Contents

Chapter 1

Background

In this chapter the building blocks for this work, almost entirely based on categories, will be defined. The aim of what follows is not only to introduce concepts that will be used later, but also to understand how category theory is general enough to give the abstraction of known notions (mainly from set theory) to reuse them in different contexts. This is not a complete tutorial on categories, but instead a sufficient compendium of definitions to make clear what will be done in the next chapters.

1.1 Basic Notions

This section is all about basic definitions and examples, to get familiar with the formalism of categories.

1.1.1 Categories

Definition 1.1.1 (Category)**.** A *category* $\mathscr{C}$ comprises:

1. A collection of *objects* $\mathcal{O}b(\mathscr{C})$;

2. A collection of *arrows* (or *morphisms*) $\mathcal{H}om(\mathscr{C})$, often called *homset*.

Two operators, *dom* and *cod*, that map every morphism $f \in \mathcal{H}om(\mathscr{C})$ to two objects, respectively, its *domain* and its *codomain*. In case $dom\ f = A$ and $cod\ f = B$, we will write $f : A \to B$. The collection of morphisms from an object A to an object B is denoted as $\mathscr{C}(A, B)$. An operator $\circ$ of *composition* maps every couple of morphisms f, g with $cod\ f = dom\ g$ (in this case f and g are said to be composable) to a morphism $g \circ f : dom\ f \to cod\ g$. The composition operator is associative, i.e., for each composable arrows f, g and h, it holds that

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

For each object A, an *identity* morphism $id_A : A \to A$ (or, when it is clear from the context, just denoted A) such that, for each $f : A \to B$:

$$id_B \circ f = f = f \circ id_A$$

The most important thing here is not the structure of the objects, but instead how this structure is preserved by the morphisms.

Example 1.1.2. A trivial example of category is the one with no objects, and hence no morphisms. Such category is denoted by $\mathbf{0}$ and is called *empty category*.

Example 1.1.3. The category with just one object and just one arrow, the identity arrow on that object, is denoted $\mathbf{1}$. In particular, the only object of this category is $\bullet$, and the only arrow is $id_\bullet$.

Given an arrow $fA \to B$ in a category $\mathscr{C}$, we say that f *factors through* $g : C \to B$ if there exists an arrow $h : A \to C$ such that $f = h \circ g$.

Definition 1.1.4. [Dual Category] Given a category $\mathscr{C}$, there exist a category $\mathscr{C}^{op}$ such that:

- $Ob(\mathscr{C}^{op}) = Ob(\mathscr{C})$;
- if $f : A \to B$ is a morphism in $\mathscr{C}$, then $f : B \to A$ is a morphism in $\mathscr{C}^{op}$.

Hence, given $f : A \to B$ and $g : B \to C$ arrows in $\mathscr{C}$, as $g \circ f : A \to C$ is an arrow in $\mathscr{C}$, then $f \circ g : C \to A$ is an arrow in $\mathscr{C}^{op}$. Such category is called *dual category* or *opposite category*.

Duality is a concept that we will encounter most of the time. Given a property P valid for a category $\mathscr{C}$, we will refer to the same property in the opposite category $\mathscr{C}^{op}$ as the *dual* of P, without explicitly constructing $\mathscr{C}^{op}$. There exist some properties that coincide exactly with their dual, and such properties are said to be *self dual* properties.

To represent morphisms of a category $\mathscr{C}$ it is possible to use *diagrams*, as the one below, in which the vertices are objects of $\mathscr{C}$, and the edges are morphisms of $\mathscr{C}$.

$$\begin{tikzcd}
X \arrow[r, "f'"] \arrow[d, "g'"'] & Z \arrow[d, "g"] \\
W \arrow[r, "f"'] & Y
\end{tikzcd}$$

The diagram is said to commute whenever $f \circ g' = g \circ f'$. Unique morphisms are represented with dashed arrows. A more rigorous definition of what a diagram is will be given later (Definition 1.2.3).

Example 1.1.5. It is easy to see that taking sets as objects and total functions as arrows, we obtain a category. In fact, given two functions $f : A \to B$ and $g : B \to C$, it is possible to compose them obtaining an arrow $g \circ f : A \to C$, and the composition is associative. For each set A there exists an identity function $id_A : A \to A$ such that $id_A(a) = a$ for each $a \in A$. This category is denoted as **Set**.

Ha senso mettere totale qui?

Remark 1.1.6. It is important to note that the Definition 1.1.1 above does not specify what kind of collections $\mathcal{O}b(\mathscr{C})$ and $\mathcal{H}om(\mathscr{C})$ are. Taking **Set** as example, the collection $\mathcal{O}b(\mathbf{Set})$ cannot be a set itself, due to Russel's paradox. It would be more appropriate referring to a category $\mathscr{C}$ which $\mathcal{O}b(\mathscr{C})$ and $\mathcal{H}om(\mathscr{C})$ are both sets as a *small category*, but it is assumed in this work, except where it is made explicit, for a category to be small. Another clarification must to be done, still considering **Set**. Given two sets A and B, it is possible to construct the set B^A of all functions from A to B. This is isomorphic to $\mathbf{Set}(A, B)$, for each pair of sets A and B. A category $\mathscr{C}$ where, for each pair of objects A and B, $\mathscr{C}(A, B)$ is a set is said to be *locally small*.

1.1.2 Mono, Epi and Iso

Between the morphisms of a category, it is possible to distinguish some that have certain properties, as functions between sets can be surjective, injective or bijective.

Definition 1.1.7 (Monomorphism)**.** An arrow $f : B \to C$ in a category $\mathscr{C}$ is a *monomorphism* if, for any pair of arrows of $\mathscr{C}$ $g : A \to B$, $h : A \to B$, the equality $f \circ g = f \circ h$ implies $g = h$. The class of monomorphisms of $\mathscr{C}$ is denoted $\mathcal{M}ono(\mathscr{C})$.

For a morphism, from an algebraic point of view, being mono means being *left cancellable.* This fact can led us to define a particular kind of class of morphisms, which will reveal useful further.

Definition 1.1.8 (Subobjects)**.** Let C be an object in a category $\mathscr{C}$. Then, if $m : A \to C$ is mono, (A, m) is said to be a *subobject* of C. Factorization of morphisms induces a preorder on subobjects of an object. $(A, m) \leq (B, n)$ whenever there exists a morphism $f : A \to B$ such that $m = n \circ f$.

sinceramente non trovo molto sensata la scelta di chiamare sottoggetto la coppia (A, m). Lo standard, per quanto ne so, è usare sottoggetto per una classe di equivalenza di mono o, al massimo, per i mono

$$
\begin{tikzcd}
A \arrow[dd, dashed, "f"'] \arrow[dr, "m"] & \\
 & B \\
C \arrow[ur, "n"'] &
\end{tikzcd}
$$

Two subobject (A, m) and (B, n) can are said to be *equivalent subobjects*, written $(A, m) \approx (B, n)$ if $(A, m) \leq (B, n)$ and $(B, n) \leq (A, m)$.

An useful fact about subobjects is how factorization behaves. In particular (A, m) and (B, n) are subobjects of C. Then, if $(A, m) \leq (B, n)$, we have $m = n \circ h$ for some h. Suppose k is another morphism such that $m = n \circ k$. We can conclude $h = k$ observing that $n \circ h = n \circ k$ implies $h = k$ when n is mono, which is by hypothesis. This is to say what follows.

Proposition 1.1.9. *Let (A, m) and (B, n) be subobjects of C in a category $\mathscr{C}$, with $(A, m) \leq (B, n)$. Then, the factorization of m through n is unique.*

Definition 1.1.10 (Epimorphism)**.** An arrow $f : A \to B$ in a category $\mathscr{C}$ is an *epimorphism* if, for any pair of arrows of $\mathscr{C}$ $g : B \to C$, $h : B \to C$, the equality $g \circ f = h \circ f$ implies $g = h$.

Definition 1.1.11 (Isomorphism)**.** An arrow $f : A \to B$ is an *isomorphism* if there is an arrow $f^{-1} : B \to A$, called the *inverse* of f, such that $f^{-1} \circ f = id_A$ and $f \circ f^{-1} = id_B$. Two objects are said to be *isomorphic* if there is an isomorphism between them.

Example 1.1.12. In **Set**, monomorphisms are injective functions, epimorphisms are surjective functions and isomorphisms are bijections.

Remark 1.1.13. Mono and epi are dual concepts. This fact is easily shown by considering how a monomorphism m in a category $\mathscr{C}$ behaves in the dual category $\mathscr{C}^{op}$. In $\mathscr{C}$ we have that $m \circ f = m \circ g$ implies $f = g$. In $\mathscr{C}^{op}$, the we can state that $f \circ m = g \circ m$ implies $f = g$, obtaining the definition of epi.

Proposition 1.1.14. *The following statements hold for every pair of composable arrows f and g for any category $\mathscr{C}$:*

1. *if both f and g are mono, then $g \circ f$ is mono;*
2. *if $g \circ f$ is mono, then f is mono;*
3. *if both f and g are epi, then $g \circ f$ is epi;*
4. *if $g \circ f$ is epi, then g is epi.*

val la pena osservare la relazione di equivalenza tra sottoggetti corrisponde ad avere un isomorfismo tra i domini

1.1.3 Categories from other categories

Starting from a category, it is possible to construct other categories with some interesting properties, as the following examples show.

The first notion to introduce is the one of subcategory.

Definition 1.1.15 (Subcategory)**.** A category $\mathscr{D}$ is a *subcategory* of a category $\mathscr{C}$ if:

1. each object of $\mathscr{D}$ is an object of $\mathscr{C}$;
2. each morphism between two objects of $\mathscr{D}$ is a morphism of $\mathscr{C}$; and
3. composites and identities of $\mathscr{D}$ are the same of $\mathscr{C}$.

If the inclusion at 2 is an equality (i.e. $\mathscr{D}(A, B) = \mathscr{C}(A, B)$ for each couple of objects A, B of $\mathscr{D}$), then $\mathscr{D}$ is said to be a *full subcategory* of $\mathscr{C}$. Another way to express that composites are the same (point 3) is to say that if $f, g \in \mathcal{H}om(\mathscr{D})$ are composable, then $g \circ f \in \mathcal{H}om(\mathscr{D})$, i.e., $\mathcal{H}om(\mathscr{D})$ is *closed under composition.*

An object of a category marks out a category itself. This is the case of slice (and coslice) categories.

Definition 1.1.16 (Slice Category)**.** Given a category $\mathscr{C}$ and an object $X \in \mathcal{O}b(\mathscr{C})$, the *slice category* $\mathscr{C}/X$ is the category that has pairs (A, f) as objects, where A is an object of C and $f : A \to X$ is an arrow in $\mathscr{C}$, and arrows $\phi : (A, f) \to (B, g)$ are given by a morphism $\phi : A \to B$ of $\mathscr{C}$ such that the following diagram commutes:

$$\begin{tikzcd} A \arrow[r, "\phi"] \arrow[dr, "f"'] & B \arrow[d, "g"] \\ & X \end{tikzcd}$$

– i.e, $g \circ \phi = f$. Composition between two arrows in $\mathscr{C}/X$ $\phi : (A, f) \to (B, g)$ and $\psi : (B, g) \to (C, h)$ is the arrow $\psi \circ \phi : (A, f) \to (C, h)$ obtained in the obvious way:

$$\begin{tikzcd} A \arrow[r, "\phi"] \arrow[rr, bend left, "\psi \circ \phi"] \arrow[dr, "f"'] & B \arrow[r, "\psi"] \arrow[d, "g"] & C \arrow[dl, "h"] \\ & X & \end{tikzcd}$$

The dual definition of *coslice category*, noted $X/\mathscr{C}$ (where $X \in \mathcal{O}b(\mathscr{C})$), is obtained by taking as objects the morhisms of $\mathscr{C}$ with domain X and as arrows the morphisms $\phi : (A, f) \to (B, g)$ such that $f : X \to A, g : B \to X$ of $\mathscr{C}$ and $g = \phi \circ f$.

Furthermore, it is possible to raise a new category from two old ones by taking their product, as the following definition shows.

Definition 1.1.17 (Product category)**.** Given two categories $\mathscr{C}, \mathscr{D}$, the *product category* $\mathscr{C} \times \mathscr{D}$ has as objects pairs of objects (A, B), where $A \in \mathcal{O}b(\mathscr{C}), B \in \mathcal{O}b(\mathscr{D})$, and as arrows pairs of morphisms (f, g), where f is an arrow in $\mathscr{C}$ and g is an arrow in $\mathscr{D}$. Composition and identities are defined pairwise: $(f, g) \circ (h, k) = (f \circ h, g \circ k)$, and $id_{(A,B)} = (id_A, id_B)$.

1.2 Functors, Natural Transformations, Adjoints

E cambierei anche l'intro a questa sezione DC

la 1.1 è un po' povera come sezione, se sposti questa sezione sui funtori prima, poi puoi arricchire la 1.1 con la sottosezione sulle comma

1.2.1 Functors

A functor is a structure preserving map between categories.

Definition 1.2.1 (Functor)**.** Let $\mathscr{C}$ and $\mathscr{D}$ be categories. A *functor* $F : \mathscr{C} \to \mathscr{D}$ is a map taking each object of $A \in \mathcal{O}b(\mathscr{C})$ to an object $F(A) \in \mathcal{O}b(\mathscr{D})$ and each arrow $f : A \to B$ of $\mathscr{C}$ to a arrow $F(f) : F(A) \to F(B)$ of $\mathscr{D}$, such that, for all objects $A \in \mathcal{O}b(\mathscr{C})$ and composable arrows f and g of $\mathscr{C}$:

- $F(id_A) = id_{F(A)}$;
- $F(g \circ f) = F(g) \circ F(f)$.

In this case, $\mathscr{C}$ is called *domain* and $\mathscr{D}$ is called *codomain* of the functor F.

Example 1.2.2. A first example of functor is the *identity functor*. Given a category $\mathscr{C}$, the identity functor $Id_{\mathscr{C}} : \mathscr{C} \to \mathscr{C}$ is the functor that maps each object on itself and each arrow onto itself.

Once defined what a functor is, we can give a more rigorous definition of diagram. Although this may seem extremely technical, it will be useful, especially in the definition of limits (Definition 1.3.3).

Definition 1.2.3 (Diagram)**.** A *diagram in a category $\mathscr{C}$ of shape $\mathscr{I}$* is a functor $D : \mathscr{I} \to \mathscr{C}$. The category $\mathscr{I}$ can be considered as the category indexing the objects and the morphisms of $\mathscr{C}$ shaped in $\mathscr{I}$.

Example 1.2.4. A diagram of shape $\Lambda = (L \xleftarrow{l} X \xrightarrow{r} R)$ is said to be a *span*, and is denoted by $(l, X, r) : L \rightharpoonup R$. A span can be viewed as the generalization of relations between sets. In fact, in **Set**, a relation $R \subseteq A \times B$ is a span, with the projections $\pi_A : R \to A$ and $\pi_B : R \to B$ as arrows.

The dual notion of span is a *cospan*, namely, a diagram of shape $\Lambda^{op} = (L \xrightarrow{l} X \xleftarrow{r} R)$, and is denoted by $(l, X, r) : L \rightharpoondown R$.

Functor are often used to generalize some structural behaviour that constructions in categories have. An important example of this fact is the universal property. The definition is not straightforward, but it gives the abstraction of a property that will be useful in further definitions

Definition 1.2.5 (Universal property)**.** Let $F : \mathscr{C} \to \mathscr{D}$ be a functor, and let $B \in \mathcal{O}b(\mathscr{D})$. A pair (u, A), with $A \in \mathcal{O}b(\mathscr{C})$ and $u : B \to F(A)$ is said to be an *universal map for B with respect to F* if for each $A' \in \mathcal{O}b(\mathscr{C})$ and each $f : B \to F(A')$ there exists a unique morphism $h \in \mathscr{C}(A, A')$ such that the following triangle commute:

$$
\begin{tikzcd}
B \arrow[r, "u"] \arrow[dr, "f"'] & F(A) \arrow[d, dashed, "F(h)"] & & A \arrow[d, dashed, "h"] \\
& F(A') & & A'
\end{tikzcd}
$$

– i.e. there exists a unique h such that $F(h) \circ u = f$. In this case, (u, A) is said to have the *universal property*.

Dually, if $G : \mathscr{C} \to \mathscr{D}$ is a functor and $B \in \mathcal{O}b(\mathscr{D})$, then a pair (A, u) is a *co-universal map for B with respect to G* if $u : G(A) \to B$ and for each $A' \in \mathcal{O}b(\mathscr{C})$ and each $f : G(A') \to B$ there exists a unique morphism $h \in \mathscr{C}(A', A)$ such that the following diagram commutes:

$$
\begin{tikzcd}
A' \arrow[d, dashed, "h"'] & & G(A') \arrow[d, dashed, "G(h)"'] \arrow[dr, "f"] & \\
A & & G(A) \arrow[r, "u"'] & B
\end{tikzcd}
$$

Some interesting properties of certain functors depend strictly on how they behave on the homsets of the domain and the codomain categories. The following definitions are about this particular type of functors.

Definition 1.2.6 (Full functor, faithful functor, fully faithful functor)**.** Let $F : \mathscr{C} \to \mathscr{D}$ be a functor, and consider the inducted function

$$F_{A,B} : \mathscr{C}(A, B) \to \mathscr{D}(F(A), F(B))$$

If, for each A, B objects of $\mathscr{C}$, $F_{A,B}$ is surjective, then F is said to be *full*, if it is injective, F is said to be *faithful*, if it is both injective and surjective, F is said to be *fully faithful*.

Observation 1.2.7. Properties such as fullness and faithfulness are so called *self-dual*, because the dual notion coincide with the same notion. This fact can be advantageous because if for example the faithfulness implies the preservation of some property, then the dual property is implied at the same way.

Example 1.2.8. Let $\mathscr{C}$ be a category and $\mathscr{D}$ a subcategory. The inclusion functor $I : \mathscr{D} \to \mathscr{C}$, mapping each object and each arrow onto itself. I is a faithful functor, because, given any pair of objects A and B of $\mathscr{D}$, $I_{A,B}$ is injective. If $\mathscr{D}$ is a full subcategory, then I is fully faithful.

Having such classification among functors turns out to be useful in many contexts. For example, consider $F(m) : F(B) \to F(C)$ be a monomorphism in a category $\mathscr{D}$, where $F : \mathscr{C} \to \mathscr{D}$ is a faithful functor. Then, if $f, g : A \to B$ are two morphisms in $\mathscr{C}$ such that $m \circ f = m \circ g$, then $F(m \circ f) = F(m) \circ F(f) = F(m) \circ F(g) = F(m \circ g)$. Since $F(m)$ is mono, then $F(f) = F(g)$, and, since $F_{A,B}$ is injective, $f = g$. Together with the fact that faithfulness is a self-dual concept, we have a proof for what follows [HS79].

Proposition 1.2.9. *Let $F : \mathscr{C} \to \mathscr{D}$ be a faithful functor. Then F reflects monomorphisms and epimorphisms.*

1.2.2 Natural Transformations

Given two functors that share domain and codomain categories, it is possible to define a transformation between them, taking each object of the domain of the functors to an arrow in the codomain of the functors that represent the action of "changing the functor acting on that object".

Definition 1.2.10 (Natural transformation)**.** Let $F, G : \mathscr{C} \to \mathscr{D}$ be two functors. A *natural transformation* η between them, denoted $\eta : F \dot{\to} G$, is a function $\eta : \mathcal{O}b(\mathscr{C}) \to \mathcal{H}om(\mathscr{D})$ taking each $A \in$

$\mathcal{O}b(\mathscr{C})$ to a morphism $\eta_A : F(A) \to G(A)$ in $\mathscr{D}$, such that, for each morphism $f : A \to B$ of $\mathscr{C}$, the following diagram commutes:

$$\begin{tikzcd} F(A) \arrow[r, "\eta_A"] \arrow[d, "F(f)"'] & G(A) \arrow[d, "G(f)"] \\ F(B) \arrow[r, "\eta_B"'] & G(B) \end{tikzcd}$$

– i.e., such that $G(f) \circ \eta_A = \eta_B \circ F(f)$.

We say that $\eta : F \dot{\to} G$ is a *natural isomorphism* if, for each $A \in \mathcal{O}b(\mathscr{C})$, η_A is an isomorphism in $\mathscr{D}$. In this case, F and G are said to be *naturally isomorphic*, and is denoted $F \cong G$.

Observation 1.2.11. It is easy to see that, given two natural transformations $\eta : F \dot{\to} G$, $\theta : G \dot{\to} H$, it is possible to compose them obtaining a new natural transformation $\xi = \theta \circ \eta : F \dot{\to} H$. This follows by the fact that the diagram

$$\begin{tikzcd} F(A) \arrow[r, "\eta_A"] \arrow[d, "F(f)"'] & G(A) \arrow[r, "\theta_A"] \arrow[d, "G(f)"] & H(A) \arrow[d, "H(f)"] \\ F(B) \arrow[r, "\eta_B"'] & G(B) \arrow[r, "\theta_B"'] & H(B) \end{tikzcd}$$

commutes because the two inner squares do. Sticking another diagram on the right of the one above, it is possible to show associativity of composition of natural transformations.

1.2.3 Functor Categories

The Observation 1.2.11 shows that natural transformations recreate on the functors the same structure that morphisms in a category have on objects. This leds us to define a particular kind of category, in which objects are functors between two categories, and arrow are natural transformations.

Definition 1.2.12 (Functor Category)**.** Let $\mathscr{C}$ and $\mathscr{D}$ be categories. The category whose objects are functors between $\mathscr{C}$ and $\mathscr{D}$ and whose arrows are natural transformations between them is said to be a *functor category*, and it is denoted by $[\mathscr{C}, \mathscr{D}]$.

A functor with a small category as domain (Remark 1.1.6) and **Set** as codomain is said to be a *presheaf* on that category. Given a category $\mathscr{C}$, it is possible to construct the functor category of the presheaves on $\mathscr{C}$, i.e. $[\mathscr{C}, \mathbf{Set}]$.

Remark 1.2.13. What we are calling here a presheaf is not totally accurate, because technically a presheaf on a small category $\mathscr{C}$ is a functor $F : \mathscr{C}^{op} \to \mathbf{Set}$. This technicality would bring more complexity, and it is beyond the scope of this work, so we will continue adopting the definition given above.

1.2.4 Comma Categories

Functor constructions allow us to generalise basic concepts already seen for categories. An important example of this fact are comma categories, a more general notion of slice categories (Definition 1.1.16).

Definition 1.2.14 (Comma category)**.** Let $\mathscr{C}$, $\mathscr{D}$ and $\mathscr{E}$ be categories, and let $S : \mathscr{C} \to \mathscr{E}$, $T : \mathscr{D} \to \mathscr{E}$ be functors (source and target):

$$\mathscr{C} \xrightarrow{\ S\ } \mathscr{E} \xleftarrow{\ T\ } \mathscr{D}$$

Then, the *comma category* $(S \downarrow T)$ is the category in which:

- the objects are triples (A, f, B), where $A \in \mathcal{O}b(\mathscr{C})$, $B \in \mathcal{O}b(\mathscr{D})$ and $f : S(A) \to T(B)$ is an arrow of $\mathscr{E}$;
- the arrows are pairs $(c, d) : (A, f, B) \to (C, g, D)$, where $c \in \mathcal{H}om(\mathscr{C})$ and $d \in \mathcal{H}om(\mathscr{D})$, such that the square below commutes;

$$\begin{tikzcd} S(A) \arrow[r, "f"] \arrow[d, "S(c)"'] & G(B) \arrow[d, "T(d)"] \\ T(C) \arrow[r, "g"'] & T(B) \end{tikzcd}$$

- composition of morphisms is obtained via pairwise composition, i.e., $(a, b) \circ (c, d) = (a \circ c, b \circ d)$.

Thus, the slice category $\mathscr{C}/X$ is the comma category given by the two functors $Id_{\mathscr{C}}$ (the identity functor), and the functor $!_X : \mathbf{1} \to \mathscr{C}$, where $\mathbf{1}$ is the one-object category defined in Example 1.1.3, and $!_X$ sends the only object of $\mathbf{1}$ to X (then the only morphism of $\mathbf{1}$ to id_X of $\mathscr{C}$):

$$\mathscr{C} \xrightarrow{Id_{\mathscr{C}}} \mathscr{C} \xleftarrow{!_X} \mathbf{1}$$

It is easy to see that $(Id_{\mathscr{C}} \downarrow !_X)$ is exactly the same of $\mathscr{C}/X$.

In the same way, it is possible to define coslice categories in terms of comma categories: the category $(!_X \downarrow Id_{\mathscr{C}})$ is exactly the coslice $X/\mathscr{C}$.

1.2.5 Adjoints

Definition 1.2.15 (Right Adjoint)**.** Let $R : \mathscr{C} \to \mathscr{D}$ be a functor. R is said *right adjoint* if, for each object A of $\mathscr{D}$, there exists an object $L(A)$ and an arrow $\eta_A : A \to R(L(A))$ in $\mathscr{C}$ such that, for each arrow $f : A \to R(B)$ of $\mathscr{D}$, there is a unique arrow $g : L(A) \to B$ such that the following diagram commutes.

$$\begin{tikzcd} A \arrow[r, "\eta_A"] \arrow[dr, "f"'] & R(L(A)) \arrow[d, dashed, "R(g)"] \\ & R(B) \end{tikzcd}$$

–i.e., $R(g) \circ \eta_A = f$.

Proposition 1.2.16. *In Definition 1.2.15, the map that takes an object A to an object $L(A)$ can be extended to a functor $L : \mathscr{D} \to \mathscr{C}$. Moreover, there exists a natural transformation $id_{\mathscr{D}} \to R \circ L$.*

Proof. Let R be the right adjoint as in Definition 1.2.15. Given $f \colon X \to Y$, we can define $L(f)$ as the unique arrow $L(X) \to L(Y)$ whose image through R fits in the diagram below.

$$\begin{tikzcd} X \arrow[r, "\eta_X"] \arrow[d, "f"'] & R(L(X)) \arrow[d, dashed, "R(L(f))"] \\ Y \arrow[r, "\eta_Y"'] & R(L(Y)) \end{tikzcd}$$

To see that in this way we get a functor it is now enough to notice the commutativity of the following diagrams.

$$\begin{tikzcd} X \arrow[r, "\eta_X"] \arrow[d, "id_X"'] & R(L(X)) \arrow[d, "R(id_{L(X)})"] \\ Y \arrow[r, "\eta_Y"'] & R(L(Y)) \end{tikzcd}$$

$$\begin{tikzcd} X \arrow[r, "f"] \arrow[d, "\eta_X"'] & Y \arrow[r, "g"] \arrow[d, "\eta_Y"'] & Z \arrow[d, "\eta_Z"] \\ R(L(X)) \arrow[r, "R(L(f))"'] & R(L(Y)) \arrow[r, "R(L(g))"'] & R(L(Z)) \end{tikzcd}$$

Finally, by construction the family given by all the $\eta_A \colon A \to R(L(A))$ is natural and we can conclude. $\square$

Remark 1.2.17. The family above mentioned is called *unit* of the adjunction.

Definition 1.2.18 (Left Adjoint)**.** Let $L : \mathscr{D} \to \mathscr{C}$ be a functor. L is a *left adjoint* if, for each object B of $\mathscr{C}$, there exists an object $R(B)$ and an arrow $\epsilon_B : L(R(B)) \to B$ in $\mathscr{D}$ such that, for each arrow $g : L(A) \to B$ of $\mathscr{C}$, there exists a unique arrow $f : A \to R(B)$ such that the following diagram commutes.

$$\begin{tikzcd} L(R(B)) \arrow[r, "\epsilon_B"] & B \\ L(A) \arrow[u, dashed, "L(f)"] \arrow[ur, "g"'] & \end{tikzcd}$$

– i.e., $\epsilon_B \circ L(f) = g$.

As we have shown before, it is possible to extend the mapping $A \to R(B)$ to a functor R, whose functoriality follows placing $\epsilon_X \circ$

$L(R(f)) = f \circ \epsilon_Y$ for each $f : X \to Y$. The family $\epsilon_B : L(R(B)) \to B$ is natural and it is called *counit* of the adjunction.

The connection between left and right adjoints is expressed in the following proposition.

Proposition 1.2.19. *Let L be the functor of Proposition 1.2.16. Then, L is a left adjoint.*

Proof. Given an object B in $\mathscr{C}$, we can consider the solid part of the diagram below. Since R is a right adjoint, we get a unique arrow whose image through R make the triangle commutative.

$$\begin{tikzcd} R(B) \arrow[r, "\eta_{R(B)}"] \arrow[dr, "id_{R(B)}"'] & R(L(R(B))) \arrow[d, dashed, "R(\epsilon_B)"] \\ & R(B) \end{tikzcd}$$

Let now A be an object of $\mathscr{D}$ and $g \colon L(A) \to B$ an arrow in $\mathscr{C}$. We can consider the composite $R(g) \circ \eta_A \colon A \to R(B)$. Then we have

$$\begin{aligned} R(\epsilon_B) \circ R(L(R(g))) \circ R(L(\eta_A)) \circ \eta_A &= R(\epsilon_B) \circ R(RL(R(g))) \circ \eta_{R(L(A))} \circ \eta_A \\ &= R(\epsilon_B) \circ \eta_{R(B)} \circ R(g) \circ \eta_A \\ &= R(g) \circ \eta_A \end{aligned}$$

Since R is a right adjoint and η its unit, it follows that $\epsilon_B \circ L(R(g) \circ \eta_A)$ coincides with g as wanted. $\square$

1.3 Limits and Universal Constructions

aggiungere i diagrammi e sistemare per l'indice

1.3.1 Limits and Colimits

Definition 1.3.1 (Cones)**.** Let $D : \mathscr{I} \to \mathscr{C}$ be a diagram in $\mathscr{C}$ of shape $\mathscr{I}$. A *cone* for D is an object X of $\mathscr{C}$, together with arrows $f_i : X \to D(i)$ indexed by $\mathscr{I}$ (i.e. one for each object i of $\mathscr{I}$), such that, for each morphism $\alpha : i \to j$ of $\mathscr{I}$, the following diagram commutes:

$$\begin{tikzcd} & X \arrow[dl, "f_i"'] \arrow[dr, "f_j"] & \\ D(i) \arrow[rr, "D(\alpha)"'] & & D(j) \end{tikzcd}$$

– i.e., $D(\alpha) \circ f_i = f_j$. We denote such cone as $\{f_i : X \to D(i)\}$.

Observation 1.3.2. Given a diagram D, the category of the cones for D, denoted $\mathbf{Cone}(D)$, is defined to have cones for D as objects and cone morphisms as arrows, where a cone morphism $\phi : C \to C'$ from $C = \{f_i : X \to D(i)\}$ to $C' = \{f'_i : X' \to D(i)\}$ is a morphism $\phi : X \to X'$ such that the following diagram commutes for each i:

$$\begin{tikzcd} X \arrow[rr, "\phi"] \arrow[dr, "f_i"'] & & X' \arrow[dl, "f'_i"] \\ & D(i) & \end{tikzcd}$$

Definition 1.3.3 (Limits)**.** Let $D : \mathscr{I} \to \mathscr{C}$ be a diagram in $\mathscr{C}$ of shape $\mathscr{I}$. A cone $\{f_i : X \to D(i)\}$ is a *limit* provided that, for any other cone $\{f'_i : X' \to D(i)\}$ for D, then there exists a unique morphism $k : X' \to X$ such that the following diagram commutes for each object i of $\mathscr{I}$:

$$\begin{tikzcd} X' \arrow[rr, dashed, "k"] \arrow[dr, "f'_i"'] & & X \arrow[dl, "f_i"] \\ & D(i) & \end{tikzcd}$$

– i.e., $f_i \circ k = f'_i$ for each object i of $\mathscr{I}$. Such limit is denoted as $(X, f_i)_{i \in \mathscr{I}}$

Observation 1.3.4. Given a diagram D, a limit for D is exactly the terminal object of the category $\mathbf{Cone}(D)$, defined in Observation 1.3.2.

The dual notions of cones and limits are that of cocones and colimits.

Definition 1.3.5. (Cocones, Colimits) A *cocone* for a diagram $D : \mathscr{I} \to \mathscr{C}$ is an object Y of $\mathscr{C}$ together with arrows $f_i : D(i) \to Y$ such that, for each $g : D(i) \to D(j)$ of $\mathscr{C}$, $f_j \circ g = f_i$. A cocone is denoted $\{f_i : D(i) \to Y\}$. A *colimit* for D is a cocone $C = \{f_i : D(i) \to Y\}$ with the universal property – i.e., if $C' = \{f'_i : D(i) \to Y'\}$ is another cone for D, then there exists a unique arrow $h : Y \to Y'$ such that, for each i, $h \circ f_i = f'_i$.

Remark 1.3.6. It makes sense to refer to a (co)limit as *the* (co)limit. Suppose $(P, p_i)_{i\in\mathscr{I}}$ and $(Q, q_i)_{i\in\mathscr{I}}$ be limits for a diagram $D : \mathscr{I} \to \mathscr{C}$. Then, there exists a unique morphism $h : Q \to P$ such that $p_i \circ k = q_i$ for each i. At the same way, there exists a unique morphisms $k : P \to Q$ such that $q_i \circ k = p_i$ for each i. From the existence of the identity, must be $k \circ h = id_Q$ and $h \circ k = id_P$, that is, P and Q are isomorphic.

Notion such limits and colimits are generalization of more particular cases that will be now introduced, that we will often call *universal constructions*.

Definition 1.3.7 (Initial Object, Terminal Object)**.** Consider the empty diagram (i.e., a diagram $D : \mathbf{0} \to \mathscr{C}$ where $\mathbf{0}$ is the empty category Example 1.1.2). Then, the limit of D is called *terminal object* and the colimit of D is called *initial object*, denoted, respectively, $\mathbb{1}_{\mathscr{C}}$ and $\mathbb{0}_{\mathscr{C}}$. (Subscripts are omitted when they are clear from the context).

Example 1.3.8. In **Set**, the initial object is the empty set $\varnothing$, because, for each set S, there exists the empty function from $\varnothing$ to S. The terminal object of **Set** is the singleton $\{\bullet\}$, because there is exactly one function from a set S to $\{\bullet\}$, namely, the function which sends each $s \in S$ to $\bullet$.

We now illustrate a result on functor categories (Definition 1.2.12) that will be useful later.

Proposition 1.3.9. *Let $\mathscr{D}$ be a category. If $\mathscr{D}$ has an initial object, then, for any category $\mathscr{C}$, $[\mathscr{C}, \mathscr{D}]$ has an initial object. If $\mathscr{D}$ has a terminal object, then, for any category $\mathscr{C}$, $[\mathscr{C}, \mathscr{D}]$ has a terminal object.*

Proof. Let $\mathbb{0}_{\mathscr{D}}$ be the initial object of $\mathscr{D}$, and consider the constant functor $I(f) = id_{\mathbb{0}_{\mathscr{D}}}$ for all $f \in \mathcal{H}om(\mathscr{C})$. Then, for any $G : \mathscr{C} \to \mathscr{D}$, $\eta : I \to G$, defining η_A as the *unique morphism from* $\mathbb{0}_{\mathscr{D}}$ *to* $G(A)$ for each $A \in \mathcal{O}b(\mathscr{C})$, is a natural transformation $I \dot{\to} G$, as the diagram below shows:

$$\begin{tikzcd}
I(A) = \mathbb{0}_{\mathscr{D}} \arrow[r, "\eta_A"] \arrow[d, "I(f) = id_{\mathbb{0}_{\mathscr{D}}}"'] & G(A) \arrow[d, "G(f)"] \\
I(A') = \mathbb{0}_{\mathscr{D}} \arrow[r, "\eta_{A'}"'] & G(A')
\end{tikzcd}$$

for each $f : A \to A'$, the square above must commute, since there is only one morphism from $\mathbb{0}_{\mathscr{D}}$ to $G(A')$. For the same reason, η is the only natural transformation from I to G, being indeed the initial object of $[\mathscr{C}, \mathscr{D}]$.

Defining $T(f) = id_{\mathbb{1}_{\mathscr{D}}}$ for each $f \in \mathcal{H}om(\mathscr{C})$. Then, $\theta : F \to T$, for any $F : \mathscr{C} \to \mathscr{D}$, defining θ_A as the *unique morphism from* $F(A)$ *to* $\mathbb{1}_{\mathscr{D}}$ is a natural transformation due to the commutativity of the following diagram for each $f : A \to A'$:

$$\begin{tikzcd}
F(A) \arrow[r, "\theta_A"] \arrow[d, "F(f)"'] & T(A) = \mathbb{1}_{\mathscr{D}} \arrow[d, "T(f) = id_{\mathbb{1}_{\mathscr{D}}}"] \\
F(A') \arrow[r, "\theta_{A'}"'] & T(A') = \mathbb{1}_{\mathscr{D}}
\end{tikzcd}$$

Hence, θ is the unique natural transformation from F to T, and T is the terminal object of $[\mathscr{C}, \mathscr{D}]$. $\square$

In particular, every presheaf has an initial and a terminal object, because **Set** does (Example 1.3.8).

Definition 1.3.10 (Product, Coproduct)**.** Let D be the following diagram:

$$A \qquad\qquad\qquad B$$

Then, a cone for D is an object X and two arrows $f : X \to A$, $g : X \to B$ (i.e., a span, defined in Example 1.2.4):

$$A \xleftarrow{f} X \xrightarrow{g} B$$

If it exists, a limit for D is called *product* of A and B, usually denoted as $(A \times B, \pi_A, \pi_B)$, while whose arrows are called *projections*. The colimit of D is called *coproduct* of A and B, usually denoted as $(\iota_A, \iota_B, A + B)$.

Example 1.3.11. **Set** has both products and coproduts. Given two sets A and B, the categorical product is the set-theoretic cartesian product $A \times B$, together with the two projections π_A and π_B, while the coproduct is the disjoint sum $A \amalg B = \{(x, 0) \mid x \in A\} \cup \{(y, 1) \mid y \in B\}$, together with the two canonical injections ι_A and ιB, where $\iota_A(a) = (a, 0)$ and $\iota_B(b) = (b, 1)$.

The notions of product and coproduct can be easily generalized, extending the definition to the product (and coproduct) of a family of objects, together with appropriate arrows (e.g., the projection arrows for each object in the product). We will denote the product of a collection of objects indexed by a (finite) category $\mathscr{I}$ as $\big(\prod_{i \in \mathcal{O}b(\mathscr{I})} X_i, (\pi_i)_{i \in \mathcal{O}b(\mathscr{I})}\big)$, and the coproduct as $\big((\iota_i)_{i \in \mathcal{O}b(\mathscr{I})}, \coprod_{i \in \mathcal{O}b(\mathscr{I})} X_i\big)$.

Definition 1.3.12 (Equalizer, Coequalizer)**.** Let D be the diagram below.

$$A \underset{g}{\overset{f}{\rightrightarrows}} B$$

The limit of D is called *equalizer*, and its colimit is called *coequalizer*.

Proposition 1.3.13. *Let $e : E \to A$ be the arrow that equalizes $f, g : A \to B$ in a category $\mathscr{C}$. Then, e is a monomorphism.*

Proof. Suppose X be an object and $x, y : X \to E$ be two morphisms in $\mathscr{C}$ such that $e \circ x = e \circ y$, and let $z = e \circ x$. Then, since e is an equalizer, $f \circ e = g \circ e$, and $f \circ z = g \circ z$. But, for the universal property of limits, there must be exactly one $u : Z \to E$ such that $z = e \circ u$. It follow that $x = u$ and $y = u$, hence $x = y$. $\square$

Of all monomorphisms, an interesting subclass of them is the one that contains only the equalizers.

Definition 1.3.14 (Regular Monomorphism)**.** A monomorphism that is an equalizer for a pair of arrows is said *regular monomorphism*. The class of all regular monomorphisms of a category $\mathscr{C}$ is denoted $\mathcal{R}eg(\mathscr{C})$.

Observation 1.3.15. Given two composable regular monos m and n, suppose that n equalizes two arrows f and g. Then, we have

$$\begin{aligned} g \circ (n \circ m) &= (g \circ n) \circ m \\ &= (f \circ n) \circ m && n \textit{ equalizer} \\ &= f \circ (n \circ m) \end{aligned}$$

Since $n \circ m$ is mono (Proposition 1.1.14), we have shown that, given a category $\mathscr{C}$, $\mathcal{R}eg(\mathscr{C})$ is closed under composition.

Definition 1.3.16 (Pullback, Pushout)**.** Let D be the cospan $(f, C, g) : A \rightharpoondown B$. A cone for D is an object P and three arrows $\phi : P \to A$, $\psi : P \to B$, and $h : P \to C$, but the latter is uniquely determined by the other ones ($f \circ \phi = h = g \circ \psi$). Thus, the following diagram is a cone:

$$\begin{tikzcd} P \arrow[r, "\psi"] \arrow[d, "\phi"'] & B \arrow[d, "g"] \\ A \arrow[r, "f"'] & C \end{tikzcd}$$

Then, the limit of D is called *pullback* of f and g. Given a span $S = (l, X, r) : L \rightharpoonup R$, shown in the diagram below,

$$\begin{tikzcd} L & X \arrow[l, "l"'] \arrow[r, "r"] & R \end{tikzcd}$$

a cocone for S is any commutative square of the form

$$\begin{tikzcd} & C & \\ L \arrow[ur, "f"] & X \arrow[l, "l"'] \arrow[r, "r"] & R \arrow[ul, "g"'] \end{tikzcd}$$

(the morphism $X \to C$ is uniquely determined by the relation $f \circ l = g \circ r$). The colimit for S is called *pushout* of l and r.

Example 1.3.17. In **Set**, given two functions $f : A \to C$ and $g : B \to C$, a pullback of f and g exists and is exactly the set $P = \{(x, y) \in A \times B \mid f(x) = g(y)\}$, with $\pi_f : P \to B$ and $\pi_g : P \to C$

defined, respectively, by $\pi_f((x,y)) = y$ and $\pi_g((x,y)) = x$. In this way, we have then, $\forall (x,y) \in P$:

$$\begin{aligned}
(f \circ \pi_g)((x,y)) &= f(\pi_g((x,y))) \\
&= f(x) && \textit{Definition of } \pi_g \\
&= g(y) && (x,y) \in P \\
&= g(\pi_f((x,y))) && \textit{Definition of } \pi_f \\
&= (g \circ \pi_f)((x,y))
\end{aligned}$$

thus, $f \circ \pi_g = g \circ \pi_f$.

Another important example to our aims is a concrete definition of what is a pushout in the category of sets, and why morally we can regard a pushout as *the way to identify part of an object with a part of another* [BW95].

Example 1.3.18. In **Set**, given two functions $f : A \to B$ and $g : A \to C$, the pushout of them is the set $X = (B \amalg C)/_{\sim}$, where $\sim$ is the least equivalence relation such that $f(a) \sim g(a)$ for each $a \in A$, with $\iota_g : B \to X$ and $\iota_f : C \to X$ as arrows, sending each element of the domain in the corresponding equivalence class in X. In particular, for each $a \in A$:

$$\begin{aligned}
(\iota_g \circ f)(a) &= \iota_g(f(a)) \\
&= [(f(a), 0)] && \textit{Definition of } \iota_g \\
&= [(g(a), 1)] && f(a) \sim g(a) \\
&= \iota_f(g(a)) && \textit{Definition of } \iota_f \\
&= (\iota_f \circ g)(a)
\end{aligned}$$

When both f and g are monos (that is, injections), then we can construct the pushout in the same way we have done above, with $(f(a), 0) \sim (g(a), 1)$ when such a exists and $(b, 0) \sim (c, 1)$ on each b and c with no preimage in A, with ι_f and ι_g injective. An easy way to see this fact is considering the following situation: let $f : A \to A \cup B$ and $g : A \to A \cup C$, with A disjoint from B and C, $f(a) = a$ and $g(a) = a$. Then the pushout is the object $A \cup B \cup C$, with the inclusions as arrows, that are also injective. A more general case is what happens considering functions $f : A \to B$ and $g : A \to C$ injective. Differently from the previous example, in this case is not possible to take just the union of codomains as the pushout, but rather the disjoint union of them and then identify the elements

$f(a)$ with $g(a)$, as we have done above. In the category of sets and functions, we have the certainty that the pullback arrows are injective. In fact, taking the equivalence relation $\sim$, we have that $f(a) \sim f(a')$ if and only if $a = a'$ by hypothesis, and then $x \sim x'$ if and only if $x = x'$, then the pushout morphisms sends each element in an equivalence class composed only by the element itself, thus are injective. This is an interesting property that in other categories may do not hold, and will be recalled later.

Given a subclass of morphisms of a category, an important property is *stability* under certain type of constructions. In our case, we are interested in stability under pullbacks and under pushouts.

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "m"'] & B \arrow[d, "n"] \\ C \arrow[r, "g"'] & D \end{tikzcd} \tag{$*$}$$

Definition 1.3.19 (Stability under pullbacks, pushouts)**.** Given a category $\mathscr{C}$, a subclass $\mathcal{A} \subseteq \mathcal{H}om(\mathscr{C})$ is said to be *stable under pullbacks* if, for every pullback square as the one in $(*)$, if $n \in \mathcal{A}$, then $m \in \mathcal{A}$. $\mathcal{A}$ is said to be *stable under pushouts* if, for every pushout square as the one in $(*)$, if $m \in \mathcal{A}$, then $n \in \mathcal{A}$.

Proposition 1.3.20. *Let $f : A \to C$, $g : B \to C$ be arrows in any category $\mathscr{C}$, and consider the following pullback square:*

$$\begin{tikzcd} P \arrow[r, "\pi_f"] \arrow[d, "\pi_g"'] & B \arrow[d, "g"] \\ A \arrow[r, "f"'] & C \end{tikzcd}$$

If g is mono, then so is π_g.

The proposition above can be dualised stating that pushouts preserves epimorphisms.

The connection between constructions as products and equalizers and limits is made clear by the following theorem. The idea behind the proof is the fact that, given a diagram $D : \mathscr{I} \to \mathscr{C}$, if each subset of objects $X = \{D(i) \mid i \in \mathcal{O}b(\mathscr{I})\} \subseteq \mathcal{O}b(\mathscr{C})$ has a product $(\prod_{i\in I} D(i), (\pi_i)_{i\in\mathcal{O}b(\mathscr{I})})$ and each pair of arrows $f, g \in \mathscr{C}(D(i), D(j))$ has an equalizer $Eq(f, g)$, then one can construct the cone taking the equalizer of the arrows that has as domain the product of the objects of the diagram, and as codomain the product of the codomains of the arrows of the diagram. This construction has the universal property because equalizers and products do. A detailed proof is in the appendix.

Theorem 1.3.21 (Limit theorem)**.** *Let $\mathscr{C}$ be a category. Then $\mathscr{C}$ has all finite limits if and only if $\mathscr{C}$ has all finite products and all finite equalizers.*

Remark 1.3.22. The theorem above (and its relative proof) can be stated in its dual form leading to a theorem on existence of colimits, and a relative criterion to calculate them (taking the dual of the proof).

Example 1.3.23. Limit theorem gives us an easy way to calculate limits. An example of this fact is how limits are computed in **Set**. Given a diagram $D : \mathscr{I} \to \mathbf{Set}$, where $\mathscr{I}$ is a small category and $I = Ob(\mathscr{I})$, its limit is the set L defined as follows:

$$L = \{(d_i)_{i\in I} \in \prod_{i\in I} D(i) \mid \forall\phi \in \mathscr{I}(i, i'), D(\phi)(d_i) = d_{i'}\}$$

with projections as arrows.

Example 1.3.24. As we have done in Example 1.3.23, we illustrate how to construct colimits in the category of sets. Given a small category $\mathscr{I}$, $I = Ob(\mathscr{I})$, and a diagram $D : \mathscr{I} \to \mathbf{Set}$, consider the equivalence relation $\sim$ defined on $\coprod_{i\in I} D(i)$ such that $d_i \sim d_{i'}$ if $d_i \in D(i), d_{i'} \in D(i')$ and there exists some $\phi \in \mathscr{I}(i, i')$ such that $D(\phi)(d_i) = d_{i'}$. Then, a colimit for D is the set

$$C = \big(\coprod_{i\in I} D(i)\big)/\sim$$

with the inclusions as arrows.

Remark 1.3.25. Since a diagram is nothing more than a functor from a "shape" category to another, it makes sense to talk about limits of functors in general, even when they are not intended to be diagrams.

Observation 1.3.26. So far we introduced categories of presheaves. In these categories, an interesting fact is that limits and colimits are computed pointwise – i.e., the limit of a diagram in a category of presheaves is exactly the limit on each of its components.

In the next sections, we will work on a special kind of diagrams with certain properties. In particular, we are interested in how a functor behaves with respect to the constructions defined so far.

Definition 1.3.27. Let $D : \mathscr{I} \to \mathscr{C}$ be a diagram, and $F : \mathscr{C} \to \mathscr{D}$ a functor. We say that F:

1. *preserves limits* of D if, given a limit $(L, l_i)_{i\in\mathscr{I}}$ for D, then $(F(L), F(l_i))_{i\in\mathscr{I}}$ is a limit for $F \circ D$.
2. *reflects limits* of D if a cone $(L, l_i)_{i\in\mathscr{I}}$ is a limit for D whenever $(F(L), F(l_i))_{i\in\mathscr{I}}$ is a limit for $F \circ D$.
3. *lifts limits (uniquely)* of D if, given a limit $(L, l_i)_{i\in\mathscr{I}}$ for $F \circ D$, there exists a (unique) limit $(L', l'_i)_{i\in\mathscr{I}}$ for D such that $(F(L'), F(l'_i))_{i\in\mathscr{I}} = (L, l_i)_{i\in\mathscr{I}}$.
4. *creates limits* of D if D has a limit and F preserves and reflects limits along it.

The dual notions are obtained in the obvious way, namely, substituting the words "limits" and "cones" with "colimits" and "cocones", respectively

Observation 1.3.28. It holds that if a functor creates limits, then lifts uniquely limits [AHS09].

Proposition 1.3.29. *A fully faithful functor reflects all limits and colimits.*

The next theorem is about a particular property that adjoint functors have.

Theorem 1.3.30. *Let $F : \mathscr{C} \to \mathscr{D}$ be a functor, and $G : \mathscr{D} \to \mathscr{C}$ its right adjoint. Then, G preserves limits.*

Remark 1.3.31. The dual of the theorem above states that, if G is a functor and F is a left adjoint, then F preserves colimits.

1.4 Adhesivity

The next section is about adhesivity. An adhesive category is intuitively a category in which pushouts of (some) monomorphisms exist and they behave more or less as they do among sets.

Definition 1.4.1. (Van Kampen property) Let $\mathcal{A}$ be a subclass of $\mathcal{H}om(\mathscr{C})$, and consider the diagram below:

$$
\begin{tikzcd}
A' \arrow[rrr, "f'"] \arrow[dr, "a"] \arrow[ddd, "m'"'] & & & B \arrow[dl, "b"'] \arrow[ddd, "n'"] \\
& A \arrow[r, "f"] \arrow[d, "m"'] & B \arrow[d, "n"] & \\
& C \arrow[r, "g"'] & D & \\
C' \arrow[rrr, "g'"'] \arrow[ur, "c"] & & & D' \arrow[ul, "d"']
\end{tikzcd}
$$

we say that the inner square is an $\mathcal{A}$-*Van Kampen* square if:

- it is a pushout;
- $a, b, c, d \in \mathcal{A}$;
- whenever the top and the left squares are pullbacks then the outer square is a pushout if and only of the right and the bottom squares are pullbacks.

We are now ready to give the notion of $\mathcal{M}$-adhesivity.

Definition 1.4.2 ($\mathcal{M}$-adhesivity)**.** Let $\mathscr{C}$ be a category and $\mathcal{M} \subseteq \mathcal{M}ono(\mathscr{C})$ containing all isomorphisms, closed under composition and stable under pullbacks and pushouts (Definition 1.3.19). Then $\mathscr{C}$ is $\mathcal{M}$-*adhesive* if

1. every cospan $C \xrightarrow{g} D \xleftarrow{m} B$ with $m \in \mathcal{M}$ can be completed to a pullback (such pullbacks are called $\mathcal{M}$-pullbacks);
2. every span $C \xleftarrow{m} A \xrightarrow{f} B$ with $m \in \mathcal{M}$ can be completed to a pushout (such pushouts are called $\mathcal{M}$-pushouts);
3. pushouts along $\mathcal{M}$-arrows are $\mathcal{M}$-Van Kampen squares.

We also say that $\mathscr{C}$ is *adhesive* when it is $\mathcal{M}ono(\mathscr{C})$-adhesive, and *quasiadhesive* when it is $\mathcal{R}eg(\mathscr{C})$-adhesive.

Observation 1.4.3. Set is adhesive.

Here it follows an interesting property of adhesive categories [Lac11].

Proposition 1.4.4. *In any adhesive category, the pushout of a monomorphism along any morphism is a monomorphism, and the resulting square is also a pullback.*

Verifying $\mathcal{M}$-adhesivity using the definition above may turn out to be very complex, so we can make use of the following result [CGM22].

Theorem 1.4.5. *Let $\mathscr{C}$ be a category, $\mathcal{M} \subseteq \mathcal{M}ono(\mathscr{C})$ containing all isomorphisms, closed under composition and stable under pullbacks and pushouts. Let now $F : \mathscr{C} \to \mathscr{D}$ be a functor with $\mathscr{D}$ $\mathcal{N}$-adhesive for some $\mathcal{N} \subseteq \mathcal{M}ono(\mathscr{D})$. If F is such that $F(\mathcal{M}) \subseteq \mathcal{N}$ and creates pullbacks and $\mathcal{M}$-pushout, then $\mathscr{C}$ is $\mathcal{M}$-adhesive.*

The idea behind this theorem is to simplify calculations to show that a certain category is adhesive for some subclass of monomorphisms, considering a functor from the category of which we want to prove adhesivity to a category we know it is adhesive, requiring that such functor has some properties.

Proof. In order to prove $\mathcal{M}$-adhesivity of $\mathscr{C}$, we have to verify the condition in Definition 1.4.2.

- Let $C \xrightarrow{g} D \xleftarrow{m} B$ with $m \in \mathcal{M}$ be a cospan in $\mathscr{C}$. Applying F, we obtain $F(C) \xrightarrow{F(g)} F(D) \xleftarrow{F(m)} B$, with $F(m) \in \mathcal{N}$ by hypothesis. Then, there exists a pullback $(P_F, p_{F(B)}, p_{(F(D))})$ in $\mathscr{D}$, which is an $\mathcal{N}$-pullback (Definition 1.3.16). Since F creates pullbacks, hence lifts them (Observation 1.3.28), there exist a pullback (P, p_B, p_D) in $\mathscr{C}$.

- Let $C \xleftarrow{m} A \xrightarrow{f} B$ with $m \in \mathcal{M}$ be a cospan in $\mathscr{C}$. Analogously to the previous point, applying the functor F we obtain $F(C) \xleftarrow{F(m)} F(A) \xrightarrow{F(f)} F(B)$ with $F(m) \in \mathcal{N}$, and there exists a $\mathcal{N}$-pushout $(q_{F(C)}, q_{F(B)}, F(Q))$ in $\mathscr{D}$. Since F reflects pushouts, (q_C, q_B, Q) is a $\mathcal{M}$-pushout in $\mathscr{C}$.

- the Van Kampen property of $\mathcal{M}$-pullbacks follows from the closure under pullbacks and pushouts of $\mathcal{M}$ and from the fact that F reflects pullbacks.

$\square$

Chapter 2

Categories of Graphs

This chapter is about graphs, and how it is possible to formalize them using categories in order to point out their properties from an abstract point of view. Starting from the set-theoretical definition of graphs, we will give an abstraction via functor categories, in which a graph is nothing but a functor between a category to another.

2.1 Graphs

A (directed graph) $\mathcal{G}$ is a mathematical structure consisting of a set of edge, a set of nodes and two functions, one assigning a *source* node and one assigning a *target* node to an edge. Formally, $\mathcal{G}$ is a quadruple $(V_\mathcal{G}, E_\mathcal{G}, s_\mathcal{G}, t_\mathcal{G})$, where $V_\mathcal{G}$ is the set of nodes, $E_\mathcal{G}$ is the set of edges, and $s_\mathcal{G}, t_\mathcal{G} : E_\mathcal{G} \to V_\mathcal{G}$ are the source and the target functions.

A *graph homomorphism* $h : \mathcal{G} \to \mathcal{H}$ is then a pair of functions $h = (h_V : V_\mathcal{G} \to V_\mathcal{H}, h_E : E_\mathcal{G} \to E_\mathcal{H})$ such that

$$h_V \circ s_\mathcal{G} = s_\mathcal{H} \circ h_E$$

and

$$h_V \circ t_\mathcal{G} = t_\mathcal{H} \circ h_E$$

that is, a structure preserving map.

We can then generalize such notion to something more abstract, considering a graph nothing more than a presheaf from the category $(E \rightrightarrows V)$ to the category of sets. Having two of such presheaves, a natural transformation from one to another encapsulates the behavior of a graph morphism due to naturality. We can now define the category of graphs.

Dimostrare come si calcolano i limiti nelle categorie di prefasci (componente per componente) e poi dare qualche esempio, porendendolo dalla versione precedente. Dimostrare anche come sono fatti i mono (mono sulle componenti).

Definition 2.1.1 (Category of Graphs)**.** We denote as **Graph** the category

$$[E \overset{s}{\underset{t}{\rightrightarrows}} V, \mathbf{Set}]$$

Remark 2.1.2. TODO: Si può generalizzare a tutte le categorie regolari per evitare di perdere le proprietà che usiamo (da eq.rel. a quot.).

2.2 Graphs with Equivalences

It is possible to endow the set of vertices of a graph with any sort of relation, requiring that such relation is preserved by homomorphisms. The ones we are interested in are *equivalence relations*. A graph with equivalence is a graph with an equivalence relation defined over its set of vertices.

Formally, A *graph with equivalence* is a pair $\mathbb{G} = (\mathcal{G}, \sim_{\mathcal{G}})$ where $\mathcal{G}$ is a graph and $\sim_{\mathcal{G}} \subseteq V_{\mathcal{G}} \times V_{\mathcal{G}}$ is an equivalence relation over its set of nodes. An homomorphism between two graphs with equivalences $h : \mathbb{G} = (\mathcal{G}, \sim_{\mathcal{G}}) \to \mathbb{H} = (\mathcal{H}, \sim_{\mathcal{H}})$ is a graph homomorphism $h = (h_V, h_E) : \mathcal{G} \to \mathcal{H}$ such that if $v_1 \sim_{\mathcal{G}} v_2$ then $h_V(v_1) \sim_{\mathcal{H}} h_V(v_2)$.

Remark su categorie in cui si può passare dalle relazioni di equivalenza al quoziente (Categorie Regolari (?)) per giustificare la def. di eq-grafo

.

As we have done in Section 2.1, we can think to a graph with equivalence as a presheaf, this time from a category $E \rightrightarrows V \to C$, where the image of C along the presheaf is the set of the equivalence classes, requiring that the morphism $V \to C$ is an epimorphism (that is, a surjective function).

Definition 2.2.1 (Category of Graphs with Equivalences)**.** The category **EqGrph** is the subcategory of

$$[E \overset{s}{\underset{t}{\rightrightarrows}} V \xrightarrow{q} C, \mathbf{Set}]$$

such that, for each $\mathbb{G} \in \mathcal{O}b(\mathbf{EqGrph})$, $\mathbb{G}(q)$ is an epimorphism.

Observation 2.2.2. Graph, defined in Definition 2.1.1 is equivalent to the full subcategory of **EqGrph** where objects are graphs $(G, =)$, i.e., in which each node is equivalent only to itself.

In the category of graphs with equivalences, universal constructions are easy to obtain.

Proposition 2.2.3. *Let* $D : \mathscr{I} \to \mathbf{EqGrph}$ *be a diagram. Then, a limit for* D *is* $(\mathbb{L}, l^i)_{i \in \mathcal{O}b(\mathscr{I})}$ *with in which* $\mathbb{L} = (\mathcal{L}, \sim_{\mathcal{L}})$ *is a graph in which (in* **Set***):*

- $(V_{\mathcal{L}}, l^i_V)$ *is the limit of all the* $V_{D(i)}$*;*
- $(E_{\mathcal{L}}, l^i_E)$ *is the limit of all the* $E_{D(i)}$*.*

And the equivalence relation on nodes is the least equivalence relation $\sim_{\mathcal{L}}$ *such that, given two vertices* v *and* w *of* $\mathbb{L}$*, if* $v \sim_{\mathcal{L}} wm$ *then* $l^i_V(v) \sim_i l^i_V(w)$ *for each* $i \in \mathcal{O}b(\mathscr{I})$*. (Here* $\sim_i$ *denotes the equivalence relation of* $D(i)$*).*

Proof. Firstly, we show that $(\mathbb{L}, l^i)_{\mathscr{I}}$ is a cone. Let $D(\alpha) : D(i) \to D(j)$, with $\alpha : i \to j$ morphism in $\mathscr{I}$. Our claim is $D(\alpha) \circ l^i = l^j$. Hence,

$$\begin{aligned}
D(\alpha) \circ l^i &= (h_V, h_E) \circ (l^i_V, l^i_E) \\
&= (h_V \circ l^i_V, h_E \circ l^i_E) \\
&= (l^j_V, l^j_E) \\
&= l^j
\end{aligned}$$

For $(\mathbb{L}, l^i)_{\mathscr{I}}$ to be a cone, the only constraint on the equivalence relation $\sim_{\mathcal{L}}$ is to be such that graph homomorphisms above can preserve it. But such equivalence relation exists in every case, considering the identity. This led us to the second point of the proof, that is, the universal property. For the cone described above to be a limit must be valid the universal property. Such condition is is satisfied taking the least equivalence relation such that, given two vertices v and w of $\mathbb{L}$, if $v \sim_{\mathcal{L}} w$ then $l^i_V(v) \sim_i l^i_V(w)$ for each $i \in \mathcal{O}b(\mathscr{I})$. Suppose the following situation:

$$\begin{tikzcd}
 & \mathbb{L}' \arrow[d, "h"] \arrow[ddl, bend right, "q^i"'] \arrow[ddr, bend left, "q^j"] & \\
 & \mathbb{L} \arrow[dl, "l^i"'] \arrow[dr, "l^j"] & \\
D(i) \arrow[rr, "D(\alpha)"] & & D(j)
\end{tikzcd}$$

The universal property the follows from the definition of least equivalence relation. □

Rivedere completamente questa proposizione e riadattarla alla definizione come prefasci.

Esempi, da copiare praticamente dalla versione precedente

prova

From Proposition 2.2.3 it is possible to derive a string characterisation of regular monomorphisms in **EqGrph**. Intuitively, a regular mono among graphs with equivalence is a morphism that not only preserves equivalences but reflects them too, that is, if $h : \mathbb{G} \to \mathbb{H}$ is mono and such that $h_V(v_1) \sim_\mathcal{H} h_V(v_2)$ implies $v_1 \sim_\mathcal{G} v_2$, then h is regular mono.

Proposition 2.2.4. *In* **EqGrph***, a monomorphims is regular if it is mono on all the components.*

The graph with equivalence $(\mathcal{G}, \sim_\mathcal{G})$ is another representation of the graph $\mathcal{G}/_{\sim_\mathcal{G}}$. Such graph is called *quotient graph*, and it can be expressed by the action of a functor over a graph with equivalence, which is defined below. The idea is to identify the graph with equivalence above with $\mathcal{G}/_{\sim_\mathcal{G}} = (V/_{\sim_\mathcal{G}}, E, s', t')$, where $s'(e) = [v]_{\sim_\mathcal{G}}$ if $s(e) = v$, and $t'(e) = [v]_{\sim_\mathcal{G}}$ if $t(e) = v$, and, consequently, having coherent action on homomorphisms.

Definition 2.2.5 (Quotient Functor)**.** The *quotient functor* is the functor Q : **EqGrph** $\to$ **Graph** such that, for each $\mathbb{G}, \mathbb{H}$ and for each morphism $\phi = (\phi_E, \phi_V, \phi_C) : \mathbb{G} \dot{\to} \mathbb{H}$:

- $Q(\mathbb{G})(E) = \mathbb{G}(E)$, $Q(\mathbb{G})(V) = \mathbb{G}(C)$;
- $Q(\mathbb{G})(s) = \mathbb{G}(q \circ s)$, $Q(\mathbb{G})(t) = \mathbb{G}(q \circ t)$;
- $Q(\phi_E) = \phi_E$, $Q(\phi_V) \circ \mathbb{G}(q) = \phi_C \circ \mathbb{H}(q)$.

We now give some considerations on quotient functor.

Proposition 2.2.6. *Quotient functor has a left adjoint and a right adjoint.*

Proof. To prove the statement we just have to find the adjoints.

il punto è che NON sai che I è un funtore. Quello che devi fare è mostrare che epsilon ha la proprietà di una counità, DOPO deduci che I è un funtore

Let I : **Graph** $\to$ **EqGrph** be the functor that sends each graph $\mathcal{G}$ onto the graph with equivalence $(\mathcal{G}, =_\mathcal{G})$, where $=_\mathcal{G}$ is the identity relation. Consider the following situation:

$$\begin{tikzcd}
(Q \circ I)(\mathcal{G}) = \mathcal{G}/_{=_\mathcal{G}} \arrow[r, "\epsilon_\mathcal{G}"] & \mathcal{G} \\
I(\mathcal{H}) = (\mathcal{H}, =_\mathcal{H}) \arrow[u, dashed, "I(f)"] \arrow[ur, "g"'] &
\end{tikzcd}$$

The graph $\mathcal{G}/_{=_\mathcal{G}}$ is exactly the graph $\mathcal{G}$, hence the arrow $\epsilon_\mathcal{G}$ is the identity arrow $id_\mathcal{G}$, so the arrow $I(f)$ is uniquely determined by g, having $\epsilon_\mathcal{G} \circ I(f) = I(f) = g$. Therefore, $\epsilon : (Q \circ I) \to Id_{\mathbf{Graph}}$ is the counit of the adjunction, and I is a left adjoint.

A right adjoint of Q is the functor $T : \mathbf{Graph} \to \mathbf{EqGrph}$ such that, for each graph $\mathcal{G}$ of **Graph**, $T(\mathcal{G}) = (\mathcal{G}, \approx_\mathcal{G})$, where $\approx_\mathcal{G}$ is the total relation among vertices of $\mathcal{G}$. We have then the following situation:

stesso problema di prima. Io riscriverei anche questa metà perché non si capisce la notazione e ad un certo punto occorre usare le proprietà dei quozienti in Set, cosa che non è evidente. UPDATE: probabilmente questo non è nu funtore aggiunto

$$\begin{tikzcd}
\mathbb{G} = (\mathcal{G}, \sim_\mathcal{G}) \arrow[r, "\eta_\mathbb{G}"] \arrow[dr, "f"'] & (T \circ Q)(\mathbb{G}) = (\mathcal{G}/_{\sim_\mathcal{G}}, \approx_{\mathcal{G}/_{\sim_\mathcal{G}}}) \arrow[d, dashed, "h"] \\
 & T(\mathcal{H}) = \mathbb{H}
\end{tikzcd}$$

Since $\mathbb{H}$ is in the image of T, it is a graph with all equivalent nodes. Then, the morphism f is a graph morphism that in addition sends each equivalence class of vertices onto the unique class in the graph $\mathbb{H}$. Then, f factors uniquely as $h \circ \eta_\mathbb{G}$, and h is a graph morphism extended to a morphism in **EqGrph** by sending the unique equivalence class on $(T \circ Q)(\mathbb{G})$ onto the unique equivalence class in $\mathbb{H}$, and $\eta : Id_{\mathbf{EqGrph}} \to (T \circ Q)$ is the unit of the adjunction. □

The following result lies on Theorem 1.3.30 and its dual.

Corollary 2.2.7. *Quotient functor preserves limits and colimits.*

Lemma 2.2.8. *Quotient functor preserves pushouts along regular monomorphisms.*

Capire qui!

COSE DA FARE >Quoziente crea i limiti (questo dovrebbe derivare direttamente dalla sezione precedente) >Quoziente crea certi pushout (questo va fatto ex novo)

Lemma 2.2.9. *In* **EqGrph**, $\mathcal{R}eg$(**EqGrph**) *is stable under pushouts and pullbacks (in the sense of Definition 1.3.19).*

Lemma 2.2.10. *The quotient functor creates pullbacks.*

Proof. We will show now that Q reflects pullbacks. Consider the following diagram in **Graph**

$$\begin{tikzcd}
 & Q(\mathbb{G}) \arrow[d, "Q(f)"] \\
Q(\mathbb{H}) \arrow[r, "Q(g)"'] & Q(\mathbb{K})
\end{tikzcd}$$

and let $(\mathcal{P}, p_{Q(f)}, p_{Q(g)})$ be the pullback. We want to show that, if $(\mathbb{P}, p_f, p_g)$ is the pullback of the arrows f and g in **EqGrph**, then $Q(\mathbb{P}) = \mathcal{P}$, $Q(p_f) = p_{Q(f)}$ and $Q(p_g) = p_{Q(g)}$. But, since Q preserves limits, $Q(\mathbb{P}) = \mathcal{P}$ (and so the arrows). Hence, from Corollary 2.2.7, we can conclude that Q creates pullbacks. □

Lemma 2.2.11. *The quotient functor creates pushouts along regular monomorphism.*

Theorem 2.2.12. EqGrph *is quasiadhesive.*

Proof. In order to apply Theorem 1.4.5, we can consider the quotient functor defined in Definition 2.2.5 Q : **EqGrph** $\to$ **Graph**. We note that Q creates limits, and that regular monos in **EqGrph** are mapped onto monos in **Graph**. In addition to Lemma 2.2.9, we can conclude that **EqGrph** is $\mathcal{R}eg$(**EqGrph**)-adhesive. □

2.3 E-Graphs

COSE DA FARE: Questa sezione ha più o meno gli stessi problemi della precedente. L'ordine da rispettare imho è il seguente: >Definizione >Calcolo dei limiti e certi colimiti (si fanno come in EqGrph) > cor del calcolo dei limiti: caratterizzare mono regolari >I crea limiti e i giusti pushout > e-graph sono quasiadesivi

E-Graphs are a particular type of graphs with equivalences, in which holds that

$$\frac{s(e) \sim s(e')}{t(e) \sim t(e')}$$

for each pair of edges e, e' of $\mathbb{G} = (G, \sim)$. In a more general case, considering a graph with equivalence as a functor $\mathbb{G} : (E \rightrightarrows V \to Q) \to$ **Set**, the inference rule above rewrites as

$$\frac{\mathbb{G}(q \circ s)(e) = \mathbb{G}(q \circ s)(e')}{\mathbb{G}(q \circ t)(e) = \mathbb{G}(q \circ t)(e')}$$

for each e, $e' \in \mathbb{G}(E)$. But this is to say that, given the two set $S = \{(e, e') \in \mathbb{G}(E) \times \mathbb{G}(E) \mid \mathbb{G}(q \circ s)(e) = \mathbb{G}(q \circ s)(e')\}$ and $T = \{(e, e') \in \mathbb{G}(E) \times \mathbb{G}(E) \mid \mathbb{G}(q \circ t)(e) = \mathbb{G}(q \circ t)(e')\}$, $S \subseteq T$. But S (with the projection arrows p_1 and p_2) is exactly the pullback of $(q \circ s, q \circ s)$, and T (together with the projections q_1, q_2) is the pullback of $(q \circ t, q \circ t)$. Then, a more general way to express that $\mathbb{G}$ is an e-graph is by saying that $\mathbb{G}$ is such that there exists a monomorphism, which is the canonical inclusion, in **Set** from S to T. To find a structure to express this fact, we have to consider a more general case.

Consider an arrow $f : X \to Y$, and let (K, π_1, π_2) be the pullback of (f, f).

$$\begin{tikzcd} K \arrow[r, "\pi_1"] \arrow[d, "\pi_2"'] & X \arrow[d, "f"] \\ X \arrow[r, "f"'] & Y \end{tikzcd}$$

Such pullback induces an arrow $\langle \pi_1, \pi_2 \rangle : K \to X \times X$. Such arrow is mono and unique, because of the universal property of the pullback, indeed a subobject.

capire quanto andare nello specifico qui

Since both S and T are subobjects of $\mathbb{G}(E) \times \mathbb{G}(E)$, we can make use of the Proposition 1.1.9. Specifically, we want, in the following situation, i to be mono and unique.

$$\begin{tikzcd} S \arrow[rr, "i"] \arrow[dr, "\langle p_1, p_2 \rangle"'] & & T \arrow[dl, "\langle q_1, q_2 \rangle"] \\ & \mathbb{G}(E) \times \mathbb{G}(E) & \end{tikzcd}$$

We have then that $\langle p_1, p_2 \rangle$ is mono, then so is $\langle q_1, q_2 \rangle \circ i$. From Proposition 1.1.14, we can conclude that i is mono too. The uniqueness follows from Proposition 1.1.9. If such i exists, then $\mathbb{G}$ is an e-graph.

Definition 2.3.1 (Category of E-Graphs)**.** The full subcategory of **EqGrph** whose objects are this particular kind of graphs is denoted as **EGG**.

Proposition 2.3.2. *The product of two e-graphs in* **EqGrph** *is an e-graph.*

Proof. Let $\mathbb{G}$, $\mathbb{H}$ be two e-graphs in **EqGrph**. Then, we want to to show that $\mathbb{G} \times \mathbb{H}$ is an e-graph too. The argument lies on the consideration that limits in presheaves categories are computed pointwise. In fact, we can □

Consider now the inclusion functor $I : \mathbf{EGG} \to \mathbf{EqGrph}$. Since **EGG** is a full subcategory of **EqGrph**, I is full and faithful (Example 1.2.8), it reflects all limits (Proposition 1.3.29). But limits are also preserved, since the limit in **EqGrph** in which objects are e-graphs is an e-graph together with morphisms that are also morphisms of **EGG** since it is a full subcategory. Then, we can conclude as follows.

Lemma 2.3.3. *The inclusion functor* $I : \mathbf{EGG} \to \mathbf{EqGrph}$ *creates limits.*

Proof. To prove that I creates limits, we have to show that both preserves and reflects limits. To see that I preserves limits, it is sufficient to note that a limit of e-graphs in **EqGrph** is again an e-graph, together with morphisms. (Note that, since **EGG** is a full subcategory of **EqGrph**, these morphisms in **EqGrph** are morphisms of **EGG** too). □

Since I is faithful, monomorphisms in **EqGrph** between graphs that are e-graphs too are monomorphisms in **EGG** too. Regular monomorphisms in **EGG** are, as in **EqGrph**, monomorphisms which reflect equivalences, hence natural transformations with all the three components mono (**??**). The following result follows by the fact that a full and faithful functor preserves equalizers. ???? Da dimostrare

Proposition 2.3.4. *Let* I *be the inclusion functor from* **EGG** *to* **EqGrph**. *Then,* $I(\mathcal{R}eg(\mathbf{EGG})) \subseteq \mathcal{R}eg(\mathbf{EqGrph})$.

At this point, by direct application of Theorem 1.4.5, it is possible to state what follows.

Corollary 2.3.5. **EGG** *is quasiadhesive.*

Appendix A

Omitted Proofs

Theorem 1.3.21. Let $\mathscr{C}$ be a category. Then $\mathscr{C}$ has all finite limits if and only if $\mathscr{C}$ has all finite products and all equalizers.

Proof. Let $D : \mathscr{I} \to \mathscr{C}$ a diagram, with $\mathscr{I}$ finite.

The *if* statement follows from definitions of products and equalizers (Definition 1.3.16, Definition 1.3.12)

To satisfy the *only if* statement, we want an object L together with morphisms $p_i : L \to D(j)$ such that:

1. $\{p_i : L \to D(i)\}$ is a cone – i.e., for each morphism of $\mathscr{I}$ $\alpha : i \to j$, $D(\alpha) \circ p_i = p_j$; and

2. for each E and $q_i : E \to D(j)$ in $\mathscr{C}$, with $D(\alpha) \circ q_i = q_j$ for each $\alpha : i \to j$ of $\mathscr{I}$, there exists a unique $f : E \to L$ such that $q_i = p_i \circ f$ for each $i \in \mathcal{O}b(\mathscr{I})$.

Consider the two products (which exist by hypothesis) $\prod_{j \in \mathcal{O}b(\mathscr{I})} D(j)$, the product of the objects of the diagram, and $\prod_{\alpha \in \mathcal{H}om(\mathscr{I})} D(cod\ \alpha)$, the product of the codomains of the morphisms in D, where π_x is the x-th projection of the product. Let now:

$$\gamma, \varepsilon : \prod_{j \in \mathcal{O}b(\mathscr{I})} D(j) \longrightarrow \prod_{\alpha \in \mathcal{H}om(\mathscr{I})} D(cod\ \alpha)$$

be defined by $\gamma_\alpha = \pi_{D(cod\ \alpha)}$ (the projection on the codomain of α) and $\varepsilon_\alpha = D(\alpha) \circ \pi_{D(dom\ \alpha)}$. Let now $e : L \to \prod_{j \in \mathcal{O}b(\mathscr{I})} D(j)$ the equalizer of γ and ε (which exists by hypothesis), and, for each $j \in \mathcal{O}b(\mathscr{I})$, $p_j : L \to D(j)$, defined by $p_j = \pi_{D(j)} \circ e$.

What we want now is to show that $(L, (p_i))_{i \in \mathscr{I}}$ is the limit of D, namely, to prove that the conditions given at the beginning are valid.

For condition 1, we have to show that, for each $\alpha : i \to j$ of $\mathscr{I}$, we have $D(\alpha) \circ p_i = p_j$:

$$\begin{aligned}
D(\alpha) \circ p_i &= D(\alpha) \circ \pi_{D(i)} \circ e && \textit{Definition of } p_j \\
&= \varepsilon_\alpha \circ e && \textit{Definition of } \varepsilon \\
&= \gamma_\alpha \circ e && e \textit{ is an equalizer of } \pi, \varepsilon \\
&= \pi_{D(j)} \circ e && \textit{Definition of } \pi \\
&= p_j && \textit{Definition of } p_j
\end{aligned}$$

For condition 2, suppose that $(E, (q_i)_{i \in \mathcal{O}b(\mathscr{I})})$ has the properties stated. By definition of product, there exists a (unique) arrow $q : E \to \prod_{j \in \mathcal{O}b(\mathscr{I})} D(j)$. For each arrow $\alpha : i \to j$, we have:

$$\begin{aligned}
\gamma_\alpha \circ q &= \pi_{D(j)} \circ q && \textit{Definition of } \pi \\
&= q_j && \textit{Definition of } q_j \\
&= D(\alpha) \circ q_i && \textit{Assumption on } q_j \\
&= D(\alpha) \circ \pi_{D(j)} \circ q && \textit{Definition of } q_i \\
&= \varepsilon_\alpha \circ q && \textit{Definition of } \varepsilon
\end{aligned}$$

Since e equalizes π and ε, there exists a unique $f : E \to L$ in $\mathscr{C}$ such that $q = e \circ f$. Then, for each $j \in \mathcal{O}b(\mathscr{I})$, we have $\pi_{D(j)} \circ q = \pi_{D(j)} \circ e \circ f$, hence, $q_i = p_i \circ f$. $\square$

Bibliography

[AHS09] J. Adámek, Horst Herrlich, and George E. Strecker. *Abstract and concrete categories: The joy of cats.* Dover Publications, 2009.

[BW95] Michael Barr and Charles Wells. *Category theory for computing science.* Prentice Hall, 2 edition, 1995.

[CGM22] Davide Castelnovo, Fabio Gadducci, and Marino Miculan. A new criterion for $\mathcal{M},\mathcal{N}$-adhesivity, with an application to hierarchical graphs, 2022.

[HS79] Horst Herrlich and George E. Strecker. *Category Theory*, volume 1 of *Sigma Series in Pure Mathematics.* Heldermann Verlag Berlin, 2 edition, 1979.

[Lac11] Stephen Lack. An embedding theorem for adhesive categories, 2011.