Introduction to Category Theory

Alexander Kurz

January 6, 2021

Contents

0 Introduction

Each of Section 1-4 was a lecture at MGS 2017 in Leicester. The notes are currently being expanded for a course at Chapman University in spring 2019. As we will go on each subsection will correspond to a lecture of approx 75 minutes.

I tried to keep the lectures on a conceptual level. There are almost no proofs. These are delegated to the exercises. If you are new to category theory and you want more than a first overview, do some of the exercises. Best is to have a favourite category at hand to instantiate the definitions and concepts.

There are many exercises and one does not have to do all of them as they usually follow a theme and are sometimes quite similar. The exercises in the text are mostly meant to be easy and should support the understanding of the definitions. Exercises in the separate subsections at the end of each section may summarise results for which we did not have time in the main part or may require special knowledge from some other area of mathematics.

1 Objects and Arrows (The Topos of Sets)

Synopsis. *Theme:* Set theory based on the notion of composition as the primitive notion. *Definitions:* Category, dual category, initial, terminal/final, isomorphism, epi, mono, split epi, split, mono, product, coproduct, equaliser, coequaliser, pullback, pushout, cartesian closure, subobject classifier, powerset. *Concepts:* Set theory without elements, universal constructions, up to canonical isomorphism, instantiating the same definitions in different categories, duality, extending universal constructions from objects to arrows. *Theorems:* Universal constructions are determined up to canonical isomorphism, the category of sets is a topos, ...

Introduction. Category theory is both an area of mathematics and a powerful language in which one can formulate mathematics in general. In this respect, category theory is similar to set theory. On the other hand, doing category theory feels entirely different from set theory. The differences between category and set theory boil down to whether one takes as the primitive symbol the elementship relation $\in$ as in set theory or identities and composition as in category theory. It is worth to emphasise here that set theory is an axiomatic theory based on first-order logic in which the only non-logical symbol is $\in$. In this lecture, we are going to learn how one do set theory without $\in$ using instead composition.

Before starting on this let me list some random comments that the reader is invited to skip.

- Set theory allows us to make precise the idea of equivalence classes and is therefore the basis of many constructions in mathematics and computer science. To give examples of mathematical objects which are equivalence classes, we can start with integers, rationals, and reals. And in programming languages the crucial distinction of syntax and semantics requires can be understood as semantics imposing and equivalence relation on the syntax.
- Category theory is important to deal elegantly with
  - constructions that are only defined up to isomorphism,
  - counting up to isomorphism in finite combinatorics,
  - mathematical structures such as groupoids and presheaves,
  - adjoint situations such as free and cofree constructions,
  - weakening isomorphism to equivalence,
  - an important class of invariants known as natural transformations,
  - models of constructive mathematics and non-classical logics,
  - models of type theory and programming languages,
  - generalised spaces such as toposes,
  - duality theory,
  - solutions to domain equations as they appear in the theory of programming languages,
  - theories of compositionality,
  - transferring properties and constructions across different areas of maths,
  - ...
- Apart from providing a great service to many fields of computer science, physics and mathematics in general, category theory is also an interesting area of maths in its own right as witnessed, for example, by topos theory, enriched category theory, and the theory of higher categories.

1.1 category, mono, epi, duality, initial, final, iso

One way to look at a category is as a graph with identities and composition:

Definition 1. A **category** $\mathcal{A}$ consists of a collection of 'objects' and for all objects $A, B \in \mathcal{A}$ a set $\mathcal{A}(\mathcal{A}, \mathcal{B})$ of 'arrows'. We write $f : A \to B$ for $f \in \mathcal{A}(\mathcal{A}, \mathcal{B})$. For all $A \in \mathcal{A}$ there is an 'identity' arrow $\mathrm{id}_A : A \to A$ and for all A, B, C there is a 'composition' operation $\circ_{ABC} : \mathcal{A}(B, C) \times \mathcal{A}(A, B) \to \mathcal{A}(A, C)$. We write $f \circ g = h$ for $\circ_{ABC}(f, g) = h$. The axioms are for all $h : A \to B$

$$h \circ \mathrm{id}_A = h = \mathrm{id}_B \circ h$$

and for all $f : C \to D, g : B \to C$, $h : A \to B$

$$(f \circ g) \circ h = f \circ (g \circ h)$$

Remark 1. It is possible to define the notion of a category as a “one-sorted” algebra of arrows only, without referring to objects (see Mac Lane, page 9). Even though that seems to be in the spirit of category theory (“everything is determined by the arrows”), it runs against the practice of category theory in which, most of the time, everything is in fact determined by the objects. We will see many instances of this later ...

Remark 2.

- A category with only one object is a monoid. That is, a category can be seen as a many-sorted monoid. Thinking about how important monoids are in computer science and how important the idea is of having many sorts or types, it is not surprising that categories pervade computer science.
- A category with at most one arrow $A \to B$ is a preorder. That is, a category can be seen as a many-arrowed preorder. Thinking about how important ordered structures are in computer science, it is not surprising that categories pervade computer science.
- Given how important monoids and orders are in computer science, it is not surprising that their common generalisation, categories, pervades computer science.

Example 3. Types as objects and functions/programs as arrows form categories. Such categories have many properties similar to the category of sets and functions, see Section**??** for more details.

Example 4. The following are examples in which it makes a lot of sense to have a second compositions, often called parallel composition. See Section **??** for more details.

- Types as objects and processes as arrows form categories.
- Sets of wires as objects and networks as arrows form categories.

Example 5. Mathematical objects of the same kind organise themselves in a category:

1. The category Set has sets as objects and functions as arrows.
2. The category Pos has partially ordered sets as objects and monotone functions as arrows.[1]
3. The category Rel has sets as objects and relations as arrows.
4. The category Monoid of monoids with monoids as objects and monoid morphisms as arrows.

[1]A partial order on a set X is a relation $\leq \subseteq X \times X$ which is reflexive, symmetric and transitive. A function f is monotne if $x \leq x' \;\Rightarrow\; f(x) \leq f(x')$.

5. If you are familiar with topological or metric spaces, then these structures also form categories. For example, we denote by Top the category of topological spaces and continuous maps.

6. ...

Example 6. Some important mathematical objects are themselves categories.

1. Every set is a discrete category.

2. Every monoid is a category with exactly one object.

3. Every group is a groupoid with exactly one object. [2]

4. Every preorder[3] is a category with at most on arrow between any two objects.

5. To see categories as generalisations of lattices is particularly useful and many basic lattic theoretic constructions have interesting categorical generalisations.

6. Generalising to enriched categories, also structures such as metric spaces and many of their generalisations are themselves categories.

7. Every logical theory is a category in which propositions are objects and proofs are arrows.

In the following we are going to see that one can perform set-theoretical definitions and constructions without talking about elements, taking instead identies and composition as primitive notions. In particular, we will see category theoretic definitions for

- injective, surjective, bijective,
- the empty set and the one-element set,
- disjoint union and cartesian product,
- subsets and quotients,
- powerset,
- natural numbers.

This line of development can be seen as a different formalisation of the same ideas that underly Zermelo-Fraenkel set theory. We can see later that also concepts such as the well-founded hierarchy and various anti-foundation axioms have nice category theoretic counterparts.

[2] A groupoid is a category in which all arrows are isomorphisms, see below for a definition of isomorphism.

[3] A preorder $(X, \leq)$ is a set X with a reflexive and transitive relation. A preorder is called a poset if $\leq$ is also antisymmetric.

Remark 7. It is an interesting (if somewhat time consuming) exercise to detail where in the following properties of categories such as Pos, Rel, Top, Monoid deviate from Set. One would see that some categories, such as Pos are quite similar to Set, whereas others, such as Rel are radically different. Algebraic and topological categories (such as Monoid and Top) are somewhere in between, but the details will depend on exactly which category of algebras or topological spaces one would be looking at.

Definition 2. An arrow m in a category is **mono** if it is left-cancellative, that is, if for all f, g we have that $m \circ f = m \circ g$ implies $f = g$.

We also say that m is a monomorphism, or that m is monic.

Exercise 8. An arrow in Set is mono iff it is injective.

What can we say about monos in the other examples of categories?

Duality is an important concept in mathematics. In many circumstances, category theory provides exactly what is needed to formalise it.

Definition 3. The **dual** $\mathcal{C}^{\mathrm{op}}$ of the category $\mathcal{C}$ has the same objects as $\mathcal{C}$ and $\mathcal{C}^{\mathrm{op}}(A, B) = \mathcal{C}(B, A)$. Given $f : A \to B$ in $\mathcal{C}$ we write f^{op} to denote $f : B \to A$ in $\mathcal{C}^{\mathrm{op}}$. Identities and composition in $\mathcal{C}^{\mathrm{op}}$ are defined as in $\mathcal{C}$.

Note that duality reverses composition: if $h = f \circ g$ then $h^{\mathrm{op}} = g^{\mathrm{op}} \circ f^{\mathrm{op}}$ (one could argue that it would be more precise to write $h = g \circ^{\mathrm{op}} f$.

Definition 4. An arrow in $\mathcal{C}$ is **epi** iff it is mono in $\mathcal{C}^{\mathrm{op}}$.

Equivalently, an arrow e is epi if for all f, g we have that $f \circ e = g \circ e$ implies $f = g$.

Exercise 9. An arrow in Set is epi iff it is surjective. This is not true in the category of posets. In a monoid monos and epis are the left or right cancellable elements. In a preorder all arrows are mono and epi.

In Set, a map that is injective and surjective is a bijection. The category theoretic generalisation is the notion of an isomorphism.

Definition 5. An arrow $f : X \to Y$ is an **isomorphism**, or **iso**, if there is $g : Y \to X$ such that $f \circ g = \mathrm{id}_Y$ and $g \circ f = \mathrm{id}_X$.

One should check that if f is an iso then its inverse g is uniquely determined.

Exercise 10.

1. In Set every arrow that is epi and mono is iso.
2. Show that in the category of monoids the inclusion $(\mathbb{N}, 0, +) \to (\mathbb{Z}, 0, +)$ is epi but not surjective. It is also mono and epi but not an isomorphism.

The above example of an epi that is not surjective may be surprising at first sight. Is there a stronger notion than epi that might capture surjectivity? In fact, there is a whole range of stronger notions and in most categories I encounter in my work, one of them does indeed capture surjective. But there is not one notion that does capture surjective in general. This should not come as a surprise since being surjective is a set theoretic property that crucially depends on the idea of an element. (An analogous discussion applies to the notion of injective.)

The strongest strengthening of epi is the notion of a split epi, also known as retraction. We will see later that split epis are exactly those arrows that are surjective on 'generalised elements'.

Definition 6. f is **split epi** if there is a 'half-inverse' g such that $f \circ g = \mathrm{id}$.

The dual notion is called a **split mono** or section. Clearly, any arrow that is split mono and split epi is an iso. But a weaker assumption suffices for this.

Exercise 11.

1. In any category, every split epi is also an epi. (Hence, by duality, split monos are monos.)
2. In any categoy, an arrow that is split epi and mono is iso. (Hence, by duality, a split mono that is epi is an iso.)
3. In the category of monoids the inclusion $(\mathbb{N}, 0, +) \to (\mathbb{Z}, 0, +)$ is (epi but) not split epi. It is also mono, but not split mono.

So far we only defined special properties of arrows. We now show how to capture the empty set and the singleton set in categorical language. This is the first time we encounter a so-called definition by universal property.

Definition 7. An object A in $\mathcal{C}$ is **initial** if for all $C \in \mathcal{C}$ there is exactly one arrow $A \to C$. An object Z in $\mathcal{C}$ is **final** (or **terminal**) if it is initial in $\mathcal{C}^{\mathrm{op}}$.

Equivalently, Z is final if for all $C \in \mathcal{C}$ there is exactly one arrow $C \to Z$.

Exercise 12 (Initial and final object in Set)**.**

1. An object in Set is initial iff it is the empty set. An object in Set is final iff it has exactly one element (is a singleton).
2. For any set A there is a bijection $\mathsf{Set}(1, A) \cong A$.

The last item allows us to capture the elements of a set A categorically as arrows $1 \to A$. This idea can be transferred to arbitrary categories. In fact, if $\mathcal{C}$ is any category with a terminal object 1, then arrows $1 \to A$ are called the **global elements** of A. As we have seen, in Set we have $A \cong \mathsf{Set}(1, A)$, that is, we can identifiy elements with global elements, but this is not true for categories in general as we can see in the following exercise.

Exercise 13 (Initial and final object in various categories)**.** 1. The initial and final object in monoids, or groups, is the one-element structure. Conclude that in these categories every object has precisely one global element.

2. In the category Rel the initial and final object coincides. What are the global elements?

3. It is not difficult, but instructive, to characterise the initial and final objects in a poset. Which objects of a poset have global elements?

4. ...

Remark 14. A remark on the impredicativity of category theoretic definitions:

- It is not entirely obvious that the definitions of epi, mono initial, final are valuable: We quantify over all sets in order to capture simple notions such as the empty set or the singleton. Such a universal quantification over all objects is potentially dangerous (think of Russel's paradox) and has been rejected in many versions of constructive mathematics. This style of definition is called impredicative. Impredicative definitions have a non-constructive flavour. For example, one may find it disturbing that I could possibly change the 'emptiness' of a set by adding or removing other sets to the category. [4]

- On the other hand, what we gain from this style of definition is that the same definitions can be instantiated in all categories. For any category at all, it makes sense to ask what epi, mono, initial, final, etc mean. While we motivated these notions from set-theoretic instances such as surjective, injective, empty set, singleton, these set-theoretic notions do not transfer succesfully from sets to other mathematically structures.

We are going to establish our first important result in category theory. It is easy to prove but contains in a nutshell an argument that we will find repeated many times later in more elaborate forms.

Proposition 15. *Any two final objects are isomorphic.*

Proof. Let A and B be two final objects. Then there are arrows $f : A \to B$ and $g : B \to A$. By uniqueness $g \circ f$ and $f \circ g$ must be both equal to the respective identities. □

Note that the proof uses all the properties of a category. In fact, we can say that a category is the minimal mathematical axiomatisation that allows us to define what an isomorphism is and to show that constructions defined by universal properties are unique up to isomorphism.

[4] Maybe there is more to investigate here? Which categories, and categorical notions, are robust under the addition and the removal of objects from the category?

1.2 (co)product, (co)equaliser

So far we only have described categorically the empty set and the one-element set. To build axiomatise the existence of larger sets we can demand that Set is closed under coproducts:

Definition 8. Let A, B be two objects in a category $\mathcal{A}$. The **coproduct** of A and B is an object $A+B$ in $\mathcal{A}$ together with two arrows $\mathrm{inl} : A \to A+B$ and $\mathrm{inr} : B \to A+B$ such that for all arrows $f : A \to C, g : B \to C$ there is a unique arrow $h : A+B \to C$ such that $h \circ \mathrm{inl} = f$ and $h \circ \mathrm{inr} = g$.

A pair of arrows $f : A \to C, g : B \to C$ is called a *co-cone* over (A, B). The pair $(\mathrm{inl} : A \to A+B, \mathrm{inr} : B \to A+B)$ is said to be *universal* among all cocones over (A, B). The arrow h is often written as $[f; g]$.

Notice the similarity of $A+B$ with the notion of a least upper bound. In fact, if the category $\mathcal{A}$ happens to be a poset, then $A+B$ is the least upper bound of A and B.

Exercise 16.

1. All coproducts of A and B are isomorphic up to 'canonical' isomorphism. [5]
2. The coproduct in Set is (isomorphic to) the disjoint union.
3. Let A^* be the monoid of words over an 'alphabet' A. Show that $A^* + B^* = (A+B)^*$. Conclude that the coproduct in the category of monoids is not disjoint.

Definition 9. $(A \times B \to A, A \times B \to B)$ is a **product** in $\mathcal{A}$ if it is a coproduct in $\mathcal{A}^{\mathrm{op}}$.

Remark 17. When we say that a category **has products**, we usually include infinite products as well as the terminal object (the empty product). This terminology will be made precise when we define the general notion of a limit.

Exercise 18. Fix a category $\mathcal{A}$.

1. Let A, B be two objects in a category $\mathcal{A}$. The product of A and B is an object $A \times B$ in $\mathcal{A}$ together with two arrows $\pi_A : A \times B \to A$ and $\pi_B : A \times B \to B$ such that for all arrows $f : C \to A, g : C \to B$ there is a unique arrow $h : C \to A \times B$ such that $\pi_A \circ h = f$ and $\pi_B \circ h = g$. The arrow h is often written as $\langle f, g \rangle$.[6]
2. Let A, B be two sets in Set. Show that $A \times B$ is isomorphic to the cartesian product $\{(a, b) \mid a \in A, b \in B\}$.
3. $(A \times B) \times C) \cong A \times (B \times C)$.
4. In your favourite axiomatisation of set theory, do you have $(A \times B) \times C) = A \times (B \times C)$?

[5] 'Canonical' here means that even though there may be many isomorphisms, there is only one that commutes with inl and inr.

[6] So while we don't have pairs of elements, we still have pairs of arrows.

5. Given $f : A \to A'$ and $g : B \to B'$, define $f \times g : A \times B \to A' \times B'$.

The last item gives a typical example showing how to extend universal constructions from objects to arrows. This gives a first example showing that often, if you know a construction on objects, then you know how to extend it to arrows.

Remark 19. The projections π_A, π_B are often called first and second projection and an element of a cartesian product is often written (x_1, x_2) and called an *ordered* pair. Note that the order here is just a convenient way of naming the components of a product. Order is not an essential feature of the product. This may become clearer when we see the product as a special kind of limit, a notion that does not depend on indices or names being ordered. In fact, the only structure needed on indices or names is equality.

Next on our list are quotients and subsets.

Definition 10. Let $f, g : A \to B$ be two arrows in a category $\mathcal{A}$. The **equaliser** of f, g is an object E together with an arrow $e : E \to A$ such that $f \circ e = g \circ e$ and for all $e' : E' \to A$ with $f \circ e' = g \circ e'$ there is a unique h such that $e \circ h = e'$. (E, e) is a **coequaliser** of (f, g) if (E, e^{op}) is an equaliser of $(f^{\mathrm{op}}, g^{\mathrm{op}})$ in $\mathcal{A}^{\mathrm{op}}$.

Exercise 20.

1. In Set we can take the equaliser to be $E = \{a \in A \mid fa = ga\}$ and e to be the inclusion $E \to A$.

2. Let $f, g : B \to A$ be two arrows in a category $\mathcal{A}$. The **coequaliser** of f, g is an object E together with an arrow $e : A \to E$ such that $e \circ f = c \circ g$ and for all $e' : A \to E'$ with $e' \circ f = e' \circ g$ there is a unique h such that $h \circ e = e'$.

3. In Set we can take the coequaliser to be the quotient of A by the equivalence relation generated by $\{(fb, gb) \mid b \in B\}$.

4. Equalisers are mono and coequalisers are epi.

5. The inclusion of natural numbers with 0 and + into the integers is an epi. But is it a coequaliser?

Categories that have all (also infinite) products and equalisers are called **complete** and the dual property (all coproducts and coqualisers) is called **cocomplete**.

We have seen previously that there are categories in which neither the notion of an epi (too weak) nor the notion of a split epi (too strong) capture surjections. In categories of algebras coequalisers typically coincide with surjections. This observation is important enough to justify

Definition 11. An arrow that is a coequaliser of some pair is called a **regular epi**. Dually, an arrow that is an equaliser of some pair is called a **regular mono**.

Exercise 21. In the categories of sets, monoids, groups, rings, etc the regular epis coincide precisely with the surjections. Find a surjection in posets that is not regular. Find a regular epi in posets that is not split.

1.3 cartesian closed categories

One way to look at what we have done so far is that we gave definitions for $0, 1, +, \times$ for sets instead of numbers. So it will not come as a suprise that next on the list is exponentiation.[7]

Another way to introduce the next definition is by asking how to capture categorically the fact that for any two sets A, B there is also the set of functions $A \to B$. A good way to proceed is to first recall how the universal property or products axiomatises the behaviour of a set of pairs and then to ask how to do the same for"function space".[8]

Definition 12. Let $\mathcal{A}$ be a category with products. The **exponentiation** of A to the power of B is an object A^B in $\mathcal{A}$ together with an 'evaluation' arrow $\mathrm{eval} : B \times A^B \to A$ such that for all $f : B \times C \to A$ there is a unique $h : C \to A^B$ such that $\mathrm{eval} \circ (\mathrm{id}_B \times h) = f$. A category with products and exponentiation is called **cartesian closed**.

Following the pattern of the previous sections, we first show that Set is cartesian closed. This is not difficult, because there is little data to work with, but may be a bit longer than you would expect.

Exercise 22. Set is cartesian closed.

Next, we show that Pos is cartesian closed. We go through the same argument as above and then discover that the key idea is that eval being monotone forces A^B to be the set of monotone functions $B \to A$.

Exercise 23. Pos is cartesian closed.

Cartesian closure is a very strong property. Most categories that you are likely to encounter as a working mathematician are complete (that is, have all products and equalisers) and cocomplete (that is, have all coproducts and coequalisers), but few will be cartesian closed. For example, the category of groups is not cartesian closed. We will learn later that in a cartesian closed category with an initial 0 it must hold that $A \times 0 \cong 0$, an equation that is familar from numbers. In categorical language we say that in a cartesian closed category the operation $A \times -$ preserves the initial object. We will see later that this is an instance of a general theorem stating that left-adjoint functors preserve colimits. With this in mind it is not difficult to show:

---

[7] It is of interest that $+, \times$, and exponention work "the same" on sets and numbers, but that these operations on Set do not have inverses and do not (in an obvious way) extend, as operations on Set, beyond the natural numbers. See eg Steve Schanuel, Negative sets have Euler characteristic and dimension, Lecture Notes in Mathematics 1488, Springer Verlag, Berlin, 1991, pp. 379-385 or Marcelo Fiore, Tom Leinster, Objects of categories as complex numbers, Adv. Math. 190 (2005), 264-277 (arXiv:0212377) or also the article on the nlab on Euler characteristics.

[8] The common terminology 'function space' comes from cartesian closed categories typically being categories of topological spaces. Top itself is not cartesian closed, but many interesting subcategories are. This includes Set and Pos but also many categories that are models of programming languages. It is not difficult to see why cartesian closure is important for programming languages that have a notion of function type. In fact, so-called typed lambda calculi can be seen as a syntax (internal language) for cartesian closed categories. The area of mathematics that studies such models of programming languages is Domain Theory.

Exercise 24. The category of groups is not cartesian closed.

This finishes the material that we were able to cover in the lecture on cartesian closed categories. I add a few more exercises that are interesting but they will be easier to solve once we learned more about the category theoretic techniques involving natural transformations and adjoint functors.

The following exercise will be the instance of a general theorem that we will learn about in the Chapter on adjunctions. If you prove it in the category of sets as a set-theoretic statement, the statement is almost obvious and you may wonder why it requires proof at all. But in the category Pos of posets it is not so obvious anymore: The set of monotone functions $B \times C \to A$ is in bijection to the set of monotone functions from B to the set of monotone functions $A \to B$. So it makes sense to prove the exercise axiomatically as a theorem of category theory so that it holds for all cartesian closed categories.

Exercise 25. Show that exponentiation induces a bijection between arrows $B \times C \to A$ and $C \to A^B$. This is also known as currying in functional programming.

We have used already above that $A \times -$ preserves initial objects. In the next exercise it will play a role that $A \times -$ also preserves coproducts.

Exercise 26. $A^{B \times C} \cong (A^B)^C$. What about other laws such as $A^{B+C} \cong A^B \times A^C$? For more on this see eg Fiore, Cosmo, and Balat. Remarks on Isomorphisms in Typed Lambda Calculi with Empty and Sum Types http://www.dicosmo.org/Papers/lics02.pdf

1.4 Stocktaking

We have built quite an arsenal of universal constructions: initial, terminal, coproduct, product, exponentiation. Moreover, all these universal constructions follow the same pattern: [...] such that for all arrows [...] there is a unique arrow [...] such that the equation [...] holds. Thus, even if this variety of constructions will seem puzzling at first, the pattern will become familiar after a while. Moreover, the the pattern is not just pleasing, it has some important consequences. We have seen in particular examples, but that is true for all of them, that

- Objects defined by universal constructions are unique up to canonical isomorphism.
- Universal constructions extend from objects to arrows.[9]

Universal constructions are at the heart of axiomatic category theory. For example, instead of giving an explicit construction of the cartesian product as a set of pairs, we can now ask for any category whether it has products. The categorical product will often, but not always, be given by sets of pairs of elements. Therefore, as usual in mathematics, it is advantageous to follow the axiomatic method and to prove properties of eg products using only its universal property.

[9] In the next section we will learn to say: Universal constructions are functorial.

It is possible to prove properties of the notions we learned in an "elementary" fashion using only the axioms of a category extended by first-order logic. This is analogous to doing set theory using only the axioms of set theory. But category theory becomes powerful only after we add functors, natural transformations, the Yoneda lemma and adjoints (or adjunctions). So bear with me, the really cool stuff is yet to come ... having said this, the subobject classifer and the natural numbers object are pretty cool as well ...

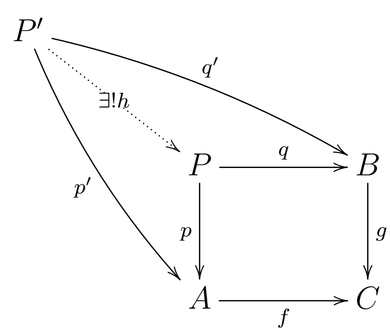
1.5 pullbacks and subobject classifiers

We made some progress towards a categorical axiomatisation of set theory, but we need two more constructions, namely powersets as well as a way to obtain infinite sets. Infinite sets are axiomatised in the next section under the name of natural number object. Powersets are axiomatised in this section with the help of the subobject classifier. But first we need to talk about pullbacks, which play an essential role in category theory in general and in the definition of subobject classifiers in particular.

Like terminal objects, products and equalisers, pullbacks are another important class of limits. For a formal definition of limits we have to wait until later. For now we can think of limits as objects defined by universal properties that specify arrows into the limit, while colimits are objects defined by universal properties that specify arrows out of the limit.

Definition 13. The **pullback** of $A \xrightarrow{f} C \xleftarrow{g} B$ consists of a pair of arrows $A \xleftarrow{p} P \xrightarrow{q} B$ with $f \circ p = g \circ q$ such that for all $A \xleftarrow{p'} P' \xrightarrow{q'} B$ with $f \circ p = g \circ q$ there is a unique arrow $h : P' \to P$ such that $p \circ h = p'$ and $q \circ h = q'$. If $f = g$, then (p, q) are called the **kernel pair** of f.

The definition can be summarised in the following commuting diagram



A *cone* over (f, g) is any pair (p', q') such that $f \circ p' = g \circ q'$. The pullback (p, q) is said to be the *universal* one among all cones over (f, g).

The next exercise shows how pullbacks and the closely related notion of kernel pair work in Set.

Exercise 27. In the category Set we have:

- The pullback of (f, g) is given (up to isomorphism) by $\{(a, b) \mid f(a) = g(b)\}$. In other words, the pullback (is isomorphic to) a subset of the product $A \times B$ satsifying the additional property $f(a) = g(b)$.
- In particular, if $f = g$ then the pullback (p, q) of (f, f) is the so-called **kernel pair** of f. In Set, the kernel pair of any arrow $f : A \to B$ is an equivalence relation on A.
- For a relation $R \subseteq A \times A$ on A with projections π_1 and π_2, the kernel pair of the coequaliser of π_1 and π_2 is the smallest equivalence relation containing R.

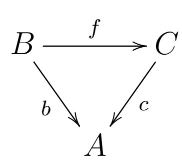
We will need that pullbacks are unique up to iso, a property that we have seen already for other limits and colimits. And, indeed, the proof follows the same pattern.

Proposition 28. *Pullbacks are unique up to isomorphism.*

Proof. Let (p, q) and (p', q') be pullbacks of (f, g). Then there are arrows h, h' such that $p \circ h = p', q \circ h = q'$ and $p' \circ h' = p, q' \circ h' = q$. Moreover, since both the identity and $h \circ h'$ are the unique arrow from the cone (p, q) to the pullback (p, q) we have $h \circ h' = \mathrm{id}$ and, similarly, $h' \circ h = \mathrm{id}$. □

In the diagram above instead of saying (p, q) is the pullback of (f, g), we can also say that p is a pullback of g along f (and that q is the pullback of f along g). Pulling back determines an arrow up to isomorphism. To make this precise we can introduce a category in which arrows are objects:

Definition 14. Let $\mathcal{A}$ be a category and A an object in $\mathcal{A}$. Then the **slice category** $\mathcal{A}/\mathcal{A}$ of $\mathcal{A}$ over A has arrows with codomain A as objects and commuting triangles



as arrows $f : b \to c$. Two arrows with codomain A are *isomorphic* if they are isomorphic as objects in $\mathcal{A}/A$.

This definition makes now precise what we mean when we say that an arrow is determined up to isomorphism by pulling back. If the arrow in question is a mono in Set then its isomorphism class has a canonical representative given by the one and only one inclusion in the isomorphism class. This justifies the way of phrasing the following exercise.

Exercise 29. In Set the pullback of $g : B \to C$ along an inclusion $A \hookrightarrow C$ is the inverse image $g^{-1}(A)$ of A under g.

In the exercise above, we used that pulling back an inclusion gives again an inclusion. Up to isomorphism this is true in general:

Proposition 30. *In any category, the pullback of a mono is a mono.*

Proof. Using the diagram above, we assume that f is a mono and want to show that q is a mono. For this we assume that $q \circ t = q \circ s$ and need to show that $t = s$. This follows from the uniqueness of arrows into the pullback once we shown that $p \circ t = p \circ s$. But this follows from $g \circ q \circ t = g \circ q \circ s$, $g \circ q = f \circ p$ and f mono. □

The subobject classifier axiomatises categorically that there is a bijection between subsets and characteristic functions.

A good way to introduce it is to explain that a characteristic function $\chi : X \to 2 = \{true, false\}$ characterises the subset of X that is the inverse image of $\{true\}$ under χ. By the remark above on pullbacks and inverse images (Exercise 29), one then obtains (for $\Omega = 2$) the special case that the following definition is designed to generalise to other categories than Set. This definition assumes that we work in a category with finite limits, that is, a category that has a terminal object and pullbacks.

Definition 15. An arrow $true : 1 \to \Omega$ from a terminal object to some object Ω is called a **subobject classifier** in the category $\mathcal{A}$, if for all objects A and all monos $m : B \to A$ there is a unique $\chi : A \to \Omega$ such that m is the pullback of $true$ along χ.

$$\begin{array}{ccc} B & \xrightarrow{m} & A \\ \downarrow & & \downarrow{\scriptstyle \chi} \\ 1 & \xrightarrow[true]{} & \Omega \end{array}$$

As usual, we should check what this definition means in Set.

Exercise 31.
1. In Set, we can take $true : 1 \to \Omega$ to be the inclusion $\{1\} \to \{0, 1\}$. The universal property of the subobject classifier then induces a bijection between subsets $B \subseteq A$ and characteristic functions $A \to \{0, 1\}$.
2. In Set, we have that Ω^A is the powerset of A for all sets A.

But what makes this definition really interesting is that it has a surprisingly rich and deep meaning in categories that can be quite different from Set. Here are two examples that we discussed in the lecture.[10]

Exercise 32. Let $\mathcal{A}$ be the caegory which as objects infinite chains

$$X_0 \leftarrow X_1 \leftarrow \ldots \leftarrow X_n \leftarrow \ldots$$

and as arrows infinite chains $f = (f_0, f_1, \ldots, f_n, \ldots)$ of arrows in $\mathcal{A}$ that make the obvious squares commute.

[10]In case you my account below not sufficiently precise, rest assured that we will fill this gap in the next chapter when we learn that they are examples of presheaf categories.

1. Write down a diagram that shows the condition on f to be a morphism in $\mathcal{A}$.

2. Explain how a chain X in $\mathcal{A}$ can be seen as a set of trees.

3. Describe the subobject classifier in $\mathcal{A}$.

Note that the subobject classifier in the category of trees has more than two truth values. In particular, the logic induced by this subobject classifier is not classical logic. Nevertheless one can define conjunction, disjunction, implication, negation and most all of the well-knwon laws of Boolean algebra. In fact, in every topos the subobject classifier carries the structure of an 'internal' Heyting algebra.

Exercise 33. Similarly to the previous exercises, a graph is a pair of sets (E, V), whose elements are called edges and vertices, respectively, together with two arrows $s, t : E \to V$ pronounced "source" and "target".

1. Explain in which sense a structure (E, V, s, t) is a graph.

2. Define the notion of morphism in the category graphs.

3. Describe the subobject classifier in the category of graphs.

Remark 34. If a category has a subobject classifier Ω and is cartesian closed, then we can take Ω^X as an internal generalisation of the powerset of X. It would be worthwhile to explore properties of Ω^X, maybe in the next lecture ...

1.6 toposes and natural number objects

Remark 35. A category with terminal object, products, equalisers, exponentiation and subobject classifier is knwon as an **elementary topos**. Elementary toposes can be seen as generalised universes of sets and as models for higher order intuitionistic logic. In particular, a two-element subobject classifier is rather the exception: the typical subobject classifier is not a Boolean algebra but a Heyting algebra.

Finally, as in Zermelo-Fraenkel set theory, we need one more construction, namely an infinite set.

Definition 16. An object N in a category $\mathcal{A}$, together with arrows $z : 1 \to N$ and $s : N \to N$, is a **natural numbers object** if for all N', $z' : 1 \to N'$ and $s' : N' \to N'$ there is a unique $f : N \to N'$ such that $fz = z'$ and $fs = s'f$.

It is instructive to compare this definition with the construction of the natural numbers in Zermelo-Fraenkel set theory.

1.7 Exercises

Some of the exercises below already appeared in the previous sections. We repeat them here if they fit well into the structure of this section.

There are two kind of exercises. In both cases the question is formulated in the langage of categories, but only in the first case also the answer is.

- Questions that ask whether a category theoretic property holds in *all* categories satisfying certain axioms, the axioms also being formulated in the language of category theory.
- Questions about particular categories which are not themselves defined in the language of category theory. Typically, the techniques required for a solution then are also from outside of category theory.

We illustrate these different kind of questions by grouping them into separate exercises: The purely category theoretic ones, and the ones that require knowledge from set-theory, or group theory, or order theory, etc.

Exercise 36 (Easy exercises in category theory)**.** Prove that the following are true in all categories. Also formulate the dual statements. We use 0 to denote an initial object and 1 to denote a final object.

1. $0 \to A$ is mono and $1 \to A$ is regular mono.
2. Split epi $\Rightarrow$ regular epi $\Rightarrow$ epi.
3. Split epi and mono implies iso.
4. Product and coproduct satisfy some of the familiar laws of addition and multiplication in any category: $1 \times A \cong A \cong A \times 1$ and $A \times B \cong B \times A$ and $(A \times B) \times C \cong A \times (B \times C)$ and, by duality, the corresponding laws for $0, +$.
5. There is a canonical arrow $A \times B + A \times C \to A \times (B + C)$.
6. ...

We will see later the general notion of limits and the theorem that a category with all products[11] and equalisers also has all limits. An important special case of this is the first item in the next exercise. To get an idea why this should be true see the first item of Exercise 27. The second item shows that to obtain all finite limits, pullbacks and a terminal object are enough. The third item is important because it relates monos to limits.[12]

[11] When we say "all products" here, we include the empty product (terminal object) and infinite products. But the definition in the text defined products as binary products only and that is the notion that should be used in the exercise.

[12] For example, it shows that a functor that preserves limits also preserves monos.

Exercise 37 (Exercises in category theory: products, pullbacks, monos)**.**

1. A category with products and equalisers also has pullbacks.

2. A category with a terminal object and pullbacks also has products and equalisers.

3. The pair $(\mathrm{id}, \mathrm{id})$ is a pullback of (m, m) if and only if m is mono.

We continue with properties that hold in Set (and some other categories that are sufficiently similar to Set such as toposes) but not in all categories.

Exercise 38 (Exercise in set theory)**.** In the category Set the following hold.

1. An arrow that is mono and epi is an isomorphism.

2. Every epi is regular, every mono is regular. [13]

3. There is a canonical isomorphism $A \times B + A \times C \cong A \times (B + C)$.

4. The catgory of sets has coproducts and coequalisers as well as products and equalisers.

5. ...

Remark 39 (Exercise in category theory)**.** The properties of Exercise 38 do not only hold for sets but also in any topos. But at this stage we have not learned enough category theory yet, so this is only recommended as an exercise for the commited reader. To look ahead one can find the necessary techniques (and answers) in Barr and Wells, Toposes, Triples and Theories.

We continue with properties of Set, but this time these properties seem quite particular to Set and I would not expect to find them in many other categories.

Exercise 40 (Exercise in set theory)**.** Prove that the following properties hold in Set.

1. Every epi is a split epi (assuming the axiom of choice).

2. Every mono with non-empty domain is split.

3. There is one and only one initial object.

4. ...

Remark 41. Let us remark here already that the properties of Exercise 40 do not hold for toposes in general. Again a commited reader may want to think about counter-examples. Maybe we will have time to come back to this later.

[13] Warning: Since we are working here in the category of sets, which is not self-dual, the statement that every mono is regular does not follow by duality from the statement that every epi is regular epi.

Exercise 42 (Exercise in set theory)**.** Prove that the category Rel of relations has (co)products. Show that Rel does not have equalisers. The first one is not too hard and instructive. The second one is more tricky, see Example 8.16 in Milius, Relations in Categories, for a solution.

Exercise 43 (Easy exercises in order theory)**.** Recall that a poset is a set X with a relation $\leq$ that is reflexive, transitive and antisymmetric.

1. Characterise in the language of ordered sets what an initial object and what a final object is.

2. When does a poset have coproducts and/or products?

3. In a poset, all arrows are mono and epi and only identities are iso.

Exercise 44 (Exercise in group theory)**.** Some of these can be more difficult.

1. In the category of (abelian) groups neither all monos nor all epis split.

2. In the category of groups all epis are surjective (MacLane, Exercise 5 in Section 5).

3. In the category of abelian groups finite products and coproducts coincide (but not infinite ones).

Exercise 45. (Varying difficulty.) Do the properties of Exercises 40 and Exercises 38 hold in the following categories? Posets, Rel, Monoids, Groups, AbelianGroups, Top, ...

The next exercise indicates that the notion of an elementary topos comes quite close to capturing the category of sets. At least, while we have seen that Set is a topos, none of the familiar categories of mathematical *structures* form a topos (here I think of structure as opposed to an unstructured set). We will have to do a bit more work in the next section to exhibit a rich source of interesting examples of toposes.

Exercise 46. Use the properties of Exercise 40 to show that none of the following are toposes: Pos, Rel, Monoid, Group, AbelianGroups, Top, ...

On the other hand, the reader who is burning to see some toposes other than Set can contemplate

Exercise 47. The category Set^2 of pairs of sets is a topos. The category of graphs (where a graph is a category without identities and composition) is a topos. The category of graphs with idenities (also known as reflexive graphs) is a topos. In all cases, the key step is to find the subobject classifier.

In particular, the subobject classifier of the category of graphs is instructive.

1.8 Further Reading

The authorative introduction to category theory is still the one by MacLane. Excellent complements are the three volumes by Borceux and the monograph by Adamek, Herrlich, Strecker. For the modern point of view the nLab is indispensable and should always be checked for reference.

To see how the ideas presented in this section were used to give a foundation of set-theory see Lawvere, An elementary theory of the category of sets, Reprints in Theory and Applications of Categories, No. 12, 2005, pp. 1–35. http://www.tac.mta.ca/tac/reprints/articles/11/tr11.pdf

Or, the same material developed into a beginner's level text book (and Lawvere is always worth reading): Sets for Mathematics by F. William Lawvere, Robert Rosebrugh.

For a general introduction to category theory as well as for details on how the ideas presented in this section lead to topos theory see Barr and Wells, Toposes, Triples and Theories, Reprints in Theory and Applications of Categories, No. 12 (2005) pp. 1-287. http://www.tac.mta.ca/tac/reprints/articles/12/tr12.pdf

A good introduction to topos theory is also MacLane, Moerdijk, Sheaves in Geometry and Logic: A First Introduction to Topos Theory. The comprehensive (and surprisingly readable) standard reference to the subject is Johnstone's Sketches of an Elephant: A Topos Theory Compendium.

2 Functors, Natural Transformations, Yoneda Lemma

Topology is the study of continuous maps and category theory is the theory of functors.

Peter Freyd, Abeliean Categories, 1964, page 1.

Synopsis.
Definitions: Functor, natural transformations, functor categories, presheaves, representable functors.
Concepts: Objects as types, functors as type constructors or parametric types, natural transformations as parametric functions; Yoneda lemma to characterise natural transformations; internal vs external category theory;
Theorems: The category of categories is cartesian closed, Yoneda lemma.

Introduction. In the previous section, we have seen how to use definitions by universal properties in order to build set theory based on the notion of function instead of elementship. [14] In this section, we will enter territory that can only be treated with considerable notational difficulty in the traditional set theoretic way. This happens in situations

[14] Actually, nothing in the axioms of a category captures the idea of a function. Indeed, relations form a category as well. Nevertheless, the definitions in the last section were found by thinking of arrows as functions. So which of the axioms of the last section capture something specific about sets and functions?

where we need to go from a one-sorted setting to a many-sorted setting, or, in other words, to a situation where we need to index our structures and, simultaneously, have a structure on the indices. In particular, we encounter *natural transformations*, the notion for which category theory was invented.

Notice: From now on examples are exercises. I am aware that these notes are a bit rough, for the missing details consult the excellent http://www.staff.science.uu.nl/~ooste110/syllabi/catsmoeder.pdf.

2.1 Functors

One way to motivate the definition of a functor is to say that it is the first step of applying category theory to category theory: If categories are mathematical structures then what are the arrows between categories?

Definition 17. Let $\mathcal{A}, \mathcal{B}$ be two categories. A **functor** $F : \mathcal{A} \to \mathcal{B}$ is a function from the objects of $\mathcal{A}$ to the objects of $\mathcal{B}$ and, for each pair A, A' of objects of $\mathcal{A}$ a function $F_{AA'} : \mathcal{A}(A, A') \to \mathcal{B}(FA, FA')$ that preserves identities and composition.

In the following we give examples that shed light on functors from different perspectives:

- functors as translations between mathematical structures
- functors defined by universal properties
- functors as data type constructors
- mathematical objects that are functors:
  - diagrams
  - presheaves
  - algebras

But before going there it is worth pointing out that the notion of functor as defined above also includes contravariant functors and functors with multiple arguments.

Example 48.
- There is a category Cat of categories and functors.
- There is a functor $(-)^{\mathrm{op}} : \mathsf{Cat} \to \mathsf{Cat}$ that maps a category to its dual category.
- A **contravariant functor** $\mathcal{A} \to \mathcal{B}$ is a functor $\mathcal{A}^{\mathrm{op}} \to \mathcal{B}$. Sometimes, for emphasis, functors are called covariant functors.
- For any two categories $\mathcal{A}, \mathcal{B}$ there is a product category $\mathcal{A} \times \mathcal{B}$ which has pairs (A, B) as objects and pairs (f, g) as morphisms. Identity and composition are defined componentwise and satisfy, in particular, $(f, \mathrm{id}_B) \circ (\mathrm{id}_A, g) = (f, g)$. If we think of arrows as programs, then this law means intuitively that the programs f, g do not have side effects.

- The product of categories is functorial, that is, it gives rise to a functor $\mathsf{Cat} \times \mathsf{Cat} \to \mathsf{Cat}$, mapping $(\mathcal{A}, \mathcal{B})$ to $\mathcal{A} \times \mathcal{B}$.
- We can eliminate the apparent circularity (using the $\times$ on Cat to define $\times$ in Cat) by making a distinction between small and large categories and agree that $\mathcal{A} \times \mathcal{B}$ is a product of small categories whereas $\mathsf{Cat} \times \mathsf{Cat}$ is a product of large categories. On occassion, one may even need an infinite hierarchy of such "universes".

A category $\mathcal{C}$ consists of hom-sets $\mathcal{C}(A, B)$. The additional structure of identities and comosition can be captured by insisting that $\mathcal{C}(A, B)$ is a functor depending on variables A and B. As the these so-called hom-functors capture exactly all the information in a category, it may not come as a surprise that they play a major role in category theory and we will see many instances of their use later on.

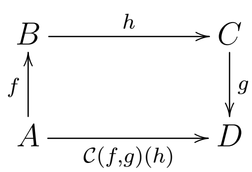
Example 49 (Hom functors and Yoneda embeddings)**.** For any category $\mathcal{C}$ there is the so-called hom-functor

$$\mathrm{Hom}_{\mathcal{C}} : \mathcal{C}^{\mathrm{op}} \times \mathcal{C} \to \mathsf{Set}.$$

We often write $\mathcal{C}(-, -)$ instead of $\mathrm{Hom}_{\mathcal{C}}$ and also drop the subscript if convenient. On arrows, this functor is defined, for $f : A \to B, g : C \to D$ as

$$\begin{aligned} \mathcal{C}(f, g) : \mathcal{C}(B, C) &\to \mathcal{C}(A, D) \\ h &\mapsto g \circ h \circ f \end{aligned}$$

as in the diagram



which also shows why the hom-functor is contravariant at f and covariant at g. Important special cases arise when we keep one coordinate fixed:

$$\mathcal{C}(B, -) : \mathcal{C} \to \mathsf{Set}$$

$$\begin{aligned} \mathcal{C}(B, g) : \mathcal{C}(B, C) &\to \mathcal{C}(B, D) \\ h &\mapsto g \circ h \end{aligned}$$

and

$$\mathcal{C}(-, C) : \mathcal{C}^{\mathrm{op}} \to \mathsf{Set}$$

$$\begin{aligned} \mathcal{C}(f, C) : \mathcal{C}(B, C) &\to \mathcal{C}(A, C) \\ h &\mapsto h \circ f \end{aligned}$$

which we will study in more detail in the next sections under the name of Yoneda embeddings.

Remark 50 (Bimodules)**.** Mathematicians, upon seeing

$$\begin{aligned}\mathcal{C}(f,g) : \mathcal{C}(B,C) &\to \mathcal{C}(A,D)\\ h &\mapsto g\circ h\circ f,\end{aligned}$$

may be reminded of a bimodule, where elements g, f of a ring are acting on the elements h of the bimodule. This is not a coincidence ...

Example 51. Functors are translations between mathematical structures, continues from Example 5.

1. If $\mathcal{A}$ is the category such as posets, monoids, groups, topological spaces, etc, then there is a so-called forgetful functor $U : \mathcal{A} \to \mathsf{Set}$.
2. The other way round, free construction are typically functors $\mathsf{Set} \to \mathcal{A}$. This typically happens if $\mathcal{A}$ is some category of algebras such as monoids, groups, rings etc.
3. ...

It is possible to characterise mathematical structures by properties of their forgetful functors. For example, Lawvere characterised categories of (universal) algebras, also called varieties of algebras, as those categories $\mathcal{A}$ equipped with a functor ...

Whenever you have a mathematical construction that transforms objects of one kind into objects of another kind, it is worth asking whether this construction is functorial. For example, this is always the case if the construction is defined by a universal property.

Example 52. Universal properties give rise to functors. Fix an object A in the category $\mathcal{A}$. Assume that $\mathcal{A}$ has products, exponents, etc as needed.

1. The identity $\mathrm{Id}_{\mathcal{A}} : \mathcal{A} \to \mathcal{A}$ is a functor.
2. $FX = A \times X$ defines a functor $\mathcal{A} \to \mathcal{A}$.
3. $F(X,Y) = X \times Y$ defines a functor $\mathcal{A} \times \mathcal{A} \to \mathcal{A}$. (That requires the definition of $\mathcal{A} \times \mathcal{A}$ first.)
4. $FX = X^A$ defines a functor $\mathcal{A} \to \mathcal{A}$.
5. $FX = A^X$ defines a functor $\mathcal{A}^{\mathrm{op}} \to \mathcal{A}$.
6. ...
7. ... add your own ...
8. ...

Remark 53. 1. Above, it is enough to define functors on objects only.

2.

An example where the definition of the action on maps does not follow (in an immediately obvious way) from a universal property:

Example 54 (The covariant powerset functor)**.** Define $\mathcal{P} : \mathsf{Set} \to \mathsf{Set}$ as $\mathcal{P}X$ the powerset of X and on maps as direct image.

Example 55 (The contravariant powerset functor)**.** Define $2^- : \mathsf{Set}^{\mathrm{op}} \to \mathsf{Set}$ so that 2^X is the powerset of X and inverse image on maps. Alternatively, and in accordance with the notation used for exponentiation, we can think of 2^X as the set of functions $X \to 2$ where 2 denotes any two element set (such as $2 = \{0, 1\}$), that is, $2^X = \mathsf{Set}(X, 2)$, which also defined the functor on maps via $2^f = \mathsf{Set}(f, 2)$, see Example **??**.

Another motivation for functors is the role they play as data type constructors: We have seen already products, coproducts, and powerset. Here is another important example.

Example 56. Define $\mathsf{List} : \mathsf{Set} \to \mathsf{Set}$ as $\mathsf{List}X$ the set of lists over X. On functions, List works 'pointwise'. Many programming languages have an operation that takes a function f to the function $\mathsf{List}(f)$. For example, in Haskell one writes `map f` instead of $\mathsf{List}(f)$.

Another useful way to think about a functor F is as an object FX parameterised by X.

Example 57. Diagrams are functors: products, pullbacks, equalisers ...

Example 58. A functor $\mathcal{C}^{\mathrm{op}} \to \mathsf{Set}$ is called a presheaf on $\mathcal{C}$. Many mathematical structures are presheaves.

1. Graphs are presheaves. The category $\mathcal{C}$ consists of two objects and two parallel arrows (plus identity arrows).

2. The topos of trees has objects $\omega^{\mathrm{op}} \to \mathsf{Set}$ where ω is the poset consisting of the natural numbers with the $\leq$ relation.

3. Simplicial sets are presheaves for $\mathcal{C}$ the category of non-empty, finite ordinals with monotone (ie order-preserving) maps. Note how this categorical definition is much simpler than the one by generators and equations for face and degeneracy maps.

4. Let $\mathbb{B}$ be the category of finite sets with bijections. Presheaves on $\mathbb{B}$ are called combinatorial species and have applications, as the name suggests, to enumerative combinatorics.

5. Let $\mathbb{I}$ be the category of finite sets with injections. Pullback preserving functors $\mathbb{I} \to \mathsf{Set}$ are known as the Schanuel topos and are equivalent to the category of nominal sets. These categories have many important application to the semantics of programming languages.

6. ...

Presheaves are many-sorted algebras with only unary operations. The generalisation to algebras of arbitrary (finitary) operations is known as Lawvere theories.

Example 59. Lawvere theory ...

2.2 Natural transformations

Now that we have seen that functors are legitimate mathematical objects in their own right, we should ask what are the morphisms between functors? We will see later, that natural transformations are precisely the notion of morphism between functors that makes the category of categories cartesian closed.

Definition 18. Let $F, G : \mathcal{A} \to \mathcal{B}$. A **natural transformation** $\tau : F \to G$ is a parameterised collection $\tau_A : FA \to GA$ satisfying $Gf \circ \tau_A = \tau_{A'} \circ Ff$ for all $f : A \to A'$.

How do we know that this definition is the 'right' definition? We will collect some evidence in the form of examples and theorems.

Example 60.

1. There is only one natural transformation $\mathrm{Id}_{\mathsf{Set}} \to \mathrm{Id}_{\mathsf{Set}}$.
2. Let $F, G : \mathsf{Set} \to \mathsf{Set}$ be given by $FX = A \times X$ and $GX = B \times X$. The natural transformations $F \to G$ are in bijection with functions $A \to B$.
3. Abstracting from the order or multiplicity of the elements of a list gives a natural transformation $\mathsf{List} \to \mathcal{P}$.

Taking these examples further, what are the natural transformations

$$\mathsf{List} \to \mathsf{List}$$

A precise answer involves some combinatorial details, but here is it familiar terms: *The natural transformations* $\mathsf{List} \to \mathsf{List}$ *are all functions* $\mathsf{List}(A) \to \mathsf{List}(A)$ *that can be programmed parametrically in* A.

Proposition 61. *Let* $\mathcal{A}, \mathcal{B}$ *be two categories. Then taking functors* $\mathcal{A} \to \mathcal{B}$ *as objects and natural transformations as arrow, we obtain a category of functors* $[\mathcal{A}, \mathcal{B}]$.

Proof. It is easy to show that there is an identity natural transformation and that natural transformations are closed under composition. □

Remark 62. In our definition of a category, the collection of objects formed a 'class', such as the universe of sets itself, and the homsets where sets. Then $[\mathcal{A}, \mathcal{B}]$ need not be a category, because "there are too many" both functors and natural transformations. There are two common solutions. One is to restrict $\mathcal{A}$ to a so-called small category, that is, a category with a set of objects, not a proper class of objects. The other is to stipulate that every collection of objects is a set in some 'universe', see the section on Foundations in MacLane, Categories for the working mathematician.

The details can be safely ignored most of the time, but what is important is the size distinction we have in the definition of categories: Where as the collection of objects is allowed to be 'large', the homsets need to be 'small'. We will be able to explain this later ...

We say that a category is **small** if the collection of objects is a set. One can think of the definition of natural transformation as exactly what is needed to make the following work:

Theorem 63. *The category of small categories is cartesian closed, that is, it has products and exponentiation.*

Proof. The proof follows the same lines as the proof of Proposition **??** that the category Pos is cartesian closed. In fact, if the categories involved happen to be posets functors F, G are just monotone functions and a natural transformation $F \to G$ is just the statement that $FA \leq GA$ for all A. ... details to be filled in ... □

Remark 64. In the last section section, we have seen that set theory can be developed in terms of category theory. But one can push this even further and the idea of the category of categories as a foundation of mathematics has been influential. Certainly, cartesian closure is sth one needs forn endeavour like this.

We started with the idea of sets and functions forming a category, with sets as objects and functions as arrows. Then we added the idea of functors as arrows between categories and saw that categories and functors form again a category. We also said that functors are parameterised objects. What then do we get in the special case where functors are parameterised sets?

These parameterised sets are known as presheaves and play a central role in all of category theory. They also have many applications in computer science.

Definition 19. Let $\mathcal{C}$ be a small category. The category $[\mathcal{C}^{\mathrm{op}}, \mathsf{Set}]$ is called the **category of presheaves** on $\mathcal{C}$.

Example 65 (Presheaves as transition systems)**.** Let ω be the category of ordinals with an arrow $n \to m$ iff $n \leq m$. The a presheaf $P : \omega^{\mathrm{op}} \to \mathsf{Set}$ is the set of trees with $P(n \to n+1)$ mapping nodes at depth $n+1$ to their parent at depth n. How would you describe natural transformations?

Example 66 (Presheaves for variable binding)**.** Let Inj be the category of finite sets with injective maps. Then the terms of the untyped lambda calculus can be seen as a presheaf $\Lambda : \mathsf{Inj} \to \mathsf{Set}$ where $\Lambda(X)$ is the set of lambda-terms with free variable in X. For a variable $x \notin X$, varibale binding $\lambda x.-$ can be described as an operation

$$\lambda x.- : \Lambda(X \cup \{x\}) \longrightarrow \Lambda(X)$$

2.3 Yoneda Lemma

In the previous examples, we have seen that as soon as we work with paremeterised sets (ie presheaves) mundane arrows such as simulations or algebraic operations become natural transformations. So we need a tool to work with natural transformations.

Theorem 67 (Yoneda Lemma)**.** *Let F be a presheaf on $\mathcal{C}$. There is a bijection between natural transformations $\mathcal{C}(-, C) \to F$ and FC.*

Definition 20. . A functor $F : \mathcal{A} \to \mathcal{B}$ is

- full if $F_{AA'} : \mathcal{A}(A, A') \to \mathcal{B}(FA, FA')$ is surjective
- faithful $F_{AA'} : \mathcal{A}(A, A') \to \mathcal{B}(FA, FA')$ is injective
- fully faithful $F_{AA'} : \mathcal{A}(A, A') \to \mathcal{B}(FA, FA')$ is an isomorphism

for all $A, A' \in \mathcal{A}$.

Corollary 68. *The Yoneda embedding $\mathcal{C} \to [\mathcal{C}^{\mathrm{op}}, \mathcal{C}]$ is fully faithful.*

Exercise 69. Use the Yoneda lemma to characterise the natural transformations between Set-functors such as $A \times X \to B \times X$ or $X^A \to X^B$, etc

2.3.1 Categories of presheaves

...

2.3.2 Completions of categories

...

2.3.3 Categorical logic

...

3 Limits and Adjoints

Synopsis. *Theme:* From posets to categoris to 2-categories, or from 2 to Set to Cat. *Definitions:* Limits, colimits, adjoints. *Concepts:* Category theory is generalised lattice theory. *Theorems:* Limits in functor categories are computed pointwise; a category with large limits and colimits is a preorder; ...

3.1 Limits and Colimits

Definition 21. Let $\mathcal{C}$ be a poset and $\mathcal{A} \subseteq \mathcal{C}$. Then $\bigwedge \mathcal{A}$, the **meet** or **greatest lower bound** of $\mathcal{A}$, is definec by the property that for all $c \in \mathcal{C}$ such that $c \leq \mathcal{A}$ we have $c \leq \bigwedge \mathcal{A}$.

To generalise this from posets to categories we need to make explicit the indexing of the elements of $\mathcal{C}$, which will be done by a *functor* $D : \mathcal{A} \to \mathcal{C}$ also called a **diagram**.

As for posets, we will require the indices to form a set, or, rather, $\mathcal{A}$ to be a **small category**, that is, a category that has a set of objects. A short and elegant way to say what a cone $C \in \mathcal{C}$ over D is, is to consider both D and C as functors, and to say that a **cone** over a diagram D is an object $C \in \mathcal{C}$ and a natural transformation $\gamma : D \to C$.

Definition 22. Let $\mathcal{C}$ be a category, let $\mathcal{A}$ be a small categeory and $D : \mathcal{A} \to \mathcal{C}$. Then $\gamma : D \to \lim D$, the **limit** of D, is defined by the universal property that for all $C \in \mathcal{C}$ and all natural transformations $\alpha : C \to D$ there is a unique $h : C \to \lim D$ such that $\gamma \circ h = \alpha$.

Definition 23. Let $\mathcal{C}$ be a poset and $\mathcal{A} \subseteq \mathcal{C}$. Then $\bigvee \mathcal{A}$, the **join** or **least upper bound** of $\mathcal{A}$, is definec by the property that for all $c \in \mathcal{C}$ such that $\mathcal{A} \leq c$ we have $\bigvee \mathcal{A} \leq c$.

Definition 24. Let $\mathcal{C}$ be a category, and $D : \mathcal{A} \to \mathcal{C}$ a diagram. Then $\gamma : \mathrm{colim} D \to D$ is the **colimit** of D if for all $C \in \mathcal{C}$ and all natural transformations $\alpha : D \to C$ there is a unique $h : \mathrm{colim} D \to C$ such that $h \circ \gamma = \alpha$.

If $\mathcal{A}, \mathcal{C}$ are categories and $D : \mathcal{A} \to \mathcal{C}$ is a functor, then the **dual functor** $D^{\mathrm{op}} : \mathcal{A}^{\mathrm{op}} \to \mathcal{C}^{\mathrm{op}}$ acts the same as D on objects and arrows of $\mathcal{A}$. If $C, D : \mathcal{A} \to \mathcal{C}$ are functors and $\tau : C \to D$ is a natural transformation, then the **dual natural transformation** $\tau^{\mathrm{op}} : D^{\mathrm{op}} \to C^{\mathrm{op}}$ is the natural transformation given by $(\tau^{\mathrm{op}})_A = (\tau_A)^{\mathrm{op}}$.

Note that arrows and natural transformation change direction under duality, but functors do not.

Important: $\gamma : D \to C$ is a colimit in $\mathcal{C}$ iff $\gamma^{\mathrm{op}} : C \to D^{\mathrm{op}}$ is a limit in $\mathcal{C}^{\mathrm{op}}$.

Example 70. See Section 1: terminal, product, equaliser are limits, initial, coproduct, coequaliser are colimits.

Whereas the definition of coproduct is abstract, in concrete examples one can give explicit descriptions.

Exercise 71. Let $D : \mathcal{I} \to \mathsf{Set}$ be a diagram. Show that $\mathrm{colim} D$ is the disjoint union of all Di, $i \in \mathcal{I}$, modulo the equivalence relation generated by $(i, x) \equiv (j, Df(x))$ where $i \in \mathcal{I}$, $x \in Di$, $f : i \to j$ is an arrow in $\mathcal{I}$.

A category is **discrete** if the only arrows it has are identities. A category has **products** if it has limits of all diagrams $D : \mathcal{A} \to \mathcal{C}$ where $\mathcal{A}$ is discrete. A category is called **complete** if it has limits (ie limits for all diagrams) and **cocomplete** if it has all colimits.

Theorem 72. *A category is complete if it has products and equalisers.*

Theorem 73. *Limits and colimits in functor categories are computed pointwise.*

Theorem 74. *The Yoneda embedding* $\mathcal{C} \to [\mathcal{C}^{\mathrm{op}}, \mathsf{Set}]$ *preserves limits.*

3.2 Colimits in Preasheaf Categories

3.3 Adjunctions

Definition 25. Let $\mathcal{A}$ and $\mathcal{B}$ be two preorders and $L : \mathcal{A} \to \mathcal{B}$ and $R : \mathcal{B} \to \mathcal{A}$ be two monotone functions. Then we say that L is the **left-adjoint** of R and R is the **right-adjoint** of L and write $L \dashv R$ iff

$$LA \leq B \;\Leftrightarrow\; A \leq RB$$

for all $A \in \mathcal{A}$ and all $B \in \mathcal{B}$.

Example 75. Let $\models$ be a relation on $Models \times Formulas$. Define $M : \mathcal{P}(Formulas) \to \mathcal{P}(Models)$ as

$$M(\mathcal{T}) = \{m \in Models \mid m \models \varphi \; \forall \varphi \in \mathcal{T}\}$$

and $T : \mathcal{P}(Models) \to \mathcal{P}(Formulas)$

$$T(\mathcal{M}) = \{\varphi \in Formulas \mid m \models \varphi \; \forall m \in \mathcal{M}\}$$

Then $MTM = M$ and $TMT = T$.

The importance of the next proposition is that it gives you different ways of showing adjointness.

Proposition 76. *$L : \mathcal{A} \to \mathcal{B}$ and $R : \mathcal{B} \to \mathcal{A}$ be two monotone functions*

1. *$L \dashv R$ iff $A \leq RLA$ and ($A \leq RB \Rightarrow LA \leq B$) for all $A \in \mathcal{A}$,$B \in \mathcal{B}$.*
2. *$L \dashv R$ iff $LRB \leq B$ and ($LA \leq B \Rightarrow A \leq RB$) for all $A \in \mathcal{A}$,$B \in \mathcal{B}$.*
3. *$L \dashv R$ iff $A \leq RLA$ and $LRB \leq B$ for all $A \in \mathcal{A}$,$B \in \mathcal{B}$.*
4. *$L \dashv R$ iff $LRL = L$ and $RLR = R$.*
5. *$L \dashv R$ iff $LA = \bigwedge\{B \mid A \leq RB\}$ for all $A \in \mathcal{A}$.*
6. *$L \dashv R$ iff $RB = \bigvee\{A \mid LA \leq B\}$ for all $B \in \mathcal{B}$.*
7. *$L \dashv R$ only if L preserves all joins and R preserves all meets.*
8. *L has a right-adjoint iff $\mathcal{A}$ has and L preserves all joins of the form $\bigvee\{A \mid LA \leq B\}$.*
9. *R has a left-adjoint iff $\mathcal{B}$ has and R preserves all meets of the form $\bigwedge\{B \mid A \leq RB\}$.*

In particular, if $\mathcal{A}$ and $\mathcal{B}$ are complete, that is, have all joins and meets, then preserving all joins is equivalent to having a right adjoint and preserving all meets is equivalent to having a left-adjoint.

Corollary 77. *If L is a left-adjoint of $R : \mathcal{B} \to \mathcal{A}$, then L is uniquely determined up to iso. If $\mathcal{B}$ is a poset, then L is uniquely determined.*

Definition 26. Let $\mathcal{A}$ and $\mathcal{B}$ be two categories and $L : \mathcal{A} \to \mathcal{B}$ and $R : \mathcal{B} \to \mathcal{A}$ be two functors. Then we say that L is the left-adjoint of R and R is the right-adjoint of L and write $L \dashv R$ if there is an isomorphism

$$\mathcal{B}(LA, B) \;\cong\; \mathcal{A}(A, RB)$$

natural in $A \in \mathcal{A}$ and all $B \in \mathcal{B}$.

Theorem 78. *The data of an adjunction can be given equivalently in any of the following ways.*

1. *For all $A \in \mathcal{A}$ an arrow $\eta_A : A \to RLA$, called the unit, such that for all $f : A \to RB$ there is a unique $f^\sharp : LA \to B$ such that $Rf^\sharp \circ \eta = f$.*
2. *For all $B \in \mathcal{B}$ an arrow $\epsilon_B : LRB \to B$, called the counit, such that for all $g : LA \to B$ there is a unique $g^\flat : A \to RB$ such that $\epsilon_B \circ Lg^\flat = g$.*
3. *Two natural transformations $\eta : \mathrm{Id} \to RL$ (the unit) and $\epsilon : LR \to \mathrm{Id}$ such that $\epsilon L \circ L\eta = \mathrm{Id}$ and $R\epsilon \circ \eta R = \mathrm{Id}$.*

Theorem 79. *Adjoints are determined up to unique isomorphism.*

Theorem 80. *Adjoints compose.*

Theorem 81. *Left adjoints preserve colimits (and, by duality, right adjoints preserve limits).*

Theorem 82. *A category that is both complete and cocomplete is a preorder.*

Theorem 83. *If R preserves limits and is determined by a small subcategory, then R is a right adjoint.*

Theorem 84. *Let $\mathcal{A} \to \mathcal{B}$ be fully faithful and have a left-adjoint. Then $\mathcal{A}$ is complete/cocomplete if $\mathcal{B}$ is.*

Example 85. Let $\mathcal{A}$ be a category of algebras given by a signature Σ and equations E such as monoids, or lattices, or Boolean algebras, etc. For the purpose of this example, consider a signature Σ for which all operations have finite arity (that is, all operations take a finite number of arguments). One can formalise this by saying that Σ is a function $\mathbb{N} \to \mathsf{Set}$ giving for each arity $n \in \mathbb{N}$ the set of opertion symbols of this arity.

Whatever the signature and equations, there always is a functor $U : \mathcal{A} \to \mathsf{Set}$ mapping an algebra to its underlying set. And there always is a left-adjoint $F : \mathsf{Set} \to \mathcal{A}$ mapping a set X to the free algebra FX over X. Explicitely, FX is constructed by taking the set of all terms with variables from X and operation symbols from Σ and quotienting by the equations E.

What does the data of an adjunction mean in this example? The unit

$$\eta_X : X \to UFX$$

says that every variable is a term. The counit

$$\epsilon_A : FUA \to A$$

not only says that every algebra is the quotient of a free algebra (Exercise: show that ϵ_A is onto). It, moreover, describes the operations on A to which each term gives rise: If t is a term in variables X, that is $t \in UFX$, the interpretation t^A of t in A should be a function $UA^X \to UA$. So given a 'valuation of variables' $v : X \to UA$, we can indeed form

$$UFX \xrightarrow{UFv} UFUA \xrightarrow{U\epsilon} UA$$

evaluating a term $t \in UFX$ to an element $a \in A$.

4 Algebra, Coalgebras, Monads, ...

Synopsis. *Theme:* Representation results (all set-functors can be presented by equation on flat terms, all set-monads can be presented by operations and equations) *Definitions:* Algebra, coalgebra, monad, algebras for a monad, Kleisli-category, ... *Concepts:* Free algebras, induction, initial and final semantics, Kleisli categories as categories of relations, ... *Theorems:* representation results plus various theorems on monads

4.1 Algebras for a functor

In universal algebra, to specify a class of algebras one starts with a signature $\Sigma : \mathbb{N} \to \mathsf{Set}$, or, equivalently, with a polynomial functor $F_\Sigma(X) = \coprod_{n\in\mathbb{N}} \Sigma(n) \bullet X^n$, where $\Sigma(n) \bullet X^n$ denotes the $\Sigma(n)$-fold coproduct of the set X^n. To regard a signature Σ as a functor $F_\Sigma : \mathsf{Set} \to \mathsf{Set}$ allows us to say that an algebra is simply an arrow

$$F_\Sigma(X) \to X$$

in the category Set and that an algebra homomorphism $f : X \to X'$ is a commuting square

$$\begin{array}{ccc} F_\Sigma(X) & \longrightarrow & X \\ {\scriptstyle F_\Sigma(f)}\downarrow & & \downarrow{\scriptstyle f} \\ F_\Sigma(X') & \longrightarrow & X' \end{array} \qquad (1)$$

Example 86. Suppose we specify a signature consisting of two binary operations $*$ and $+$ and one nullary operation e. Thus, the corresponding $\Sigma : \mathbb{N} \to \mathsf{Set}$ is defined by putting $\Sigma(2) = \{*, +\}$, $\Sigma(0) = \{e\}$ and $\Sigma(n) = \emptyset$ otherwise. The appropriate polynomial endofunctor $F_\Sigma : \mathsf{Set} \to \mathsf{Set}$ then collapses to

$$F_\Sigma(X) = \{e\} \bullet X^0 + \{*, +\} \bullet X^2$$

since the signature Σ is empty for $n \notin \{0, 2\}$. A typical element of $F_\Sigma(X)$ therefore can be conceived as having one of the following three forms:

$$e, x * y, x + y$$

where we denoted by 0 the unique element of the set X^0 and (x, y) denotes an arbitrary element of X^2.

Thus, elements of $F_\Sigma(X)$ are precisely the *flat terms* in variables X for the signature Σ. A mapping $a : F_\Sigma(X) \to X$ that makes X into an algebra for F_Σ is then simply the *interpretation* of flat terms in X. Thus, the mapping a sends the above three typical elements to their "meanings" in X.

It is now straightforward to verify that the commutative square (1) encodes precisely the fact that the mapping $f : X \to X'$ respects the operations e, $*$ and $+$.

In category theory, the notion of algebra for a signature is generalised to the notion of an algebra for a functor. Looking at (1) above, we see that it makes sense to speak of algebras $FX \to X$ and their homomorphisms whenever we have a functor $\mathcal{C} \to \mathcal{C}$ on an arbitrary category $\mathcal{C}$.

Definition 27. Let $T : \mathcal{C} \to \mathcal{C}$ be a functor. The category $\mathsf{Alg}(T)$ has as objects **T-algebras** $TA \to A$ and arrows $(TA \stackrel{\alpha}{\to} A) \longrightarrow (TA' \stackrel{\alpha'}{\to} A')$ are arrows $f : A \to A'$ in $\mathcal{C}$ such that $f \circ \alpha = \alpha' \circ Tf$.

What is gained by this generalisation?

Answer 1. Maybe not too much, as long as one stays in sets, that is, as long as one takes $\mathcal{C} = \mathsf{Set}$. Let us call a functor $\mathsf{Set} \to \mathsf{Set}$ *finitary* if it is fully determined by its action on finite sets. Without going into the category theoretic definition of finitary, it suffices to say here that an arbitrary functor $F : \mathsf{Set} \to \mathsf{Set}$ is finitary iff there is a signature Σ such that F is a quotient of some F_Σ,

$$F_\Sigma \twoheadrightarrow F\ .$$

It follows that for any finitary $F : \mathsf{Set} \to \mathsf{Set}$, an F-algebra $FX \to X$ is nothing but an algebra

$$F_\Sigma X \twoheadrightarrow FX \longrightarrow X$$

for the signature Σ (and the *equations* defining the quotient $F_\Sigma \twoheadrightarrow F$). To summarize, the study of algebras for (finitary) functors $\mathsf{Set} \to \mathsf{Set}$ does not lead beyond the study of varieties in universal algebra, that is, algebras defined by operations and equations. In fact, algebras for a functor are algebras for operations and equations only involving flat terms. For a detailed account on algebras for a functor see the monograph by Adamek and Trnkova.

Answer 2. Quite a lot is gained when moving to other categories $\mathcal{C}$ than Set. Ever since the work of Scott and others on domain theory and program semantics, type constructors T have been viewed as functors and semantic domains as (particular) algebras $TX \to X$, see e.g. the handbook article by Abramsky and Jung on Domain Theory. Typically, the category $\mathcal{C}$ is a category of partial orders or metric spaces, possibly with some completeness requirements.

Another interesting choice for $\mathcal{C}$ is the category which is *dual* to Set. One then obtains the notion of a *coalgebra* for a functor $T : \mathsf{Set} \to \mathsf{Set}$ as a function

$$X \to TX.$$

As opposed to what we have seen in Answer 1 above, the fact that a finitary T is a quotient $F_\Sigma \twoheadrightarrow T$ of a polynomial functor F_Σ does not allow us to reduce the notion of a T-coalgebra to the notion of a coalgebra $X \to F_\Sigma X$ for a signature Σ. Going beyond polynomial functors will lead to new and interesting examples, as we are going to see next.

On the other hand, we still have an induction principle for all T-algebras: Let $\iota : TI \to I$ be an initial object in $\mathsf{Alg}(T)$ (assuming that such an object exists). Then initiality means that for all $\alpha : TA \to A$ there is a unique $f : I \to A$ such that

$$\begin{array}{ccc} TI & \xrightarrow{\iota} & I \\ {\scriptstyle Tf}\downarrow & & \downarrow{\scriptstyle f} \\ TI & \xrightarrow{\alpha} & A \end{array} \tag{2}$$

This is **induction**: To define an arrow on all ‘terms’ from the recursive data-type I to the data-type A, it is enough to specify one-step (or flat) T-operations α on A.

Exercise 87. Show how the diagram (2) specialises to account for induction on the natural numbers.

4.2 Coalgebras for a functor

Examples of coalgebras below show that coalgebras for polynomial functors F_Σ are of interest, but also that new phenomena such as *bisimulation* come into focus when going beyond polynomial functors.

Example 88. *Coalgebras for polynomial functors describe infinite trees.* For example, an element x in a coalgebra $\xi : X \to X + X$ can be seen as an infinite stream of left/right decisions: in state x, taking a transition by applying ξ yields a successor state $\xi(x)$ in either the left or the right component of $TX = X + X$.

Similarly, a state in a coalgebra $X \to A + B \times X \times X$ represents a possibly infinite tree with leaves labelled by elements of A and non-leaf nodes labelled by elements of B. Here, the polynomial functor is $TX = A + B \times X \times X$.

Example 89. *Coalgebras for the powerset functor are transition systems.* Here the functor T assigns the powerset PX to every set X. Thus a coalgebra $\xi : X \to TX$ can be seen as describing the behaviour of a nondeterministic transition system: the "next state" $\xi(x)$ of a state x is, in fact, the subset $\xi(x) \subseteq X$ of all possible states into which x can evolve.

Example 90. *Coalgebras for the distribution functor are probabilistic transition systems.* Denote by DX the set of all functions $p : X \to [0;1]$ that have a finite support (i.e., such that $p(x) = 0$ for all but finitely many $x \in X$) and that satisfy $\sum_{x \in X} p(x) = 1$. Then a coalgebra $\xi : X \to DX$ describes a transition system with $\xi(x) : X \to [0;1]$ giving the probability $\xi(x)(x')$ that x evolves to x'.

In universal coalgebra, a notion coined by Rutten in the eponymous article, therefore, it is important to develop the theory of T-coalgebras parametric in a functor T, much in the same way as universal algebra is done parametrically in a signature Σ. Some questions that arise in that context are:

- For which functors $T : \mathsf{Set} \to \mathsf{Set}$ is there a final coalgebra?
- Can the behavioural equivalence given by the final coalgebra be characterised in terms of bisimulations?
- In universal algebra every signature Σ gives rise to an equational logic. Can we associate a coalgebraic logic to every functor $T : \mathsf{Set} \to \mathsf{Set}$?
- How much of this can be done axiomatically, replacing Set by general categories $\mathcal{C}$?

Of course, there are many further topics in coalgebra, for example, the use of coalgebra to solve recursive equations or to describe and derive congruence formats of process algebras or to extend and apply coalgebraic logic to description logics and knowledge representation.

4.3 Algebras for a monad

We need notation for 'whiskering' a natural transformation with a functor. Let $\tau : F \to G : \mathcal{A} \to \mathcal{B}$ and let $L : \mathcal{A}' \to \mathcal{A}$ and $R : \mathcal{B} \to \mathcal{B}'$. Then we have natural transformations $(\tau L)_{A'} = \tau_{LA'}$ and $(R\tau)_A = R(\tau_A)$. [15] Of course, we now can also write $R\tau L$. This notation also suggests to write τA instead of τ_A.

Definition 28. A **monad** (M, η, μ) on a category $\mathcal{C}$ is a functor $M : \mathcal{C} \to \mathcal{C}$ and two natural transformations $\eta : \mathrm{Id} \to M$ and $\mu : MM \to M$ such that $\mu \circ M\eta = \mu \circ \eta M = \mathrm{Id}$ and $\mu \circ \mu M = \mu \circ M\mu$.

Definition 29. An **algebra for a monad** (M, η, μ) is an algebra $\alpha : MA \to A$ for the functor M satisfying $\alpha \circ \eta A = \mathrm{id}_A$ and $\alpha \circ \mu A = \alpha \circ M\alpha$. We denote the category of algebras for a monad, also known as the category of **Eilenberg-Moore algebras**, by $\mathsf{MAlg}(M, \eta, \mu)$ or simply by $\mathsf{MAlg}(M)$.

[15] Exercise: Show that τL and $R\tau$ are natural transformations, given that τ is natural.

The next two examples show that monads are ubiquitous.

Example 91. Given a signature Σ and equations E, the category of algebras for $\langle\Sigma, E\rangle$ is a category of algebras for a monad.

Example 92. Given and adjunction $F \dashv U : \mathcal{A} \to \mathcal{C}$, then UF is a monad.

In the following we are going to look at two representation theorem, that tell us that all monads arise in this way.

Theorem 93. *Let $U : \mathsf{MAlg} \to \mathcal{C}$ be the forgetful functor. Then U has a left adjoint and $UF = M$.*

Is this the only adjunction that generates M? Not at all, but it is the terminal one, in a sense. The initial one is the following.

Definition 30. Given a category $\mathcal{C}$, let M be a map (which is not required to be functorial at this stage) from objects of $\mathcal{C}$ to objects of $\mathcal{C}$ (think of $M = \mathcal{P}$ and $\mathcal{C} = \mathsf{Set}$). Let $\eta_X : X \to MX$ be a collection of arrows in $\mathcal{C}$ and let $(-)^\sharp_{Y,Z}$ be a collection of functions $\mathcal{C}(Y, MZ) \to \mathcal{C}(MY, MZ)$. Then $(M, \eta, (-)^\sharp)$ is called a **Kleisli triple** if

$$\eta^\sharp = \mathrm{id}$$
$$f^\sharp \circ \eta = f$$
$$(g^\sharp \circ f)^\sharp = g^\sharp \circ f^\sharp$$

Example 94. The assignment $X \mapsto \mathcal{P}X$, together with the collection $\eta_X(x) = \{x\}$ of singleton maps and with

$$g^\sharp(b) = \bigcup\{g(y) \mid y \in b\}$$

is easily seen to form a Kleisli triple.

In fact, the above axiomatisation allows us to extend the assignment $X \mapsto MX$ uniquely to a functor $M : \mathcal{C} \to \mathcal{C}$ such that the collection η_X becomes a natural transformation from Id to M, and, when one defines $\mu_X : MMX \to MX$ by putting $\mu_X = (\mathrm{id}_{MX})^\sharp$, then μ is a natural transformation from MM to M. The above axioms guarantee that (M, η, μ) is a monad. Conversely, every monad (M, η, μ) yields a Kleisli triple by defining $f^\sharp = \mu_Z \circ Mf : MY \to MZ$ for $f : Y \to MZ$. (All details about this are eg discribed in the monograph by Manes on algebraic theories but also many other places such as in MacLane.)

Kleisli triples gives rise to categories, the Kleisli categories, which tend to resemble categories of relations.

Definition 31. Given a Kleisli triple $(M, \eta, (-)^\sharp)$ on a category $\mathcal{C}$, the **Kleisli category** $Kl(M)$ has the same objects as $\mathcal{C}$ and arrows $X \rightharpoonup Y$ in $Kl(M)$ are arrows $X \to MY$ in $\mathcal{C}$. The identity on X in $Kl(M)$ is given by η_X and the composition $g \cdot f$ in $Kl(M)$ is given by the composition $g^\sharp \circ f$ in $\mathcal{C}$.

As to be expected for a 'category of relations' there is an identity-on-objects functor

$$(-)_\star : \mathcal{C} \to Kl(M)$$

taking an arrow $f : X \to Y$ to a 'map' $\eta_Y \circ f : X \rightharpoonup Y$. On the other hand, there need to be no analogue of the converse of a relation nor of the order between relations.

Theorem 95. *Let $F \dashv U : \mathcal{A} \to \mathcal{C}$. Then there are 'comparison functors' $Kl(UF) \xrightarrow{K} \mathcal{A} \xrightarrow{L} \mathsf{MAlg}(UF)$ commuting with the respective forgetful functors.*

Theorem 96. *For any monad M on set, one can find a class of operations Σ and a class of equations E, such that $\mathsf{MAlg}(M) \cong \mathsf{Alg}(\Sigma, E)$.*

5 2-categories

Synopsis. *Theme:* Arrows between arrows: 2-cells

Introduction. We advertised in the first chapter the idea of doing set-theory without elements. Adding a level of arrows allowes us to see category theory as "abstract set theory", or "formal set theory" where abstract or formal refers to sets just being objects without actual elements. This terminology is analogue to the classical distinction of a concrete group as a group of permutations and an abstract group.

Further Reading.

Ross Street. Categorical structures, Handbook of Algebra Volume 1 (editor M. Hazewinkel; Elsevier Science, Amsterdam 1996; ISBN 0 444 82212 7) 529–577; http://maths.mq.edu.au/~street/45.pdf.

6 Two remarks

We will do some 'mental gymnastics' motivated by Pawel's lectures and discuss the small vs large issue, introducing Kan extensions.

6.1 Monoids

We started by saying that category theory is a theory of composition. Let us go back to basics. The basic theory of composition is given by the equational theory of monoids $(A, e, \cdot)$ where e is the identity and $\cdot$ is associative. The paragdigmatic monoid, the free monoid over a set A, is the monoid A^* of finite words with the empty word and concatenation as operations. But monoids alone cannot serve as a theory of composition, because the set A of things we want to compose may have some more structure than just being mere letters or words. We may want to compose functions, or relations, or processes, or computer programs, or some kind of systems, etc

So going back to the definition of a monoid as consisting of

$$1 \to A$$
$$A \times A \to A$$

we see that its very definition presupposes some structure such as 1 and $\times$ and, depending on the things we want to compose, this structure will take different forms.

So really we should consider monoids in a category. This gives us some more generality, so for example, we can now say that a topological monoid is a monoid in the category of topological spaces.

But this is not general enough. It is not always the case that composition

$$A \times A \to A$$

should be defined on a cartesian or categorical product $A \times A$. In fact, what is required to make the idea of a monoid work, is only that of a 'monoidal product'

$$A \otimes A \to A.$$

This can be formalised as a functor $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ satisfying itself the properties of a monoid and giving us the notion of a (strict) monoidal category.

Example 97. The category of endo-functors $\mathcal{C} \to \mathcal{C}$ with composition is a monoidal category.

We motivated monoidal categories by saying that they provide the minimal environment in which one can talk meaningfully about monoids. So what a monoid in the category of monoids?

Example 98. A monoid in the category of endofunctors is a monad.

We started with monoids as a theory of composition, introduced categories and monoidal categories to say what a monoid is and then recovered monads as particular monoids. Now let us go one step further and recover categories as monads.

Before we can do this, we need to think of monads in a category (or rather 2-category): (M, η, μ) is given by an arrow $M : C \to C$ and 2-cells $\mathrm{Id} \to M$ and $\mu : MM \to M$.

Example 99. Let Span be the category which has sets as objects and spans $X \leftarrow S \to Y$ as arrows $X \to Y$. The 2-cells $S \to S'$ are 1-cells that commute with the legs of the spans. What is a monad in Span?

As we have learned from Pawel this is more than only 'mental gymnastics' but this is genuinly useful ... all what Pawel showed us is ultimately based on this.

6.2 Small vs large, generation and Kan extensions

We have seen the small vs large issues in many places.

- In the definition of a category the homs are small but the collection of objects may be large (unless we have a small category).
- In universal properties we quantify over a large set of objects. And this is where the power of universal properties resides.
- Completeness and cocompleteness of categories was defined wrt to small limits and colimits ... since a category with all large limits and colimits is a preorder.
- A functor that is determined on a small subcategory and preserves limits is a right adjoint.
- If functor $T : \mathsf{Set} \to \mathsf{Set}$ is determined on a small subcategory then
  - T is a quotient of a signature functor
  - initial and free T-algebras exist
  - final and cofree T-algebras exists
- For all functors $T : \mathsf{Set} \to \mathsf{Set}$ there is a final T-coalgebra, but its carrier is not a (small) set, so does not exists in Set, but it is a proper class (large set). Thus the final coalgebra is then really the final coalgebra not for T but for an extension of T to an enlargement of Set.

One way of tackling the problem is to think about generators in a category. For example, the category Set is generated by 1: Every set is the quotient of a coproduct of 1.

Example 100. 1. In the universal definition of a product one does not need to quantify over all objects in Set, in fact, 1 is enough.

2. Every colimit preserving functor $\mathsf{Set} \to \mathsf{Set}$ is a left-adjoint.

3. Every left-adjoint $\mathsf{Set} \to \mathsf{Set}$ is of the form $X \mapsto A \times X$ form some $A \in \mathsf{Set}$.

4. The inclusion $\{1\} \to \mathsf{Set}$ is dense, that is, there is a bijection between colimit preserving functor $\mathsf{Set} \to \mathcal{C}$ and functors $\{1\} \to \mathsf{Set}$.

Definition 32. Given $F : \mathcal{A} \to \mathcal{B}$ and $K : \mathcal{A} \to \mathcal{C}$ the left Kan-extension of F along K is a functor $\mathrm{Lan}_K F : \mathcal{C} \to \mathcal{B}$ together with a natural transformation $\gamma : F \to (\mathrm{Lan}_K F)K$ such that for all $H : \mathcal{C} \to \mathcal{B}$ and all $\alpha : F \to HK$ there is a unique $\beta : \mathrm{Lan}_K F \to H$ such that $\alpha = \beta \circ \gamma$.

This is often remembered as

$$\frac{F \to HK}{\mathrm{Lan}_K F \to H}$$

Example 101. 1. What is the left Kan-extension of $F : \{*\} \to \mathsf{Set}$, $F(*) = A$ along $\{*\} \to \mathsf{Set}$?

2. Let Fin be the category of finite sets. What is the left Kan extension of $X \mapsto A \times X$, $X \mapsto \mathsf{List}(X)$, ... along $\mathsf{Fin} \to \mathsf{Set}$?

3. Let Fin be the category of finite sets. What is the left Kan extension of $X \mapsto \mathcal{P}(X)$, ... along $\mathsf{Fin} \to \mathsf{Set}$?

Definition 33. A functor $F : \mathsf{Set} \to \mathsf{Set}$ is called **finitary** iff it is the left Kan extension of its restriction along $\mathsf{Fin} \to \mathsf{Set}$.

Example 102. 1. The forgetful functor from algebras to sets is finitary if all operations take only a finite number of arguments.

2. If $T : \mathsf{Set} \to \mathsf{Set}$ is a finitary functor, then the initial T-algebra can by built as the colimit

$$0 \to T0 \to T(T0) \to \ldots \to T^n(0) \to \ldots \mathrm{colim}_{n\in\mathbb{N}} T^n(0)$$

3. If $T : \mathsf{Set} \to \mathsf{Set}$ is a finitary functor, then the final T-coalgebra can by built as the limit

$$1 \leftarrow \ldots T^n(1) \ldots \leftarrow \lim_{n\in\mathbb{N}} T^n(1) \leftarrow \ldots T^m(\lim_{n\in\mathbb{N}} T^n(1)) \ldots \leftarrow \lim_{m\in\mathbb{N}} T^m(\lim_{n\in\mathbb{N}} T^n(1))$$