Lecture 9: Natural Transformations and Universal Properties

Lecturer: Max S. New
Scribe: Steven Schaefer

February 8, 2023

1 Introduction to the Functor Category

Inside of Simple Type Theory (STT) connectives give us a way to build up new types from already existing types. Up to this point, we have interpeted the STT connectives $+$ and $\times$. Here, inside of $\mathcal{C}$ we hope to give an interpretation of function types built from $\Rightarrow$.

Given posets P, Q we can construct the set of monotone functions from P to Q,

$$|Q^P| = \{f : P \to Q \mid f \text{ monotone}\}$$

Additonally we can order these by using the order of Q,

$$f \leq g := \forall x \in P. fx \leq_Q gx$$

When moving from posets to categories, we can introduce very similar notions. For categories $\mathcal{C}$ and $\mathcal{D}$, we may introduce a new category $\mathcal{D}^{\mathcal{C}}$ that generalizes the construction Q^P.

The objects of $\mathcal{D}^{\mathcal{C}}$ are functors from $\mathcal{C}$ to $\mathcal{D}$. For functors F and G, we construct a morphism $F \to G$ by providing a morphism $\alpha_a : F_0 a \to G_0 a$ for every object in $a \in \mathcal{C}$. When moving from posets to categories, we need to additionally account for equational conditions on our constructions. In this case we add a condition we call *naturality*: that for any $f : \mathcal{C}_1(a, b). Gf \circ \alpha_a = \alpha_b \circ Ff$.

$$\begin{aligned}
\mathcal{D}^{\mathcal{C}}_0 :=& \{F : \mathcal{C} \to \mathcal{D} \mid F \text{ functor}\} \\
\mathcal{D}^{\mathcal{C}}_1 :=& \prod_{a \in \mathcal{C}_0} (\alpha_a : \mathcal{D}_1(F_0 a, G_0 a)) \\
& \quad \text{such that } \forall f : \mathcal{C}_1(a, b). Gf \circ \alpha_a = \alpha_b \circ Ff
\end{aligned}$$

1.1 An aside on generalization

Note: this does indeed generalize the poset construction. Viewing the poset as a category, $fx \leq gx$ is exactly the data of a morphism $fx \to gx$. When we lifted the order of Q to an ordering on Q^P, $f \leq g$ required $fx \leq_Q gx$ for all x in the poset P — that is, we required there to be a morphism $fx \to gx$ for all $x \in P$. This is exactly what the definiton of $\mathcal{D}_1^{\mathcal{C}}$ requires!

Changing to categorical terminology and swapping $\mathcal{C}$ out for P, $\mathcal{D}$ out for Q, and $\leq_Q$ out for $\mathcal{D}_1$ in the definition of $f \leq g$ yields precisely the definition of the hom-set $\mathcal{D}_1^{\mathcal{C}}$ minus the side conditions.

2 Natural Transformations

When introducing $\mathcal{D}^{\mathcal{C}}$ we had the side condition on morphisms,

$$\forall f : \mathcal{C}_1(a, b). Gf \circ \alpha_a = \alpha_b \circ Ff$$

This condition is called *naturality* and it plays a fundamental role in category theory. Maps α that obey this condition are called *natural transformations*. We may think of natural transformations as transforming a functor $F : \mathcal{C} \to \mathcal{D}$ into a functor $G : \mathcal{C} \to \mathcal{D}$ while respecting the morphism structure of $\mathcal{C}$. It is useful to think of them as a morphism of functors — in a sense that is made rigorous inside of the functor category $\mathcal{D}^{\mathcal{C}}$.

For functors $F, G : \mathcal{C} \to \mathcal{D}$, and a natural transformation $\alpha : \mathcal{D}^{\mathcal{C}}(F, G)$, the naturality condition says that for each $f : \mathcal{C}(a, b)$, the following diagram commutes:

$$a \xrightarrow{f} b$$

$$\begin{array}{ccc} Fa & \xrightarrow{Ff} & Fb \\ \downarrow{\scriptstyle \alpha_a} & & \downarrow{\scriptstyle \alpha_b} \\ Ga & \xrightarrow{Gf} & Gb \end{array}$$

To show that $\mathcal{D}^{\mathcal{C}}$ is actually a category, we need to define appropriate identity and composition of natural transformations. We see that each functor has an identity natural transformation that acts on it, $\mathrm{id}_F : F \to F$. At the object $a \in \mathcal{C}_0$, $\mathrm{id}_F^a : Fa \to Fa$:

$$\mathrm{id}_F^a = \mathrm{id}_{Fa}$$

We need to show that this is natural. Naturality in this case says for each $f : \mathcal{C}(a, b)$ we get a commuting square:

$$\begin{array}{ccc} Fa & \xrightarrow{Ff} & Fb \\ {\scriptstyle \mathrm{id}_{Fa}}\downarrow & & \downarrow{\scriptstyle \mathrm{id}_{Fb}} \\ Fa & \xrightarrow[Ff]{} & Fb \end{array}$$

which commutes because id_{Fa} is the identity morphism.

Likewise, we propose a composition: given natural transformations $\alpha : F \to G$ and $\beta : G \to H$, we can form the composition $\beta \circ \alpha : F \to H$:

$$(\beta \circ \alpha)^a := \beta^a \circ \alpha^a$$

Naturality becomes showing the following square commutes:

$$\begin{array}{ccc} Fa & \xrightarrow{Ff} & Fb \\ {\scriptstyle \alpha_a}\downarrow & & \downarrow{\scriptstyle \alpha_b} \\ Ga & & Gb \\ {\scriptstyle \beta_a}\downarrow & & \downarrow{\scriptstyle \beta_b} \\ Ha & \xrightarrow[Hf]{} & Hb \end{array}$$

Why does this commutes? If we insert the Gf in the middle we get two commuting diagrams:

$$\begin{array}{ccc} Fa & \xrightarrow{Ff} & Fb \\ {\scriptstyle \alpha_a}\downarrow & & \downarrow{\scriptstyle \alpha_b} \\ Ga & \xrightarrow[Gf]{} & Gb \\ {\scriptstyle \beta_a}\downarrow & & \downarrow{\scriptstyle \beta_b} \\ Ha & \xrightarrow[Hf]{} & Hb \end{array}$$

Then this is a commuting diagram as well because each of the sub-squares commutes. We could write this argument out completely algebraically:

$$\begin{aligned} (\beta \circ \alpha)_b \circ Ff &= \beta_b \circ \alpha_b \circ Ff \\ &= \beta_b \circ Gf \circ \alpha_a \\ &= Hf \circ \beta_a \circ \alpha_a \\ &= Hf \circ (\beta \circ \alpha)_a \end{aligned}$$

But you may prefer the diagrammatic style.

Then these definitions satisfy unit and associativity laws because they are given pointwise.

2.1 Application Functor

We now define a functor $\mathbf{app} : \mathcal{D}^{\mathcal{C}} \times \mathcal{C} \to \mathcal{D}$.

$$\begin{aligned} \mathbf{app}_0(F, a) &:= F_0 a \\ \mathbf{app}_1(\alpha : F \to G, f : a \to b) &:= Gf \circ \alpha_a \end{aligned}$$

Above, because α is a morphism in $\mathcal{D}^{\mathcal{C}}$, it must be a natural transformation. Further, because α is natural we made and arbitrary choice when defining $\mathbf{app}_1$. By the naturality of α,

$$Gf \circ \alpha_a = \alpha_b \circ Ff$$

Each of these presentations are justified in believing that they should be the definition of $\mathbf{app}_1$. But alas, we can only choose one, so we might as well take the left-hand side.

Intuitively, we want **app** to behave like a generalized form of function application. Recall that $\Rightarrow E$ takes a function and an element of that function's domain and returns an element of the codomain. Here, **app** is performing the very same task. Given a pair of data from $\mathcal{D}^{\mathcal{C}}$ — which encodes how to act on data from $\mathcal{C}$ — and data from $\mathcal{C}$ — which gives us the arguments to be acted upon — **app** will return a result in $\mathcal{D}$.

It is a good, recommended exercise in unrolling definitions and diagram chasing to prove **app** as defined is functorial. Instead of doing that here, let's show that *if* **app** is to be a functor, then α needs to be a *natural* transformation, which should give some motivation for why the naturality condition is the right one:

$$\begin{aligned}
Gf \circ \alpha_a &= \mathbf{app}_1(\alpha, f) && \text{(defintion of } \mathbf{app}\text{)}\\
&= \mathbf{app}_1(\alpha \circ \mathrm{id}_F, \mathrm{id}_b \circ f) && \text{(identity)}\\
&= \mathbf{app}_1((\alpha, \mathrm{id}_b) \circ (\mathrm{id}_b, f)) && \text{(composition in product category)}\\
&= \mathbf{app}_1(\alpha, \mathrm{id}_b) \circ \mathbf{app}_1(\mathrm{id}_b, f) && \text{(functoriality)}\\
&= G\,\mathrm{id}_b \circ \alpha_b \circ Ff \circ \mathrm{id}_{Fa} && \text{(defintion of } \mathbf{app}\text{)}\\
&= \alpha_b \circ Ff && \text{(identity/functoriality)}
\end{aligned}$$

3 Universal Properties

Many of the connectives in a Heyting algebra — and thus in models of intuitionistic propostional logic — were constructed as the greatest or least element of some set. For instance, $a \wedge b$ is constructed as the element c such that if x is a lower bound for $\{a, b\}$ then $x \leq c$.

Moreover, in any preorder the $a \wedge b$ is unique up to order equivalence. In this instance, the join $a \wedge b$ is the unique element satisfying some property, and we have isolated it entirely in order theoretic terms. As we so often do in this class, we now want to generalize from order theory to any category. That is, we want to be able to uniquely determine objects of a category that obey properties related to our connectives. Uniquely determining objects in this categorical manner is precisely what it means to show that an object obeys a *universal property*.

3.1 Terminal Objects

In a category $\mathcal{C}$, an object $T \in \mathcal{C}_0$ is *terminal* if for any object $V \in \mathcal{C}_0$ there is a unique morphism from V to T.

Note here that we are both making a claim on both the existence of such a morphism and its uniqueness. We write existence as $\exists$, and unique existence as $\exists!$.

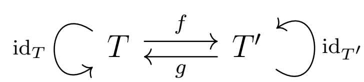
Recall that in a poset, we have an arrow $x \to y$ if and only if $x \leq y$. Within a poset, a terminal element must be $\top$. Every element has a morphism to $\top$ because $x \leq \top$ for all x, so there is at least one morphism into $\top$ from any object. Additionally, posets are thin categories so there is at most one morphism between any two objects. Thus, the size of the hom-set $\mathcal{C}(x, \top)$ is exactly 1 for all elements x.

From a slight weakening to our reasoning on posets, a terminal object in any preorder must be order equivalent to $\top$.

In **Set**, the category of sets, $\{*\}$ is terminal. In particular, the only function into $\{*\}$ is the constant function that outputs $*$ for any input. Moreover, any singleton set is terminal. For instance, $\{5\}$ and {LaTeX} each satisfy the criteria to be terminal objects in **Set**, and the unique maps into each of these are again just the constant maps.

Even though there may be several objects that behave like terminal objects, there is morally only one. What do we mean by this? Terminal objects are unique up to unique isomorphism. That is, up to the categories notion of equivalence — isomorphism — any two terminal objects are the same in exactly one way.

We can show that terminal objects are unique up to unique isomorphism via the following diagram:



Because T is terminal there is a unique morphism $g : T' \to T$. Likewise, because T' is terminal there is a unique morphism $f : T \to T'$. We may compose these to get $f \circ g : T \to T$. However, because T is terminal there is only one morphism $T \to T$. In particular, there is $\mathrm{id}_T : T \to T$. We have shown that there are two inhabitants of $\mathcal{C}(T, T)$ but $|\mathcal{C}(T, T)| = 1$, thus these two inhabitants must coincide. That is,

$$f \circ g = \mathrm{id}_T$$

Through a symmetric argument, $g \circ f = \mathrm{id}_{T'}$. This is precisely the statement that f and g form an isomorphism between T and T'. Finally, we have the even stronger statement that f and g form the *only isomorphism* between T and T' because they form the only morphisms between the terminal objects in this case.

We refer to this uniqueness up to unique isomorphism as being the universal property of terminal objects. A very common proof strategy to show an isomorphism of two objects is to show that they obey the same universal property, or to show two morphisms are equal because they both satisfy a unique existence condition as above.

We often write 1 to denote terminal objects. The unit type, singleton set, and one-element monoid are terminal in the appropriate categories.

3.1.1 Initial Objects

As always in category theory, we get a dual notion for free by consider the opposite category $\mathcal{C}^{op}$. By considering the dual notion of a terminal object, we arrive at the definition for an initial object.

I is an *initial object* if for any object $V \in \mathcal{C}_0$ there is a unique morphism $I \to V$. This is *just* a terminal object in $\mathcal{C}^{op}$. Thus, we can invoke the universal property of terminal objects instantiated with I to prove that I is determined up to unique isomorphism.

We often write 0 to represent initial objects. The empty set, one-element monoid, bottom element of a poset are all initial objects in their respective categories.

The one-element monoid thus has the interesting property of being *both* initial and terminal in the category of monoids: it is terminal because every homomorphism must send everything to the one element of the codomain, and it is initial, because there is a unique homomorphism to every monoid because homomorphisms preserve identity.

If an object is both initial and terminal it is referred to as a *zero object.* A question arose in class asking about the fact that functors do not necessarily preserve both initial objects and terminal objects. For instance, the monoid with one element in **Mon**. However, consider the forgetful functor $U : \mathbf{Mon} \to \mathbf{Set}$. While 0 is a zero object in **Mon**, $U0$ is not the zero object in **Set**. To see this, we can just calculate:

$$U0 = \{e\}$$

$U\ 0$ is a singleton set, and thus terminal in **Set**. Notably $U0 \neq \emptyset$, so it is not initial. Why might this forgetful functor only preserve terminal objects?

Disclaimer: the answer to this question skips ahead a few lectures, and was not mentioned in class; however, I thought it might be valuable to include it for anyone interested.

Answer: The forgetful functor $U : \mathbf{Mon} \to \mathbf{Set}$ is the right adjoint to the free monoid functor $F : \mathbf{Set} \to \mathbf{Mon}$. That is, for a set S and a monoid M, there is a bijection of hom-sets,

$$\mathbf{Set}_1(S, UM) \cong \mathbf{Mon}_1(FS, M)$$

All this says is that a monoid homomorphism out of a free monoid generated by S is exactly the same data as choosing where to send the generators — i.e. choose the image for all elements of S and the monoid axioms completely determine the rest of the homomorphism.

It is a deep theorem that is currently over my head that states that right adjoint functors (in this case U) preserve limits (and a terminal object is an instance of a limit). Dually, we also get the statement that left adjoint functors (in this case F) preseve colimits (in this case initial objects).

We can see this theorem in action by verifying that F sends the initial object of **Set** to the initial object of **Mon**. Recalling that $\emptyset$ is initial in **Set**, we calculate the free monoid generated by the empty set.

$$F\emptyset = \{e\}$$

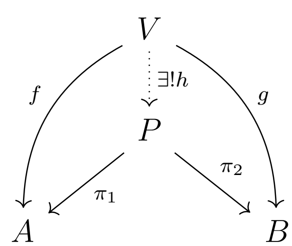
We have no generators to build our monoid with, so the only element of $F\emptyset$ can be the identity element of the monoid, leaving us with the trivial monoid. This is the initial object just as expected!

3.2 Products

A product structure of $A, B \in \mathcal{C}$ comprises the following information:

- An object P
- Morphisms $\pi_1 : P \to A$, $\pi_2 : P \to B$ such that for all objects $V \in \mathcal{C}_0$ with $f : V \to A$, $g : V \to B$ there exists a unique morphism $h : V \to P$ such that $\pi_1 \circ h = f$ and $\pi_2 \circ h = g$.

That is, we have the following diagram:



In **Set** we can check that $A \times B$ with the usual projection functions obeys the universal property laid out above. Here we are taking $A \times B$ to have the same encoding into ZF as laid out in a previous lecture. Namely, $(a, b) \in A \times B$ is given as the set,

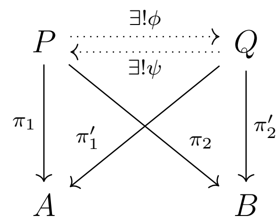
$$\{\{a\}, \{a, b\}\}$$

where the A coordinate is the element of the singleton set and the B coordinate is the element of the two element set that is not also in the singleton set.

Just like with the terminal object example from earlier, we don't have just one object that satisfies the criteria laid out above to be considered product. There are many objects that work. We could have swapped coordinates as $B \times A$ with modified projection functions satisfies the univeral property of products; or perhaps we could have taken another encoding into ZF that still obeys this property. At the end of the day, we don't really care about which particular encoding or product we take. These different choices of encodings are just implementation details, whereas the universal property lays out an interface that all products must obey. Moreover, as we are about to show, all of these products are uniquely isomorphic to each other so it really does not matter which one we consider as *the product* of A and B.

Suppose we have two candidate products P and Q, with projections $\pi_1 : P \to A, \pi_2 : P \to B$ and $\pi_1' : Q \to A, \pi_2' : Q \to B$ such that P and Q the product conditions. That is, for all $V \in \mathcal{C}_0$ with $f : V \to A$ and $g : V \to B$ we have a unique morphism $h_P : V \to P$ such that $\pi_1 \circ h_P = f$ and $\pi_2 \circ h_P = g$. Likewise, for Q we get a unique morphism $h_Q : V \to Q$ such that $\pi_2' \circ h_Q = f$ and $\pi_1' \circ h_Q = g$.

Consider the below diagram. Because Q has projections π_1', π_2', the fact that P is a product gives us a unique morphism $\phi : P \to Q$ that the morphisms from Q to A and B factor through ϕ. In particular, $\pi_1 \circ \phi = \pi_1'$ and $\pi_2 \circ \phi = \pi_2'$.



Through a symmetric argument, becuase Q is a product we have a unique morphism $\psi : Q \to P$ such that morphisms from P into A and B factor through ψ. In particular, $\pi_1' \circ \psi = \pi_1$ and $\pi_2' \circ \psi = \pi_2$.

Consider now the composition $\psi \circ \phi : P \to P$. This is a morphism $P \to P$ such that morphisms $P \to A$, $P \to B$ must factor through this map, as

$$\pi_i \circ \psi \circ \phi = \pi_i' \phi = \pi_i$$

Additionally, $\mathrm{id}_P : P \to P$ is another morphism $P \to P$ such that morphisms $f : P \to A$, $g : P \to B$ must factor through id_P. To see this, note that $f = f \circ \mathrm{id}_P$ and $g = g \circ \mathrm{id}_P$. The universal property of products states that there must be exactly one morphism that preserves the projections π_i in this manner. Thus the two morphisms that we have demonstrated to do this are in fact the same!

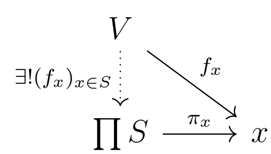
$$\mathrm{id}_P = \psi \circ \phi$$

Through a very similar argument, $\mathrm{id}_Q = \phi \circ \psi$. This gives that P and Q are isomorphic, and we see that this isomorphism is unique.

3.2.1 General Products

Above we demonstrated the universal property of binary products. However, we can generalize this idea to be indexed over any subset of objects. Instead of just $A \times B$, which corresponds to the subset $\{A, B\} \subset \mathcal{C}_0$, we take any $S \subset \mathcal{C}_0$ and build their product $\prod S$.

For all $x \in S$, we require projections $\pi_x : \prod S \to x$ and that for all $V \in \mathcal{C}_0$ such that there are morphisms $f_x : V \to x$, there is a unique morphism $(f_x)_{x \in S}$ such that $\pi_x (f_x)_{x \in S} = f_x$. That is, we want this diagram to commute for all $x \in S$,



We note that terminal objects are nullary products — products over the empty set.