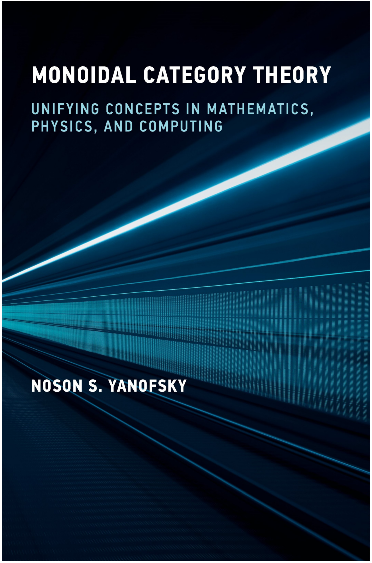
Monoidal Category Theory: Unifying concepts in Mathematics, Physics, and Computing



MONOIDAL CATEGORY THEORY

UNIFYING CONCEPTS IN MATHEMATICS,
PHYSICS, AND COMPUTING

NOSON S. YANOFSKY

Chapter 3:

Structures Within Categories

Foreshadowing

- Chapter 3: Structures Within Categories
  - Section 3.1: Products and Coproducts
  - Section 3.2: Limits and Colimits
  - Section 3.3: Slices and Coslices
  - Section 3.4: Mini-course: Self-Referential Paradoxes

Foreshadowing

- Chapter 3: Structures Within Categories
  - Section 3.1: Products and Coproducts
    - Products
    - Coproducts

- We motivate the notion of a product in a category.
- We define a product and look at its properties.
- We describe duality and come to the notion of a coproduct with its properties.

Foreshadowing

A category may be a lot more than simply a collection of objects and morphisms between objects. Rather, there might be various relationships between certain objects and morphisms. These relationships can make certain objects have important properties. In this chapter we describe many such relationships. We also describe operations one can perform with the objects and morphisms in a category.

Motivating examples

- We begin with one of the simplest operations one can perform with the objects of a category.
- Given objects a and b in a category, one can sometimes form their product $a \times b$.
- The product means different things in different categories.
- While a product is one of the simplest types of structure, the ideas in this section arise over and over again in the rest of this book.

Motivating examples

Before we come to the formal definition of what we mean by a product in an arbitrary category, let us carefully look at the category of sets and examine the idea of a Cartesian product of sets.

Example

- *Let* $S = \{x, y, z\}$ *and* $T = \{0, 1\}$.
- *The Cartesian product is defined as the set of pairs of elements where the first element is from* S *and the second element is from* T.
- $$S \times T = \{(x, 0), (y, 0), (z, 0), (x, 1), (y, 1), (z, 1)\}.$$
- *The set* $S \times T$ *contains the information of both* S *and* T.
- *There are functions* $\pi_S \colon S \times T \longrightarrow S$ *and* $\pi_T \colon S \times T \longrightarrow T$ *that "forgets" one of the elements of the pair. These functions are called* **projection functions**.
- *They are defined on the elements as in the next slide.*

Motivating examples

The Cartesian product of two sets and its projection functions.

Example (Continued)

$$\begin{tikzcd}
& S \times T \arrow[dl, "\pi_S"'] \arrow[dr, "\pi_T"] & \\
S & & T \\
x & (x,0) \arrow[l, mapsto, "\pi_S"'] \arrow[r, mapsto, "\pi_T"] & 0 \\
& (y,0) \arrow[dl, mapsto, "\pi_S"'] \arrow[ur, mapsto, "\pi_T"] & \\
y & (z,0) \arrow[ddl, mapsto, "\pi_S"'] \arrow[uur, mapsto, "\pi_T"] & 1 \\
& (x,1) \arrow[uuul, mapsto, "\pi_S"'] \arrow[ur, mapsto, "\pi_T"] & \\
z & (y,1) \arrow[uul, mapsto, "\pi_S"'] \arrow[uur, mapsto, "\pi_T"] & \\
& (z,1). \arrow[ul, mapsto, "\pi_S"'] \arrow[uuur, mapsto, "\pi_T"] &
\end{tikzcd}$$

Motivating examples

How does the Cartesian product of two sets and the projections relate to other sets? Let us look at three examples of other sets and morphisms. All three examples are depicted by the following diagram

$$\begin{tikzcd}
 & V \arrow[ddl, "k_S"'] \arrow[d, "h"', "\exists!"] \arrow[ddr, "k_T"] & \\
 & S \times T \arrow[dl, "\pi_S"] \arrow[dr, "\pi_T"'] & \\
S & & T.
\end{tikzcd}$$

Motivating examples

Example

- *Consider the set* $V = \{(y, 1), (z, 0), (z, 1)\}$.
- *This set has some — but not all — pairs of elements from* S *and* T.
- *There are also projection maps* $k_S \colon V \longrightarrow S$ *and* $k_T \colon V \longrightarrow T$ *which have the same values as* π_S *and* π_T *on those pairs.*
- *While* V *feels like a product, it is not.* V *is a subset of* $S \times T$ *and there is an inclusion function* $h \colon V \longrightarrow S \times T$ *which makes the above diagram commute.*
- *The inclusion function is unique.*
- *From this vantage point, think of* $S \times T$ *as the set of pairs that is the largest or "best fitting" set of pairs.*

Motivating examples

Example

- *Consider the set* $V = \{a, b, c, d, e\}$ *and two functions* $k_S \colon V \longrightarrow S$ *and* $k_T \colon V \longrightarrow T$.
- *For each element of* V, *these two functions pick an element in* S *and an element in* T.
- *For example,* $k_S(c) = z$ *and* $k_T(c) = 0$. *With* k_S *and* k_T *one can make a new function* $h \colon V \longrightarrow S \times T$ *that uses the information of* k_S *and* k_T.
- *This new function associate* c *with* $(z, 0)$ *in* $S \times T$.
- *That is, there is a unique implied function* $h \colon V \longrightarrow S \times T$ *that makes the above diagram commute.*

Motivating examples

Example

- *In the definition of* $S \times T$ *we used the notation* $(x, 1)$.
- *While this is common, other notations could also be used. For example, there is* $[x, 1]$ *or* $\langle x, 1\rangle$ *or even* $\{x, 1, \{x\}\}$.
- *If one was to use this last example as a pair of elements, then the set of all pairs of elements from* S *and* T *would look like* V

  $$\{\{x, 0, \{x\}\}, \{y, 0, \{y\}\}, \{z, 0, \{z\}\}, \{x, 1, \{x\}\}, \{y, 1, \{y\}\}, \{z, 1, \{z\}\}\}.$$

- *Such a set* V *also has two projection functions* $k_S\colon V \longrightarrow S$ *and* $k_T\colon V \longrightarrow T$. *They are defined, for example, as* $k_S(\{y, 1, \{y\}\}) = y$.
- *While* $S \times T$ *has the same number of elements as* V *they are not the same sets. There is an obvious unique* $h\colon V \longrightarrow S \times T$ *that takes* $\{x, 1, \{x\}\}$ *to* $(x, 1)$. *In this case,* h *is a bijection (isomorphism).*

Motivating examples

Definition

- *The point of these three examples is that if there is a V and functions k_S and k_T, then there is a unique function h making the appropriate diagram commute.*
- *This property that $S \times T$ and its projection maps, π_S and π_T, have is called a* **universal property**.
- *It is a way of saying that $S \times T$ is the best set that has the information of S and T and maps to S and T.*
- *This property of the product is satisfied by all the objects and maps of the category that fit into the diagram above.*

Motivating examples

Example

Given a set of real numbers, say

$$S = \left\{ \frac{1}{2}, \frac{2}{3}, \frac{3}{4}, \ldots, \frac{n}{n+1}, \ldots \right\}$$

one can ask for the **greatest lower bound** *or the* **infimum** *of this set. This is a number X that is*

- *a lower bound, i.e, a number, X, that is less than or equal to all the numbers in S, and*
- *it is greatest of all the lower bounds, i.e., this number X is greater than (or equal to) all the lower bounds.*

It is not hard to see that $X = \frac{1}{2}$ is the greatest lower bound of the set S.

Motivating examples

Example (Continued)

Within the partial order of real numbers, a greatest lower bound corresponds to the fact that there is

- *a morphism* $X \longrightarrow s$ *for all* $s \in S$*, and*
- *any number* Y *that has a morphism* $Y \longrightarrow s$ *for all* $s \in S$ *will also have a morphism* $Y \longrightarrow X$*.*

Another way to say this is that X *is*

- *a lower bound, and*
- *the "best fitting" lower bound.*

Motivating examples

Example (Continued)

A product is similar to a greatest lower bound, but it pertains to an arbitrary category and not just to a partial order category. A product is

- *an object with projections, and*
- *it is the "best fitting" object with projections, i.e., if there is any other object with projections, then there is a unique morphism from the other object to the product.*

Definition of a product

$$\begin{tikzcd} a & & b. \end{tikzcd}$$

Definition

- *Let $\mathbb{A}$ be a category with objects a and b.*

Definition of a product

$$\begin{tikzcd}
& a \times b \arrow[dl, "\pi_a"'] \arrow[dr, "\pi_b"] & \\
a & & b.
\end{tikzcd}$$

Definition

- *Let* $\mathbb{A}$ *be a category with objects* a *and* b.
- *A* **product** *of* a *and* b *is an object* $a \times b$ *with* **projection morphisms** π_a *and* π_b.

Definition of a product

$$\begin{tikzcd}
& & c \arrow[lldd, "k_a"'] \arrow[rrdd, "k_b"] & & \\
& & a \times b \arrow[lld, "\pi_a"] \arrow[rrd, "\pi_b"'] & & \\
a & & & & b.
\end{tikzcd}$$

Definition

- *Let $\mathbb{A}$ be a category with objects a and b.*
- *A* **product** *of a and b is an object $a \times b$ with* **projection morphisms** *π_a and π_b.*
- *These satisfy the following universal property: for every object c and any two maps k_a and k_b,*

Definition of a product

$$\begin{tikzcd}
 & c \arrow[ddl, "k_a"'] \arrow[d, "h"', "\exists!"] \arrow[ddr, "k_b"] & \\
 & a \times b \arrow[dl, "\pi_a"] \arrow[dr, "\pi_b"'] & \\
a & & b.
\end{tikzcd}$$

Definition

- *Let* $\mathbb{A}$ *be a category with objects* a *and* b.
- *A* **product** *of* a *and* b *is an object* $a \times b$ *with* **projection morphisms** π_a *and* π_b.
- *These satisfy the following universal property: for every object* c *and any two maps* k_a *and* k_b,
- *there exists a unique map* $h\colon c \longrightarrow a \times b$ *which makes both triangles commute.*

Definition of a product

$$\begin{tikzcd}
 & c \arrow[ddl, "k_a"'] \arrow[d, "h"', "\exists!"] \arrow[ddr, "k_b"] & \\
 & a \times b \arrow[dl, "\pi_a"] \arrow[dr, "\pi_b"'] & \\
a & & b.
\end{tikzcd}$$

Definition

- *Let* $\mathbb{A}$ *be a category with objects* a *and* b*.*
- *A* ***product*** *of* a *and* b *is an object* $a \times b$ *with* ***projection morphisms*** π_a *and* π_b*.*
- *These satisfy the following universal property: for every object* c *and any two maps* k_a *and* k_b*,*
- *there exists a unique map* $h\colon c \longrightarrow a \times b$ *which makes both triangles commute.*

We call h the **induced** morphism of k_a and k_b and we write it as $h = \langle k_a, k_b \rangle$. We write the h with the $\exists!$ to stress that h is a unique such map that satisfies the universal property.

Examples of a product

Let us look at some examples.

Example

In $\mathbb{Set}$ the product of sets S and T is the Cartesian product of sets.

Example

In a partial order $(P, \leq)$, the product of two elements, p and q is the **meet**. *This is an element $p \wedge q$ such that*

- *$p \wedge q \leq p$ and $p \wedge q \leq q$, and furthermore*
- *if there is any other element c such that $c \leq p$ and $c \leq q$, then $c \leq p \wedge q$.*

Notice that $p \wedge q$ is exactly the greatest lower bound of p and q. If $(P, \leq)$ is a total order than $p \wedge q$ is the minimum of p and q.

Example

In $\mathbb{CompFunc}$, the category of computable functions, the objects

Examples of a product

Example

In the category of graphs, the product of two graphs G *and* H *is a graph whose vertices are pairs of vertices from* G *and* H *and there is an edge from* (g, h) *to* (g', h') *if there is an edge from* g *to* g' *in* G *and an edge from* h *to* h' *in* H*. It is not hard to see that the projection functions are graph homomorphisms.*

Properties of a product

- The property is universal because it characterizes being a product within the *whole* category.
- Be aware of what was done in this definition. We defined a special object in the category by not looking at what was in the objects, but by examining the relationship of that object with all the objects in the category. (This is an Important Categorical Idea.
- Notice, a category $\mathbb{A}$ with objects a and b might not have an object that satisfies the above universal property. In that case we say that the “product does not exist.” Within a category where the product always exists, we say that the category “has binary products.”
- Notice that we defined “a” product rather than “the” product. There might be more than one product for any two objects. This is not a problem because the next theorem.

Properties of a product

$$\begin{tikzcd}
a & & b
\end{tikzcd}$$

Theorem

- *Consider objects a and b.*

Properties of a product

$$\begin{tikzcd}
 & c \arrow[dl, "\pi_a"'] \arrow[dr, "\pi_b"] & \\
a & & b
\end{tikzcd}$$

Theorem

- *Consider objects a and b.*
- *Assume that there exists an object c and projection maps π_a and π_b so that c is a product of a and b.*

Properties of a product

$$\begin{tikzcd}
 & c \arrow[dl, "\pi_a"'] \arrow[dr, "\pi_b"] & \\
a & & b \\
 & c' \arrow[ul, "\pi'_a"] \arrow[ur, "\pi'_b"'] &
\end{tikzcd}$$

Theorem

- *Consider objects a and b.*
- *Assume that there exists an object c and projection maps π_a and π_b so that c is a product of a and b.*
- *Furthermore, assume that there also exists an object c', and projection maps π'_a and π'_b so that c' is a product of a and b.*

Properties of a product

$$\begin{tikzcd}
 & c \arrow[dl, "\pi_a"'] \arrow[dr, "\pi_b"] \arrow[dd, "\exists!"', "\cong"] & \\
a & & b \\
 & c' \arrow[ul, "\pi'_a"] \arrow[ur, "\pi'_b"'] &
\end{tikzcd}$$

Theorem

- *Consider objects* a *and* b.
- *Assume that there exists an object* c *and projection maps* π_a *and* π_b *so that* c *is a product of* a *and* b.
- *Furthermore, assume that there also exists an object* c', *and projection maps* π'_a *and* π'_b *so that* c' *is a product of* a *and* b.
- *Then there is a unique isomorphism from* c *to* c' *that commutes with the projections.*

Properties of a product

$$\begin{tikzcd}
 & c \arrow[ld, "\pi_a"'] \arrow[rd, "\pi_b"] \arrow[dd, "\exists!"', "\cong"] & \\
a & & b \\
 & c' \arrow[lu, "\pi'_a"] \arrow[ru, "\pi'_b"'] &
\end{tikzcd}$$

Theorem

- *Consider objects a and b.*
- *Assume that there exists an object c and projection maps π_a and π_b so that c is a product of a and b.*
- *Furthermore, assume that there also exists an object c', and projection maps π'_a and π'_b so that c' is a product of a and b.*
- *Then there is a unique isomorphism from c to c' that commutes with the projections.*

Properties of a product

- This means the product is unique **up to a unique isomorphism**.
- The proof is very similar to the proof that initial objects are unique up to a unique isomorphism.

Properties of a product

From the definition of the product $a \times b$, for any object c and any pair of morphisms $k_a \colon c \longrightarrow a$ and $k_b \colon c \longrightarrow b$, there is an induced morphism $\langle k_a, k_b \rangle \colon c \longrightarrow a \times b$. This gives us an isomorphism

$$\begin{tikzcd}
Hom_{\mathbb{A}}(c, a \times b) \arrow[rr, shift left, "\pi_a \circ (\ \ )\ ,\ \pi_b \circ (\ \ )"] & & Hom_{\mathbb{A}}(c, a) \times Hom_{\mathbb{A}}(c, b). \arrow[ll, shift left, "\langle\ ,\ \rangle"]
\end{tikzcd}$$

Properties of a product

This isomorphism of Hom sets can be taken as a the definition of a product. To put it formally:

Definition

Let $\mathbb{A}$ be a category with objects a and b. A **product** *of a and b is an object $a \times b$ such that there exists an isomorphism of Hom sets:*

$$Hom_{\mathbb{A}}(c, a \times b) \cong Hom_{\mathbb{A}}(c, a) \times Hom_{\mathbb{A}}(c, b).$$

(Notice that the $\times$ in the left is a product of the category and the $\times$ on the right is the Cartesian product of Hom sets.) We need one more requirement, namely this isomorphism should be **natural***. We will spell out this requirement later.*

Examples of a product

Example

In the partial order $(P, \leq)$, the Hom sets are either the empty set or a set with one element.

$$Hom_P(p, q \wedge r) \cong Hom_P(p, q) \times Hom_P(p, r)$$

The left side is a one-element set if and only if each part of the right side is a one-element set.

Example

In $\mathbb{PO}$, let $(P, \leq)$ and $(Q, \leq)$ be partial orders. Then $(P \times Q, \sqsubseteq)$ is the partial order whose elements are ordered pairs of elements from P and Q and whose order is defined as follows:

$$(p, q) \sqsubseteq (p', q') \textit{ if and only if } p \leq p' \textit{ and } q \leq q'$$

It is not hard to show that this is a partial order. This partial order is

Examples of a product

Example

In $\mathbb{Top}$, if (X, τ) and (Y, σ) are topological spaces, then $(X \times Y, \delta)$ is their product where $X \times Y$ is the Cartesian product of X and Y and δ is the **product topology** *(see any introductory topology textbook).*

Example

In $\mathbb{Group}$, if $(G, \star, 1, (\ )^{-1})$ and $(G', \star', 1', [\ ]^{-1})$ are groups, then their product is $(G \times G', \cdot, (1, 1'), ((\ )^{-1}, [\ ]^{-1}))$. The elements are pairs of elements (x, x') where x is in G and x' is in G'. The multiplication $\cdot$ is given pointwise, i.e., on each component:

$$(x, x') \cdot (y, y') = (x \star y, x' \star' y').$$

The unit of the group is $(1, 1')$. The inverse of (x, x') is $((x)^{-1}, [x']^{-1})$.

Properties of a product

Definition

In any category, consider the special case of taking a product of an object a with itself, i.e., $a \times a$. The induced morphism for $\langle id_a, id_a \rangle$ as in

$$\begin{tikzcd}
& a \arrow[ddl, "id_a"'] \arrow[d, "\Delta"'] \arrow[ddr, "id_a"] & \\
& a \times a \arrow[dl, "\pi_a"] \arrow[dr, "\pi_a"'] & \\
a & & a
\end{tikzcd}$$

is denoted as Δ (Greek letter delta) and is called the **diagonal morphism**.

Such morphisms will be very important in the rest of the course.

Properties of a product

Example

- *In $\mathbb{S}\mathrm{et}$, the diagonal morphism $\Delta\colon S \longrightarrow S \times S$ takes each element s in S to (s, s) in $S \times S$.*
- *In $\mathbb{G}\mathrm{roup}$, the diagonal morphism $\Delta\colon G \longrightarrow G \times G$ is a group homomorphism that is defined as $\Delta(g) = (g, g)$. To see that this is a homomorphism, consider*

  $$\Delta(x \star y) = (x \star y, x \star y) = (x, x) \star' (y, y) = \Delta(x) \star' \Delta(y)$$

  where $\star'$ is the multiplication in $G \times G$.
- *In $\mathbb{T}\mathrm{op}$, Δ works the same way it works in $\mathbb{S}\mathrm{et}$.*

Properties of a product

.

Definition

Properties of a product

$$\begin{tikzcd} a & a \times b \arrow[l, "\pi_a"'] \arrow[r, "\pi_b"] & b \end{tikzcd}$$

.

Definition

- *Let $a \times b$ be a product of a and b.*

Properties of a product

$$\begin{tikzcd}
a & a \times b \arrow[l, "\pi_a"'] \arrow[r, "\pi_b"] & b
\end{tikzcd}$$

$$\begin{tikzcd}
a' & a' \times b' \arrow[l, "\pi'_a"] \arrow[r, "\pi'_b"'] & b'.
\end{tikzcd}$$

Definition

- *Let $a \times b$ be a product of a and b.*
- *Let $a' \times b'$ be a product of a' and b'.*

Properties of a product

$$\begin{tikzcd}
a \arrow[d, "f"'] & a \times b \arrow[l, "\pi_a"'] \arrow[r, "\pi_b"] & b \arrow[d, "g"] \\
a' & a' \times b' \arrow[l, "\pi'_a"] \arrow[r, "\pi'_b"'] & b'.
\end{tikzcd}$$

Definition

- *Let* $a \times b$ *be a product of* a *and* b.
- *Let* $a' \times b'$ *be a product of* a' *and* b'.
- *Consider morphisms* $f\colon a \longrightarrow a'$ *and* $g\colon b \longrightarrow b'$.

Properties of a product

$$\begin{tikzcd}
a \arrow[d, "f"'] & a \times b \arrow[l, "\pi_a"'] \arrow[r, "\pi_b"] \arrow[d, "f{\times}g"'] & b \arrow[d, "g"] \\
a' & a' \times b' \arrow[l, "\pi'_a"] \arrow[r, "\pi'_b"'] & b'.
\end{tikzcd}$$

Definition

- *Let $a \times b$ be a product of a and b.*
- *Let $a' \times b'$ be a product of a' and b'.*
- *Consider morphisms $f \colon a \longrightarrow a'$ and $g \colon b \longrightarrow b'$.*
- *Since $f \circ \pi_a$ and $g \circ \pi_b$ satisfy the universal property of $a' \times b'$, there is an induced map.*

Properties of a product

$$\begin{tikzcd}
a & a \times b \arrow[l, "\pi_a"'] \arrow[r, "\pi_b"] \arrow[d, "f\times g"'] & b \arrow[d, "g"] \\
a' \arrow[from=u, "f"'] & a' \times b' \arrow[l, "\pi'_a"] \arrow[r, "\pi'_b"'] & b'.
\end{tikzcd}$$

Definition

- *Let* $a \times b$ *be a product of* a *and* b.
- *Let* $a' \times b'$ *be a product of* a' *and* b'.
- *Consider morphisms* $f : a \longrightarrow a'$ *and* $g : b \longrightarrow b'$.
- *Since* $f \circ \pi_a$ *and* $g \circ \pi_b$ *satisfy the universal property of* $a' \times b'$, *there is an induced map.*
- *This map is denoted* $f \times g$ *and is called the* **product** *of* f *and* g *and* $f \times g = \langle f \circ \pi_a, g \circ \pi_b \rangle$.

Properties of a product

There are two operations of morphisms: composition and taking the product. How do these two operations respect each other?

$$\begin{tikzcd}
& d \arrow[d, "h"] \arrow[dddl, "fh"'] \arrow[dddr, "gh"] \arrow[dd, bend left=60, "{\langle fh,gh\rangle}"] & \\
& c \arrow[d, "{\langle f,g\rangle}"] \arrow[ddl, "f"'] \arrow[ddr, "g"] & \\
& a \times b \arrow[dl, "\pi_a"'] \arrow[dr, "\pi_b"] & \\
a & & b.
\end{tikzcd}$$

Theorem

- *Let $a \times b$ be the product of a and b.*
- *Consider f, and g*
- *This induces $\langle f, g\rangle$.*
- *Consider h.*
- *This induces fh and gh.*
- *This induces $\langle fh, gh\rangle$*
- *We have $\langle f \circ h, g \circ h\rangle = \langle f, g\rangle \circ h$.*

That is, the composition with *h* on the right distributes over the entries in the bracket.

Properties of a product

There are many consequences of this theorem.

Theorem

Let $f\colon a \longrightarrow a'$, $g\colon b \longrightarrow b'$ *and* $h\colon c \longrightarrow a \times b$. *Then we have*

$$(f \times g) \circ h = \langle f \circ \pi_a, g \circ \pi_b \rangle \circ h = \langle f \circ \pi_a \circ h, g \circ \pi_b \circ h \rangle.$$

Properties of a product

The following relationship between the × and ∘ operations will be very similar to other relationships between binary operations we will find throughout this text. It is an instance of something called the **interchange law** and is written like this

$$(f' \times g') \circ (f \times g) = (f' \circ f) \times (g' \circ g) \colon a \times b \longrightarrow a'' \times b''.$$

Properties of a product

Theorem

Let $f\colon a \longrightarrow a'$, $f'\colon a' \longrightarrow a''$, $g\colon b \longrightarrow b'$ *and* $g\colon b' \longrightarrow b''$. *Assume that* $a \times b$, $a' \times b'$, *and* $a'' \times b''$ *exist. Then the two curved lines are equal.*

$$\begin{tikzcd}
a \arrow[d, "f"'] & & a \times b \arrow[ll, "\pi"'] \arrow[rr, "\pi"] \arrow[d, "f\times g"] \arrow[dd, bend right=60, "(f'\circ f)\times(g'\circ g)"'] \arrow[dd, bend left=60, "(f'\times g')\circ(f\times g)"] & & b \arrow[d, "g"] \\
a' \arrow[d, "f'"'] & & a' \times b' \arrow[ll, "\pi"'] \arrow[rr, "\pi"] \arrow[d, "f'\times g'"] & & b' \arrow[d, "g'"] \\
a'' & & a'' \times b'' \arrow[ll, "\pi"] \arrow[rr, "\pi"'] & & b''.
\end{tikzcd}$$

Properties of a product

Let us spend a few words meditating on the interchange law.

- Think of the functions f, f', g, and g' as processes.
- Consider the composition operation, $\circ$, as doing one process after another (sequential processes) and consider the product operation, $\times$, as doing two processes independently (parallel processes).
- The interchange law (see Important Categorical Idea) tells us how sequential and parallel process get along.
- On the one hand, we can think of first performing the parallel process $f \times g$ and then sequentially performing the parallel process $f' \times g'$.
- On the other hand, we can think of parallel processing the compositions $f' \circ f$ and $g' \circ g$.
- The interchange law tells us that both ways of looking at these processes are correct.

Central Idea

Important Categorical Idea

The Interchange Law.

- *In general, consider when there are two binary operations, say $\otimes$ and $\oplus$, operating on a collection.*
- *If the two operations satisfies the equality*

  $$(a \otimes b) \oplus (c \otimes d) = (a \oplus c) \otimes (b \oplus d)$$

  then we call it an **interchange law** *or* **interchange rule***.*
- *It means that each operation respects the other operation, that is, each operation is a homomorphism in terms of the other operation.*
- *This idea arises many times in category theory and will be at the heart of the theory of monoidal categories.*

Properties of a product

Example

The interchange law is of interest in the category $\mathbb{CompFunc}$ *of computable functions. If* $f\colon T_1 \longrightarrow T_2$ *and* $g\colon T_3 \longrightarrow T_4$*, then one should think of* $f \times g\colon T_1 \times T_3 \longrightarrow T_2 \times T_4$ *as a type of parallel processing. The function* $f \times g$ *does each computation separately. In contrast, for composable maps* f *and* f'*, one can think of* $f' \circ f$ *as sequential processing since one has to preform one process and then the other.*

Properties of a product

We have seen the product of two objects. What about the product of three objects?

Theorem

Let a, b and c be three objects in a category. If the products $a \times (b \times c)$ *and* $(a \times b) \times c$ *exist, then there is a unique isomorphism* $\alpha \colon a \times (b \times c) \longrightarrow (a \times b) \times c$ *and an inverse* $\alpha' \colon (a \times b) \times c \longrightarrow a \times (b \times c)$ *such that* α *and* α' *each commute with all the projection maps as in this diagram:*

$$\begin{tikzcd}
 & a \times (b \times c) \arrow[rr, shift left, "\alpha"] \arrow[dl, "\pi"'] \arrow[d, "g"'] \arrow[drr, "\pi"' near start] & & (a \times b) \times c \arrow[ll, shift left, "\alpha'"] \arrow[dll, "\pi" near start] \arrow[d, "g'"] \arrow[dr, "\pi"] & \\
a & a \times b \arrow[l, "\pi"] \arrow[r, "\pi"'] & b & b \times c \arrow[l, "\pi"] \arrow[r, "\pi"'] & c.
\end{tikzcd}$$

Properties of a product

Example

In $\mathbb{S}\mathrm{et}$*, the associativity isomorphism for sets* S, T, *and* U *is* $\alpha\colon S \times (T \times U) \longrightarrow (S \times T) \times U$ *is defined for* $s \in S$, $t \in T$, *and* $u \in U$ *as*

$$\alpha((s,(t,u))) = ((s,t),u).$$

Properties of a product

In a category with products, for every two objects a and b, what is the relationship between $a \times b$ and $b \times a$?

Theorem

In a category with products, for every two objects a and b, there is a **braid morphism** $br\colon a \times b \longrightarrow b \times a$ *which is induced by projection maps as follows*

$$\begin{tikzcd}
 & a \times b \arrow[ddl, "\pi_b"'] \arrow[d, "br"'] \arrow[ddr, "\pi_a"] & \\
 & b \times a \arrow[dl, "\pi_b"] \arrow[dr, "\pi_a"'] & \\
b & & a.
\end{tikzcd}$$

This means that $br = \langle \pi_b^{ab}, \pi_a^{ab} \rangle$*. This map is an isomorphism which shows that* $a \times b$ *is isomorphic to* $b \times a$.

Properties of a product

Theorem

Let t be a terminal object in a category. If $a \times t$ exists, then it is isomorphic to a.

Proof.

The proof follows from the following commutative diagram:

$$\begin{tikzcd}
 & a \times t \arrow[ddl, "\pi_a"'] \arrow[d, "\pi_a"] \arrow[ddr, "!"] & \\
 & a \arrow[dl, "id_a"] \arrow[d, "\langle id_a, ! \rangle"] \arrow[dr, "!"'] & \\
a & a \times t \arrow[l, "\pi_a"] \arrow[r, "!"'] & t.
\end{tikzcd}$$

□

Central Idea

One of the amazing aspects of category theory is that if you have some idea or construction and you turn all the arrows around to go in the opposite direction, then you have a related construction.

Important Categorical Idea

Duality.

- *Many properties and structures in categories come in pairs.*
- *For a given definition of a structure, one can make another* **dual** *definition with the arrows going in the opposite direction.*
- *This idea is called* **duality** *and happens very often.*
- *If a structure is called "X," then the dual structure is called "coX."*

The **coproduct** is what happens when you turn around the arrows of a product.

Motivating examples

Example

First let us look at the coproduct in $\mathbb{Set}$ and see what we can learn about it.

- *Let $S = \{x, y, z\}$ and $T = \{0, 1\}$.*
- *The coproduct in $\mathbb{Set}$ is the disjoint union of these sets. In this example the coproduct is just the union of both sets*

$$S \amalg T = \{x, y, z, 0, 1\}.$$

- *There are two inclusion functions $inc_S\colon S \longrightarrow S \amalg T$ and $inc_T\colon T \longrightarrow S \amalg T$. (Note, these are in the opposite direction of the projection functions).*

Motivating examples

Example

How does the disjoint union relate to other sets? Let us look at three examples of other sets and functions. Keep the following diagram in mind.

$$\begin{tikzcd}
S \arrow[dr, "inc_S"] \arrow[ddr, "k_S"'] & & T \arrow[dl, "inc_T"'] \arrow[ddl, "k_T"] \\
& S \amalg T \arrow[d, "h"'] & \\
& V. &
\end{tikzcd}$$

Motivating examples

Example

- *Consider the set* $V = \{r, w, x, y, z, a, 0, 1, 2\}$. *The sets* S *and* T *are subsets of* V *and there are inclusion functions* $k_S \colon S \longrightarrow V$ *and* $k_T \colon T \longrightarrow V$. *While* V *has a feel of a disjoint union,* $S \amalg T$ *is the real disjoint union because it contains nothing besides* S *and* T. *There is an inclusion function* $h \colon S \amalg T \longrightarrow V$ *that makes the above triangles commute. From this vantage point, think of* $S \amalg T$ *as the smallest or "best fitting" set that contains* S *and* T. *If there is any other set that contains* S *and* T, *then* $S \amalg T$ *is the best and includes into it.*
- *Consider the set* $V = \{a, b, c, d, e\}$ *and two functions* $k_S \colon S \longrightarrow V$ *and* $k_T \colon T \longrightarrow V$. *Let us say* $k_S(x) = b$ *and* $k_T(0) = e$. *There is a function* $h \colon S \amalg T \longrightarrow V$ *that unites the information of* k_S *and* k_T. *Such a function has* $h(x) = b$ *and* $h(0) = e$. *The function* h *defined like this ensures that the two triangles commute.*

Motivating examples

Example

- *Consider the set* $V = \{x', y', z', 0', 1'\}$. *While this is not the disjoint union of* S *and* T, *there still are obvious functions* $k_S\colon S \longrightarrow V$ *and* $k_T\colon T \longrightarrow V$ *that take elements to their prime versions. There is an isomorphism of sets* $h\colon S \amalg T \longrightarrow V$ *that takes every element* $h(x) = x'$.

Coproduct

With these examples in mind, lets us define a coproduct in any category.

Definition

Let $\mathbb{A}$ be a category with objects a and b. A **coproduct** *of a and b is an object $a+b$ (also written $a \amalg b$) with morphisms*

$$\begin{tikzcd}
a \arrow[dr, "inc_a"] & & b \arrow[dl, "inc_b"'] \\
& a+b &
\end{tikzcd}$$

Coproduct

Definition (Continued.)

These maps satisfy the following universal property: if there is any other object c with two morphisms

$$\begin{tikzcd}
a \arrow[dr, "k_a"'] & & b \arrow[dl, "k_b"] \\
& c &
\end{tikzcd}$$

Coproduct

Definition (Continued.)

Then there exists a unique map $h\colon a+b \longrightarrow c$ *which makes the following two triangles commute*

$$\begin{tikzcd}
a \arrow[dr, "inc_a"] \arrow[ddr, "k_a"'] & & b \arrow[dl, "inc_b"'] \arrow[ddl, "k_b"] \\
& a+b \arrow[d, "h"'] & \\
& c. &
\end{tikzcd}$$

Coproduct

The morphisms $k_a\colon a \longrightarrow c$ and $k_b\colon b \longrightarrow c$ induce the morphism $a+b \longrightarrow c$. We will write this morphism as $[k_a, k_b]\colon a+b \longrightarrow c$. Notice that

$$[k_a, k_b] \circ inc_a = k_a \qquad \text{and} \qquad [k_a, k_b] \circ inc_b = k_b.$$

Coproduct

Example

As we showed just before the definition, in $\mathbb{Set}$ *the coproduct is simply the disjoint union.*

Example

In computable functions $\mathbb{CompFunc}$, *the coproduct of two types* T_1 *and* T_2 *is simply the disjoint union of* T_1 *and* T_2. *There are obvious inclusion functions. The universal property is also easy to see: if there is a computable function* $f_1 \colon T_1 \longrightarrow T'$ *and a computable function* $f_2 \colon T_2 \longrightarrow T'$ *then there is a computable function* $h \colon T_1 \amalg T_2 \longrightarrow T'$. *The function h depends on the input. If the input is of type* T_1, *then h executes* f_1 *on it. If the input is of type* T_2, *then h executes function* f_2 *on the input, i.e.,*

$$h(x) = \begin{cases} f_1(x) & \textit{if } x \textit{ is of type } T_1 \\ f_2(x) & \textit{if } x \textit{ is of type } T_2 \ . \end{cases}$$

Coproduct

Many facts about coproducts are similar to products.

- Coproducts are unique up to a unique isomorphism.
- If a category has the property that for any two objects a coproduct exists, then we say that the category "has binary coproducts".
- For the coproduct, the following two maps are inverse of each other

$$\begin{tikzcd} Hom_{\mathbb{A}}(a+b,c) \arrow[r, shift left, "(\ )\circ inc_a\ ,\ (\ )\circ inc_b"] & Hom_{\mathbb{A}}(a,c) \times Hom_{\mathbb{A}}(b,c) \arrow[l, shift left, "{[\ ,\ ]}"] \end{tikzcd}$$

  and hence the Hom sets are isomorphic.
- We can use this as an equivalent definition of a coproduct.

Coproduct

More facts about coproducts that are similar to products.

- There is a unique isomorphism

$$\alpha\colon a + (b + c) \longrightarrow (a + b) + c.$$

- There is a **codiagonal morphism**, ∇, and a **cobraid morphism**, *cobr* for the coproduct. They are given by the following two diagrams

$$\begin{tikzcd}
a \arrow[rrd, "inc_a"] \arrow[rrdd, "id_a"'] & & & & a \arrow[lld, "inc_a"'] \arrow[lldd, "id_a"] \\
& & a + a \arrow[d, "\nabla"'] & & \\
& & a & &
\end{tikzcd}
\qquad
\begin{tikzcd}
a \arrow[rrd, "inc_a"] \arrow[rrdd, "inc_a"'] & & & & b \arrow[lld, "inc_b"'] \arrow[lldd, "inc_b"] \\
& & a + b \arrow[d, "cobr"'] & & \\
& & b + a & &
\end{tikzcd}$$

Foreshadowing

- Chapter 3: Structures Within Categories
    - Section 3.2: Limits and Colimits
        - Equalizers and Coequalizer
        - Pullbacks and Pushouts
        - General Limits and Colimits

The Product and Coproduct Again

Limits and colimits are generalizations of products and coproducts. Before we move on to describe general limits, it is important to understand a product from a different perspective. One can think of a product of objects *a* and *b* in the category as **completeing**the diagram or a **completion** of a diagram where *a* and *b* are points of a diagram.

$$\begin{tikzcd}
a & & & & b
\end{tikzcd}$$

In other words, the product and the projection maps from the product to *a* and *b* are like a pot cover on *a* and *b*.

$$\begin{tikzcd}
 & a \times b \arrow[dl, "\pi_a"'] \arrow[dr, "\pi_b"] & \\
a & & b.
\end{tikzcd}$$

The Product and Coproduct Again

The universal property, says $a \times b$ is the “best” way to complete the diagram. That means if there is any other completion c of a and b with maps $k_a \colon c \longrightarrow a$ and $k_b \colon c \longrightarrow b$ then there is a unique map $h \colon c \longrightarrow a \times b$ making all the triangles in this diagram

$$\begin{tikzcd}
 & c \arrow[d, "h"', "\exists!"] \arrow[ddl, "k_a"'] \arrow[ddr, "k_b"] & \\
 & a \times b \arrow[dl, "\pi_a"] \arrow[dr, "\pi_b"'] & \\
a & & b.
\end{tikzcd}$$

commute.

The Product and Coproduct Again

In a similar way, we might say that $a \amalg b$ is the best way to complete the diagram “from the bottom”:

$$\begin{tikzcd}
a \arrow[dr, "inc_a"] \arrow[ddr, "k_a"'] & & b \arrow[dl, "inc_b"'] \arrow[ddl, "k_b"] \\
& a \amalg b \arrow[d, "h"', "\exists!"] & \\
& c. &
\end{tikzcd}$$

We say we are **cocompleting** the diagram or this is a **cocompletion** of the diagram.

The Product and Coproduct Again

- While binary products and coproducts are the completion of diagrams with two objects, limits and colimits are the completions and cocompletions of any type of diagram.
- For example, given a diagram with four objects a_1, a_2, a_3 and a_4,, we can talk of their product $a_1 \times a_2 \times a_3 \times a_4$. (See next slide.)
- It will have projection maps $\pi_1 \colon a_1 \times a_2 \times a_3 \times a_4 \longrightarrow a_1$, $\pi_2 \colon a_1 \times a_2 \times a_3 \times a_4 \longrightarrow a_2$, $\pi_3 \colon a_1 \times a_2 \times a_3 \times a_4 \longrightarrow a_3$, and $\pi_4 \colon a_1 \times a_2 \times a_3 \times a_4 \longrightarrow a_4$.
- This product satisfies the following universal property: if there exists an object c and morphisms $g_1 \colon c \longrightarrow a_1$, $g_2 \colon c \longrightarrow a_2$, $g_3 \colon c \longrightarrow a_3$ and $g_4 \colon c \longrightarrow a_4$, then there exists a unique $h \colon c \longrightarrow a_1 \times a_2 \times a_3 \times a_4$ such that the expected diagrams commute.
- In a similar way we can talk about the colimit of this diagram. We denote the colimit as $a_1 + a_2 + a_3 + a_4$.

The Product and Coproduct Again

$$\begin{tikzcd}
& & c \arrow[ddll] \arrow[ddl] \arrow[d] \arrow[ddr] \arrow[ddrr] & & \\
& & a_1 \times a_2 \times a_3 \times a_4 \arrow[dll] \arrow[dl] \arrow[dr] \arrow[drr] & & \\
a_1 \arrow[drr] \arrow[ddrr] & a_2 \arrow[dr] \arrow[ddr] & & a_3 \arrow[dl] \arrow[ddl] & a_4 \arrow[dll] \arrow[ddll] \\
& & a_1 + a_2 + a_3 + a_4 \arrow[d] & & \\
& & d & &
\end{tikzcd}$$

The limit and colimit of four objects.

The Product and Coproduct Again

In the same way, we can define the product and coproduct of any number of objects. In general, for a collection of objects $\{a_i\}$ where i is an index in a set S, then the product and coproduct are denoted as

$$\prod_{i \in S} a_i \qquad \text{and} \qquad \coprod_{i \in S} a_i.$$

Basic Limits

Before we go onto more complicated diagrams, take a moment and think about completing simpler diagrams.

Exercise

Show that the limit of a diagram with just one object a is isomorphic to a. Show this is true for colimits also.

Exercise

Show that a limit of the empty diagram is a terminal object of the category. Show that the colimit of the empty category is the initial object of the category.

Basic Limits

Products are special types of limits and coproducts are special types of colimits. Till now the diagrams we dealt with were discrete (without morphisms). Let's look at diagrams with some morphisms. Consider the simple diagram:

$$\begin{tikzcd} a \arrow[r, "f"] & b. \end{tikzcd}$$

The limit of this diagram will be an object c with two maps $\pi_a \colon c \longrightarrow a$ and $\pi_b \colon c \longrightarrow b$ such that the following triangle commutes

$$\begin{tikzcd} & c \arrow[dl, "\pi_a"'] \arrow[dr, "\pi_b"] & \\ a \arrow[rr, "f"] & & b. \end{tikzcd}$$

It satisfies the universal property that if there is any element d with maps $g_a \colon d \longrightarrow a$ and $g_b \colon d \longrightarrow b$ such that $f \circ g_a = g_b$ then there is a unique $h \colon d \longrightarrow c$ such that

Basic Limits

We will examine the different types of limits and colimits in the category of sets.

Example

In $\mathbb{Set}$, given a set function $f\colon S \longrightarrow T$, the limit will be set R with two functions $p_S\colon R \longrightarrow S$ and $p_T\colon R \longrightarrow T$ such that $f \circ p_S = p_T$. This set and these maps will satisfy a universal property. It is not hard to see that the set

$$R = \{(s, f(s)) \in S \times T\}$$

with the obvious projection functions satisfies the requirement of being a limit. This set is usually called the **graph of** f.

Basic Limits

Example

In $\mathbb{S}\mathrm{et}$, given a set function $f\colon S \longrightarrow T$, the colimit will be set V with two functions $inc_S\colon S \longrightarrow V$ and $inc_T\colon T \longrightarrow V$ such that $inc_T \circ f = inc_S$. This set and these maps will satisfy a universal property. The requirement is satisfied by the set

$$V = (S + T)/\sim$$

where $\sim$ is the relation on the disjoint union that has $s \in S$ equivalent to $f(s) \in T$. The functions inc_S and inc_T are the obvious inclusion functions.

Basic Limits

Consider the diagram

$$\begin{tikzcd} a \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & b. \end{tikzcd}$$

A limit for this diagram will be an object c and two maps $e\colon c \longrightarrow a$ and $j\colon c \longrightarrow b$ such that $f \circ e = j = g \circ e$ as in

$$\begin{tikzcd} & c \arrow[dl, "e"'] \arrow[dr, "j"] & \\ a \arrow[rr, shift left, "f"] \arrow[rr, shift right, "g"'] & & b. \end{tikzcd}$$

We write the requirement as $f \circ e = g \circ e$ (without mentioning j) and then write the required commutative diagram as

$$\begin{tikzcd} c \arrow[r, "e"] & a \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & b. \end{tikzcd}$$

Basic Limits

The universal property says that, for any object d and map $k\colon d \longrightarrow a$ such that $f \circ k = g \circ k$, there is a unique map $h\colon d \longrightarrow c$ such that the triangle in the following diagram commutes:

$$\begin{tikzcd}
d \arrow[d, "h"', "\exists!"] \arrow[rrd, "k"] & & & \\
c \arrow[rr, tail, "e"] & & a \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & b.
\end{tikzcd}$$

We call such a limit an **equalizer** of f and g.

Basic Limits

Example

In $\mathbb{S}\mathrm{et}$, an equalizer of set functions $f\colon S \longrightarrow T$ and $g\colon S \longrightarrow T$ is a set

$$R = \{s \in S : f(s) = g(s)\}.$$

There is an inclusion map $e\colon R \hookrightarrow S$. This set and inclusion map satisfy the universal property.

Basic Limits

Let us look carefully of a colimit of the same diagram. Such a colimit is called a coequalizer.

- Consider the same diagram.

$$\begin{tikzcd} a \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & b \end{tikzcd}$$

Basic Limits

Let us look carefully of a colimit of the same diagram. Such a colimit is called a coequalizer.

- Consider the same diagram.
- A **coequalizer** is an object c

$$\begin{tikzcd}
a \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & b & & c
\end{tikzcd}$$

Basic Limits

Let us look carefully of a colimit of the same diagram. Such a colimit is called a coequalizer.

$$\begin{tikzcd} a \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & b \arrow[r, "p"] & c \end{tikzcd}$$

- Consider the same diagram.
- A **coequalizer** is an object c
- and a map $p\colon b \longrightarrow c$ such that $p \circ f = p \circ g$.

Basic Limits

Let us look carefully of a colimit of the same diagram. Such a colimit is called a coequalizer.

$$\begin{tikzcd}
a \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & b \arrow[r, "p"] \arrow[dr, "k"'] & c \\
& & d
\end{tikzcd}$$

- Consider the same diagram.
- A **coequalizer** is an object c
- and a map $p\colon b \longrightarrow c$ such that $p \circ f = p \circ g$.
- Furthermore, if there is any d and map $k\colon b \longrightarrow d$ such that $k \circ f = k \circ g$,

Basic Limits

Let us look carefully of a colimit of the same diagram. Such a colimit is called a coequalizer.

$$\begin{tikzcd}
a \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & b \arrow[r, "p"] \arrow[dr, "k"'] & c \arrow[d, "h"', "\exists!"] \\
& & d
\end{tikzcd}$$

- Consider the same diagram.
- A **coequalizer** is an object c
- and a map $p\colon b \longrightarrow c$ such that $p \circ f = p \circ g$.
- Furthermore, if there is any d and map $k\colon b \longrightarrow d$ such that $k \circ f = k \circ g$,
- then there is a unique $h\colon c \longrightarrow d$ such that the triangle commutes.

Basic Limits

Example

In $\mathbb{Set}$*, a coequalizer for functions* $f\colon S \longrightarrow T$ *and* $g\colon S \longrightarrow T$ *is the set* $V = T/\sim$ *where* $\sim$ *is the equivalence relation that is generated by* $f(s) \sim g(s)$ *for all* $s \in S$*. The map* $p\colon T \longrightarrow V$ *takes every element* $t \in T$ *to the equivalence class it belongs to in* V*, i.e.,* $t \mapsto [t]$*.*

Basic Limits

Let us move on to more complicated diagrams. We will no longer make a dashed line around the drawings that we are completing.

- Consider the following diagram.

$$\begin{tikzcd}
 & b \arrow[d, "g"] \\
a \arrow[r, "f"'] & c
\end{tikzcd}$$

Basic Limits

Let us move on to more complicated diagrams. We will no longer make a dashed line around the diagrams that we are completing.

- Consider the following diagram.
- The limit is an object $a \times_c b$ called a **pullback**

$$\begin{tikzcd}
a \times_c b & b \arrow[d, "g"] \\
a \arrow[r, "f"'] & c
\end{tikzcd}$$

Basic Limits

Let us move on to more complicated diagrams. We will no longer make a dashed line around the diagrams that we are completing.

- Consider the following diagram.
- The limit is an object $a \times_c b$ called a **pullback**
- and three maps π_a, π_c, and π_b such that $f \circ \pi_a = \pi_c = g \circ \pi_b$. (There is no purpose in displaying π_c.)

$$\begin{tikzcd}
a \times_c b \arrow[r, "\pi_b"] \arrow[d, "\pi_a"'] & b \arrow[d, "g"] \\
a \arrow[r, "f"'] & c
\end{tikzcd}$$

Basic Limits

Let us move on to more complicated diagrams. We will no longer make a dashed line around the diagrams that we are completing.

$$\begin{tikzcd}
e \arrow[rrd, "k_b"] \arrow[rdd, "k_a"'] & & \\
& a \times_c b \arrow[r, "\pi_b"] \arrow[d, "\pi_a"'] & b \arrow[d, "g"] \\
& a \arrow[r, "f"'] & c
\end{tikzcd}$$

- Consider the following diagram.
- The limit is an object $a \times_c b$ called a **pullback**
- and three maps π_a, π_c, and π_b such that $f \circ \pi_a = \pi_c = g \circ \pi_b$. (There is no purpose in displaying π_c.)
- They satisfy the following universal property: for any object e and morphisms k_a, k_c, and k_b such that $f \circ k_a = k_c = g \circ k_b$,

Basic Limits

Let us move on to more complicated diagrams. We will no longer make a dashed line around the diagrams that we are completing.

$$\begin{tikzcd}
e \arrow[dr, "h", "\exists!"'] \arrow[drr, "k_b"] \arrow[ddr, "k_a"'] & & \\
& a \times_c b \arrow[r, "\pi_b"] \arrow[d, "\pi_a"'] & b \arrow[d, "g"] \\
& a \arrow[r, "f"'] & c
\end{tikzcd}$$

- Consider the following diagram.
- The limit is an object $a \times_c b$ called a **pullback**
- and three maps π_a, π_c, and π_b such that $f \circ \pi_a = \pi_c = g \circ \pi_b$. (There is no purpose in displaying π_c.)
- They satisfy the following universal property: for any object e and morphisms k_a, k_c, and k_b such that $f \circ k_a = k_c = g \circ k_b$,
- there is a unique $h\colon e \longrightarrow a \times_c b$ such that the triangles commute.

Basic Limits

- Notice that a product is a special type of pullback:
- The product is a pullback where the target c is a terminal object of the category. In that case, there is a unique morphism from every element, and hence, the square always commutes.
- Notice also that an equalizer is also a special type of pullback:
- It is the case where $b = a$, i.e., the two maps of the pullback diagram have the same source.

Basic Limits

Example

In $\mathbb{Set}$, a pullback is sometimes called a **fiber product of sets***. Let $f\colon S \longrightarrow T$ and $g\colon R \longrightarrow T$ then the pullback is the set*

$$P = \{(s, r) \in S \times R : f(s) = g(r)\}.$$

There are projection functions $\pi_S\colon P \longrightarrow S$ and $\pi_R\colon P \longrightarrow R$ which satisfy the universal properties.

Basic Limits

Example

A special case of a fiber product is when $g\colon R \hookrightarrow T$ *is an inclusion of a subset. We then have the pullback diagram*

$$\begin{tikzcd}
\{s \in S : f(s) \in R\} \arrow[r] \arrow[d, hook] & R \arrow[d, hook, "g"] \\
S \arrow[r, "f"'] & T.
\end{tikzcd}$$

The fiber product of f *and* g *is isomorphic to the set*

$$f^{-1}(R) = \{s \in S : f(s) \in R\}$$

which is called the **preimage** *of* f *for the subset* T*. This is an example of an instance of a general theorem that the pullback of a monic is a monic.*

Basic Limits

The colimit of the diagram

$$\begin{tikzcd}
a \arrow[r, "f"] \arrow[d, "g"'] & b \\
c &
\end{tikzcd}$$

is called a **pushout** of f and g.

Basic Limits

- The colimit consists of an object $c +_a b$
- and maps h_a, h_b, and h_c such that $h_b \circ f = h_a = h_c \circ g$.
- The universal property says that for any object e and any maps k_b, k_a, and k_c such that $k_b \circ f = k_a = k_c \circ g$
- there is a unique h that makes the following diagram commute

$$\begin{tikzcd}
a \arrow[r, "f"] \arrow[d, "g"'] & b \arrow[d, "h_b"] \arrow[ddr, "k_b"] & \\
c \arrow[r, "h_c"] \arrow[drr, "k_c"'] & c +_a b \arrow[dr, "h", "\exists!"'] & \\
 & & e.
\end{tikzcd}$$

Basic Limits

Example

In $\mathbb{S}\mathrm{et}$*, a pushout of the maps* $f\colon S \longrightarrow R$ *and* $g\colon S \longrightarrow T$ *is a set* $P = (R+T)/\sim$ *where* $\sim$ *is the equivalence relation on the set* $R+T$ *generated by the relation for all* $s \in S$, $f(s) \sim g(s)$*. There are obvious inclusions of* R *and* T *into* P*.*

Basic Limits

- Notice that a coproduct is a special type of pushout:
- It is the case where a is an initial object of the category, and hence, the square always commutes.
- Similarly, a coequalizer is a special type of pushout where $a = c$.

Limits and Colimits

Now that we have seen many examples of limits and colimits, let us formally define a limit and colimit of any diagram in a category.

Definition

- *For an arbitrary diagram D in a category, a* **limit** *is an object of the category, denoted* $\varprojlim D$*, and morphisms from that object to every object in D such that the appropriate diagrams commute. The object and projections must satisfy the universal properties outlined before.*
- *A* **colimit** *of a diagram D in the category will be an object, denoted* $\varinjlim D$*, with morphisms from every object in the diagram to the colimit that makes the appropriate commutative diagram. The object and morphisms must satisfy the universal properties outlined above.*

Limits and Colimits

Since there are maps from the limit to all the objects of the diagram, and there are maps from all the objects of the diagram to the colimit, we might write this as

$$\varprojlim D \longrightarrow D \longrightarrow \varinjlim D.$$

Limits and Colimits

There is a way of getting a limit by using a product and then taking an equalizer. First a simple case. Let us revisit the case where we took a limit of single arrow $f\colon a \longrightarrow b$. Consider the object $a \times b$ which is the product of the two objects in the diagram. There are two projection maps as in the following

$$\begin{tikzcd}
& a \times b \arrow[dl, "\pi_a"'] \arrow[dr, "\pi_b"] & \\
a \arrow[rr, "f"'] & & b.
\end{tikzcd}$$

Limits and Colimits

Notice that there are two maps from $a \times b$ to b: π_b and $f \circ \pi_a$. We can now take the equalizer of these two maps as follows

$$\begin{tikzcd}
c \arrow[r, "e"] & a \times b \arrow[r, shift left, "f \circ \pi_a"] \arrow[r, shift right, "\pi_b"'] & b.
\end{tikzcd}$$

Setting $p_a = \pi_a \circ e$ and $p_b = \pi_b \circ e = f \circ \pi_a \circ e$ as in

$$\begin{tikzcd}
 & c \arrow[d, "e"'] \arrow[ddl, "p_a"'] \arrow[ddr, "p_b"] & \\
 & a \times b \arrow[dl, "\pi_a"] \arrow[dr, "\pi_b"'] & \\
a \arrow[rr, "f"'] & & b.
\end{tikzcd}$$

it is easy to see that c is not only the equalizer of the two maps, but it is also the limit of the diagram $f \colon a \longrightarrow b$.

Limits and Colimits

We can generalize this from a single arrow to all finite diagrams. any diagram with a finite number of objects and morphisms.

Theorem

Limits of finite diagrams can be obtained using finite products and equalizers.

See proof in the book.

Limits and Colimits

Theorem

Limits of finite diagrams can be obtained using pullbacks and a terminal object.

Proof.

If there is a terminal object and pullbacks, we saw that we can create all finite products. Similarly equalizers can be seen as a type of pullback. ☐

There is, of course, a dual theorem.

Theorem

Colimits of finite diagrams can be obtained using coproducts and coequalizers. Equivalently, all finite colimits can be obtained using pushouts and an initial object.

Foreshadowing

- Chapter 3: Structures Within Categories
  - Section 3.3 Slices and Coslices
    - Slices
    - Coslices

Slice Categories

There are certain constructions that make the morphisms of one category into the objects of another category.

Definition

Given a category $\mathbb{A}$ and an object a of that category, the **slice category**, *$\mathbb{A}/a$, read "$\mathbb{A}$ over a" is a category whose objects are pairs $(b, f\colon b \longrightarrow a)$ where b is an object of $\mathbb{A}$ and f is a morphism of $\mathbb{A}$ whose target is a. The morphisms of $\mathbb{A}/a$ from $(b, f\colon b \longrightarrow a)$ to $(b', f'\colon b' \longrightarrow a)$ are morphisms $g\colon b \longrightarrow b'$ of $\mathbb{A}$ that make the following triangle commute.*

$$\begin{tikzcd}
b \arrow[dd, "g"'] \arrow[dr, "f"] & \\
 & a. \\
b' \arrow[ur, "f'"'] &
\end{tikzcd}$$

Slice Categories

Example

Some examples of slice categories:

- *For* $\mathbf{R}$*, the real numbers thought of as an element of* $\mathbb{Set}$*, the category of* $\mathbb{Set}/\mathbf{R}$ *is the collection of all* $\mathbf{R}$*-valued functions.*
- *For a partial order category* $A(P, \leq)$ *and* $p \in P$ *the category* $A(P, \leq)/p$ *is the partial order category of all elements below* p*, denoted* $p\downarrow$*.*

Coslice Categories

There is the dual notion of a slice category:

Definition

Given a category $\mathbb{A}$ and an object a of that category, the **coslice category**, *$a/\mathbb{A}$, read "$\mathbb{A}$ under a," is a category whose objects are pairs $(b, f\colon a \longrightarrow b)$ where b is an object of $\mathbb{A}$ and f is a morphism of $\mathbb{A}$ whose source is a. The morphisms of $a/\mathbb{A}$ from $(b, f\colon a \longrightarrow b)$ to $(b', f'\colon a \longrightarrow b')$ are morphisms $g\colon b \longrightarrow b'$ of $\mathbb{A}$ that make the following triangle commute.*

$$\begin{tikzcd}
 & b \arrow[d, "g"] \\
a \arrow[ur, "f"] \arrow[r, "f'"'] & b'.
\end{tikzcd}$$

Coslice Categories

Example

Some examples of a coslice category:

- *Consider the one-element set* $\{*\}$. *The category* $\{*\}/\mathbb{Set}$ *has objects that are sets with a function that picks out a distinguished element of the set. So an object in the category is effectively a pair* (S, s_0) *where* S *is a set and* s_0 *is a distinct element of that set. The morphisms from* (S, s_0) *to* (T, t_0) *are set functions that preserve the distinguished element. That is,* $f \colon S \longrightarrow T$ *such that* $f(s_0) = t_0$. *This is the category of* **pointed sets**.

Coslice Categories

Example (Continued)

- *Similarly, there is the category* $\{*\}/\mathbb{Top}$*, the category of* **pointed topological spaces**. *Where the objects are topological spaces with a distinguished object and morphisms are continious maps that preserve the distinguished point.*
- *For a partial order category* $A(P, \leq)$ *and* p*, an element in the partial order, the category* $p/A(P, \leq)$ *is* $p\uparrow$*, the partial order of the elements above* p*.*

Mini-course:

Self-Referential Paradoxes

Foreshadowing

- Chapter 3: Structures Within Categories
  - Section 3.4: Mini-course: Self-Referential Paradoxes
    - The Barber Paradox
    - Russell's Paradox
    - Hetrological Paradox
    - A philosophical interlude on paradoxes
    - Cantor's Inequality
    - The Main Theorem About Self-Referential Paradoxes
    - Turing's Halting Problem
    - The Contrapositive of the Main Theorem About Self-Referential Paradoxes
    - Fixed Points in Logic
    - Gödel's Incompleteness Theorem
    - Tarski's Theorem
    - Parikh Sentences
    - Epimenides and the Liar
    - Time Travel Paradoxes

Foreshadow

With the simple idea of a product in a category, we know enough category theory to describe some of the most profound and influential theorems in mathematics and computer science of the past hundred and fifty years. In the next few pages, we will meet

- Georg Cantor's theorem that shows there are different types of infinity;
- Bertrand Russell's paradox which proves that simple set theory is inconsistent;
- Kurt Gödel's famous incompleteness theorems that demonstrates a limitation of the notion of proof;
- Alan Turing's realization that there are some problems that can never be solved by a computer;
- and much more.

Foreshadow

What is truly amazing is that all these diverse and important theorems are consequences of a single simple theorem of category theory. This demonstrates the true power of category theory! What is still more shocking is that the central idea of this simple theorem goes back some 2,500 years ago to a conundrum about language called the Epimenides paradox. This conundrum shows that language can talk about itself, i.e. has self-reference. In the coming pages, we will see that not only language, but many systems, have the ability to have objects in the system refer to other objects within the system and even to themselves. This is the core of self-reference. It is shown that sets, language, logic, computers, and many other systems have the ability of self-reference. With self reference, one goes on to form **self-referential paradoxes** which are contradictions that come through an object using self reference to negate itself. This is the explanation of all the important theorems above.

Some Preliminaries

Before we leap into all the examples, there is one technical definition that we have to describe.

- Let S be a set and $2 = \{0, 1\}$ be a set with two values which will correspond to true and false.
- We saw that a function $g\colon S \longrightarrow 2$ is a characteristic function and describes the subset of S that g takes to 1.
- Now consider a set function $f\colon S \times S \longrightarrow 2$. The function f accepts two elements of S and outputs either 0 or 1.
- For any element s_0 of S, consider the function f where the second input is always s_0. We say that s_0 is “hardwired into the function.” This gives us a function

  $$f(\ \ , s_0)\colon S \longrightarrow 2$$

  with only one input.

Some Preliminaries

- Since this function goes from S to 2, it is also a characteristic function and describes the subset of S. The subset is

$$\{s \in S : f(s, s_0) = 1\}.$$

- For different f's and different elements of S, there will be different characteristic functions which describe different subsets.
- We now ask a simple question: given $g\colon S \longrightarrow 2$ and $f\colon S \times S \longrightarrow 2$, is there an s_0 in S such that g characterizes the same subset as $f(\ \ , s_0)$.
- To restate, for a given g and f, does there exist an $s_0 \in S$ such that $g(\ \ ) = f(\ \ , s_0)$?
- If such an s_0 exists, then we say g can be **represented** by f, or g is **representable** by f.

Barber Paradox

Bertrand Russell was not only a great logician, mathematician, and philosopher. He was also a great expositor. In order to explain some of the central ideas of self-referential systems to a general audience, he supposedly conjured up the **barber paradox**. Imagine an isolated village on top of a mountain in the Austrian alps where it is difficult for villagers to leave and for itinerant barbers to come to the village. This village has exactly one barber and there is a strict rule that is enforced:

A villager cuts his own hair if and only if he does not go to the single barber.

This makes sense. After all, if the villager will cut his own hair, why should he go to the barber? On the other hand, if the villager goes to the barber, he will not need to cut his own hair.

Barber Paradox

This works out very well for all the villagers except for one: the barber. Who cuts the barber's hair? If the barber cuts his own hair, then he is violating the village ordinance by cutting his own hair and having his hair cut by the barber. If he goes to the barber, then he is also cutting his own hair. This is illegal! What is an honest law abiding barber to do?

Barber Paradox

Now we formalize the problem. Let the set *Vill* consist of all the villagers in the village. Let us also remember our set $2 = \{0, 1\}$. The function $f\colon \mathit{Vill} \times \mathit{Vill} \longrightarrow 2$ describes who cuts whose hair in the village. It is defined for villagers v and v' as

$$f(v, v') = \begin{cases} 1 & : \text{ if the hair of } v \text{ is cut by } v' \\ 0 & : \text{ if the hair of } v \text{ is not cut by } v'. \end{cases}$$

We can now express the village ordinance as saying that for all v

$$f(v, v) = 1 \quad \text{if and only if} \quad f(v, \text{barber}) = 0.$$

The problem arises because this is true for all v including $v =$ barber. In this case we get:

$$f(\text{barber}, \text{barber}) = 1 \quad \text{if and only if} \quad f(\text{barber}, \text{barber}) = 0.$$

This is clearly a contradiction and cannot be true.

Barber Paradox

Let us be more categorical. Because $\mathbb{S}\mathrm{et}$ has products, there is the diagonal function $\Delta\colon \mathit{Vill} \longrightarrow \mathit{Vill} \times \mathit{Vill}$ that is defined as $\Delta(v) = (v, v)$. This function is at the core of self reference. It helps us see what f says when you evaluate an element v with itself, i.e., $f(v, v)$ and $f(\text{barber}, \text{barber})$. There is also a negation function $\mathit{NOT}\colon 2 \longrightarrow 2$ defined as $\mathit{NOT}(0) = 1$ and $\mathit{NOT}(1) = 0$ that swaps true and false. This function will be used to negate properties.

Barber Paradox

Composing f with Δ and NOT gives us $g\colon Vill \longrightarrow 2$ as in the following commutative diagram:

$$\begin{tikzcd}
 & Vill \times Vill \arrow[r, "f"] & 2 \arrow[dr, "NOT"] & \\
Vill \arrow[ur, "\Delta"] \arrow[rrr, "g"'] & & & 2.
\end{tikzcd}$$

That is,

$$g = NOT \circ f \circ \Delta.$$

For a villager v, $g(v) = NOT(f(\Delta(v))) = NOT(f(v, v))$. So $g(v) = 1$ if and only if $NOT(f(v, v)) = 1$ if and only if $f(v, v) = 0$ if and only if the hair of v is not cut by v. In other words $g(v) = 1$ if and only if v does not cut his own hair. In terms of self reference, the function g is the characteristic function of the subset of villagers who do not cut their own hair.

Barber Paradox

We now ask the simple question: can g be represented by f? In other words, is there a villager v_0 such that $g(\ ) = f(\ , v_0)$? The function g describes all those villagers who do not cut their own hair. It stands to reason that the barber is the villager who can represent g. After all, $f(\quad,\text{barber})$ describes all the villagers who get their hair cut by the barber. We are asking if $g(\ )$ is the same function as $f(\ ,\text{barber})$ and whether they characterize the same subset of villagers? Another way to pose this question is as follows: is the set of villagers who do not cut their own hair the same as the set of villagers who get their hair cut by the barber?

Barber Paradox

The answer is no. While it is true that for most $v \in Vill$

$$g(v) = f(v, \text{barber}),$$

it is not true for $v = \text{barber}$. If it were true for the barber, then we would have

$$g(\text{barber}) = f(\text{barber}, \text{barber}).$$

But the definition of g is given as $g(\text{barber}) = NOT(f(\text{barber}, \text{barber}))$. We conclude that g is not represented by $f(\quad, \text{barber})$, in fact it is not represented by any $f(\quad, v_0)$. That is, the set of villagers who do not cut their own hair cannot be the same as the set of villagers who get their hair cut by anyone.

Barber Paradox

It will be helpful to describe this problem in matrix form. Let us consider the set $Vill$ as $\{v_1, v_2, v_3, \ldots, v_n\}$. We can then describe the function $f\colon\ Vill \times Vill \longrightarrow 2$ as a matrix. Let us say that the barber is v_4. Notice that every row has exactly one 1 (every villager gets their haircut in only one place): either along the diagonal (the villager cuts their own hair) or in the v_4 column (the villager goes to the barber). Since it can only be one or the other, the numbers along the diagonal $1, 0, 1, ?, 0, \ldots, 1$ are almost the exact opposite of the numbers along the v_4 column $0, 1, 0, ?, 1, \ldots, 0$. This is a restatement of the rule of the village. There is only one problem: what is in the (v_4, v_4) position? We put a question mark in the matrix because that entry cannot be the opposite of itself. This way of seeing the problem will arise over and over again. Here we can see why these paradoxes are related to proofs called **diagonal arguments**.

Barber Paradox

$$\begin{array}{cc|ccccccc}
 & & \multicolumn{7}{c}{\textbf{Cutter}} \\
 & f & v_1 & v_2 & v_3 & v_4 & v_5 & \cdots & v_n \\
\hline
 & v_1 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 \\
 & v_2 & 0 & 0 & 0 & 1 & 0 & \cdots & 0 \\
\textbf{Cuttee} & v_3 & 0 & 0 & 1 & 0 & 0 & \cdots & 0 \\
 & v_4 & 0 & 0 & 0 & ? & 0 & \cdots & 0 \\
 & v_5 & 0 & 0 & 0 & 1 & 0 & \cdots & 0 \\
 & \vdots & \vdots & & & \vdots & & \ddots & \vdots \\
 & v_n & 0 & 0 & 0 & 0 & 0 & \cdots & 1
\end{array}$$

The function f as a matrix. Notice the diagonal is the opposite of the barber, column v_4.

Barber Paradox — Resolution

What is the resolution to this paradox? There are many attempts to solve this paradox, but they are not very successful. For example, the barber resigns as barber before cutting his own hair. (But that means that there is no barber in the town). Or the wife of the barber cuts the barber's hair. (But that means that there are two barbers in the town.) Or the barber is bald. Or the barber is a long-haired hippie. Or the rule is ignored while the barber cuts his own hair, etc. All these are saying the same thing: the village with this (contrived and) important rule cannot exist. Because if the village with this rule existed, there would be a contradiction. There are no contradictions in the physical world. The only way the world can be free of contradictions is if this proposed village with this strict rule does not exist.

Russell's Paradox

- Bertrand Russell described the barber paradox to help explain a deeper, more important problem that he formulated and is called **Russell's paradox**.
- This paradox concerns sets which are considered the foundation of much of mathematics.
- As is known, sets contain elements. The elements can be anything. In particular an element in a set can be a set itself. Here are a few sets to consider.
    - $A = \{x, y, z\}$ simply has three elements.
    - $B = \{s, t, \{x, y\}\}$ has as the set $\{x, y\}$ as an element.
    - $C = \{s, t, u, A\}$ contains the set A as an element.
- It is not strange to have sets as elements of sets.
    - The set of all classes given in a university can be thought of as containing elements where each element is the set of students in a class.
    - For a set of certain people, one can imagine every person as the set of their cells.
    - Consider the set $D = \{x, y, D\}$. This set contains itself.

Russell's Paradox

A set containing itself is not so strange. Here are three examples of sets that contain themselves:

- The set of all ideas discussed in this book.
- The set that contains all the sets that have more than three objects.
- The set of abstract ideas.

Russell's Paradox

If you do not like sets that contain themselves, you might want to consider the R which is the collection of all sets that do not contain themselves. Formally,

$$R = \{\text{set } S : S \text{ does not contain } S\} = \{S : S \notin S\}.$$

Going back to our simple examples of sets, we have $A \in R$, $B \in R$, $C \in R$ while $D \notin R$.

Now ask yourself the simple question: does R contain itself? In symbols, we ask if $R \in R$? Let us consider the possible answers. If $R \in R$, then since R fails to satisfy the requirements of being a member of R, we get that $R \notin R$. In contrast, if $R \notin R$, then since R satisfies the requirement of belonging to R, we have that $R \in R$. This is a contradiction.

Russell's Paradox

Let us formulate this. There is a collection of all sets called *Set*. There is also a two-place function $f\colon \mathit{Set} \times \mathit{Set} \longrightarrow 2$ that describes which sets are elements of which sets.

$$f(S, S') = \begin{cases} 1 & : \text{if } S \in S' \\ 0 & : \text{if } S \notin S'. \end{cases}$$

Russell's Paradox

We can use this f to describe all sets that do not contain themselves as follows. Consider g formed by composing the following maps:

$$\begin{tikzcd}
 & Set \times Set \arrow[r, "f"] & 2 \arrow[dr, "NOT"] & \\
Set \arrow[ur, "\Delta"] \arrow[rrr, "g"'] & & & 2
\end{tikzcd}$$

The value $g(S)$ is defined to be $NOT(f(S, S))$. This means $g(S) = 1$ if and only if $f(S, S) = 0$. In terms of self reference, g is the characteristic function of those sets that do not contain themselves.

Russell's Paradox

Now we ask the simple question: does there exist a set R such that $g(\quad)$ is represented by f as $f(\quad, R)$. That is, we want a set R such that

$$g(S) = 1 \quad \text{if and only if} \quad f(S, R) = 1$$

and

$$g(S) = 0 \quad \text{if and only if} \quad f(S, R) = 0.$$

This means that R contains only the sets that do not contain themselves. The problem is that if such a set R exists, then we can ask about $g(R)$, i.e., is $R \in R$. On the one hand, $g(R)$ is defined as $NOT(f(R, R))$ and on the other hand, if f represents g with R, then $g(R) = f(R, R)$. That is,

$$f(R, R) = g(R) = NOT(f(R, R)).$$

This is a contradiction.

Russell's Paradox

Let us look at Russell's paradox from a matrix/array point of view. Let us consider the infinite collection *Set* as $\{S_1, S_2, S_3, \ldots\}$. We can then describe the function $f\colon Set \times Set \longrightarrow 2$ as the following matrix.

Russell’s Paradox

		Subset					
	f	S_1	S_2	S_3	S_4	S_5	$\cdots$
Element	S_1	NOT(1)=0	0	0	0	1	$\cdots$
	S_2	0	NOT(0)=1	0	1	0	$\cdots$
	S_3	0	0	NOT(1)=0	0	0	$\cdots$
	S_4	1	0	0	NOT(1)=0	0	$\cdots$
	S_5	0	1	0	1	NOT(0)=1	$\cdots$
	$\vdots$	$\vdots$			$\vdots$		$\ddots$

The function *f* as a matrix. The function *g* is the changed diagonal and it is different from every column. This is a way of saying that the diagonal (which is *g*) cannot be represented by any column of the array.

Russell’s Paradox — Resolution

- Let us consider how to deal with this paradox.
- The only way to avoid this contradiction is to accept that the function g cannot be represented by any element of Set.
- This translates into meaning that the collection of all sets that do not contain themselves does not form a set, i.e., this collection is not an element of Set.
- While such a collection seems to be a well-defined notion, we have shown that if we say that this collection is an element of Set, then there is a contradiction. Mathematicians went on and used this to make a distinction between a “set” and a “class.”
- They declared that classes are collections that are not sets.
- This distinction plays major roles in logic and higher mathematics.

Heterological Paradox

Now for a linguistic paradox. The **heterological paradox**, also called **Grelling's paradox** after Kurt Grelling, who first formulated it, is about adjectives (words that modify nouns.) Consider several adjectives and ask if they describe themselves, that is, if the adjective has the property of the adjective. "English" is English. In contrast, "French" is not French ("Francais" is Francais.) "German" is not German ("Deutsch" is Deutsch.) The word "abbreviated" is not abbreviated, "unabbreviated" is unabbreviated and "hyphenated" is not hyphenated, etc. We see that some adjectives describe themselves and some adjectives do not describe themselves. Call all adjectives that describe themselves "autological." In contrast, call all adjectives that do not describe themselves as "heterological." We can start making a table of many adjectives.

Heterological Paradox

autological	**heterological**
English	non-English
Francais	French
Deutsch	German
noun	verb
unhyphenated	hyphenated
unabbreviated	abbreviated
polysyllabic	monosyllabic
⋮	⋮

Heterological Paradox

It seems that we can split all adjectives into these two groups. Is that true? Let us ask a simple question. Is “heterological” autological or heterological? That is, does the adjective “heterologoical” belong on the left side or the right side of the table? Let us go through the two possibilities.

- If “heterological” is autological, then it is not heterological and it does not describe itself. This means it is heterological.
- If “heterological” is heterological, then the adjective does describe itself and that makes it autological and not heterological.

We have a contradiction.

Heterological Paradox

Let us formulate this paradox categorically. There is a set *Adj* of adjectives and a function $f\colon Adj \times Adj \longrightarrow 2$ which is defined for adjectives a and a' as follows:

$$f(a,a') = \begin{cases} 1 & : \text{if } a \text{ is described by } a' \\ 0 & : \text{if } a \text{ is not described by } a'. \end{cases}$$

Use f to formulate g as the composition of the following three maps:

$$\begin{tikzcd}
 & Adj \times Adj \arrow[r, "f"] & 2 \arrow[dr, "NOT"] & \\
Adj \arrow[ur, "\Delta"] \arrow[rrr, "g"'] & & & 2.
\end{tikzcd}$$

Heterological Paradox

In terms of self reference, the function g is the characteristic function of those adjectives that do not describe themselves. Can g be represented by some element in Adj? Is there some adjective, say “heterological,” that can be used in f to represent g? That is, is it true that $g(\quad) = f(\quad,$ “heterological”$)$? We are asking if the subset of adjectives that do not describe themselves can be described by the word “heterological.” If this was true, we would have that for all adjectives A,

$$g(A) = f(A, \text{“heterological”}).$$

But this cannot be true because then we would have that

$$g(\text{“heterological”}) = f(\text{“heterological”}, \text{“heterological”}).$$

But that would give a contradiction because by the definition of g we have

$$g(\text{“heterological”}) = NOT(f(\text{“heterological”}, \text{“heterological”})).$$

Heterological Paradox — Resolution

- The only conclusion we can come to is that $g(\ \ )$ cannot be represented by f. That is, the set of all adjectives that do not describe themselves cannot be represented by “heterological”. However, that is exactly the definition of “heterological”!
- How do we avoid this little paradox? There are two usual ways of resolving this paradox.
    - Many philosophers say that the word “heterological” cannot exist. After all, we just showed that it is not always well-defined. We cannot determine if a certain adjective (“hetrological”) is heterological or not.
    - Another more obvious solution is to just ignore the problem. Human language is inexact and full of contradictions. Every time we use an oxymoron, we are stating a contradiction. Every time we ask for another piece of cake while lamenting the fact that we cannot lose weight, we are stating a contradiction. We can safely ignore the fact that heterological is not well-defined for only one adjective.

A philosophical interlude on paradoxes

The word “paradox” has many different definitions. For a logician, a paradox is a process where an assumption is made, and through valid reasoning, a contradiction is derived. We can visualize this as

$$\text{Assumption} \Longrightarrow \text{Contradiction.}$$

The logician then concludes that since the reasoning was valid and the contradiction cannot happen, it must be that the assumption was wrong. This is very similar to what mathematicians call “proof by contradiction” and philosophers call “*reductio ad absurdum*.” In a sense, a paradox is a method of showing that the assumption is not part of rational thought. If one accepts the assumption, one will come to a contradiction which is so dangerous to rational thought.

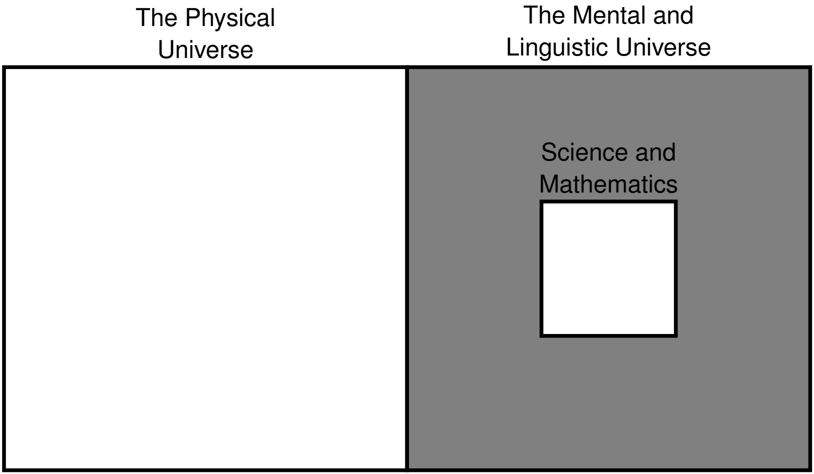
A philosophical interlude on paradoxes

We have so far seen the same pattern of proof in three different areas:

- villagers
- sets
- adjectives

In all three of the above paradoxes, the assumption is that the g function can be represented by the f function. A contradiction is then derived and we conclude that g is not represented by f. These three examples highlight three different realms where the alleged contradictions might be found. Let us examine these carefully.

A philosophical interlude on paradoxes



Contradictions can occur in gray areas. White areas cannot have contradictions.

A philosophical interlude on paradoxes

The Physical Universe. A village with a particular rule is part of the physical universe. The physical universe does not have any contradictions. Facts and properties simply are and no object can have two opposing properties. Whenever we come to such contradictions, we have no choice but to conclude that the assumption was wrong.

A philosophical interlude on paradoxes

The Mental and Linguistic Universe. In contrast to the physical universe, the human mind and human language — that the mind uses to express itself — are full of contradictions. We are not perfect machines. We have a lot of different contradictory parts and desires. An oxymoron is a small contradiction that we all use in our speech. We all have conflicting thoughts in our head and these thoughts are expressed in our speech. So when an assumption brings us to a contradiction in our thought or language, we do not need to take it very seriously. If an adjective is in two opposite classifications, it does not really bother us. In such a case, we cannot go back to our assumption and say it is wrong. The entire paradox can be ignored.

A philosophical interlude on paradoxes

Science and Mathematics. There are, however, parts of human thought and language where we cannot tolerate contradictions: science and mathematics. These areas of exact thought are what we use to discuss the physical world (and more). If science and math are to discuss / describe / model / predict the contradiction-free physical universe, then we better make sure that no contradictions occur there. We first saw this in the early years of elementary school when our teachers proclaimed that we are not permitted to divide by zero. Why not? We can divide by any other number. Why not zero? If we were permitted to divide by zero, an easy contradiction could be derived. Since math and science cannot have contradictions, young fledglings are not permitted to divide by zero. To summarize, science and mathematics are products of the human mind and language which we do not permit to have contradictions. If an assumption leads us to a contradiction in science or mathematics, then we must abandon the assumption.

A philosophical interlude on paradoxes

What is gained by looking at self-referential paradoxes from the categorical point of view? Many have felt that these different instances of self-referential paradoxes have a similar pattern (witness Bertrand Russell supposedly inventing the barber paradox to illustrate Russell's paradox.) However, no one has ever formalized this feeling. The major advance that category theory has to offer the subject is to actually show that all these different self-referential paradoxes are really instances of a single categorical theorem. F. William Lawvere described a simple formalism that showed many of the major self-referential paradoxes and more. This shows that the logic of self-referential paradoxes is inherent in many systems. It also shows the unifying power of category theory.

A philosophical interlude on paradoxes

Another positive aspect of our categorical formalism. Lawvere showed us how to have an exact mathematical description of the paradoxes while avoiding messy statements about what exists and what does not exist.

- In the categorical setting, the barber paradox does not say that a village with a rule does not exists.
- With Russell's paradox, a category theorist does not say that a certain collection does not form a set.
- Similarly with the heterological paradox, we avoid the silly analysis as to whether a word exists or not.

Although in our presentation of the paradoxes we mention the way philosophers have thought about these issues, in our categorical discussion, we successfully avoid metaphysical gobbledygook. For this alone, we should be appreciative of the categorical formalism.

Cantor's Inequalities

Let us continue our list of instances of self-referential paradoxes.

- At the end of the 19th century Georg Cantor proved some important theorems about the sizes of sets.
- He first showed that every set is smaller than its powerset (set of subsets). That is, every set S is smaller than the set $\mathcal{P}(S)$.
- A more categorical way of saying this is that for any set S, there cannot exist a surjection $h\colon S \longrightarrow \mathcal{P}(S)$.
- Yet another way of saying this, is that for every purported surjection $h\colon S \longrightarrow \mathcal{P}(S)$, there will be some subset of S that will not be in the image of h. One can think of this as a proof by contradiction: we are going to assume (wrongly) that there is such a surjection h and derive a contradiction (because we will find something that is not in the image of h.) Since this is formal mathematics, no such contradiction can exist and hence our assumption that such a surjection exists must be false.

Cantor's Inequalities

Given such an $h \colon S \longrightarrow \mathcal{P}(S)$, let us define $f_h \colon S \times S \longrightarrow 2$ for $s, s' \in S$ as follows

$$f_h(s, s') = \begin{cases} 1 & : s \in h(s') \\ 0 & : s \notin h(s'). \end{cases}$$

Use f_h to construct g_h as follows

$$\begin{tikzcd} & S \times S \arrow[rr, "f_h"] & & 2 \arrow[dr, "NOT"] & \\ S \arrow[ur, "\Delta"] \arrow[rrrr, "g_h"'] & & & & 2. \end{tikzcd}$$

The function g_h is the characteristic function of the subset $C_h \subseteq S$ where each element s does not belong to $h(s)$, i.e.,

$$C_h = \{s \in S : s \notin h(s)\} \subseteq S.$$

Notice that f_h, g_h and C_h depend on h. If we change h, we will get different functions and sets.

Cantor's Inequalities

We claim that the subset C_h of S is not in the image of h, i.e., C_h is a "witness" or a "certificate" that h is not surjective. If C_h was in the image of h, there would be some $s_0 \in S$ such that $h(s_0) = C_h$. In that case g_h would be represented by f_h with s_0. That is, for all $s \in S$

$$g_h(s) = f_h(s, s_0)$$

but this would also be true for $s_0 \in S$ which would mean that

$$g_h(s_0) = f_h(s_0, s_0).$$

However, by the definition of g_h, we have that

$$g_h(s_0) = NOT(f_h(s_0, s_0)).$$

Since this cannot be, our assumption that $h(s_0) = C_h$ is wrong, and there is a subset of S that is not in the image of h.

Cantor's Inequalities

Let us look at Cantor's inequality from a matrix/array point of view. We write the collection S as $\{s_1, s_2, s_3, \ldots\}$. The function $f_h \colon S \times S \longrightarrow 2$ can be described as the following matrix Notice that the diagonal is different than every column of the array. This is a way of saying that the diagonal (which is g_h) cannot be represented by any column of the array.

Cantor’s Inequalities

		Subset of S					
	f_h	$h(s_1)$	$h(s_2)$	$h(s_3)$	$h(s_4)$	$h(s_5)$	$\cdots$
Elements of S	s_1	NOT(1)=0	0	0	0	1	$\cdots$
	s_2	0	NOT(0)=1	0	1	0	$\cdots$
	s_3	0	0	NOT(1)=0	0	0	$\cdots$
	s_4	1	0	0	NOT(1)=0	0	$\cdots$
	s_5	0	1	0	1	NOT(0)=1	$\cdots$
	$\vdots$	$\vdots$			$\vdots$		$\ddots$

The function f_h as a matrix. The function g_h represents C_h and is the changed diagonal. It is different from every column.

Cantor's Inequalities - Resolution

- This is part of mathematics and the only resolution for this paradox is to accept the fact no such surjective h exists and that $|S| < |\mathcal{P}(S)|$.
- Notice that this applies to any set.
- For finite S, this is obvious since $|S| = n$ implies $|\mathcal{P}(S)| = 2^n$.
- However, this is true for infinite S also. What this shows is that $\mathcal{P}(S)$ is a different level of infinity than S.
- One can iterate this process and get $\mathcal{P}(S)$, $\mathcal{P}(\mathcal{P}(S))$, $\mathcal{P}(\mathcal{P}(\mathcal{P}(S)))$, $\ldots$ each of which is at a different level of infinity.

Cantor’s Inequalities

Related to the Cantor’s inequality above is the theorem that the set of natural numbers $\mathbf{N} = \{0, 1, 2, 3, \ldots\}$ is smaller than the interval of all real numbers between 0 and 1, i.e., $(0, 1) \subseteq \mathbf{R}$.

Definition

The infinity that corresponds to the natural numbers is called **countable infinity** *while any larger infinity — such as the interval of real numbers — is called* **uncountable infinity***.*

Cantor's Inequalities

This proof will be slightly different than the previous examples that we saw. We include it because it has features that are closer to the upcoming general theorem. Rather than working with the set $2 = \{0, 1\}$, this proof works with the set $10 = \{0, 1, 2, 3, \ldots, 9\}$. Also, rather than working with the function $NOT\colon 2 \longrightarrow 2$ which swaps both elements of 2, we now work with the function $\alpha\colon 10 \longrightarrow 10$ which is defined as follows:

$$\alpha(0) = 1, \quad \alpha(1) = 2, \quad \alpha(2) = 3, \quad \ldots, \quad \alpha(8) = 9, \quad \alpha(9) = 0,$$

i.e., $\alpha(n) = n + 1 \;\; Mod \;\; 10$. The most important feature of α is that every output is different than its input. There are many such functions from 10 to 10. We choose this one.

Cantor's Inequalities

The proof that $\mathbf{N}$ is smaller than $(0,1)$ works by showing every function $h\colon \mathbf{N} \longrightarrow (0,1)$ defines a real number in $(0,1)$ that is not in the image of h. We can think of this again as a proof by contradiction. We assume (wrongly) that there is a surjection $h\colon \mathbf{N} \longrightarrow (0,1)$ and come to a contradiction which proves that no such h can possibly exist.

With such an h we can define a function $f_h\colon \mathbf{N}\times\mathbf{N} \longrightarrow 10$ that describes the decimal expansions of the real numbers that h describes. For $m, n \in \mathbf{N}$,

$$f_h(m,n) = \text{the } m\text{th digit of } h(n).$$

This means that f_h gives every output digit of the purported function h. The next Figure will help explain f_h. The natural numbers on the left tell you the position. The function h assigns to every natural number on the top, a real number below it. The numbers on the left are the first inputs to f_h, and the numbers on the top are the second inputs to f_h.

Cantor's Inequalities

		The numbers $h(n)$							
	f_h	0	1	2	3	4	5	6	$\cdots$
		0	0	0	0	0	0	0	$\cdots$
		.	.	.	.	.	.	.	$\cdots$
Position	0	0	0	7	2	7	7	4	$\cdots$
	1	0	1	2	2	7	6	7	$\cdots$
	2	0	3	0	3	0	0	0	$\cdots$
	3	0	6	2	0	1	2	0	$\cdots$
	4	0	1	0	2	3	1	3	$\cdots$
	5	0	1	0	3	0	1	5	$\cdots$
	$\vdots$	$\vdots$			$\vdots$			$\vdots$	$\ddots$

Cantor's Inequalities

With such an f_h, one can go on to describe a function g_h with the — by now familiar — construction

$$\begin{tikzcd}
 & \mathbf{N} \times \mathbf{N} \arrow[r, "f_h"] & 10 \arrow[dr, "\alpha"] & \\
\mathbf{N} \arrow[ur, "\Delta"] \arrow[rrr, "g_h"'] & & & 10
\end{tikzcd}$$

The function g_h also depends on h. The next Figure will help explain the function g_h. It is the same as the last Figure but with the elements along the diagonal changed by α. That is, the nth digit of the nth number is changed. The changed numbers are the outputs to the function g_h. Thinking of the outputs of g_h as the digits of a real number, we are describing a real number between 0 and 1. We call this number G_h. In our case,

$$G_h = 0.121142\ldots$$

Cantor's Inequalities

		The number $h(n)$							
	f_h	0	1	2	3	4	5	6	$\cdots$
		0	0	0	0	0	0	0	$\cdots$
		.	.	.	.	.	.	.	$\cdots$
Position	0	$\alpha(0) = 1$	0	7	2	7	7	4	$\cdots$
	1	0	$\alpha(1) = 2$	2	2	7	6	7	$\cdots$
	2	0	3	$\alpha(0) = 1$	3	0	0	0	$\cdots$
	3	0	6	2	$\alpha(0) = 1$	1	2	0	$\cdots$
	4	0	1	0	2	$\alpha(3) = 4$	1	3	$\cdots$
	5	0	1	0	3	0	$\alpha(1) = 2$	5	$\cdots$
	$\vdots$	$\vdots$			$\vdots$			$\vdots$	$\ddots$

The changed diagonal, g_h, of a purported surjection h from $\mathbf{N}$ to $(0, 1)$.

Cantor's Inequalities

The claim is that g_h is not represented by f_h. This means that the number represented by g_h will not be the number represented by $f_h(\quad , n_0)$ for any n_0. Another way to say this is that the number G_h will not be any column in the scheme described in the Figure. This is obviously true.

- G_h was formed to be different than the first column because the number in the first position is different.
- It is different than the second column because it was formed to be different at the second position.
- It is different than the third column because it was formed to be different at the third digit, etc.

Cantor's Inequalities

In terms of self reference, G_h is a real number that says

"This number is not the number in column n because the nth digit is different from the nth column's nth digit."

or

"This number is not in the image of h."

Conclusion: G_h is not on our list and hence h is not surjective.

Cantor's Inequalities

Let us show the end of the proof formally. If there was some n_0 that represented g_h, then for all m

$$g_h(m) = f_h(m, n_0)$$

(i.e., G_h is the same as column n_o.) But if this was true for all m, then it is true for n_0 also (that is, it is true by every digit including the one on the diagonal.) But that says that

$$g_h(n_0) = f_h(n_0, n_0).$$

However, g_h was defined for n_0 as

$$g_h(n_0) = \alpha(f_h(n_0, n_0)).$$

We conclude that no such n_0 exists, and g_h describes a number in $(0, 1)$ which is not in the image of h. That is, h cannot be surjective and the set $\mathbf{N}$ is smaller than the set $(0, 1)$.

The Main Theorem About Self-Referential Paradoxes

- We have seen the same idea over and over in many different contexts.
- All these examples are instances of a single theorem of category theory.
- This again shows the unifying power and versatility of category theory.
- We should note that our examples till now have all been about sets and set maps.
- However, we will see that there are instances of the same phenomena in other categories.
- It pays to describe the theorem in its most general setting.

The Main Theorem About Self-Referential Paradoxes

We begin with some needed preliminaries.

Definition

First a simple definition in $\mathbb{Set}$*. Consider a set* Y *and a set function* $\alpha\colon Y \longrightarrow Y$*. We call* $s_0 \in Y$ *a* **fixed point** *of* α *if* $\alpha(s_0) = s_0$*. That is, the output is the same (or fixed) as the input. We write the element* s_0 *by talking about a function* $p\colon \{*\} \longrightarrow Y$ *such that* $p(*) = s_0$*. Remember that* $\{*\}$ *is the terminal object in* $\mathbb{Set}$ *and helps pick out elements of sets. Saying that* s_0 *is a fixed point of* α *amounts to saying that* $\alpha \circ p = p$*, i.e., the following diagram commutes:*

$$\begin{tikzcd}
\{*\} \arrow[rr, "p"] \arrow[dr, "p"'] & & Y \arrow[dl, "\alpha"] \\
& Y. &
\end{tikzcd}$$

The Main Theorem About Self-Referential Paradoxes

Definition

Let us generalize this to any category $\mathbb{A}$ *with a terminal object* 1. *Let* y *be an object in* $\mathbb{A}$ *and* $\alpha\colon y \longrightarrow y$ *be a morphism in* $\mathbb{A}$. *Then we say* $p\colon 1 \longrightarrow y$ *is a fixed point of* α *if* $\alpha \circ p = p$.

The Main Theorem About Self-Referential Paradoxes

Definition

- *Let us start in the category of* $\mathbb{Set}$*.*
- *Remember that if* $f\colon S \times S \longrightarrow Y$ *is a set function and* s_0 *is an element in* S*, then* $f(\ \ , s_0)\colon S \longrightarrow Y$ *is a function of one input.*
- *We say* $g\colon S \longrightarrow Y$ *is* **represented** *by* f *if there exists an* s_0 *in* S *such that* $g(\ \ ) = f(\ \ , s_0)$*.*
- *In other words, for every* x *in* S*, we have that* $g(x) = f(x, s_0)$*.*
- *What does it mean for* $g\colon S \longrightarrow Y$ *to* not *be represented by* f*?*
- *That means for all* s *in* S*,* $g(\ \ ) \neq f(\ \ , s)$*.*
- *In detail, for all* $s \in S$*, there is some* x *in* S *such that* $g(x) \neq f(x, s)$*.*

The Main Theorem About Self-Referential Paradoxes

Definition

- *Let us generalize representability to any category* $\mathbb{A}$ *with a terminal object* 1 *and binary products.*
- *We need the isomorphism* $i\colon a \longrightarrow a \times 1$*.*
- *Let* $f\colon a \times a \longrightarrow y$ *and* $g\colon a \longrightarrow y$ *be morphisms in* $\mathbb{A}$*.*
- *Then* g *is representable by* f *if there is a morphism* $p\colon 1 \longrightarrow a$ *such that* $g = f \circ (id_a \times p) \circ i\colon a \longrightarrow y$*.*
- *We can see this as*

$$\begin{tikzcd}
a \arrow[r, "i", "\cong"'] \arrow[rrr, bend left, "g"] & a \times 1 \arrow[r, "id_a \times p"] & a \times a \arrow[r, "f"] & y.
\end{tikzcd}$$

- g *is* not *representable if for all* $p\colon 1 \longrightarrow a$ *we have that* $g \neq f \circ (id_a \times p) \circ i$*.*

The Main Theorem About Self-Referential Paradoxes

Now for the main theorem as given by Lawvere in 1969.

Theorem

Cantor's Theorem.

- *Let* $\mathbb{A}$ *be a category with a terminal object and binary products.*
- *Let* y *be an object in the category and* $\alpha \colon y \longrightarrow y$ *be a morphism in the category.*
- *If* α *does not have a fixed point, then for all objects* a *and for all* $f \colon a \times a \longrightarrow y$ *there exists a* $g \colon a \longrightarrow y$ *such that* g *is not representable by* f.

The Main Theorem About Self-Referential Paradoxes

Proof.

Let $\alpha\colon y \longrightarrow y$ not have a fixed point, then for any a and for any $f\colon a \times a \longrightarrow y$ we can compose f with Δ and α to form g as follows

$$\begin{tikzcd}
& a \times a \arrow[r, "f"] & y \arrow[dr, "\alpha"] & \\
a \arrow[ur, "\Delta"] \arrow[rrr, "g"'] & & & y.
\end{tikzcd}$$

We claim that g is not representable by f. Assume (wrongly) that g is represented by f with $p\colon 1 \longrightarrow a$.

With a few steps, we get that α has a fixed point:

$$(f \circ \Delta \circ p) = g \circ p = \alpha \circ (f \circ \Delta \circ p).$$

□

Turing's Halting Problem

Let us look at more examples of Cantor's Theorem in other categories.

In the early 1930's, long before the engineers actually created computers, Alan Turing, the "father of computer science," showed what computers *cannot* do. Loosely speaking, researchers[1] proved that no program can decide whether or not any program will go into an infinite loop or not. Already from this inexact statement one can see the self reference: programs deciding properties of programs.

---

[1] Most writers attribute the halting problem to Alan Turing. This is historically not true. He was the first to prove that certain problems were not decidable by machines, but he did not mention or prove the undecidability of the halting problem. This was originally done by Martin Davis. See more about this in Cristian Calude's book *To Halt or Not to Halt: That is the question*.

Turing's Halting Problem

Let us state a more exact version of Turing's theorem. First some preliminaries.

- Programs come in many different forms.
- Here we are concerned with programs that only accept a single natural number as input.
- To every such program, there is a unique natural number that describes that program.
- In order to see this, realize that all computer programs are stored as a binary string. Every binary string can be seen as a natural number.
- We call the program associated with the natural number *n*, "program *n*".
- This idea — that programs which act on numbers can be represented by numbers — shows that programs can be self referential.

Turing's Halting Problem

Programs that accept a single number can halt or they can go into an infinite loop. The **halting problem** asks for a program to accept a program and a number and determine if that program with that number will halt or go into an infinite loop. To be more exact, the halting problem asks for two numbers (i) a number of a program that accepts a single number and (ii) an input to that program. Turing's theorem says that no such program can possibly exist. This is not a limitation of modern technology or of our current ability. Rather, this is an inherent limitation of computation.

Turing's Halting Problem

The proof is, once again, a proof by contradiction. Assume (wrongly) that there does exist a program that accepts a program number and an input, and can determine if that program will halt or go into an infinite loop when that number is entered into that program. Formally, such a program will describe a total computable function, i.e., a morphism in $\mathbb{CompFunc}$. The function named $Halt\colon \mathbf{N} \times \mathbf{N} \longrightarrow Bool$ defined on natural numbers $m, n \in \mathbf{N}$ is

$$Halt(m,n) = \begin{cases} 1 & : \text{if input } m \text{ into program } n \text{ halts} \\ 0 & : \text{if input } m \text{ into program } n \text{ goes into an infinite loop.} \end{cases}$$

This function can be described by the chart in the next slide where the natural numbers on the top are the number of the programs and the numbers on the left are the input numbers. The numbers in the chart tell the value of *Halt*.

Turing's Halting Problem

		Program						
	Halt	0	1	2	3	4	5	⋯
Input	0	0	1	0	0	0	1	⋯
	1	1	1	1	1	1	1	⋯
	2	0	1	0	0	0	0	⋯
	3	0	1	0	0	1	0	⋯
	4	0	0	1	1	1	1	⋯
	5	1	0	1	0	1	1	⋯
	⋮			⋮		⋮		⋱

A purported halting function.

Turing's Halting Problem

- It is not hard to see that the function $\Delta \colon \mathbf{N} \longrightarrow \mathbf{N} \times \mathbf{N}$ defined as $\Delta(n) = (n, n)$ is a computable function. That is, one can program a computer to accept a single number and output the pair of the same numbers.
- Consider the partial NOT function $ParNOT \colon Bool \longrightarrow Bool$ defined as follows:

$$ParNOT(n) = \begin{cases} 1 & : \text{if } n = 0 \\ \uparrow & : \text{if } n = 1 \end{cases}$$

  where ↑ means it will go into an infinite loop. The *ParNOT* is also a computable function.
- Since *Halt* is assumed computable, and the function Δ and *ParNOT* are computable, then their composition is also computable function.

Turing's Halting Problem

$$\begin{tikzcd}
 & \mathbf{N} \times \mathbf{N} \arrow[rr, "Halt"] & & Bool \arrow[dr, "ParNOT"] & \\
\mathbf{N} \arrow[ur, "\Delta"] \arrow[rrrr, "Halt'"'] & & & & Bool
\end{tikzcd}$$

It is important to stress that this diagram is in $\mathbb{CompFunc}$ and not in $\mathbb{Set}$. The new computable function, *Halt′*, accepts a number *n* as input and does the opposite of what program *n* on input *n* does. That is, if program *n* on input *n* halts, then *Halt′*(*n*) will go into an infinite loop. Otherwise, if program *n* on input *n* goes into an infinite loop, then *Halt′*(*n*) will halt. In terms of self reference, *Halt′* describes a program that the does the opposite of what program *n* is supposed to do on input *n*. We can see the way *Halt′* is defined with the chart on the next slide.

Turing’s Halting Problem

		Program						
	Halt	0	1	2	3	4	5	$\cdots$
Input	0	$\alpha(0) = 1$	1	0	0	0	1	$\cdots$
	1	1	$\alpha(1) = \uparrow$	1	1	1	1	$\cdots$
	2	0	1	$\alpha(0) = 1$	0	0	0	$\cdots$
	3	0	1	0	$\alpha(0) = 1$	1	0	$\cdots$
	4	0	0	1	1	$\alpha(1) = \uparrow$	1	$\cdots$
	5	1	0	1	0	1	$\alpha(1) = \uparrow$	$\cdots$
	$\vdots$			$\vdots$		$\vdots$		$\ddots$

The changed diagonal of the purported halting function, where $\alpha = \mathit{ParNOT}$.

Turing's Halting Problem

Since *Halt′* is a computable function, the program for this computable function must have a number and be somewhere on our list of computable functions. However, it is not. *Halt′* was formed to be different than every column in the chart. What is wrong? We know that Δ and *ParNOT* are computable. We assumed that *Halt* was computable. It must be our assumption of the computability of the the *Halt* function was wrong and *Halt* is not computable.

Let us formally show that *Halt′* is different than every column in the chart. Imagine that *Halt′* is computable and the number of *Halt′* is n_0. This means that *Halt′* is the n_0 column of our chart.

Turing's Halting Problem

Another way to say this is that $Halt'$ is representable by $Halt(\quad, n_0)$, i.e., for all n,

$$Halt'(n) = Halt(n, n_0).$$

Now let us ask about $Halt'(n_0)$? We get

$$Halt'(n_0) = Halt(n_0, n_0).$$

But we defined $Halt'(n_0)$ to be

$$Halt'(n_0) = ParNOT(Halt(n_0, n_0))$$

so we have a contradiction.
In a sense, we can say that the computational task that $Halt'$ (and in particular $Halt'(n_0)$) performs is:

> "If you ask whether this program will halt or go into an infinite loop, then this program will give the wrong answer."

Since computers cannot give the wrong answer, $Halt'$ cannot exist and hence $Halt$ cannot exist.

The Contrapositive of the Main Theorem

Cantor's theorem is very important and we need to express it in many equivalent ways. Let us consider the contrapositive of Cantor's theorem.

In less technical terms, the matrix form of Cantor's theorem says:

- If α does not have a fixed point, then the diagonal — which is changed by α — is different than every column in the matrix.

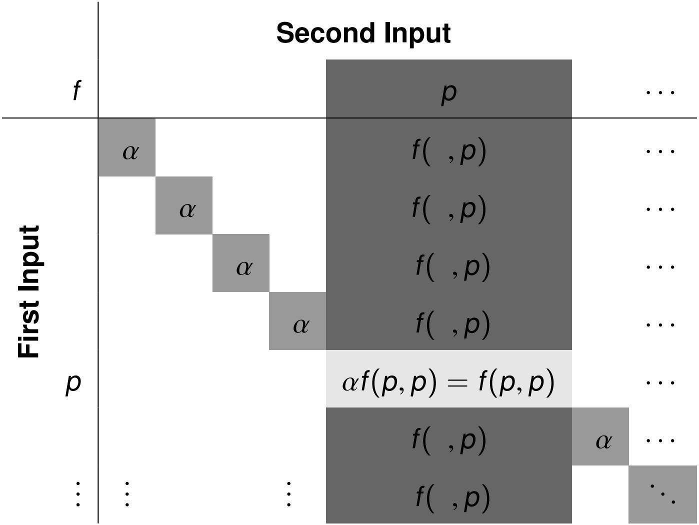
The contrapositive then says:

- If α forms a changed diagonal that is the same as some column, then the point where the diagonal and the column meet will be a fixed point of α.

The Contrapositive of the Main Theorem

The intuition for the contrapositive can be viewed in the next slide. The function g uses α and forms the changed diagonal of the matrix. The fact that g is representable by some column, say p, means that they are two ways of talking about the same thing: as a diagonal and as a column. At the crossing point, there is a fixed point where $\alpha(f(p,p)) = f(p,p)$.

The Contrapositive of the Main Theorem



Second Input

First Input

f					p		$\cdots$
	α				$f(\ \ ,p)$		$\cdots$
		α			$f(\ \ ,p)$		$\cdots$
			α		$f(\ \ ,p)$		$\cdots$
				α	$f(\ \ ,p)$		$\cdots$
p					$\alpha f(p,p) = f(p,p)$		$\cdots$
					$f(\ \ ,p)$	α	$\cdots$
$\vdots$	$\vdots$			$\vdots$	$f(\ \ ,p)$		$\ddots$

The Contrapositive of the Main Theorem

With this intuition in hand, let us state the contrapositive.

Theorem

- *Let* $\mathbb{A}$ *be a category with a terminal object and binary products.*
- *Let* y *be an object in the category and* $\alpha\colon y \longrightarrow y$ *be a morphism in the category.*
- *If there is an object* a *and a morphism* $f\colon a \times a \longrightarrow y$*, such that* $g = \alpha \circ f \circ \Delta$ *is representable by* f*, then* α *has a fixed point.*

The Contrapositive of the Main Theorem

Proof.

Let $\mathbb{A}$, a, and f be as described in the theorem. Let $g = \alpha \circ f \circ \Delta$ be representable by $p\colon 1 \longrightarrow a$, i.e., $g = f \circ (id_a \times p) \circ i$ (where i is the isomorphism $a \longrightarrow a \times 1$.) This commutative diagram is helpful:

$$\begin{tikzcd}
1 \arrow[r, "p"] \arrow[rrr, bend right, "\Delta\circ p"'] & a \arrow[r, "i", "\cong"'] \arrow[rrr, bend left, "g"] & a \times 1 \arrow[r, "id_a \times p"] & a \times a \arrow[r, "f"] & y
\end{tikzcd}$$

□

The Contrapositive of the Main Theorem

Proof.

Since g is represented by f, and g is defined as $g = \alpha \circ f \circ \Delta$, we have both equalities

$$f \circ (id_a \times p) \circ i = g = \alpha \circ f \circ \Delta.$$

Precomposing all three maps by p (i.e., plugging p into the equations) gives us

$$f \circ (id_a \times p) \circ i \circ p = g \circ p = \alpha \circ (f \circ \Delta \circ p).$$

The left side shortens to

$$(f \circ \Delta \circ p) = g \circ p = \alpha \circ (f \circ \Delta \circ p).$$

And here we see that $f \circ \Delta \circ p$ is a fixed point of α. □

Fixed Points in Logic

When we apply the contrapositive of Cantor's theorem to find fixed points in logic. First, some elementary logic.

- We are working in a system that can handle basic arithmetic.
- We will deal with logical formulas that accept at most one value which is a number.
- A logical formula that accepts one value will be called a *predicate* and will be written as $\mathcal{A}(x)$, $\mathcal{B}(x)$, $\mathcal{C}(x)$, etc.
- A logical formula that accepts no value (and hence is true or false) will be called a *sentence* and will be written as A, B, and C etc.
- Rather than considering the set of all predicates and the set of all sentences, we will be interested in equivalence classes of these sets.

Fixed Points in Logic

Since logical formulas are made out of a finite string of symbols (like programs), all logical formulas can be encoded as a natural number. We will write the natural number of a logical formula $\mathcal{A}(x)$ as $\ulcorner \mathcal{A}(x) \urcorner$ and the number of a sentence A as $\ulcorner A \urcorner$. By assigning a number to each formula, formulas can be made inputs to formulas. That means that logical formulas about numbers can then *evaluate* logical formulas about numbers. It is these numbers that will help logical formulas about numbers be self-referential.

Fixed Points in Logic

We are going to get fixed points of logical predicates.

Theorem

For every logical predicate that takes a number, $\mathcal{E}(x)$, there is a way of constructing a **fixed point** *which is a logical sentence C such that*

$$\mathcal{E}(\ulcorner C \urcorner) \equiv C$$

The process that goes from a $\mathcal{E}(x)$ to C is called a **fixed point machine**. *In a sense, C is a logical sentence that says*

"This logical statement has property $\mathcal{E}$."

With this fixed point machine we will find some of the most fascinating aspects of logic. These slides will skip the proof of the fixed point theorem and go directly to the applications.

Gödel’s Incompleteness Theorem

Now let us use this fixed-point machine for some interesting predicates $\mathcal{E}(x)$ and get self-referential statements. First some ideas about logic. Not only can we assign a unique number to every predicate and to every sentence, we can also assign a unique number to every statement in logic. This follows from the fact that statement are sequences of symbols. Every symbol can be given a unique number and every sequence of numbers can be given a unique number. Similarly, we can assign a unique natural number to every proof. After all, a proof is a sequence of statements. The numbers assigned to statements and proof are called the **Gödel numbers** of those statements and proofs.

Gödel's Incompleteness Theorem

Let $\mathrm{Prov}(x, y)$ be the two place predicate that is true when "y is the Gödel number of a proof of a statement whose Gödel number is x". Now we form the statement

$$\mathcal{E}(x) = (\forall y)\neg\mathrm{Prov}(y, x).$$

This is true for x when no number y is a proof of statement x, i.e., this is true for statement x when no proof of x exists. With the fixed point machine, we find a statement G (for Gödel) such that

$$G \equiv \mathcal{E}(\ulcorner G \urcorner) = (\forall y)\neg\mathrm{Prov}(y, \ulcorner G \urcorner)$$

G is a logical statement that essentially says

"This statement is not provable for any proof y"

or succinctly

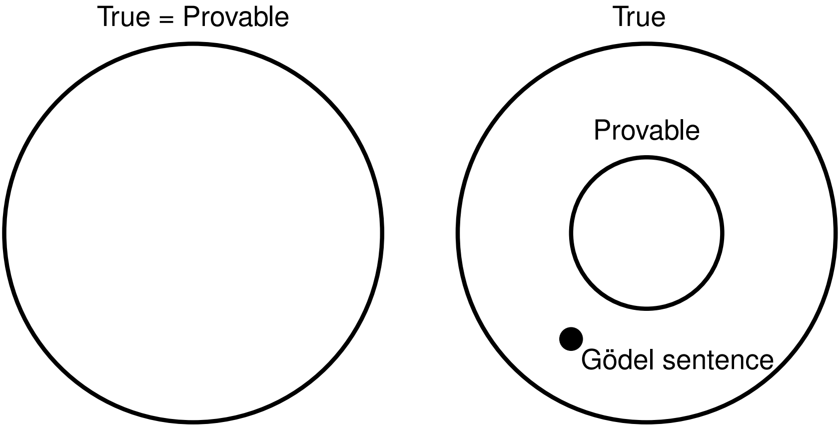
"This statement is unprovable."

Gödel's Incompleteness Theorem

Let us assume that the logical system is sound (i.e., you cannot prove false statements). If G was false then there would be a proof of G and hence there would be a proof of a false statement. In that case the system is not sound. On the other hand, if G is true, then it essentially says that G is true but unprovable.

This is one of the most important theorems in twentieth century mathematics and it is worthwhile to spend a few minutes meditating on its significance. Before Gödel came along, it was believed that any statement that is true is also provable. After Gödel, we can see that the set of provable statements is a proper subset of the set of statements that are true. There are statements that are true which cannot be proven within the particular system.

Gödel's Incompleteness Theorem



On the left: what was believed before Gödel's Theorem. On the right: what we know after Gödel's Theorem.

Tarski's Theorem

Alfred Tarski's theorem shows that a logical system cannot tell which of its predicates are true. Assume (wrongly) that there is some logical predicate $\mathcal{T}(x)$ that accepts a number and tells if the statement of that number is true. This formula will be true when "x is the Gödel number of a true statement in the theory". We can then use $\mathcal{T}(x)$ to form the statement

$$\mathcal{E}(x) = \neg\mathcal{T}(x)$$

This says that $\mathcal{E}(x)$ is true when $\mathcal{T}(x)$ is false. Now place $\mathcal{E}(x)$ into the fixed point machine. We will get a statement C such that

$$C \equiv \mathcal{E}(\ulcorner C \urcorner) = \neg\mathcal{T}(\ulcorner C \urcorner).$$

The logical sentence C essentially says

"This statement is false."

Tarski's Theorem

It is a logical version of the liar paradox. Logical sentence C is true if and only if it is false. A logical system cannot be consistent and have such a statement. The only assumption we made was that $\mathcal{T}(x)$ can be formulated in the language of the system. It follows that this assumption is false. So while Gödel showed us that certain logical systems have limitations with respect to the notion of provability, Tarski showed us that those logical systems cannot deal with their own truthfulness.

Parikh Sentences

Rohit Parikh used the fixed point machine to formulate some fascinating sentences that express properties about the length of proofs. Consider the two-place predicate $\mathrm{Prflen}(m, x)$ which is true if "there exists a proof of length m (in symbols) of a statement whose Gödel number is x." A computer can actually decide if this is true or false because there are only a finite number of proofs of length m. Now consider the logical formula

$$\mathcal{E}_n(x) = \neg(\exists m < n \quad \mathrm{Prflen}(m, x)).$$

The statement $\mathcal{E}_n(x)$ is true if the formula with Gödel number x does not have a proof whose length is less than n. With the fixed point machine, we find a sentence C_n that satisfies

$$C_n \equiv \mathcal{E}_n(\ulcorner C_n \urcorner) = \neg(\exists m < n \quad \mathrm{Prflen}(m, \ulcorner C_n \urcorner)).$$

Parikh Sentences

The logical sentence C_n essentially says

> "This statement does not have a proof of length less than n."

As long as the logical system is sound, C_n will be true and will not have a proof of length less than n. This is interesting in itself, however, Parikh went further. He showed that although C_n does not have a short proof (you can make n as large as you want,) there does exist a short proof of the fact that C_n is provable. With the predicate $\mathcal{P}(x) = \exists y \mathrm{Prov}(y, x)$, we will show that although there is no short proof of C_n, there is a short proof of $\mathcal{P}(\ulcorner C_n \urcorner)$.

Parikh Sentences

The short proof is basically a formalization of the following short argument.

1. If C_n does not have any proof, then C_n is true, i.e., $\neg\mathcal{P}(\ulcorner C_n \urcorner) \longrightarrow C_n$.
2. If C_n is true, we can check all proofs of length less then n and prove C_n, i.e., $C_n \longrightarrow \mathcal{P}(\ulcorner C_n \urcorner)$
3. From 1 and 2, we have shown that if C_n does not have a proof then we can prove C_n, i.e., $\neg\mathcal{P}(\ulcorner C_n \urcorner) \longrightarrow \mathcal{P}(\ulcorner C_n \urcorner)$. (This is not a contradiction. It just shows that $\neg\mathcal{P}(\ulcorner C_n \urcorner)$ cannot be true.)
4. We conclude that there is a proof of C_n, i.e., $\mathcal{P}(\ulcorner C_n \urcorner)$.

Parikh Sentences

Parikh went even further. He iterated $\mathcal{P}$ such that $\mathcal{P}^t(A) = \mathcal{P}(\ulcorner \mathcal{P}^{t-1}(A) \urcorner)$ and for every C there is a sequence

$$\mathcal{P}^0(\ulcorner C \urcorner) = C, \quad \mathcal{P}^1(\ulcorner C \urcorner), \quad \mathcal{P}^2(\ulcorner C \urcorner), \quad \ldots, \quad \mathcal{P}^k(\ulcorner C \urcorner), \ldots.$$

Parikh then showed that for every k there is a formula C_n^k that does not have a short proof, nor does the fact that it is provable have a short proof, nor that it is provable that it is provable ... have a short proof. However, eventually it will have a short proof. In symbols: There is no short proof of $\mathcal{P}^t(\ulcorner C_n^k \urcorner)$ for $t < k$ but there is a short proof of $\mathcal{P}^k(\ulcorner C_n^k \urcorner)$. This is just one of the many gems found in Rohit Parikh's papers.

Epimenides paradox

Before we close this mini-course it pays to look at two famous paradoxes that are not exactly instances of Cantor's theorem but are close enough that they are easy to describe. The two examples are (i) the Epimenides paradox (which is related to the liar's paradox) and (ii) the time travel paradox.

Chronologically, the granddaddy of all the self-referential paradoxes is the **Epimenides paradox**. Epimenides (6th or 7th century BC), a philosopher from Crete, was a curmudgeon who did not like his neighbors in Crete. He is quoted as saying that "All Cretans are liars." The problem is that Epimenides himself is a Cretan. He is talking about himself and his statement. If his statement is true, then this very utterance is also a lie and hence is not true. On the other hand, if what he is saying is false, then he is not a liar and what he said is true. This seems to be a contradiction.

Epimenides paradox

On deeper analysis, one sees that Epimenides' utterance is not a contradiction. If it is true that all Cretans are liars, that does not mean that every sentence that every Cretan ever made is false. A liar is someone who lied once; not necessarily someone who lies all the time. We have all lied and hence we are all liars! So the statement could be true and it will not negatively effect his own statement. On the other hand, if the statement is false, that implies that there is at least one pious Cretan who always tells the truth all the time. Presumably Epimenides thinks that he is that righteous truth teller. The statement could be true or false and it will not be a contradiction.

Epimenides paradox

Even though the paradox of Epimenides is flawed, there are other similar types of sentences that are paradoxical. The sentences

- "I always lie."
- "This sentence is false."
- **"The only sentence that is in boldface on this page is not true."**

are all declarative statements that are true if and only if they are false and hence they are contradictions. They are simple examples of paradoxical statements. There are many such statements and they are all instances of the **Liar Paradox**.

Let's back up a little and see what it going on here. Consider the statement "This sentence is false." It is an English sentence that refers to itself. Usually a declarative sentence refers to some object. For our purpose, language gets interesting because a sentence has the ability to refer to itself.

Epimenides paradox

Here are true sentences that refer to themselves:

- “This sentence has five words.”
- *“This sentence is in italics.”*
- “This is an example of a sentence that shows no originality.”

There are also false sentences that refer to themselves:

- “This sentence has six words.”
- “This sentence is in italics.”
- “This is an example of a sentence that shows originality.”
- “The world will little note, nor long remember what we say here...” (from Abraham Lincoln’s Gettysburg Address — arguably one of the most famous speeches ever made.)

The liar paradoxes are about sentences that go further. They are not false statements about themselves. Rather they are sentences that negate themselves.

Epimenides paradox

Let's look at the self reference of language in a formal way. There is a set of English sentences which we call *Sent*. We can describe a function $f\colon Sent \times Sent \longrightarrow 2$. The function f is defined for sentences s and s' as

$$f(s,s') = \begin{cases} 1 & : s \text{ is negated by } s' \\ 0 & : s \text{ is not negated } s'. \end{cases}$$

We are interested in sentences that (do not negate other sentences but) negate themselves. In order to deal with such sentences we are going to compose f with the function Δ and *NOT* as follows

$$\begin{tikzcd}
 & Sent \times Sent \arrow[r, "f"] & 2 \arrow[dr, "NOT"] & \\
Sent \arrow[ur, "\Delta"] \arrow[rrr, "g"'] & & & 2.
\end{tikzcd}$$

Epimenides paradox

The value $g(S)$ is defined as $NOT(f(S, S))$. The function g is the characteristic function of the subset of sentences that negate themselves. Till here, we have been mimicking the set-up of Cantor's theorem. However, this is where the resemblance stops. It is not clear what we would mean by talking about g being representable by f. What would it mean for a sentence S to represent a subset of sentences? Nevertheless, the sentences that negate themselves are declarative sentences that are contradictions.

Time Travel Paradoxes

A significant amount of science fiction is about time travel paradoxes. If time travel was possible, a time traveler might go back in time and shoot his bachelor grandfather, guaranteeing that the time traveler was never born. Homicidal behavior is not necessary to achieve such paradoxical results. The time traveler might just make sure that his parents never meet, or he might simply go back in time and make sure that he does not enter the time machine. These actions would imply a contradiction and hence cannot happen. The time traveler should not shoot his own grandfather (moral reasons notwithstanding) because if he shoots his own grandfather, he will not exist and will not be able to travel back in time to shoot his own grandfather. So by performing an action he is guaranteeing that the action cannot be performed. The event of shooting your own bachelor grandfather is self-referential. Usually, one event affects other events, but here an event affects itself. Since the physical universe does not permit contradictions, we must deny the assumption that time travel exists.

Time Travel Paradoxes

By now it is easy to formalize this paradox. There is a collection, *Events*, of all physical events. There is a function $f\colon \mathit{Events} \times \mathit{Events} \longrightarrow 2$ which is defined for two events e and e' as

$$f(e, e') = \begin{cases} 1 & : \text{if } e \text{ is negated by } e' \\ 0 & : \text{if } e \text{ is not negated } e'. \end{cases}$$

Time Travel Paradoxes

Some events negate other events and some events do not. If e and e' are not in each other's space-time cone then they will not effect each other. We now move on to describe the subset of all physical events that negate themselves in the usual way:

$$\begin{tikzcd}
 & Events \times Events \arrow[rr, "f"] & & 2 \arrow[d, "NOT"] \\
Events \arrow[ur, "\Delta"] \arrow[rrr, "g"'] & & & 2.
\end{tikzcd}$$

The function g is the characteristic function of those events that negate themselves. Such events cannot exist. Till here the pattern has been the same with Cantor's theorem. However, we do not go on to talk about representing g by f. What would it mean for $f(\quad, e)$ to represent a subset of events? While time travel paradoxes do not necessarily fit into Cantor's theorem, there is a sense of self-reference here and the events described by g cannot possibly exist.

Time Travel Paradoxes

Is there a resolution to the time travel paradox? Since events are part of the physical world, the obvious resolution is that time travel is not a possibility. However, we need not be so drastic. We can be a little subtle. Einstein's theory of relativity tells us that time travel is not possible in the usual way we think of the universe. However, Einstein's friend and neighbor, Kurt Gödel, wrote an interesting paper on relativity theory. The paper constructed a model of the universe in which time travel would be possible. In this "Gödel universe" it would be very hard, but not impossible, for time travel to be a reality. Gödel, the greatest logician of the past thousand years, was aware of the logical problems of time travel.

Time Travel Paradoxes

A writer, Rudy Rucker, tells of an interview with Gödel in which Rucker asks about the time-travel paradoxes. Gödel responded by saying “Time-travel is possible, but no person will ever manage to kill his past self ... The *a priori* is greatly neglected. Logic is very powerful.” That means that the universe simply will not allow you to kill your past self. Just as the barber paradox shows that certain villages with strict rules cannot exist, so too the physical universe will not allow you to perform an action that will cause a contradiction. This leads us to even more mind-blowing questions. What would happen if someone took a gun back in time to shoot an earlier version of himself? How will the universe stop him? Will he not have the free will to perform the dastardly deed? Will the gun fail to shoot? If the bullet fires and is properly aimed, will the bullet stop short of his body? It is indeed bewildering to live in a world with self reference.

Self-Referential Paradoxes

Our tour is over. We hoped you enjoyed it!

Next stop: Chapter 4. How different categories relate with each other.