi]$. We may also say that $[\Gamma \mid \varphi \vdash \psi]$ is "true in M". A *model* of a theory $\mathbb{T}$ is an interpretation which believes all the axioms of $\mathbb{T}$.

Theorem 5.3.1 The sequent calculus is sound: If $\mathbb{T} \vdash \sigma$ then σ is true in all $\mathbb{T}$-models.

Proof. It suffices to show that each of the rules preserve truth. For the three structural rules, this follows from the fact that $\mathsf{Sub}([\![\Gamma]\!])$ is a poset, and that $[\![\overline{s}]\!]^*$ is a functor between posets.

Rule	**Proof of soundness**
$\dfrac{}{\varphi \vdash \varphi}$	$[\![\varphi]\!] \leq [\![\varphi]\!]$ holds since $\leq$ is reflexive on $\mathsf{Sub}([\![\Gamma]\!])$.
$\dfrac{\varphi \vdash \psi \quad \psi \vdash \chi}{\varphi \vdash \chi}$	If $[\![\varphi]\!] \leq [\![\psi]\!]$ and $[\![\psi]\!] \leq [\![\chi]\!]$, then $[\![\varphi]\!] \leq [\![\chi]\!]$ since $\leq$ is transitive on $\mathsf{Sub}([\![\Gamma]\!])$.
$\dfrac{\Gamma \mid \varphi \vdash \psi}{\Delta \mid \varphi(\overline{s}) \vdash \psi(\overline{s})}$	If $[\![\varphi]\!] \leq [\![\psi]\!]$ then $[\![\overline{s}]\!]^*([\![\varphi]\!]) \leq [\![\overline{s}]\!]^*([\![\psi]\!])$ since $[\![\overline{s}]\!]^*$ is a covariant functor from $\mathsf{Sub}([\![\Gamma]\!])$ to $\mathsf{Sub}([\![\Delta]\!])$.

For the remaining rules, soundness follows directly from the corresponding adjunction.

Rule(s)	**Adjunction(s)**
$\begin{array}{c} \top \vdash \varphi(x,x) \\ \hline\hline x = y \vdash \varphi(x,y) \end{array}$	$\exists_\Delta \dashv \Delta^*$
$\begin{array}{c} \\ \hline \bot \vdash \varphi \end{array} \quad \begin{array}{c} \\ \hline \varphi \vdash \top \end{array}$	$\bot \dashv * \dashv \top$
$\begin{array}{c} \varphi \vdash \chi \quad \psi \vdash \chi \\ \hline\hline \varphi \vee \psi \vdash \chi \end{array} \quad \begin{array}{c} \varphi \vdash \psi \quad \varphi \vdash \chi \\ \hline\hline \varphi \vdash \psi \wedge \chi \end{array}$	$\vee \dashv \Delta \dashv \wedge$
$\begin{array}{c} \varphi \wedge \psi \vdash \chi \\ \hline\hline \varphi \vdash \psi \to \chi \end{array}$	$(- \wedge \psi) \dashv (\psi \to -)$
$\begin{array}{c} \varphi(\overline{x},y) \vdash \psi(\overline{x}) \\ \hline\hline \exists y.\varphi(\overline{x},y) \vdash \psi(\overline{x}) \end{array} \quad \begin{array}{c} \varphi(\overline{x}) \vdash \psi(\overline{x},y) \\ \hline\hline \varphi(\overline{x}) \vdash \forall y.\psi(\overline{x},y) \end{array}$	$\exists_\pi \dashv \pi^* \dashv \forall_\pi$

For the connectives and quantifiers, we directly required these adjunctions to exist in Section 5.1. The adjunction for equality is new: Here Δ is the monic $\llbracket\, (x,x) \,\rrbracket : \llbracket\, x \,\rrbracket \rightarrowtail \llbracket\, x,y \,\rrbracket$ and

$$\begin{aligned} \Delta^* : \mathsf{Sub}(\llbracket\, x,y \,\rrbracket) &\to \mathsf{Sub}(\llbracket\, x \,\rrbracket) \\ \llbracket\, \varphi(x,y) \,\rrbracket &\mapsto \llbracket\, \varphi(x,x) \,\rrbracket, \end{aligned}$$

$$\begin{aligned} \exists_\Delta : \mathsf{Sub}(\llbracket\, x \,\rrbracket) &\to \mathsf{Sub}(\llbracket\, x,y \,\rrbracket) \\ \llbracket\, H(x) \,\rrbracket &\mapsto \llbracket\, H(x) \,\rrbracket \wedge \llbracket\, x = y \,\rrbracket \end{aligned}$$

so by applying the adjunction $\exists_\Delta \dashv \Delta^*$ (from Lemma 5.2.6) we get soundness not only of the $=$-rule but in fact of the stronger $=_H$-rule:

$$\begin{array}{c} H(x) \vdash \varphi(x,x) \\ \hline\hline H(x) \wedge x = y \vdash \varphi(x,y) \end{array} =_H$$

□

Soundness of the regular and coherent sequent calculus requires also that we show soundness of the rules $\exists_H$ and $\vee_H$. This follows from the next two lemmas.

Lemma 5.3.2 The *Frobenius rule*

$$\begin{array}{c} \\ \hline H(\overline{x}) \wedge \exists y.\varphi(\overline{x},y) \vdash \exists y.(H(\overline{x}) \wedge \varphi(\overline{x},y)) \end{array}$$

is sound for models in regular categories.

Proof. By applying Beck–Chevalley to the diagram

$$\begin{tikzcd} \llbracket\, \overline{x},y \mid H(\overline{x}) \,\rrbracket \arrow[r, tail, "{(m,\mathrm{id})}"] \arrow[d, "\pi'"'] & \llbracket\, \overline{x},y \,\rrbracket \arrow[d, "\pi"] \\ \llbracket\, \overline{x} \mid H(\overline{x}) \,\rrbracket \arrow[r, tail, "m"'] & \llbracket\, \overline{x} \,\rrbracket \end{tikzcd}$$

we get

$$\begin{tikzcd}
\mathsf{Sub}(\llbracket \overline{x}, y \mid H(\overline{x}) \rrbracket) \arrow[d, "\exists_{\pi'}"'] & \mathsf{Sub}(\llbracket \overline{x}, y \rrbracket) \arrow[l, "(m, \mathrm{id})^*"'] \arrow[d, "\exists_{\pi}"] \\
\mathsf{Sub}(\llbracket \overline{x} \mid H(\overline{x}) \rrbracket) & \mathsf{Sub}(\llbracket \overline{x} \rrbracket) \arrow[l, "m^*"]
\end{tikzcd}$$

and since the action of $(m, \mathrm{id})^*$ and m^* is exactly $\llbracket H(\overline{x}) \rrbracket \wedge -$ this gives

$$\llbracket H(\overline{x}) \wedge \exists y.\varphi(\overline{x}, y) \rrbracket = \llbracket \exists y.(H(\overline{x}) \wedge \varphi(\overline{x}, y)) \rrbracket.$$

□

Lemma 5.3.3 The *distributive rule*

$$\frac{}{H \wedge (\varphi \vee \psi) \vdash (H \wedge \varphi) \vee (H \wedge \psi)}$$

is sound for models in coherent categories.

Proof. Let m be the monic $\llbracket \Gamma \mid H \rrbracket \rightarrowtail \llbracket \Gamma \rrbracket$, and note that the action of m^* is $\llbracket H \rrbracket \wedge -$. Now

$$m^*(\llbracket \varphi \rrbracket \vee \llbracket \psi \rrbracket) = m^*(\llbracket \varphi \rrbracket) \vee m^*(\llbracket \psi \rrbracket)$$

by stability of $\vee$ under pullback, which gives $\llbracket H \wedge (\varphi \vee \psi) \rrbracket = \llbracket (H \wedge \varphi) \vee (H \wedge \psi) \rrbracket$. □

The rules $\exists_H$ and $\vee_H$ can easily be derived from the Frobenius and distributive rule respectively.

6 The classifying category

We will now construct a category $\mathcal{C}_{\mathbb{T}}$ which will fill the same role as $\mathsf{Syn}(\mathbb{T})$ did for algebraic theories: there will be a canonical interpretion of $\mathbb{T}$ inside $\mathcal{C}_{\mathbb{T}}$ which we can prove to be a *universal model.* Then in the next chapter we will show that a model of $\mathbb{T}$ in $\mathcal{C}$ is the same thing as a (structure-preserving) functor from $\mathcal{C}_{\mathbb{T}}$ to $\mathcal{C}$.

We can identity a function between sets $f : X \to Y$ with its graph

$$\{(x, y) \in X \times Y \mid f(x) = y\}.$$

A set S is the graph of a function if and only if

1. $S \subseteq X \times Y$
2. For each $x \in X$ there exists a unique $y \in Y$ such that $(x, y) \in S$

We generalize this idea as follows:

Let $\varphi(\overline{x})$ and $\psi(\overline{y})$ be formulas. A $\mathbb{T}$-*provably functional formula* from $\varphi(\overline{x})$ to $\psi(\overline{y})$ is a formula $\theta(\overline{x}, \overline{y})$ such that the following three sequents are provable:

1. Well-typed: $\theta(\overline{x}, \overline{y}) \vdash \varphi(\overline{x}) \wedge \psi(\overline{y})$. (this corresponds to the condition $S \subseteq X \times Y$ above)
2. Total: $\varphi(\overline{x}) \vdash \exists\overline{y}.\theta(\overline{x}, \overline{y})$ ($\exists\overline{y}$ is shorthand for $\exists y_1 \exists y_2 ... \exists y_n$)
3. Functional: $\theta(\overline{x}, \overline{y}) \wedge \theta(\overline{x}, \overline{z}) \vdash \overline{y} = \overline{z}$.

($\overline{y} = \overline{z}$ is shorthand for $y_1 = z_1 \wedge ... \wedge y_n = z_n$)

The classifying category $\mathcal{C}_{\mathbb{T}}$ is now defined by:

Objects: α-equivalence classes of formulas-in-context.

Morphisms: $\mathbb{T}$-provable equivalence classes of $\mathbb{T}$-provably functional formulas (thus θ and θ' represent the same equivalence class if $\theta \dashv\vdash \theta'$).

Composition is defined as composition of relations:

$$\begin{tikzcd}
{[\,\overline{x} \mid \varphi\,]} \arrow[r, "\theta"] \arrow[rr, dashed, bend right, "\theta \circ \gamma = \exists\overline{y}.(\theta \wedge \gamma)"'] & {[\,\overline{y} \mid \psi\,]} \arrow[r, "\gamma"] & {[\,\overline{z} \mid \chi\,]}
\end{tikzcd}$$

We omit most of the details involved in showing that composition is well-defined and that $\mathcal{C}_{\mathbb{T}}$ is a category, but we state the required properties as lemmas for the record.

Lemma 6.1 The composition operation respects renaming and provable equivalence.

Lemma 6.2 If θ and γ are $\mathbb{T}$-provably functional, so is $\theta \circ \gamma$.

Lemma 6.3 Composition is associative in $\mathcal{C}_{\mathbb{T}}$.

Proof. Let θ, γ and δ be functional formulas as in this diagram:

$$\begin{tikzcd}
\varphi(\overline{x}) \arrow[r, "\theta(\overline{x}, \overline{y})"] & \psi(\overline{y}) \arrow[r, "\gamma(\overline{y}, \overline{z})"] & \chi(\overline{z}) \arrow[r, "\delta(\overline{z}, \overline{w})"] & \xi(\overline{w})
\end{tikzcd}$$

Showing associativity amounts to proving the sequent

$$\exists\overline{y}(\theta \wedge \exists\overline{z}(\gamma \wedge \delta)) \vdash \exists\overline{z}(\exists\overline{y}(\theta \wedge \gamma) \wedge \delta)$$

and its converse. This follows from the $\exists_H$-rule, since $\overline{z}$ does not appear in $\theta(\overline{x}, \overline{y})$ and $\overline{y}$ does not appear in $\delta(\overline{z}, \overline{w})$. ☐

Lemma 6.4 Let $\varphi(\overline{x})$ be any object in $\mathcal{C}_{\mathbb{T}}$. The identity morphism $\mathrm{id}_{\varphi(\overline{x})}$ is given by:

$$\begin{tikzcd}\varphi(\overline{x}) \arrow[r, "\varphi(\overline{x}) \wedge (\overline{x} = \overline{y})"] & \varphi(\overline{y})\end{tikzcd}$$

Note that $\varphi(\overline{x})$ and $\varphi(\overline{y})$ are two different representatives of the same object.

Proof. This amounts to showing that for any morphism $\theta(\overline{y}, \overline{z})$ the composition

$$(\varphi(\overline{x}) \wedge (\overline{x} = \overline{y})) \circ \theta(\overline{y}, \overline{z}) = \exists\overline{y}(\varphi(\overline{x}) \wedge (\overline{x} = \overline{y}) \wedge \theta(\overline{y}, \overline{z}))$$

is provably equivalent to $\theta(\overline{x}, \overline{z})$, and then similarly that for any $\theta(\overline{z}, \overline{x})$ the composition

$$\theta(\overline{z}, \overline{x}) \circ (\overline{x} = \overline{y}) = \exists\overline{x}(\theta(\overline{z}, \overline{x}) \wedge (\overline{x} = \overline{y}))$$

is provably equivalent to $\theta(\overline{z}, \overline{y})$. Again we omit the details. ☐

Remark 6.5. To connect the construction of $\mathcal{C}_{\mathbb{T}}$ back to the syntactic category of an algebraic theory (where the morphisms where given by In the syntactic category of an algebraic theory, objects were contexts and morphsms were substitutions $\overline{s} : \overline{x} \to \overline{y}$. In the category $\mathcal{C}_{\mathbb{T}}$, both objects and morphisms are formulas, and the substitution $\overline{s}$ can instead be represented by:

$$\begin{tikzcd}{[\,\overline{x} \mid \top\,]} \arrow[r, "\overline{s}(\overline{x}) = \overline{y}"] & {[\,\overline{y} \mid \top\,]}\end{tikzcd}$$

Indeed this will be the interpretation of $\overline{s}$ in the universal model that we define in Section 6.1.

Remark 6.6. In the algebraic case, we first constructed the syntactic category of the signature, and then formed the syntactic category of the theory as a quotient category by provable equality. If we take $\mathbb{T}$ to be the empty regular theory, then $\mathcal{C}_{\mathbb{T}}$ can be thought of as $\mathcal{C}_{\Sigma}$, the classifying category of the signature. Note however that morphisms here still need to be taken as $\mathbb{T}$-provable equivalence classes – otherwise there is no identity morphism and composition is not associative. For this reason constructing $\mathcal{C}_{\Sigma}$ is not any conceptually simpler than constructing $\mathcal{C}_{\mathbb{T}}$ directly.

Lemma 6.7 $\mathcal{C}_{\mathbb{T}}$ has finite products.

Proof. $[\,- \mid \top\,]$ is a terminal object.

The product of $\varphi(\overline{x})$ and $\psi(\overline{y})$ is given by $\overline{x}, \overline{y} \mid \varphi(\overline{x}) \wedge \psi(\overline{y})$. ☐

Let $\varphi(\overline{x})$ be an arbitrary object of $\mathcal{C}_{\mathbb{T}}$ and let $\chi(\overline{x})$ be a formula such that $\chi \vdash \varphi$. Then the map

$$\begin{tikzcd}\chi(\overline{x}) \arrow[d, tail, "\chi(\overline{x}) \wedge \overline{x} = \overline{x}'"'] \\ \varphi(\overline{x}')\end{tikzcd}$$

is a monic. This is a particularly simple form of subobject: we may informally think of it as an inclusion, analagous to the set inclusion $\{\overline{x} \mid \chi(\overline{x})\} \hookrightarrow \{\overline{x} \mid \varphi(\overline{x})\}$. When we have a subobject of this form we will notate it by simply $\chi(\overline{x}) \hookrightarrow \varphi(\overline{x})$, leaving the variable renaming implicit. In particular the identity morphism (see Lemma 6.4) is the inclusion $\varphi(\overline{x}) \hookrightarrow \varphi(\overline{x})$.

Lemma 6.8 Any subobject in $\mathcal{C}_{\mathbb{T}}$ can be represented by an inclusion.

Proof. Given an arbitrary monic

$$\begin{tikzcd}
\varphi(\overline{x}) \arrow[r, tail, "\theta(\overline{x},\overline{y})"] & \psi(\overline{y})
\end{tikzcd}$$

we have an inclusion $\exists\overline{x}.\theta(\overline{x},\overline{y}) \hookrightarrow \psi(\overline{y})$. The diagram

$$\begin{tikzcd}
\varphi(\overline{x}) \arrow[r, leftrightarrow, "\theta"] \arrow[dr, "\theta"'] & \exists\overline{x}.\theta(\overline{x},\overline{y}) \arrow[d, hook] \\
 & \psi(\overline{y})
\end{tikzcd}$$

commutes, showing that $\varphi(\overline{x})$ and $\exists\overline{x}.\theta(\overline{x},\overline{y})$ represent the same subobject. $\square$

Two subobject representatives in the inclusion form are equal if and only if the formulas are provably equivalent.

Lemma 6.9 Pullbacks of monics in $\mathcal{C}_{\mathbb{T}}$ are given by

$$\begin{tikzcd}
\exists\overline{y}.(\theta(\overline{x},\overline{y}) \wedge \chi(\overline{y})) \arrow[d, hook] & \chi(\overline{y}) \arrow[d, hook] \\
\varphi(\overline{x}) \arrow[r, "\theta(\overline{x},\overline{y})"'] & \psi(\overline{y})
\end{tikzcd}$$

or in equation form $\theta^*(\chi) = \exists\overline{y}.(\theta \wedge \chi)$.

Remark 6.10. The construction above relies on which representative θ we choose. If we want there to be a canonical choice, we need to add a requirement that our signature is well-ordered, so that the set of formulas can be well-ordered.

We now want to define operations $\top_\varphi, \wedge_\varphi, \bot_\varphi, \vee_\varphi$ and $\rightarrow_\varphi$ on each $\mathsf{Sub}(\varphi(\overline{x}))$. We assume that the inputs are represented as inclusions, and produce the output as an inclusion as well.

The operations $\bot_\varphi, \wedge_\varphi$ and $\vee_\varphi$ can be defined in the most straightforward way:

$$\begin{aligned}
\bot_\varphi &= \bot \\
\chi_1 \wedge_\varphi \chi_2 &= \chi_1 \wedge \chi_2 \\
\chi_1 \vee_\varphi \chi_2 &= \chi_1 \vee \chi_2
\end{aligned}$$

The same does not work for $\top_\varphi$ and $\rightarrow_\varphi$, because $\top$ and $\chi_1 \rightarrow \chi_2$ may not "include" into φ. To fix this we define:

$$\begin{aligned}
\top_\varphi &= \varphi \\
\chi_1 \rightarrow_\varphi \chi_2 &= (\chi_1 \rightarrow \chi_2) \wedge \varphi
\end{aligned}$$

To treat all the connectives on equal footing, we can phrase the general definition as follows: For each connective C, define an operation C_φ on $\mathsf{Sub}(\varphi)$ by $C_\varphi(\overline{\chi}) = C(\overline{\chi}) \wedge \varphi$.

It is easily checked that these operations satisfy the the corresponding adjunctions. For instance for $\to_\varphi$ we have

$$\begin{array}{c} \chi_1 \wedge_\varphi \chi_2 \vdash \chi_3 \\ \hline\hline \chi_1 \wedge \chi_2 \vdash \chi_3 \\ \hline\hline \chi_1 \vdash \chi_2 \to \chi_3 \\ \hline\hline \chi_1 \vdash (\chi_2 \to \chi_3) \wedge \varphi \\ \hline\hline \chi_1 \vdash \chi_2 \to_\varphi \chi_3 \end{array}$$

where the third step uses that $\chi_1 \vdash \varphi$.

Next we check that the operations are stable under pullback: for any $\theta : \varphi \to \psi$ we should have

$$\theta^*\big(C_\psi(\chi_1, \chi_2)\big) = C_\varphi(\theta^*(\chi_1), \theta^*(\chi_2))$$

which works out to

$$\exists\overline{y}.(\theta \wedge C(\chi_1, \chi_2) \wedge \psi) = C(\exists\overline{y}.(\theta \wedge \chi_1), \exists\overline{y}.(\theta \wedge \chi_2)) \wedge \varphi.$$

This is equality as subobjects, so what we need to show is that the two formulas are equivalent. We will not show this in detail, but intuitively the reason this holds is that the functionality of $\theta(\overline{x}, \overline{y})$ implies that any subformula of the form $[\,\overline{x} \mid \exists\overline{y}.(\theta \wedge \ldots)\,]$ must pick out "the same $\overline{y}$".

Next, given an arbitrary product projection

$$\varphi(\overline{x}) \times \psi(\overline{y}) \underset{\pi}{\to} \varphi(\overline{x})$$

we define $\exists_\pi, \forall_\pi : \mathsf{Sub}(\varphi(\overline{x}) \wedge \psi(\overline{y})) \to \mathsf{Sub}(\varphi(\overline{x}))$ as follows:

$$\begin{aligned} \exists_\pi(\chi(\overline{x}, \overline{y})) &:= \exists\overline{y}.(\psi(\overline{y}) \ \wedge \ \chi(\overline{x}, \overline{y})) \\ \forall_\pi(\chi(\overline{x}, \overline{y})) &:= \forall\overline{y}.(\psi(\overline{y}) \to \chi(\overline{x}, \overline{y})) \end{aligned}$$

This is directly analogous to the formulas for $\exists_\pi$ and $\forall_\pi$ in Set that we saw in Example 5.1.4. It is straightforward to check that $\exists_\pi \dashv \pi^* \dashv \forall_\pi$, and that the Beck–Chevalley condition is satisfied.

Remark 6.11. For an arbitrary object $[\,\Gamma \mid \varphi\,]$ we have an inclusion

$$[\,\Gamma \mid \varphi\,] \hookrightarrow [\,\Gamma \mid \top\,].$$

Lemma 6.12 The sequent $[\,\Gamma \mid \varphi \vdash \psi\,]$ is provable if and only if $[\,\Gamma \mid \varphi\,] \leq [\,\Gamma \mid \psi\,]$ in $\mathsf{Sub}([\,\Gamma \mid \top\,])$.

Proof. If $\varphi \vdash \psi$ is provable, then we have an inclusion $\varphi \hookrightarrow \psi$, and since the diagram

$$\begin{tikzcd} \varphi \arrow[r, hook] \arrow[dr, hook] & \psi \arrow[d, hook] \\ & \top \end{tikzcd}$$

commutes it follows that $[\,\varphi\,] \leq [\,\psi\,]$. Conversely, if $[\,\varphi\,] \leq [\,\psi\,]$, then there exists some functional formula $\theta : \varphi \to \psi$ such that

$$\begin{tikzcd}
\varphi \arrow[r, "\theta"] \arrow[dr, hook] & \psi \arrow[d, hook] \\
& \top
\end{tikzcd}$$

commutes. It then follows that θ is monic, and in fact θ must also be an inclusion, i.e. we have

$$\theta(\overline{x}, \overline{x}') \vdash \overline{x} = \overline{x}'.$$

$\square$

6.1 The universal model

Just like we did in the algebraic case, we now define an interpretation $\mathcal{U}$ of $\mathbb{T}$ inside $\mathcal{C}_{\mathbb{T}}$:

1. $\llbracket A \rrbracket^{\mathcal{U}} = [\, x{:}A \mid \top \,]$ (an object of $\mathcal{C}_{\mathbb{T}}$)
2. $\llbracket f \rrbracket^{\mathcal{U}} = [\, \overline{x}, y \mid f(\overline{x}) = y \,]$ (a functional formula from $[\, \overline{x} \mid \top \,]$ to $[\, y \mid \top \,]$)
3. $\llbracket R \rrbracket^{\mathcal{U}} = [\, \overline{x} \mid R(\overline{x}) \,]$ (a subobject of $[\, \overline{x} \mid \top \,]$)

Lemma 6.1.1 For any substitution $\overline{s} : \overline{x} \to \overline{y}$

$$\llbracket \overline{s} \rrbracket^{\mathcal{U}} = [\, \overline{x}, \overline{y} \mid \overline{s}(\overline{x}) = \overline{y} \,]$$

Proof. Analogous to algebraic case. Note that these are morphisms in $\mathcal{C}_{\mathbb{T}}$, so equality means $\mathbb{T}$-provable equivalence. $\square$

Lemma 6.1.2 For any formula $[\, \Gamma \mid \varphi \,]$,

$$\llbracket \Gamma \mid \varphi \rrbracket^{\mathcal{U}} = [\, \Gamma \mid \varphi \,]$$

as subobjects of $\llbracket \Gamma \rrbracket^{\mathcal{U}} = [\, \Gamma \mid \top \,]$.

Proof. Induction over the structure of φ. $\square$

Lemma 6.1.3 The interpretation $\mathcal{U}$ is a $\mathbb{T}$-model.

Proof. Let $\Gamma \mid \varphi \vdash \psi$ be any axiom of $\mathbb{T}$. Then by preceding lemmas

$$\llbracket \Gamma \mid \varphi \rrbracket^{\mathcal{U}} = [\, \Gamma \mid \varphi \,] \leq [\, \Gamma \mid \psi \,] = \llbracket \Gamma \mid \psi \rrbracket^{\mathcal{U}}.$$

$\square$

Theorem 6.1.4 The sequent calculus from Section 4.3 is complete for models in Heyting categories: if $M \vDash [\, \Gamma \mid \varphi \vdash \psi \,]$ holds for all models M in all Heyting categories, then $\mathbb{T} \vdash [\, \Gamma \mid \varphi \vdash \psi \,]$.

Proof. If $M \vDash [\, \Gamma \mid \varphi \vdash \psi \,]$ holds for all models, then in particular $\mathcal{U} \vDash [\, \Gamma \mid \varphi \vdash \psi \,]$, so that $[\Gamma \mid \varphi] \leq [\Gamma \mid \psi]$ in $\mathsf{Sub}([\Gamma \mid \top])$. By Lemma 6.12 this implies that $\mathbb{T} \vdash [\, \Gamma \mid \varphi \vdash \psi \,]$. $\square$

We now observe that if $\mathbb{T}$ is a regular theory, the construction of $\mathcal{C}_{\mathbb{T}}$ could also have been carried out in the regular fragment of logic. We then get a *different* category $\mathcal{C}_{\mathbb{T}}^{\mathsf{Reg}}$: the objects are restricted to regular formulas and we have a different notion of $\mathbb{T}$-provable equality where the rules $\{\bot, \vee, \to, \forall\}$ are no longer allowed in proofs. If we try to use just the subset $\{\mathsf{refl}, \mathsf{cut}, \mathsf{sub}, =, \top, \wedge, \exists\}$ of rules we quickly run into trouble, because we can no longer prove associativity of composition (Lemma 6.3), which relied on the $\exists_H$-rule. However, once we replace

$\exists$ by $\exists_H$ and $=$ by $=_H$, all the remaining lemmas can be proved without issue. This $\mathcal{C}_{\mathbb{T}}^{\mathsf{Reg}}$ will no longer have the structure of a Heyting category, but it will still have the structure $\{\top, \wedge, \exists\}$ that is necessary for a regular category. The construction of the universal model $\mathcal{U}$ then carries over directly and the completeness theorem follows.

Theorem 6.1.5 The (modified) sequent calculus is complete for models in regular categories: if $M \vDash \sigma$ holds for all models in all regular categories then σ can be proved from the axioms of $\mathbb{T}$ using just the rules $\{\mathsf{refl}, \mathsf{cut}, \mathsf{sub}, =_H, \top, \wedge, \exists_H\}$.

Similarly, we can construct a coherent classifying category $\mathcal{C}_{\mathbb{T}}^{\mathsf{Coh}}$ where the objects are coherent formulas and $\mathbb{T}$-provable equality is defined by the rules $\{\mathsf{refl}, \mathsf{cut}, \mathsf{sub}, =_H, \top, \wedge, \bot, \vee_H, \exists_H\}$. This then gives the analogous completeness theorem for coherent models.

Theorem 6.1.6 The sequent calculus with rules $\{\mathsf{refl}, \mathsf{cut}, \mathsf{sub}, =_H, \top, \wedge, \bot, \vee_H, \exists_H\}$ is complete for models in coherent categories.

7 Functorial semantics

7.1 Functors preserving structure

In the algebraic case, the extra structure of our categories was simply “having finite products”, and the appropriate functors between such categories were those that preserve finite products. A Heyting category has not only finite products but all finite limits, and we require those to be preserved: if $(L, f_1, \dots, f_n)$ is a limit of the diagram D, then $(F(L), F(f_1), \dots, F(f_n))$ should be a limit of the diagram $F(D)$.

If F preserves finite limits then it also preserves monics, since m is monic precisely if

$$\begin{tikzcd} X \arrow[r, "\mathrm{id}"] \arrow[d, "\mathrm{id}"'] & X \arrow[d, "m"] \\ X \arrow[r, "m"'] & Y \end{tikzcd}$$

is a pullback. Thus a subobject $S \rightarrowtail X$ induces a subobject $F(S) \rightarrowtail F(X)$.

We say that F preserves $\top, \wedge, \bot, \vee, \to$ if

$$\begin{gathered} F(\top) = \top, \qquad F(S \wedge T) = F(S) \wedge F(T), \\ F(\bot) = \bot, \qquad F(S \vee T) = F(S) \vee F(T), \\ F(S \to T) = F(S) \to F(T). \end{gathered}$$

Note that these are equalities between subobjects of $F(X)$, which is weaker than equality between objects, but stronger than isomorphism between objects. To be more explicit, we could use notation such as

$$F(S \underset{X}{\wedge} T) \underset{F(X)}{=} F(S) \underset{F(X)}{\wedge} F(T)$$

to indicate where each operation is happening and where the equality is taken.

Similarly F preserves $\exists$ and $\forall$ if for any projection $\pi : A \times B \to A$ and subobject $S \rightarrowtail A \times B$ we have

$$F(\exists_\pi S) = \exists_\pi F(S), \qquad F(\forall_\pi S) = \forall_\pi F(S)$$

as subobjects of $F(A)$.

Definition 7.1.1 A regular/coherent/Heyting functor is a functor between regular/coherent/Heyting categories that preserves all the regular/coherent/Heyting structure.

Remark 7.1.2. Note that preserving finite limits and $\{\top, \wedge, \bot, \vee, \to, \exists_\pi, \forall_\pi\}$ is equivalent to preserving finite limits and $\{\bot, \vee, \exists_f, \forall_f\}$ by the results in Section 5.2.

Definition 7.1.3 The category of regular/coherent/Heyting functors from $\mathcal{C}$ to $\mathcal{D}$ (with natural transformations as morphisms) is denoted by $\mathsf{Reg}(\mathcal{C}, \mathcal{D})$, $\mathsf{Coh}(\mathcal{C}, \mathcal{D})$ and $\mathsf{Heyt}(\mathcal{C}, \mathcal{D})$ respectively.

Lemma 7.1.4 If $\mathcal{C}$ and $\mathcal{D}$ are regular/coherent/Heyting categories, M is a model of a regular/ coherent/first-order theory $\mathbb{T}$ inside $\mathcal{C}$, and $F : \mathcal{C} \to \mathcal{D}$ is a regular/coherent/Heyting functor, then $F(M)$ is a model of $\mathbb{T}$ in $\mathcal{D}$.

7.2 Model homomorphisms

A *model homomorphism* $h : M \to N$ consists of, for each sort A, a morphism

$$h_A : [\![A]\!]^M \to [\![A]\!]^N.$$

Just like in the algebraic case (see Definition 3.3.6), these morphisms must commute with function symbols. Additionally, for each relation symbol R there should exist a morphism h_R making this diagram commute:

$$\begin{tikzcd}
{[\![R]\!]^M} \arrow[r, tail] \arrow[d, dashed, "h_R"'] & {[\![A_1]\!]^M \times \dots \times [\![A_n]\!]^M} \arrow[d, "h_{A_1} \times \dots \times h_{A_n}"] \\
{[\![R]\!]^N} \arrow[r, tail] & {[\![A_1]\!]^N \times \dots \times [\![A_n]\!]^N}
\end{tikzcd}$$

If this morphism exists, it is unique, since the lower morphism is a monic.

The $\mathbb{T}$-models and $\mathbb{T}$-model homomorphisms in a fixed category $\mathcal{C}$ form a category $\mathbb{T}_{\mathsf{mod}}(\mathcal{C})$. Composition of homomorphisms is defined componentwise.

We shall also be interested in two subcategories of this category:

- $\mathbb{T}_{\mathsf{mod}}(\mathcal{C})^i$ where the morphisms are restricted to isomorphisms.
- $\mathbb{T}_{\mathsf{mod}}(\mathcal{C})^e$ where the morphisms are restricted to *elementary embedddings*.

A model homomorphism $h : M \to N$ is an elementary embedding if for every $[\, A_1, \dots, A_n \mid \varphi \,]$ there exists a morphism h_φ making the following diagram commute:

$$\begin{tikzcd}
{[\![\varphi]\!]^M} \arrow[r, tail] \arrow[d, dashed, "h_\varphi"'] & {[\![A_1]\!]^M \times \dots \times [\![A_n]\!]^M} \arrow[d, "h_{A_1} \times \dots \times h_{A_n}"] \\
{[\![\varphi]\!]^N} \arrow[r, tail] & {[\![A_1]\!]^N \times \dots \times [\![A_n]\!]^N}
\end{tikzcd}$$

This is exactly the same condition as for relation symbols, so we are simply extending this requirement to non-atomic formulas.

7.3 Models are functors

In Theorem 3.7.1 we summarized the functorial semantics for algebraic theories by the equivalence of categories

$$\mathbb{T}_{\mathsf{mod}}(\mathcal{C}) \simeq \mathsf{FP}(\mathsf{Syn}(\mathbb{T}), \mathcal{C}).$$

We are now ready to state the corresponding results for regular, coherent and first-order theories.

Theorem 7.3.1 For every regular theory $\mathbb{T}$ and regular category $\mathcal{C}$:

$$\mathbb{T}_{\mathsf{mod}}(\mathcal{C}) \simeq \mathsf{Reg}\big(\mathcal{C}_{\mathbb{T}}^{\mathsf{Reg}}, \mathcal{C}\big).$$

Theorem 7.3.2 For every coherent theory $\mathbb{T}$ and coherent category $\mathcal{C}$:

$$\mathbb{T}_{\mathsf{mod}}(\mathcal{C}) \simeq \mathsf{Coh}\big(\mathcal{C}_{\mathbb{T}}^{\mathsf{Coh}}, \mathcal{C}\big).$$

Theorem 7.3.3 For every first-order theory $\mathbb{T}$ and Heyting category $\mathcal{C}$:

$$\mathbb{T}_{\mathsf{mod}}(\mathcal{C})^{e} \simeq \mathsf{Heyt}\big(\mathcal{C}_{\mathbb{T}}^{\mathsf{Heyt}}, \mathcal{C}\big).$$

Note that for the last equivalence we need to restrict to elementary embeddings! We will come back to this point later.

The proof outline is the same for all three equivalences. We will give the main steps of the construction but omit all the details that need to be verified.

The first step is to define the functors

$$\begin{tikzcd}
\mathbb{T}_{\mathsf{mod}}(\mathcal{C}) \arrow[rr, bend left, "(-)^{\sharp}"] & & \mathsf{Reg}\big(\mathcal{C}_{\mathbb{T}}^{\mathsf{Reg}}, \mathcal{C}\big) \arrow[ll, bend left, "\mathsf{ev}_{\mathcal{U}}"]
\end{tikzcd}$$

analogously to how we did in the algebraic case. The functor $\mathsf{ev}_{\mathcal{U}}$ is simply evaluation at $\mathcal{U}$: it maps a regular functor F to the model $F(\mathcal{U})$, and a natural transformation $\eta : F \Rightarrow G$ to the model homomorphism $\mathsf{ev}_{\mathcal{U}}(F) \to \mathsf{ev}_{\mathcal{U}}(G)$ defined by the components

$$\mathsf{ev}_{\mathcal{U}}(\eta)_A = \eta_{[A]} : F([A]) \to G([A]).$$

The fact that these components commute with function and relation symbols follows directly from the naturality of η.

In the other direction, given a model M of $\mathbb{T}$ in $\mathcal{C}$, we define a functor $M^{\sharp} : \mathcal{C}_{\mathbb{T}} \to \mathcal{C}$ by

$$M^{\sharp}([\,\Gamma \mid \varphi\,]) = [\![\,\Gamma \mid \varphi\,]\!]^{M}.$$

To determine how $M^{\sharp}$ should act on morphisms $[\,\overline{x}, \overline{y} \mid \theta\,]$, we will need to make use of all three properties that we have required of θ:

1. Well-typed: $\theta(\overline{x}, \overline{y}) \vdash \varphi(\overline{x}) \land \psi(\overline{y})$.
2. Total: $\varphi(\overline{x}) \vdash \exists \overline{y}.\theta(\overline{x}, \overline{y})$.
3. Functional: $\theta(\overline{x}, \overline{y}) \land \theta(\overline{x}, \overline{z}) \vdash \overline{y} = \overline{z}$.

We observe first that the interpretation of θ is a subobject of $[\![\,\overline{x}, \overline{y}\,]\!]^{M} = [\![\,\overline{x}\,]\!]^{M} \times [\![\,\overline{y}\,]\!]^{M}$. The "well-typed" condition implies that this interpretation factors through $[\![\,\varphi \land \psi\,]\!] = [\![\,\varphi\,]\!] \times [\![\,\psi\,]\!]$:

$$\begin{tikzcd}
{[\![\,\theta\,]\!]} \arrow[r, tail, "m"] & {[\![\,\varphi\,]\!] \times [\![\,\psi\,]\!]} \arrow[dl, "\pi_1"] \arrow[dr, "\pi_2"] & \\
{[\![\,\varphi\,]\!]} & & {[\![\,\psi\,]\!]}
\end{tikzcd}$$

Next the "total" condition implies that $\pi_1 \circ m$ is epic, and the "functional" condition implies that $\pi_1 \circ m$ is monic. Finally it can be shown that in a regular category, a morphism that is both epic and monic is necessarily an isomorphism.

We then define $M^\sharp(\theta : \varphi \to \psi)$ to be the composition $\pi_2 \circ m \circ (\pi_1 \circ m)^{-1} : [\![\varphi]\!] \to [\![\psi]\!]$.

So far we have not made $M^\sharp$ into a functor: we have only defined it on objects. But this is all we need for the equivalence – we can now check that $\mathsf{ev}_{\mathcal{U}}(M^\sharp) \cong M$ for any model M, so that $\mathsf{ev}_{\mathcal{U}}$ is essentially surjective on objects.

For the verification that $\mathsf{ev}_{\mathcal{U}}$ is full and faithful, we refer to [AB24, Theorem 3.2.32]. The key takeaway is that natural transformations map onto exactly those model homomorphism that satisfy the "elementary embedding" condition which we defined in the last section:

$$\begin{tikzcd}
{[\![\varphi]\!]^M} \arrow[r, tail] \arrow[d, dashed, "h_\varphi"'] & {[\![A_1]\!]^M \times ... \times [\![A_n]\!]^M} \arrow[d, "h_{A_1} \times ... \times h_{A_n}"] \\
{[\![\varphi]\!]^N} \arrow[r, tail] & {[\![A_1]\!]^N \times ... \times [\![A_n]\!]^N}
\end{tikzcd}$$

In the regular and coherent case, it can be shown that *any* model homomorphism is an elementary embedding, and so theorems Theorem 7.3.1 and Theorem 7.3.2 follow. However in the Heyting case this is not true, so we must add the restriction to elementary embeddings in Theorem 7.3.3.

A Appendices

A.1 Completeness for set models

The setup of functorial semantics gives a nice way to prove completeness for general models. Some extra work needs to be done to recover the classical completeness theorem for set-based models. We reproduce here the proof given in [AB24, Section 1.1.4] of set-completeness for algebraic models.

Lemma A.1.1 The Yoneda embedding

$$\begin{aligned} \text{よ} : \mathsf{Syn}(\mathbb{T}) &\to \mathsf{Set}^{\mathsf{Syn}(\mathbb{T})^{\mathsf{op}}} \\ x &\mapsto \mathsf{Syn}(\mathbb{T})(-, x) \end{aligned}$$

is a universal model for $\mathbb{T}$. That is:

$$\text{よ} \vDash (t_1 = t_2) \quad \Longleftrightarrow \quad \mathbb{T} \vdash (t_1 = t_2)$$

Proof. The first step is to verify that よ preserves products (in fact it preserves all limits). It then corresponds to a model, which by soundness believes anything provable. Next we use the fact that the Yoneda embedding is faithful: $\text{よ} \vDash (t_1 = t_2)$ means by definition that $\text{よ}(t_1) = \text{よ}(t_2)$ which by faithfulness implies that t_1 and t_2 represent the same morphism in the syntactic category, which by definition means that $\mathbb{T} \vdash (t_1 = t_2)$. □

Lemma A.1.2 Let $\mathcal{C}$ be a small category. For each $x \in \mathsf{ob}(\mathcal{C})$, there is an evaluation functor

$$\begin{aligned} \mathsf{ev}_x : \mathsf{Set}^{\mathcal{C}} &\to \mathsf{Set} \\ F &\mapsto F(x) \\ \eta &\mapsto \eta_x \end{aligned}$$

and the family $\{\mathsf{ev}_x\}_{x \in \mathsf{ob}(\mathcal{C})}$ is jointly faithful: if

$$\mathsf{ev}_x(\eta_1) = \mathsf{ev}_x(\eta_2)$$

holds for all x, then $\eta_1 = \eta_2$.

Proof. Checking functoriality is straightforward. After unpacking what jointly faithful means in this case, we see that it is simply claiming that if two natural transformations have the same components, then they are the same. □

Theorem A.1.3 The equational calculus is complete for algebraic set-models.

Proof. Consider the family $\mathsf{ev}_x \circ \text{よ}$ of set-models, where x ranges over all the objects of $\mathcal{C}_{\mathbb{T}}$. □

Set-completeness for regular and coherent theories can be derived via analagous embedding lemmas – see [Joh02, Section D1.5].

Note that set-completeness for Heyting theories is false: The law of excluded middle is true in any set-model but not provable from our sequent calculus. However if the axiom for LEM is added, then completeness can be derived as in [Joh02, Lemma D1.5.13].

A.2 Syntax of bound variables

We chose to define our syntax in a way where bound variables are required to be distinct from free variables. Thus for instance an expression such as

$$x{:}A \mid x = x \wedge \forall x.(x = x).$$

would not be a valid formula. We also cannot nest quantifiers that reuse the same variable. Thus the following is not a valid formula, in any context:

$$\forall x.(\forall x.(x = x)).$$

However, we can re-use the same bound variable for two different quantifers as long as they are not nested. The following formula is valid in any context not containing the variable x:

$$\forall x.(x = x) \wedge \forall x.(x = x).$$

Claiming that a formula is valid is claiming that there is a construction of it step by step using the clauses above. However, in contrast to provability, there is an efficient algorithm to check whether a given formula can be constructed or not. To illustrate, note that the formula

$$x{:}A \mid x = x \wedge \forall x.(x = x)$$

is of the form $x{:}A \mid \varphi \wedge \psi$, so a construction of it *must* first have constructed $x{:}A \mid \varphi$ and $x{:}A \mid \psi$. Similarly $x{:}A \mid \forall x.(x = x)$ must have first constructed $x{:}A, x{:}A \mid x = x$. But this is impossible, since x is not allowed to appear twice in the context.

A.3 The internal language of a first-order theory

We briefly sketch how to adapt the construction of the internal language (Section 3.8) to the first-order case. Once again, we restrict attention to small categories, but the same construction would work for large categories provided we first modify our definition of signature to allow proper classes.

Given a small Regular/Coherent/Heyting category $\mathcal{C}$, define a signature $\Sigma_{\mathcal{C}}$ as follows:

1. For each $A \in \mathsf{ob}(\mathcal{C})$, include a sort A in the signature.
2. For each morphism $A_1 \times \ldots \times A_n \to B$, include a function symbol $f : A_1, \ldots, A_n \to B$.
3. For each subobject of $A_1 \times \ldots \times A_n$, include a relation symbol $R \rightarrowtail A_1, \ldots, A_n$.

The same remarks apply as in Section 3.8: A single morphism can give rise to many different function, and a single subobject can give rise to many different relation symbols.

We have a canonical interpretation of $\Sigma_{\mathcal{C}}$ inside $\mathcal{C}$, which simply inverts the above procedure. We define a theory $\mathbb{T}_{\mathcal{C}}$ by including $\Gamma \mid \varphi \vdash \psi$ as an axiom whenever $[\![\Gamma \mid \varphi]\!] \leq [\![\Gamma \mid \psi]\!]$ is true in the canonical interpretation.

Now by construction the canonical interpretation is a model of $\mathbb{T}_{\mathcal{C}}$, so it corresponds to a functor $[\![\cdot]\!] : \mathcal{C}_{\mathbb{T}_{\mathcal{C}}} \to \mathcal{C}$, and it can be shown that this functor is an equivalence.

Bibliography

[Joh02] P. Johnstone, *Sketches of an Elephant - A Topos Theory Compendium*, vol. 2. Oxford University Press, 2002.

[AB24] S. Awodey and A. Bauer, "Introduction to Categorical Logic," Sep. 15, 2024. Accessed: Jul. 14, 2025. [Online]. Available: https://awodey.github.io/catlog/notes/catlogdraft.pdf

[Lei14] T. Leinster, *Basic Category Theory.* Cambridge University Press, 2014.