Equivalences

Samuel Mimram

2025

École polytechnique

Constructing identities

For now, we cannot construct all the useful identities we would need.

In fact, nothing forces (yet) equality to be non-trivial and thus useful:
for instance, we have a set-theoretic model.

In order to fix this, we will add the univalence axiom which roughly states:

equivalent types are equal.

We first need to make precise what we mean by equivalence.

Contractible types

The types

and

respectively correspond to

- one point x
- two points a and b which are equal

We therefore expect them to be identified, i.e. equal.

However the following space should not be identified to them

because there are 2 (actually, $\mathbb{Z}$) identifications of a and b.

Contractible types

The types

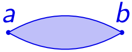
and

respectively correspond to

- one point x
- two points a and b which are equal

We therefore expect them to be identified, i.e. equal.

However the following space should not be identified to them

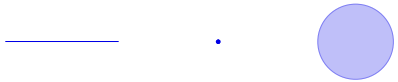
because there are 2 (actually, $\mathbb{Z}$) identifications of a and b. But 

is trivial.

Equivalences

When do we expect two types to be equivalent?

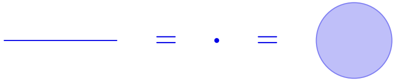
- contractible types:



Equivalences

When do we expect two types to be equivalent?

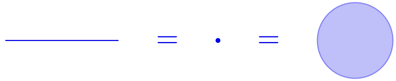
- contractible types: always



Equivalences

When do we expect two types to be equivalent?

- contractible types: always

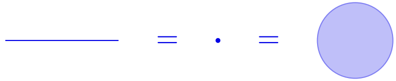


- propositions:

Equivalences

When do we expect two types to be equivalent?

- contractible types: always



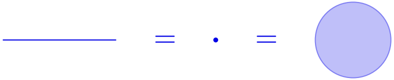
- propositions: when they are equivalent

$$(A \to B) \wedge (B \to A)$$

Equivalences

When do we expect two types to be equivalent?

- contractible types: always



- propositions: when they are equivalent

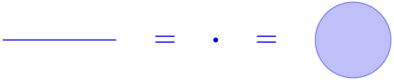
$$(A \to B) \land (B \to A)$$

- sets:

Equivalences

When do we expect two types to be equivalent?

- contractible types: always



- propositions: when they are equivalent

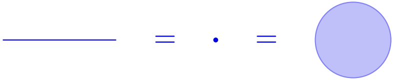
$$(A \to B) \wedge (B \to A)$$

- sets: when isomorphic

Equivalences

When do we expect two types to be equivalent?

- contractible types: always



- propositions: when they are equivalent

$$(A \to B) \wedge (B \to A)$$

- sets: when isomorphic
- ...

Equivalences

We expect that the following notion is suitable:

Definition
A map $f : A \to B$ is an **isomorphism** when there exists $g : B \to A$ such that

$$g \circ f = \mathrm{id}_A \qquad \text{and} \qquad f \circ g = \mathrm{id}_B\,.$$

Excepting that we need to make sure that details are right:

- we have to make sure that we can compare functions in the right way (we don't want to assume funext)
- we have to make sure that we can say "exists", i.e. being a quasi-inverse is a proposition.

Equivalences

We expect that the following notion is suitable:

Definition
A map $f : A \to B$ is an **isomorphism** when there exists $g : B \to A$ such that

$$g \circ f = \mathrm{id}_A \qquad \text{and} \qquad f \circ g = \mathrm{id}_B \,.$$

Excepting that we need to make sure that details are right:

- we have to make sure that we can compare functions in the right way (we don't want to assume funext)
- we have to make sure that we can say "exists", i.e. being a quasi-inverse is a proposition.

This is not the case here.

Equivalences

For the first point (comparison of functions), suppose that we want to show that $\mathrm{id} : A \to A$ is an isomorphism with id as inverse.

We need to be able to show

$$\mathrm{id} \circ \mathrm{id} = \mathrm{id}$$

which we cannot without function extensionality.

Homotopy

For the first problem there is an easy fix. Instead of requiring

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

we require

$$(x : A) \to g\,(f\,x) = x$$

what we will write

$$g \circ f \sim \mathrm{id}$$

Bi-invertible maps

For the second problem there is also an easy fix:

Definition
A map $f : A \to B$ is an **isomorphism** when there exists $g : B \to A$ such that

$$g \circ f = \mathrm{id}_A \qquad \text{and} \qquad f \circ g = \mathrm{id}_B\,.$$

For the second problem there is also an easy fix:

Definition
A map $f : A \to B$ is **quasi-invertible** when there exists $g : B \to A$ such that

$$g \circ f \sim \mathrm{id}_A \qquad \text{and} \qquad f \circ g \sim \mathrm{id}_B \,.$$

For the second problem there is also an easy fix:

Definition
A map $f : A \to B$ is **bi-invertible** when there exists $g : B \to A$ and $g' : B \to A$ such that

$$g \circ f \sim \mathrm{id}_A \qquad \text{and} \qquad f \circ g' \sim \mathrm{id}_B\,.$$

Bi-invertible maps

For the second problem there is also an easy fix:

Definition
A map $f : A \to B$ is **bi-invertible** when there exists $g : B \to A$ and $g' : B \to A$ such that

$$g \circ f \sim \mathrm{id}_A \qquad \text{and} \qquad f \circ g' \sim \mathrm{id}_B \, .$$

Let us understand why this is the case.

Part I

Equivalences in category theory

TODO: the fundamental groupoid is characterized by the fundamental group (when path connected)

Categories

Recall that a **category** C consists of

- a set of objects $x \in C$
- sets of morphisms $C(x, y)$ for $x, y \in C$
- composition operations

$$x \xrightarrow{f} y \xrightarrow{g} z \qquad \rightsquigarrow \qquad x \xrightarrow{g \circ f} z$$

- identities
- satisfying axioms

Categories

Recall that a **category** C consists of

- a set of objects $x \in C$
- sets of morphisms $C(x, y)$ for $x, y \in C$
- composition operations

$$x \xrightarrow{f} y \xrightarrow{g} z \qquad \rightsquigarrow \qquad x \xrightarrow{g \circ f} z$$

- identities
- satisfying axioms

A **groupoid** is a category in which every morphism is invertible.

Typical example: the fundamental groupoid of a space.

Functors between categories

A functor $F : C \to D$ between categories consists in

- a function $F : C \to D$ from the objects of C to those of D,
- for every $x, y : C$, a function $C(x, y) \to D(Fx, Fy)$
- in a way which respects composition and identities

Functors between categories

A functor $F : C \to D$ between categories consists in

- a function $F : C \to D$ from the objects of C to those of D,
- for every $x, y : C$, a function $C(x, y) \to D(Fx, Fy)$
- in a way which respects composition and identities

A functor $F : C \to D$ is an **isomorphism** when there exists $G : D \to C$ such that $G \circ F = \mathrm{Id}_C$ and $F \circ G = \mathrm{Id}_D$.

Functors between categories

A functor $F : C \to D$ between categories consists in

- a function $F : C \to D$ from the objects of C to those of D,
- for every $x, y : C$, a function $C(x, y) \to D(Fx, Fy)$
- in a way which respects composition and identities

A functor $F : C \to D$ is an **equivalence** when there exists $G : D \to C$ such that $G \circ F \sim \mathrm{Id}$ and $F \circ G \sim \mathrm{Id}_D$.

Lemma

Given invertible morphisms $f : x' \to x$ and $h : y \to y'$ in a category C, we have $C(x, y) \cong C(x', y)$:

$$\begin{tikzcd} x \arrow[r, "f"] \arrow[rr, bend left=60, dotted] \arrow[rr, bend right=60, dotted] & x' \arrow[r, bend left] \arrow[r, bend right] \arrow[r, phantom, "C(x',y)"] & y \end{tikzcd}$$

Groupoids

Lemma
Given invertible morphisms $f : x' \to x$ and $h : y \to y'$ in a category C, we have $C(x, y) \cong C(x', y)$:

$$
\begin{tikzcd}
x \arrow[r, "f"] \arrow[rr, bend left=60, dotted] \arrow[rr, bend right=60, dotted] & x' \arrow[r, bend left] \arrow[r, bend right] \arrow[r, phantom, "C(x',y)"] & y
\end{tikzcd}
$$

Proof.
We have maps

$$
\begin{aligned}
C(x', y) &\to C(x, y) \\
g &\mapsto g \circ f
\end{aligned}
\qquad\qquad
\begin{aligned}
C(x, y) &\to C(x', y) \\
g &\mapsto g \circ f^{-1}
\end{aligned}
$$

$\square$

A groupoid is **connected** when there is a morphism $f_{x,y} : x \to y$ between any two objects x, y.

A groupoid is **connected** when there is a morphism $f_{x,y} : x \to y$ between any two objects x, y.

Proposition

Given a groupoid C and an object x, we have a group $C(x,x)$.

When C is connected, we have $\phi_{y,z} : C(y,z) \xrightarrow{\sim} C(x,x)$ and composition corresponds to multiplication: for $f \in C(y,z)$ and $g : C(z,w)$, we have $\phi(g \circ f) = \phi g \circ \phi f$.

In particular $C(y,y) \cong C(x,x)$ as a group.

Part II

Equivalences in topology

Let's investigate the situation on the topological side.

Topological spaces are difficult to study and classify, which should instead study some equivalence classes of such spaces.

Identity

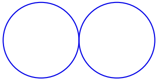
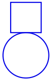
The strict notion of equality is clearly too fine: for instance, we want to identify the spaces

$$\overset{x}{\bullet} \qquad \overset{y}{\bullet} \qquad\qquad \text{and} \qquad\qquad \overset{a}{\bullet} \qquad \overset{b}{\bullet}$$

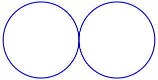
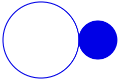
Homeomorphisms

An **homeomorphism** $f : A \to B$ is a continuous map admitting a continuous inverse.

For instance,

 $\cong$ 

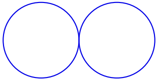
but

 $\not\cong$ 

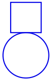
Homeomorphisms

An **homeomorphism** $f : A \to B$ is a continuous map admitting a continuous inverse.

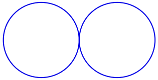
For instance,



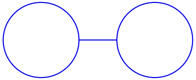
$\cong$



but

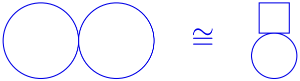


$\ncong$



An **homeomorphism** $f : A \to B$ is a continuous map admitting a continuous inverse.

For instance,



but

This notion is too strong!

Homotopy

A **homotopy** between two maps $f, g : A \to B$ is a continuous map

$$\alpha : \mathsf{I} \to A \to B$$

with $\mathsf{I} = [0, 1]$, such that

$$\alpha\, 0 = f \qquad\qquad \alpha\, 1 = g$$

Homotopy

A **homotopy** between two maps $f, g : A \to B$ is a continuous map

$$\alpha : \mathsf{I} \to A \to B$$

with $\mathsf{I} = [0, 1]$, such that

$$\alpha\, 0 = f \qquad\qquad \alpha\, 1 = g$$

We can also see this as a continuous map

$$\alpha : A \to I \to B$$

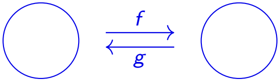
which means that for every $x \in A$ we should have paths

$$f\, x = g\, x$$

Homotopy equivalences

A **homotopy equivalence** $f : A \to B$ is a continuous map such that there is a continuous map $g : B \to A$ with $g \circ f \sim \mathrm{id}$ and $f \circ g \sim \mathrm{id}$.

For instance



Weak and homotopy equivalences in topology

fundamental group (quotient by homotopy = path between paths)

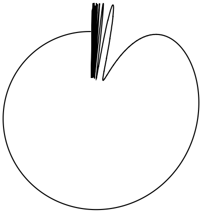
higher groups

weak homotopy equivalences

existence of a whe is reflexive and transitive, but not symmetric in general

homotopy equivalences

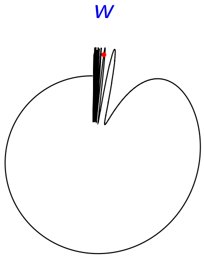
homotopy equivalence in topology



Consider the **extended topologist's sine curve** aka the **Warsaw circle** W, which consists of the following points in $\mathbb{C}$:

$$W = \left\{ \left(1 + \frac{1}{2}\sin\left(\frac{1}{\theta}\right)\right) e^{\mathrm{i}\,\theta} \mid \theta \in]0, 2\pi] \right\}$$

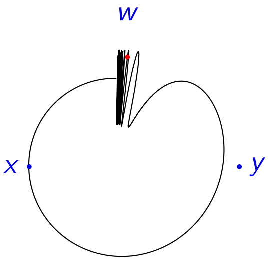
The Warsaw circle



Consider the **extended topologist's sine curve** aka the **Warsaw circle** W, which consists of the following points in $\mathbb{C}$:

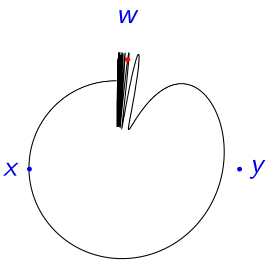
$$W \quad = \quad \left\{ \left(1 + \frac{1}{2}\sin\left(\frac{1}{\theta}\right)\right) \mathrm{e}^{\mathrm{i}\,\theta} \;\middle|\; \theta \in]0, 2\pi] \right\}$$

The Warsaw circle



Given two points x and y, there is no *continuous* path from x to y going through w.

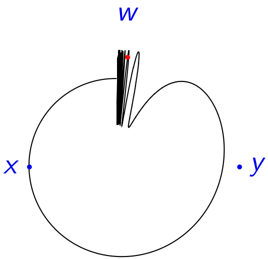
The Warsaw circle



Given two points x and y, there is no *continuous* path from x to y going through w.

As a consequence, there is no non-trivial map $p : \mathsf{S}^1 \to W$.

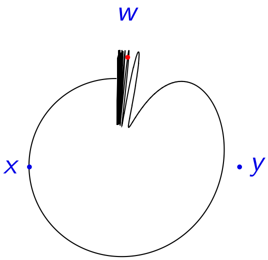
The Warsaw circle



Given two points x and y, there is no *continuous* path from x to y going through w.

As a consequence, there is no non-trivial map $p : \mathsf{S}^1 \to W$.

Therefore $\pi_1(W) = 1$.



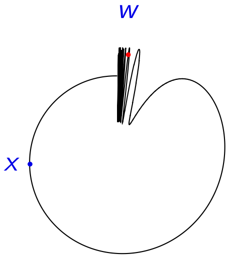
Given two points x and y, there is no *continuous* path from x to y going through w.

As a consequence, there is no non-trivial map $p : \mathsf{S}^1 \to W$.

Therefore $\pi_1(W) = 1$.

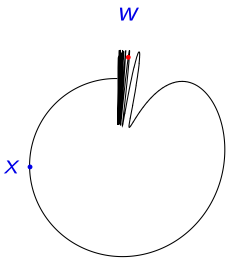
More generally $\pi_k(W) = 1$, and W is thus weakly equivalent to 1.

The Warsaw circle



Consider the map $f : 1 \to W$ pointing at x. We have seen that this is a weak homotopy equivalence.

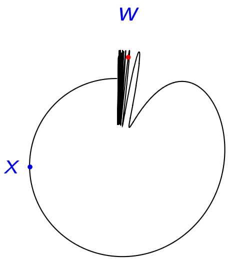
The Warsaw circle



Consider the map $f : 1 \to W$ pointing at x. We have seen that this is a weak homotopy equivalence.

But this is not a homotopy equivalence:

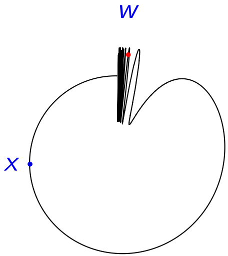
The Warsaw circle



Consider the map $f : 1 \to W$ pointing at x. We have seen that this is a weak homotopy equivalence.

But this is not a homotopy equivalence: we would need a map $g : W \to 1$ such that $f \circ g \sim \mathrm{id}_W$,

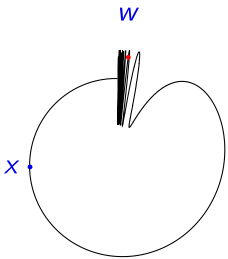
The Warsaw circle



Consider the map $f : 1 \to W$ pointing at x. We have seen that this is a weak homotopy equivalence.

But this is not a homotopy equivalence: we would need a map $g : W \to 1$ such that $f \circ g \sim \mathrm{id}_W$, i.e. a family of paths $p_y : y \rightsquigarrow x$ continuous in y,

The Warsaw circle



Consider the map $f : 1 \to W$ pointing at x. We have seen that this is a weak homotopy equivalence.

But this is not a homotopy equivalence: we would need a map $g : W \to 1$ such that $f \circ g \sim \mathrm{id}_W$, i.e. a family of paths $p_y : y \rightsquigarrow x$ continuous in y, but one of those paths would have to go through w!

Lemma
Every homotopy equivalence is a weak homotopy equivalence.

Proof.
We have

$$\begin{tikzcd}
\pi_n(A,a) \arrow[rr, "\sim"] \arrow[dr] & & \pi_n(A,g(f(a))) \arrow[dr] & \\
& \pi_n(A,f(a)) \arrow[ur] \arrow[rr, "\sim"'] & & \pi_n(A,f(g(f(a))))
\end{tikzcd}$$

By the 2-out-of-6 property, all the diagonal maps are equivalences (the first one in particular). □

Definition

S^1, S^2, Mobius, torus, Klein

Theorem ([Whi49a, Whi49b])
A map $f : A \to B$ between CW-complexes A and B is a homotopy equivalence iff it is a weak homotopy equivalence.

Theorem ([Whi49a, Whi49b])
A map $f : A \to B$ between CW-complexes A and B is a homotopy equivalence iff it is a weak homotopy equivalence.

Theorem (CW approximation)
For every space A there exists a space $\tilde{A}$ and a weak homotopy equivalence $\tilde{A} \to A$.

Part III

Equivalences

Homotopies

A **homotopy** between two functions $f\, g : (x : A) \to B(x)$ is a function of type

$$(f \sim g) \quad \hat{=} \quad (x : A) \to f\, x = g\, x$$

Homotopies

Lemma ([Uni13, Lemma 2.4.2])
Homotopy is an equivalence relation: we have

- *reflexivity:* $f \sim f$
- *symmetry:* $f \sim g \to g \sim f$
- *transitivity:* $f \sim g \to g \sim h \to f \sim h$

for $f\ g\ h : (x : A) \to B(x)$.

Lemma ([Uni13, Lemma 2.4.2])
Homotopy is an equivalence relation: we have

- *reflexivity:* $f \sim f$
- *symmetry:* $f \sim g \to g \sim f$
- *transitivity:* $f \sim g \to g \sim h \to f \sim h$

for $f\ g\ h : (x : A) \to B(x)$.

Proof.
We can consider:

$$: \quad (x : A) \to f\,x = f\,x$$

Homotopies

Lemma ([Uni13, Lemma 2.4.2])
Homotopy is an equivalence relation: we have

- *reflexivity:* $f \sim f$
- *symmetry:* $f \sim g \to g \sim f$
- *transitivity:* $f \sim g \to g \sim h \to f \sim h$

for $f\ g\ h : (x : A) \to B(x)$.

Proof.
We can consider:

$$\lambda x.\,\mathsf{refl} \quad : \quad (x : A) \to f\ x = f\ x$$

Lemma ([Uni13, Lemma 2.4.2])
Homotopy is an equivalence relation: we have

- *reflexivity:* $f \sim f$
- *symmetry:* $f \sim g \to g \sim f$
- *transitivity:* $f \sim g \to g \sim h \to f \sim h$

for $f\ g\ h : (x : A) \to B(x)$.

Proof.
We can consider:

$$\lambda x.\,\mathsf{refl} \quad : \quad (x : A) \to f\,x = f\,x$$

and

$$: \quad ((x : A) \to f\,x = g\,x) \to (x : A) \to g\,x = f\,x$$

Lemma ([Uni13, Lemma 2.4.2])
Homotopy is an equivalence relation: we have

- *reflexivity:* $f \sim f$
- *symmetry:* $f \sim g \to g \sim f$
- *transitivity:* $f \sim g \to g \sim h \to f \sim h$

for $f\ g\ h : (x : A) \to B(x)$.

Proof.
We can consider:

$$\lambda x.\,\mathsf{refl} \quad : \quad (x : A) \to f\,x = f\,x$$

and

$$\lambda p.\lambda x.(p\,x)^{-} \quad : \quad ((x : A) \to f\,x = g\,x) \to (x : A) \to g\,x = f\,x$$

Homotopies

Lemma ([Uni13, Lemma 2.4.2])
Homotopy is an equivalence relation: we have

- *reflexivity:* $f \sim f$
- *symmetry:* $f \sim g \to g \sim f$
- *transitivity:* $f \sim g \to g \sim h \to f \sim h$

for $f\ g\ h : (x : A) \to B(x)$.

Proof.
We can consider:

$$\lambda x.\,\mathrm{refl} \quad : \quad (x : A) \to f\,x = f\,x$$

and

$$\lambda p.\lambda x.(p\,x)^{-} \quad : \quad ((x : A) \to f\,x = g\,x) \to (x : A) \to g\,x = f\,x$$

and

$$: \quad ((x : A) \to f\,x = g\,x) \to ((x : A) \to g\,x = h\,x) \to (x : A) \to f\,x =$$

Homotopies

Lemma ([Uni13, Lemma 2.4.2])
Homotopy is an equivalence relation: we have

- *reflexivity:* $f \sim f$
- *symmetry:* $f \sim g \to g \sim f$
- *transitivity:* $f \sim g \to g \sim h \to f \sim h$

for $f\ g\ h : (x : A) \to B(x)$.

Proof.
We can consider:

$$\lambda x.\,\mathsf{refl} \quad : \quad (x : A) \to f\,x = f\,x$$

and

$$\lambda p.\lambda x.(p\,x)^{-} \quad : \quad ((x : A) \to f\,x = g\,x) \to (x : A) \to g\,x = f\,x$$

and

$$\lambda p.\lambda q.\lambda x \to p\,x \cdot q\,x \quad : \quad ((x : A) \to f\,x = g\,x) \to ((x : A) \to g\,x = h\,x) \to (x : A) \to f\,x =$$

Homotopies: a 2-category

We have a **2-category** categories, functors and natural transformations:

$$\begin{tikzcd} \mathcal{C} \arrow[r, bend left, "F"{name=U}] \arrow[r, bend right, "G"'{name=D}] & \mathcal{D} \arrow[from=U, to=D, phantom, "\alpha\Downarrow"] \end{tikzcd}$$

Similarly, we have something like a 2-category of types, functions and homotopies:

$$\begin{tikzcd} A \arrow[r, bend left, "f"{name=U}] \arrow[r, bend right, "g"'{name=D}] & B \arrow[from=U, to=D, phantom, "\alpha\Downarrow"] \end{tikzcd}$$

with $\alpha : f \sim g$.

Homotopies: a 2-category

The fact that $\sim$ is an "equivalence relation" can be pictured as

$$\text{reflexivity:} \qquad A \xrightarrow{f} B \quad \rightsquigarrow \quad \begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[r, bend right, "f"'] \arrow[r, phantom, "\mathrm{id}\Downarrow"] & B \end{tikzcd}$$

Homotopies: a 2-category

The fact that $\sim$ is an "equivalence relation" can be pictured as

$$\text{reflexivity:} \quad A \xrightarrow{f} B \quad \rightsquigarrow \quad \begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[r, bend right, "f"'] \arrow[r, phantom, "\mathrm{id}\Downarrow"] & B \end{tikzcd}$$

$$\text{transitivity:} \quad \begin{tikzcd} A \arrow[r, bend left=60, "f"] \arrow[r, "g" description] \arrow[r, bend right=60, "h"'] \arrow[r, phantom, shift left=3, "\alpha\Downarrow"] \arrow[r, phantom, shift right=3, "\beta\Downarrow"] & B \end{tikzcd} \quad \rightsquigarrow \quad \begin{tikzcd} A \arrow[r, bend left=60, "f"] \arrow[r, bend right=60, "h"'] \arrow[r, phantom, "\beta\circ\alpha\Downarrow"] & B \end{tikzcd}$$

Homotopies: a 2-category

The fact that $\sim$ is an "equivalence relation" can be pictured as

reflexivity: $A \xrightarrow{f} B \quad \rightsquigarrow \quad \begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[r, bend right, "f"'] \arrow[r, phantom, "\mathrm{id}\Downarrow"] & B \end{tikzcd}$

transitivity: $\begin{tikzcd} A \arrow[r, bend left=60, "f"] \arrow[r, "g" description] \arrow[r, bend right=60, "h"'] \arrow[r, phantom, bend left=30, "\alpha\Downarrow"] \arrow[r, phantom, bend right=30, "\beta\Downarrow"] & B \end{tikzcd} \quad \rightsquigarrow \quad \begin{tikzcd} A \arrow[r, bend left=60, "f"] \arrow[r, bend right=60, "h"'] \arrow[r, phantom, "\beta\circ\alpha\Downarrow"] & B \end{tikzcd}$

symmetry: $\begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[r, bend right, "g"'] \arrow[r, phantom, "\alpha\Downarrow"] & B \end{tikzcd} \quad \rightsquigarrow \quad \begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[r, bend right, "g"'] \arrow[r, phantom, "\Uparrow\alpha^{-1}"] & B \end{tikzcd}$

Homotopies: whiskering

Proposition
As in a 2-category, we also have ***whiskering*** *compositions:*

- *left whiskering:*

$$\begin{tikzcd} A \arrow[r, "f"] & B \arrow[r, bend left, "g"] \arrow[r, bend right, "g'"'] \arrow[r, phantom, "\alpha\Downarrow"] & C \end{tikzcd} \quad \rightsquigarrow \quad \begin{tikzcd} A \arrow[r, bend left, "g\circ f"] \arrow[r, bend right, "g'\circ f"'] \arrow[r, phantom, "\alpha_f \Downarrow"] & C \end{tikzcd}$$

- *right whiskering:*

$$\begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[r, bend right, "f'"'] \arrow[r, phantom, "\alpha\Downarrow"] & B \arrow[r, "g"] & C \end{tikzcd} \quad \rightsquigarrow \quad \begin{tikzcd} A \arrow[r, bend left, "g\circ f"] \arrow[r, bend right, "g\circ f'"'] \arrow[r, phantom, "g\alpha \Downarrow"] & C \end{tikzcd}$$

Proposition
As in a 2-category, we also have ***whiskering*** *compositions:*

- *left whiskering:*

$$\begin{tikzcd} A \arrow[r, "f"] & B \arrow[r, bend left, "g"] \arrow[r, bend right, "g'"'] \arrow[r, phantom, "\alpha\Downarrow"] & C \end{tikzcd} \quad \rightsquigarrow \quad \begin{tikzcd} A \arrow[r, bend left, "g \circ f"] \arrow[r, bend right, "g' \circ f"'] \arrow[r, phantom, "\alpha_f \Downarrow"] & C \end{tikzcd}$$

- *right whiskering:*

$$\begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[r, bend right, "f'"'] \arrow[r, phantom, "\alpha\Downarrow"] & B \arrow[r, "g"] & C \end{tikzcd} \quad \rightsquigarrow \quad \begin{tikzcd} A \arrow[r, bend left, "g \circ f"] \arrow[r, bend right, "g \circ f'"'] \arrow[r, phantom, "g\alpha \Downarrow"] & C \end{tikzcd}$$

Proof.
Define $\alpha_f\, x \triangleq$ and $g\, \alpha\, x \triangleq$. □

Homotopies: whiskering

Proposition
As in a 2-category, we also have ***whiskering*** *compositions:*

- *left whiskering:*

$$\begin{tikzcd} A \arrow[r, "f"] & B \arrow[r, bend left, "g"] \arrow[r, bend right, "g'"'] \arrow[r, phantom, "\alpha\Downarrow"] & C \end{tikzcd} \quad\rightsquigarrow\quad \begin{tikzcd} A \arrow[r, bend left, "g \circ f"] \arrow[r, bend right, "g' \circ f"'] \arrow[r, phantom, "\alpha_f \Downarrow"] & C \end{tikzcd}$$

- *right whiskering:*

$$\begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[r, bend right, "f'"'] \arrow[r, phantom, "\alpha\Downarrow"] & B \arrow[r, "g"] & C \end{tikzcd} \quad\rightsquigarrow\quad \begin{tikzcd} A \arrow[r, bend left, "g \circ f"] \arrow[r, bend right, "g \circ f'"'] \arrow[r, phantom, "g\alpha \Downarrow"] & C \end{tikzcd}$$

Proof.
Define $\alpha_f\, x \mathrel{\hat{=}} \alpha\,(f\, x)$ and $g\,\alpha\,x \mathrel{\hat{=}}$. $\square$

Homotopies: whiskering

Proposition
As in a 2-category, we also have ***whiskering*** *compositions:*

- *left whiskering:*

$$\begin{tikzcd} A \arrow[r, "f"] & B \arrow[r, bend left, "g"{name=U}] \arrow[r, bend right, "g'"'{name=D}] \arrow[from=U, to=D, Rightarrow, "\alpha"'] & C \end{tikzcd} \quad \rightsquigarrow \quad \begin{tikzcd} A \arrow[rr, bend left, "g \circ f"{name=U}] \arrow[rr, bend right, "g' \circ f"'{name=D}] \arrow[from=U, to=D, Rightarrow, "\alpha_f"'] & & C \end{tikzcd}$$

- *right whiskering:*

$$\begin{tikzcd} A \arrow[r, bend left, "f"{name=U}] \arrow[r, bend right, "f'"'{name=D}] \arrow[from=U, to=D, Rightarrow, "\alpha"'] & B \arrow[r, "g"] & C \end{tikzcd} \quad \rightsquigarrow \quad \begin{tikzcd} A \arrow[rr, bend left, "g \circ f"{name=U}] \arrow[rr, bend right, "g \circ f'"'{name=D}] \arrow[from=U, to=D, Rightarrow, "g\alpha"'] & & C \end{tikzcd}$$

Proof.
Define $\alpha_f\, x \triangleq \alpha\,(f\,x)$ and $g\,\alpha\,x \triangleq \mathsf{ap}\,g\,(\alpha\,x)$. □

Homotopies satisfy the following **naturality** property:

Lemma ([Uni13, Lemma 2.4.3])
Given functions $f\ g : A \to B$, *homotopy* $\alpha : f \sim g$ *and equality* $p : x = y$ *in* A, *we have*

$$\alpha\, x \cdot \mathsf{ap}\ g\ p = \mathsf{ap}\ f\ p \cdot \alpha\, y$$

i.e.

$$\begin{tikzcd}
f\,x \arrow[r, equal, "\alpha\,x"] \arrow[d, equal, "\mathsf{ap}\ f\ p"'] & g\,x \arrow[d, equal, "\mathsf{ap}\ g\ p"] \\
f\,y \arrow[r, equal, "\alpha\,y"'] & g\,y
\end{tikzcd}$$

Homotopy: naturality

Homotopies satisfy the following **naturality** property:

Lemma ([Uni13, Lemma 2.4.3])
Given functions $f\ g : A \to B$, *homotopy* $\alpha : f \sim g$ *and equality* $p : x = y$ *in* A, *we have*

$$\alpha\, x \cdot \mathsf{ap}\ g\ p = \mathsf{ap}\ f\ p \cdot \alpha\, y$$

i.e.

$$\begin{tikzcd}
f\ x \arrow[r, equal, "\alpha\, x"] \arrow[d, equal, "\mathsf{ap}\ f\ p"'] & g\ x \arrow[d, equal, "\mathsf{ap}\ g\ p"] \\
f\ y \arrow[r, equal, "\alpha\, y"'] & g\ y
\end{tikzcd}$$

Proof.
By path induction on p, it is enough to show

$$\alpha\, x \cdot \mathsf{ap}\ g\ \mathsf{refl} = \qquad\qquad \mathsf{ap}\ f\ \mathsf{refl} \cdot \alpha\, x \qquad \square$$

Homotopy: naturality

Homotopies satisfy the following **naturality** property:

Lemma ([Uni13, Lemma 2.4.3])
Given functions $f\ g : A \to B$, *homotopy* $\alpha : f \sim g$ *and equality* $p : x = y$ *in* A, *we have*

$$\alpha\, x \cdot \mathsf{ap}\ g\ p = \mathsf{ap}\ f\ p \cdot \alpha\, y$$

i.e.

$$\begin{tikzcd}
f\,x \arrow[r, equal, "\alpha\,x"] \arrow[d, equal, "\mathsf{ap}\ f\ p"'] & g\,x \arrow[d, equal, "\mathsf{ap}\ g\ p"] \\
f\,y \arrow[r, equal, "\alpha\,y"'] & g\,y
\end{tikzcd}$$

Proof.
By path induction on p, it is enough to show

$$\alpha\, x \cdot \mathsf{ap}\ g\ \mathsf{refl} = \ = \alpha\, x \cdot \mathsf{refl} = \alpha\, x = \mathsf{refl} \cdot \alpha\, x = \ \mathsf{ap}\ f\ \mathsf{refl} \cdot \alpha\, x \qquad \square$$

Quasi-invertible maps

This suggests that we consider the following notion of equivalence:

Definition
For a map $f : A \to B$, a **quasi-inverse** is $g : B \to A$ such that

$$g \circ f \sim \mathrm{id}_A \qquad\qquad f \circ g \sim \mathrm{id}_B$$

Quasi-invertible maps

This suggests that we consider the following notion of equivalence:

Definition
For a map $f : A \to B$, a **quasi-inverse** is $g : B \to A$ such that

$$g \circ f \sim \mathrm{id}_A \qquad\qquad f \circ g \sim \mathrm{id}_B$$

We write

$$\mathrm{hasQInv}(f) \quad \triangleq \quad \Sigma(g : B \to A).(g \circ f \sim \mathrm{id}\, A) \times (f \circ g \sim \mathrm{id}_B)$$

for the type of proofs that f admits a quasi-inverse.

Quasi-invertible maps

This suggests that we consider the following notion of equivalence:

Definition
For a map $f : A \to B$, a **quasi-inverse** is $g : B \to A$ such that

$$g \circ f \sim \mathrm{id}_A \qquad\qquad f \circ g \sim \mathrm{id}_B$$

We write

$$\mathrm{hasQInv}(f) \quad \hat{=} \quad \Sigma(g : B \to A).(g \circ f \sim \mathrm{id}\, A) \times (f \circ g \sim \mathrm{id}_B)$$

for the type of proofs that f admits a quasi-inverse.

The reason why this is not called an equivalence is that this <u>not</u> the right notion.

Quasi-invertible maps

This suggests that we consider the following notion of equivalence:

Definition
For a map $f : A \to B$, a **quasi-inverse** is $g : B \to A$ such that

$$g \circ f \sim \mathrm{id}_A \qquad\qquad f \circ g \sim \mathrm{id}_B$$

We write

$$\mathrm{hasQInv}(f) \quad \hat{=} \quad \Sigma(g : B \to A).(g \circ f \sim \mathrm{id}\, A) \times (f \circ g \sim \mathrm{id}_B)$$

for the type of proofs that f admits a quasi-inverse.

The reason why this is not called an equivalence is that this <u>not</u> the right notion.

Being quasi-invertible is not a property: we might have two non-equal inverses g and g'!

We cannot say “is quasi-invertible”!

Quasi-invertible maps are not right

Consider

$$f : A \to B$$

which can be pictured as

$$\begin{tikzcd} A \arrow[r, bend left, "f"] & B \end{tikzcd}$$

Quasi-invertible maps are not right

Consider

$$f : A \to B \qquad\qquad g : B \to A$$

which can be pictured as

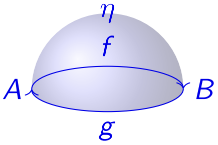
$$\begin{tikzcd} A \arrow[r, bend left, "f"] & B \arrow[l, bend left, "g"] \end{tikzcd}$$

Quasi-invertible maps are not right

Consider

$$f : A \to B \qquad g : B \to A \qquad \eta : g \circ f \sim \mathrm{id}_A$$

which can be pictured as

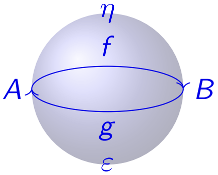


Quasi-invertible maps are not right

Consider

$$f : A \to B \qquad g : B \to A \qquad \eta : g \circ f \sim \mathrm{id}_A \qquad \varepsilon : f \circ g \sim \mathrm{id}_B$$

which can be pictured as

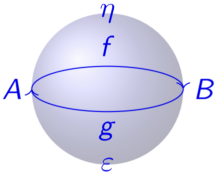


Quasi-invertible maps are not right

Consider

$$f : A \to B \qquad g : B \to A \qquad \eta : g \circ f \sim \mathrm{id}_A \qquad \varepsilon : f \circ g \sim \mathrm{id}_B$$

which can be pictured as



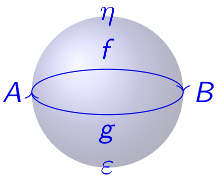
One solution: fill in the sphere!

Quasi-invertible maps are not right

Consider

$$f : A \to B \qquad g : B \to A \qquad \eta : g \circ f \sim \mathrm{id}_A \qquad \varepsilon : f \circ g \sim \mathrm{id}_B$$

which can be pictured as



One solution: fill in the sphere!

Definition
A **half-adjoint equivalence** is a map $f : A \to B$ such that there exists

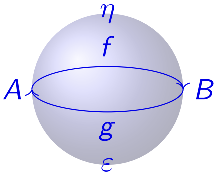
$$g : B \to A \qquad \eta : g \circ f \sim \mathrm{id}_A \qquad \varepsilon : f \circ g \sim \mathrm{id}_B \qquad f\eta \sim \varepsilon f$$

Quasi-invertible maps are not right

Consider

$$f : A \to B \qquad g : B \to A \qquad \eta : g \circ f \sim \mathrm{id}_A \qquad \varepsilon : f \circ g \sim \mathrm{id}_B$$

which can be pictured as



One solution: fill in the sphere!

Definition
A **half-adjoint equivalence** is a map $f : A \to B$ such that there exists

$$g : B \to A \qquad \eta : g \circ f \sim \mathrm{id}_A \qquad \varepsilon : f \circ g \sim \mathrm{id}_B \qquad f\eta \sim \varepsilon f$$

We will first see a simpler solution.

Bi-invertible equivalences

Definition
A map $f : A \to B$ is a **bi-invertible equivalence** where there are $g : B \to A$ and $h : B \to A$ such that

$$g \circ f \sim \mathrm{id}_A \qquad\qquad f \circ h \sim \mathrm{id}_B$$

Definition
A map $f : A \to B$ is a **bi-invertible equivalence** where there are $g : B \to A$ and $h : B \to A$ such that

$$g \circ f \sim \mathrm{id}_A \qquad\qquad f \circ h \sim \mathrm{id}_B$$

We write

$$\mathrm{isEquiv}(f) \quad \hat{=} \quad \Sigma(g : B \to A).(g \circ f \sim \mathrm{id}_A) \times \Sigma(h : B \to A).(f \circ h \sim \mathrm{id}_B)$$

Bi-invertible equivalences

Definition
A map $f : A \to B$ is a **bi-invertible equivalence** where there are $g : B \to A$ and $h : B \to A$ such that

$$g \circ f \sim \mathrm{id}_A \qquad\qquad f \circ h \sim \mathrm{id}_B$$

We write

$$\mathrm{isEquiv}(f) \quad \hat{=} \quad \Sigma(g : B \to A).(g \circ f \sim \mathrm{id}_A) \times \Sigma(h : B \to A).(f \circ h \sim \mathrm{id}_B)$$

Theorem
Being a bi-invertible equivalence is a property: $\mathrm{isProp}(\mathrm{isEquiv}(f))$.

Bi-invertible equivalences

Definition
A map $f : A \to B$ is a **bi-invertible equivalence** where there are $g : B \to A$ and $h : B \to A$ such that

$$g \circ f \sim \mathrm{id}_A \qquad\qquad f \circ h \sim \mathrm{id}_B$$

We write

$$\mathrm{isEquiv}(f) \quad \triangleq \quad \Sigma(g : B \to A).(g \circ f \sim \mathrm{id}_A) \times \Sigma(h : B \to A).(f \circ h \sim \mathrm{id}_B)$$

Theorem
Being a bi-invertible equivalence is a property: $\mathrm{isProp}(\mathrm{isEquiv}(f))$*.*

Proof.
Wait some more! $\square$

Consider

$$f : A \to B$$

which can be pictured as

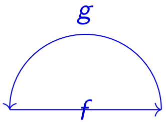
$$\xrightarrow{\quad f \quad}$$

Bi-invertible equivalences

Consider

$$f : A \to B \qquad g : B \to A$$

which can be pictured as

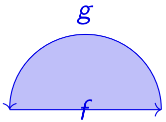


Bi-invertible equivalences

Consider

$$f : A \to B \qquad g : B \to A \qquad \eta : g \circ f \sim \mathrm{id}_A$$

which can be pictured as

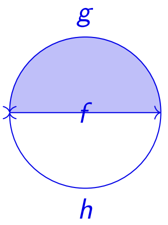


Bi-invertible equivalences

Consider

$$f : A \to B \qquad g : B \to A \qquad \eta : g \circ f \sim \mathrm{id}_A \qquad h : B \to A$$

which can be pictured as

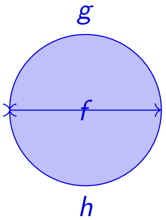


Bi-invertible equivalences

Consider

$$f : A \to B \qquad g : B \to A \qquad \eta : g \circ f \sim \mathrm{id}_A \qquad h : B \to A \qquad \varepsilon : f \circ h \sim \mathrm{id}_B$$

which can be pictured as



The data is contractible!

Quasi- vs bi-invertibles

Oddly, we have that

Proposition
Having a quasi-inverse and being bi-invertible are equivalent:

$$\mathsf{hasQInv}(f) \to \mathsf{isEquiv}(f) \qquad\qquad \mathsf{isEquiv}(f) \to \mathsf{hasQInv}(f)$$

Quasi- vs bi-invertibles

Oddly, we have that

Proposition
Having a quasi-inverse and being bi-invertible are equivalent:

$$\mathsf{hasQInv}(f) \to \mathsf{isEquiv}(f) \qquad\qquad \mathsf{isEquiv}(f) \to \mathsf{hasQInv}(f)$$

This is very useful because quasi-invertible is a self-dual notion contrarily to bi-invertible!

Quasi- vs bi-invertibles

Oddly, we have that

Proposition
Having a quasi-inverse and being bi-invertible are equivalent:

$$\mathsf{hasQInv}(f) \to \mathsf{isEquiv}(f) \qquad \mathsf{isEquiv}(f) \to \mathsf{hasQInv}(f)$$

This is very useful because quasi-invertible is a self-dual notion contrarily to bi-invertible!

Note: the pair of maps does not itself form an equivalence...

From quasi-invertibles to equivalences

A map

$$f : A \to B$$

which is quasi-invertible by

$$(g, \eta, \varepsilon)$$

is also an equivalence by

From quasi-invertibles to equivalences

A map

$$f : A \to B$$

which is quasi-invertible by

$$(g, \eta, \varepsilon)$$

is also an equivalence by

$$(g, \eta, g, \varepsilon)$$

Interlude
In a group, a left inverse of an element a is equal to any right inverse.

Interlude
In a group, a left inverse of an element a is equal to any right inverse.

Proof.
Suppose given a, b and c such that $ba = 1$ and $ac = 1$. We want to show $b = c$.

We have

□

Interlude
In a group, a left inverse of an element a is equal to any right inverse.

Proof.
Suppose given a, b and c such that $ba = 1$ and $ac = 1$. We want to show $b = c$.

We have

$$1 = ba$$

□

Interlude
In a group, a left inverse of an element a is equal to any right inverse.

Proof.
Suppose given a, b and c such that $ba = 1$ and $ac = 1$. We want to show $b = c$.

We have

$$\begin{aligned} 1 &= ba \\ c &= bac \end{aligned}$$

□

Interlude
In a group, a left inverse of an element a is equal to any right inverse.

Proof.
Suppose given a, b and c such that $ba = 1$ and $ac = 1$. We want to show $b = c$.

We have

$$\begin{aligned} 1 &= ba \\ c &= bac = b \end{aligned}$$

□

From equivalences to quasi-invertibles

Conversely, suppose given a bi-invertible equivalence $(f, g, \eta, h, \varepsilon)$.

We can build a quasi-invertible

From equivalences to quasi-invertibles

Conversely, suppose given a bi-invertible equivalence $(f, g, \eta, h, \varepsilon)$.

We can build a quasi-invertible

$$(f, g, \eta, g, \varepsilon')$$

where we still have to define

$$\varepsilon' : f \circ g \sim \mathrm{id}$$

from

$$\eta : g \circ f \sim \mathrm{id} \qquad\qquad \varepsilon : f \circ h \sim \mathrm{id}$$

From equivalences to quasi-invertibles

The left inverse g has to be the same as the right inverse h:

$$g$$

and we can therefore define ε' as the composite

$$fg \sim fh \overset{\varepsilon}{\sim} \mathrm{id}$$

From equivalences to quasi-invertibles

The left inverse g has to be the same as the right inverse h:

$$g = gfh$$

and we can therefore define ε' as the composite

$$fg \sim fh \overset{\varepsilon}{\sim} \mathrm{id}$$

The left inverse g has to be the same as the right inverse h:

$$g = gfh = h$$

and we can therefore define ε' as the composite

$$fg \sim fh \overset{\varepsilon}{\sim} \mathrm{id}$$

From equivalences to quasi-invertibles

The left inverse g has to be homotopic to the right inverse h:

$$g \overset{g\varepsilon^{-}}{\sim} gfh \overset{\eta_h}{\sim} h$$

and we can therefore define ε' as the composite

$$fg \sim fh \overset{\varepsilon}{\sim} \mathrm{id}$$

From equivalences to quasi-invertibles

The left inverse g has to be homotopic to the right inverse h:

$$g \overset{g\varepsilon^-}{\sim} gfh \overset{\eta_h}{\sim} h$$

and we can therefore define ε' as the composite

$$fg \overset{fg\varepsilon^-}{\sim} fgfh \overset{f\eta_h}{\sim} fh \overset{\varepsilon}{\sim} \mathrm{id}$$

From equivalences to quasi-invertibles

The left inverse g has to be homotopic to the right inverse h:

$$g \overset{g\varepsilon^{-}}{\sim} gfh \overset{\eta_h}{\sim} h$$

and we can therefore define ε' as the composite

$$fg \overset{fg\varepsilon^{-}}{\sim} fgfh \overset{f\eta_h}{\sim} fh \overset{\varepsilon}{\sim} \mathrm{id}$$

More explicitly:

$$\varepsilon' x \quad \hat{=} \quad \mathrm{ap}\,(f \circ g)((\varepsilon\, x)^{-}) \cdot \mathrm{ap}\, f\,(\eta\,(h\, x)) \cdot \varepsilon\, x$$

Equivalences

We write

$$A \simeq B$$

for the type of **equivalences** between A and B

Equivalences

We write

$$A \simeq B$$

for the type of **equivalences** between A and B:

$$A \simeq B \quad \hat{=} \quad \Sigma(f : A \to B).\, \mathsf{isEquiv}(f)$$

Part IV

First properties of equivalences

Proposition ([Uni13, Lemma 2.4.12])
Being equivalent is an equivalence relation:

- $A \simeq A$,
- $A \simeq B$ *implies* $B \simeq A$,
- $A \simeq B$ *and* $B \simeq C$ *implies* $A \simeq C$.

Proposition ([Uni13, Lemma 2.4.12])
Being equivalent is an equivalence relation:

- $A \simeq A$,
- $A \simeq B$ *implies* $B \simeq A$,
- $A \simeq B$ *and* $B \simeq C$ *implies* $A \simeq C$.

Proof.
The identity $\mathrm{id} : A \to A$ is an equivalence because it admits id as inverse.

Proposition ([Uni13, Lemma 2.4.12])
Being equivalent is an equivalence relation:

- $A \simeq A$,
- $A \simeq B$ *implies* $B \simeq A$,
- $A \simeq B$ *and* $B \simeq C$ *implies* $A \simeq C$.

Proof.
A function $f : A \to B$ is bi-invertible,

Proposition ([Uni13, Lemma 2.4.12])
Being equivalent is an equivalence relation:

- $A \simeq A$,
- $A \simeq B$ *implies* $B \simeq A$,
- $A \simeq B$ *and* $B \simeq C$ *implies* $A \simeq C$.

Proof.
A function $f : A \to B$ is bi-invertible, thus quasi-invertible with inverse $f^{-1} : B \to A$,

Proposition ([Uni13, Lemma 2.4.12])
Being equivalent is an equivalence relation:

- $A \simeq A$,
- $A \simeq B$ *implies* $B \simeq A$,
- $A \simeq B$ *and* $B \simeq C$ *implies* $A \simeq C$.

Proof.
A function $f : A \to B$ is bi-invertible, thus quasi-invertible with inverse $f^{-1} : B \to A$, thus f^{-1} is quasi-invertible,

Proposition ([Uni13, Lemma 2.4.12])
Being equivalent is an equivalence relation:

- $A \simeq A$,
- $A \simeq B$ *implies* $B \simeq A$,
- $A \simeq B$ *and* $B \simeq C$ *implies* $A \simeq C$.

Proof.
A function $f : A \to B$ is bi-invertible, thus quasi-invertible with inverse $f^{-1} : B \to A$, thus f^{-1} is quasi-invertible, thus f^{-1} is an equivalence.

Proposition ([Uni13, Lemma 2.4.12])
Being equivalent is an equivalence relation:

- $A \simeq A$,
- $A \simeq B$ *implies* $B \simeq A$,
- $A \simeq B$ *and* $B \simeq C$ *implies* $A \simeq C$.

Proof.
Given $f : A \to B$ and $g : B \to C$, we have that $gf : A \to C$ admits $f^{-1}g^{-1} : C \to A$ as inverse. $\square$

Equivalences between propositions

Lemma ([Uni13, Lemma 3.3.3])
Two propositions A and B are equivalent if and only if they are logically equivalent:

$$(A \leftrightarrow B) \qquad \leftrightarrow \qquad (A \simeq B)$$

Lemma ([Uni13, Lemma 3.3.3])
Two propositions A and B are equivalent if and only if they are logically equivalent:

$$(A \leftrightarrow B) \qquad \leftrightarrow \qquad (A \simeq B)$$

Proof.
The right-to-left implication is immediate.

Lemma ([Uni13, Lemma 3.3.3])
Two propositions A and B are equivalent if and only if they are logically equivalent:

$$(A \leftrightarrow B) \qquad \leftrightarrow \qquad (A \simeq B)$$

Proof.
The right-to-left implication is immediate.

For left-to-right, suppose given $f : A \to B$ and $g : B \to A$.

Equivalences between propositions

Lemma ([Uni13, Lemma 3.3.3])
Two propositions A and B are equivalent if and only if they are logically equivalent:

$$(A \leftrightarrow B) \qquad \leftrightarrow \qquad (A \simeq B)$$

Proof.
The right-to-left implication is immediate.

For left-to-right, suppose given $f : A \to B$ and $g : B \to A$.

For $x : A$, we have $g(f(x)) = x$ because A is a proposition

Lemma ([Uni13, Lemma 3.3.3])
Two propositions A and B are equivalent if and only if they are logically equivalent:

$$(A \leftrightarrow B) \qquad \leftrightarrow \qquad (A \simeq B)$$

Proof.
The right-to-left implication is immediate.

For left-to-right, suppose given $f : A \to B$ and $g : B \to A$.

For $x : A$, we have $g(f(x)) = x$ because A is a proposition and thus $g \circ f \sim \mathrm{id}$.

Equivalences between propositions

Lemma ([Uni13, Lemma 3.3.3])
Two propositions A and B are equivalent if and only if they are logically equivalent:

$$(A \leftrightarrow B) \qquad \leftrightarrow \qquad (A \simeq B)$$

Proof.
The right-to-left implication is immediate.

For left-to-right, suppose given $f : A \to B$ and $g : B \to A$.

For $x : A$, we have $g(f(x)) = x$ because A is a proposition and thus $g \circ f \sim \mathrm{id}$.

Similarly, $f \circ g \sim \mathrm{id}$. □

Equivalences between propositions

We can even upgrade previous lemma:

Lemma
Given two propositions A and B, we have

$$(A \leftrightarrow B) \qquad \leftrightarrow \qquad (A \simeq B)$$

Equivalences between propositions

We can even upgrade previous lemma:

Lemma
Given two propositions A and B, we have

$$(A \leftrightarrow B) \qquad \simeq \qquad (A \simeq B)$$

Equivalences between propositions

We can even upgrade previous lemma:

Lemma
Given two propositions A and B, we have

$$(A \leftrightarrow B) \qquad \simeq \qquad (A \simeq B)$$

Proof.
We have that

$$(A \leftrightarrow B) \quad \hat{=} \quad (A \to B) \times (B \to A)$$

is a proposition and

$$(A \simeq B) \quad \hat{=} \quad \Sigma(f : A \to B).\,\mathrm{isEquiv}(f)$$

is a proposition. □

Lemma ([Uni13, Lemma 3.11.3])
We have

$$\mathrm{isContr}(A) \quad \simeq \quad (A \simeq 1)$$

Equivalences between contractible types

Lemma ([Uni13, Lemma 3.11.3])
We have

$$\mathsf{isContr}(A) \quad \simeq \quad (A \simeq 1)$$

Proof.
We have that $\mathsf{isContr}(A)$ is a proposition as well as

$$(A \simeq 1) \quad \hat{=} \quad \Sigma(f : A \to 1).\, \mathsf{isEquiv}(f)$$

and therefore it is enough to show that

$$(A \simeq 1) \leftrightarrow \Sigma(f : A \to 1).\, \mathsf{isEquiv}(f)$$

Equivalences between contractible types

Lemma ([Uni13, Lemma 3.11.3])
We have

$$\mathrm{isContr}(A) \quad \simeq \quad (A \simeq 1)$$

Proof.
Suppose that A is contractible. The we can build an equivalence

$$f : A \leftrightarrow 1 : g$$

where f is the terminal arrow, and g sends $\star$ to the center of contraction a of A.

We then have $g \circ f(x) = x$ and $f \circ g(y) = y$ because both A and 1 are propositions.

Lemma ([Uni13, Lemma 3.11.3])
We have

$$\mathrm{isContr}(A) \quad \simeq \quad (A \simeq 1)$$

Proof.
Suppose given $f : A \simeq 1$. We chose $a \hat{=} f^{-1}(\star)$ as center of contraction. Given $x : A$, we have

$$x = f^{-1}(f(x)) = f^{-1}(\star) = a$$

□

Standard equivalences

Lemma
Coproduct and product satisfy the expected equivalences:

$$0 \sqcup A \simeq A \simeq A \sqcup 0 \qquad (A \sqcup B) \sqcup C \simeq A \sqcup (B \sqcup B) \qquad A \sqcup B \simeq B \sqcup A$$

$$1 \times A \simeq A \simeq A \times 1 \qquad (A \times B) \times C \simeq A \times (B \times C) \qquad A \times B \simeq B \times A$$

$$0 \times A \simeq 0 \simeq A \times 0 \qquad A \times (B \sqcup C) \simeq (A \times B) \sqcup (A \times C)$$

Standard equivalences

Lemma
Coproduct and product satisfy the expected equivalences:

$$0 \sqcup A \simeq A \simeq A \sqcup 0 \qquad (A \sqcup B) \sqcup C \simeq A \sqcup (B \sqcup B) \qquad A \sqcup B \simeq B \sqcup A$$

$$1 \times A \simeq A \simeq A \times 1 \qquad (A \times B) \times C \simeq A \times (B \times C) \qquad A \times B \simeq B \times A$$

$$0 \times A \simeq 0 \simeq A \times 0 \qquad A \times (B \sqcup C) \simeq (A \times B) \sqcup (A \times C)$$

Lemma ([Uni13, Section 2.15])
Arrow types satisfy the expected equivalences:

$$A \to 1 \simeq 1 \qquad X \to A \times B \simeq (X \to A) \times (X \to B)$$

$$0 \to X \simeq 1 \qquad A \sqcup B \to X \simeq (A \to X) \times (B \to X)$$

$$1 \to C \simeq C \qquad (A \times B) \to C \simeq A \to (B \to C)$$

The universal property of products

Let's focus on $(X \to A \times B) \simeq (X \to A) \times (X \to B)$: we have

$$\begin{aligned} \phi : (X \to A \times B) &\to (X \to A) \times (X \to B) \\ f &\mapsto \end{aligned}$$

The universal property of products

Let's focus on $(X \to A \times B) \simeq (X \to A) \times (X \to B)$: we have

$$\begin{aligned}\phi : (X \to A \times B) &\to (X \to A) \times (X \to B) \\ f &\mapsto (\lambda x.\, \mathrm{fst}(f\, x), \lambda x.\, \mathrm{snd}(f\, x))\end{aligned}$$

The universal property of products

Let's focus on $(X \to A \times B) \simeq (X \to A) \times (X \to B)$: we have

$$\begin{aligned} \phi : (X \to A \times B) &\to (X \to A) \times (X \to B) \\ f &\mapsto (\lambda x.\, \mathrm{fst}(f\,x), \lambda x.\, \mathrm{snd}(f\,x)) \end{aligned}$$

and

$$\begin{aligned} \psi : (X \to A) \times (X \to B) &\to X \to A \times B \\ (f, g) &\quad x \mapsto \end{aligned}$$

The universal property of products

Let's focus on $(X \to A \times B) \simeq (X \to A) \times (X \to B)$: we have

$$\begin{aligned}
\phi : (X \to A \times B) &\to (X \to A) \times (X \to B) \\
f &\mapsto (\lambda x.\, \mathsf{fst}(f\,x), \lambda x.\, \mathsf{snd}(f\,x))
\end{aligned}$$

and

$$\begin{aligned}
\psi : (X \to A) \times (X \to B) &\to X \to A \times B \\
(f, g) &\quad x \mapsto (f\,x, g\,x)
\end{aligned}$$

Let's focus on $(X \to A \times B) \simeq (X \to A) \times (X \to B)$: we have

$$\begin{aligned} \phi : (X \to A \times B) &\to (X \to A) \times (X \to B) \\ f &\mapsto (\lambda x.\, \mathsf{fst}(f\,x), \lambda x.\, \mathsf{snd}(f\,x)) \end{aligned}$$

and

$$\begin{aligned} \psi : (X \to A) \times (X \to B) &\to X \to A \times B \\ (f, g) &\quad x \mapsto (f\,x, g\,x) \end{aligned}$$

satisfying

$$\begin{aligned} \psi \circ \phi(f) &= \psi(\lambda x.\, \mathsf{fst}(f\,x), \lambda x.\, \mathsf{snd}(f\,x)) = \lambda x. (\mathsf{fst}(f\,x), \mathsf{snd}(f\,x)) = \lambda x. f\,x \\ \phi \circ \psi(f, g) &= \phi(\lambda x. (f\,x, g\,x)) = (\lambda x.\, \mathsf{fst}(f\,x, g\,x), \lambda x.\, \mathsf{snd}(f\,x, g\,x)) = (\lambda x. f\,x, \lambda x. g\,x) = (f, \end{aligned}$$

The universal property of products

The equivalence

$$(X \to A \times B) \simeq (X \to A) \times (X \to B)$$

can be understood as encoding the universal property of products:

$$\begin{tikzcd}
 & X \arrow[d, dotted] \arrow[ddl, bend right, "f"'] \arrow[ddr, bend left, "g"] & \\
 & A \times B \arrow[dl, "\mathsf{fst}"] \arrow[dr, "\mathsf{snd}"'] & \\
A & & B
\end{tikzcd}$$

The universal property of coproducts

....

TODO: for coproducts : introduction / elimination / computation / uniqueness

this only encodes non-dependent variant and does not inform us about definitional equalities

Lemma ([Uni13, Section 2.15])
We also have the expected variants for dependent types:

- *given* $B\,C : A \to \mathcal{U}$*:*

$$(x : A) \to B\,x \times C\,x \quad \simeq \quad ((x : A) \to B\,x) \times ((x : A) \to C\,x)$$

Variants for dependent types

Lemma ([Uni13, Section 2.15])
We also have the expected variants for dependent types:

- *given* $B\,C : A \to \mathcal{U}$*:*

$$(x : A) \to B\,x \times C\,x \quad \simeq \quad ((x : A) \to B\,x) \times ((x : A) \to C\,x)$$

- *given* $C : A \times B \to \mathcal{U}$*:*

$$(x : A \times B) \to C\,x \quad \simeq \quad (a : A) \to (b : B) \to C\,(a, b)$$

Variants for dependent types

Lemma ([Uni13, Section 2.15])
We also have the expected variants for dependent types:

- *given* $B\,C : A \to \mathcal{U}$*:*

$$(x : A) \to B\,x \times C\,x \quad \simeq \quad ((x : A) \to B\,x) \times ((x : A) \to C\,x)$$

- *given* $C : A \times B \to \mathcal{U}$*:*

$$(x : A \times B) \to C\,x \quad \simeq \quad (a : A) \to (b : B) \to C\,(a, b)$$

- *given* $B : A \to \mathcal{U}$ *and* $C : (\Sigma A.B) \to \mathcal{U}$*:*

$$(x : \Sigma A.B) \to C\,x \quad \simeq \quad (a : A) \to (b : B\,a) \to C\,(a, b)$$

Skolemizing

In model theory, there is a *Skolemization* operation, which allows removing existential quantification for prenex formulas.

Skolemizing

In model theory, there is a *Skolemization* operation, which allows removing existential quantification for prenex formulas.

The idea is that we can replace a formula

$$\forall x. \exists y. P(x, y)$$

by a formula

$$\forall x. P(x, f(x))$$

for some function f.

Skolemizing

In model theory, there is a *Skolemization* operation, which allows removing existential quantification for prenex formulas.

The idea is that we can replace a formula

$$\forall x.\exists y.P(x, y)$$

by a formula

$$\forall x.P(x, f(x))$$

for some function f.

For instance, we can replace

$$\forall x.\exists y.y \times x = 1 = x \times y$$

by

$$\forall x.x^{-1} \times x = 1 = x \times x^{-1}$$

The type-theoretic axiom of choice

A dependent variant of $A \times (B \sqcup C) \simeq (A \times B) \sqcup (A \times C)$ is

Proposition ([Uni13, Theorem 2.15.7])
Given $A : X \to \mathcal{U}$ *and* $B : (x : X) \to A\,x \to \mathcal{U}$, *we have*

$$\Pi(x : X).\Sigma(a : A\,x).B\,x\,a \quad \simeq \quad \Sigma(f : (x : X) \to A\,x).\Pi(x : X).B\,x\,(f\,x)$$

The type-theoretic axiom of choice

A dependent variant of $A \times (B \sqcup C) \simeq (A \times B) \sqcup (A \times C)$ is

Proposition ([Uni13, Theorem 2.15.7])
Given $A : X \to \mathcal{U}$ *and* $B : (x : X) \to A\,x \to \mathcal{U}$*, we have*

$$\Pi(x : X).\Sigma(a : A\,x).B\,x\,a \quad \simeq \quad \Sigma(f : (x : X) \to A\,x).\Pi(x : X).B\,x\,(f\,x)$$

This is sometimes called the **type-theoretic axiom of choice**.

The type-theoretic axiom of choice

A dependent variant of $A \times (B \sqcup C) \simeq (A \times B) \sqcup (A \times C)$ is

Proposition ([Uni13, Theorem 2.15.7])
Given $A : X \to \mathcal{U}$ and $B : (x : X) \to A\,x \to \mathcal{U}$, we have

$$\Pi(x : X).\Sigma(a : A\,x).B\,x\,a \quad \simeq \quad \Sigma(f : (x : X) \to A\,x).\Pi(x : X).B\,x\,(f\,x)$$

This is sometimes called the **type-theoretic axiom of choice**.

The usual axiom of choice is rather

$$\Pi(x : X).\|\Sigma(a : A\,x).B\,x\,a\|_{-1} \quad \to \quad \|\Sigma(f : (x : X) \to A\,x).\Pi(x : X).B\,x\,(f\,x)\|_{-1}$$

Equivalence is a congruence

Lemma
Equivalence is a congruence with usual type formers: for $A \simeq A$ *and* $B \simeq B'$

$$A \times B \simeq A' \times B'$$

and similarly for $\sqcup$ *and* $\to$.

Equivalence is a congruence

Lemma
Equivalence is a congruence with usual type formers: for $A \simeq A$ and $B \simeq B'$

$$A \times B \simeq A' \times B'$$

and similarly for $\sqcup$ and $\to$.

Note: this is quite painful / by hand. With univalence, will become automatic!

We can also characterize path types in constructed types.

Paths in products types

Recall that, for $x : A$ and $x' : B$, we have a function

$$(x = x') \to (\mathsf{fst}\ x = \mathsf{fst}\ x') \times (\mathsf{snd}\ x = \mathsf{snd}\ x')$$

given by $p \mapsto$

Paths in products types

Recall that, for $x : A$ and $x' : B$, we have a function

$$(x = x') \to (\mathrm{fst}\ x = \mathrm{fst}\ x') \times (\mathrm{snd}\ x = \mathrm{snd}\ x')$$

given by $p \mapsto (\mathrm{ap}\ \mathrm{fst}\ p, \mathrm{ap}\ \mathrm{snd}\ p)$.

Paths in products types

Recall that, for $x : A$ and $x' : B$, we have a function

$$(x = x') \to (\mathsf{fst}\ x = \mathsf{fst}\ x') \times (\mathsf{snd}\ x = \mathsf{snd}\ x')$$

given by $p \mapsto (\mathsf{ap}\ \mathsf{fst}\ p, \mathsf{ap}\ \mathsf{snd}\ p)$.

Theorem ([Uni13, Theorem 2.6.2])
The above function is an equivalence.

Paths in products types

Recall that, for $x : A$ and $x' : B$, we have a function

$$(x = x') \to (\mathrm{fst}\ x = \mathrm{fst}\ x') \times (\mathrm{snd}\ x = \mathrm{snd}\ x')$$

given by $p \mapsto (\mathrm{ap}\ \mathrm{fst}\ p, \mathrm{ap}\ \mathrm{snd}\ p)$.

Theorem ([Uni13, Theorem 2.6.2])
The above function is an equivalence.

Proof.
We need to define a function

$$\mathrm{pair}^{=} : (\mathrm{fst}\ x = \mathrm{fst}\ x') \times (\mathrm{snd}\ x = \mathrm{snd}\ x') \to x = x'$$

Paths in products types

Recall that, for $x : A$ and $x' : B$, we have a function

$$(x = x') \to (\mathsf{fst}\ x = \mathsf{fst}\ x') \times (\mathsf{snd}\ x = \mathsf{snd}\ x')$$

given by $p \mapsto (\mathsf{ap}\ \mathsf{fst}\ p, \mathsf{ap}\ \mathsf{snd}\ p)$.

Theorem ([Uni13, Theorem 2.6.2])
The above function is an equivalence.

Proof.
We need to define a function

$$\mathsf{pair}^{=} : (a = a') \times (b = b') \to (a, b) = (a', b')$$

(by product induction we assume that $x \triangleq (a, b)$ and $x' \triangleq (a', b')$)

Recall that, for $x : A$ and $x' : B$, we have a function

$$(x = x') \to (\mathsf{fst}\ x = \mathsf{fst}\ x') \times (\mathsf{snd}\ x = \mathsf{snd}\ x')$$

given by $p \mapsto (\mathsf{ap}\ \mathsf{fst}\ p, \mathsf{ap}\ \mathsf{snd}\ p)$.

Theorem ([Uni13, Theorem 2.6.2])
The above function is an equivalence.

Proof.
We need to define a function

$$\mathsf{pair}^{=} : (a = a') \times (b = b') \to (a, b) = (a', b')$$

(by product induction we assume that $x \triangleq (a, b)$ and $x' \triangleq (a', b')$) which we can do by path induction on both components.

Paths in products types

Recall that, for $x : A$ and $x' : B$, we have a function

$$(x = x') \to (\mathsf{fst}\ x = \mathsf{fst}\ x') \times (\mathsf{snd}\ x = \mathsf{snd}\ x')$$

given by $p \mapsto (\mathsf{ap}\ \mathsf{fst}\ p, \mathsf{ap}\ \mathsf{snd}\ p)$.

Theorem ([Uni13, Theorem 2.6.2])
The above function is an equivalence.

Proof.
We need to define a function

$$\mathsf{pair}^{=} : (a = a') \times (b = b') \to (a, b) = (a', b')$$

(by product induction we assume that $x \triangleq (a, b)$ and $x' \triangleq (a', b')$) which we can do by path induction on both components.

The fact that they are mutually inverse is (again) done by path induction. $\square$

Paths in products types

This equivalence means that we have

- an elimination rule

Paths in products types

This equivalence means that we have

- an elimination rule
- an introduction rule

Paths in products types

This equivalence means that we have

- an elimination rule
- an introduction rule
- a computation rule

Paths in products types

This equivalence means that we have

- an elimination rule
- an introduction rule
- a computation rule
- a uniqueness rule

Paths in Σ-types

Given $A : \mathcal{U}$ and $B : A \to \mathcal{U}$, and $(x, y)\,(x', y') : \Sigma A.B$, recall that we have a function

$$((x, y) = (x', y')) \to \Sigma(p : x = x').(y =^B_p y')$$

and a function

$$\mathsf{pair}^= : \Sigma(p : x = x').(y =^B_p y') \to ((x, y) = (x', y'))$$

Paths in Σ-types

Given $A : \mathcal{U}$ and $B : A \to \mathcal{U}$, and $(x, y)\,(x', y') : \Sigma A.B$, recall that we have a function

$$((x, y) = (x', y')) \to \Sigma(p : x = x').(y =^B_p y')$$

and a function

$$\mathsf{pair}^= : \Sigma(p : x = x').(y =^B_p y') \to ((x, y) = (x', y'))$$

Theorem ([Uni13, Theorem 2.7.2])
We have

$$(x, y) = (x', y') \quad \simeq \quad \Sigma(p : x = x').(y =^B_p y')$$

[Uni13, Section 2.14] [Uni13, Section 9.8]

SIP

Part V

Being an equivalence is a proposition

Let's try to prove that being an equivalence is a proposition.

Being an equivalence is a proposition

Theorem ([Uni13, Lemma 4.2.9])
For a map $f : A \to B$, being an equivalence is a proposition.

Proof (first draft).
Our goal is to show that

$$\mathsf{isEquiv}(f) \quad \hat{=} \quad \Sigma(g : B \to A).(g \circ f \sim \mathsf{id}) \times \Sigma(g : B \to A).(f \circ g \sim \mathsf{id})$$

is a proposition.

Being an equivalence is a proposition

Theorem ([Uni13, Lemma 4.2.9])
For a map $f : A \to B$, being an equivalence is a proposition.

Proof (first draft).
Our goal is to show that

$$\mathrm{isEquiv}(f) \quad \hat{=} \quad \Sigma(g : B \to A).(g \circ f \sim \mathrm{id}) \times \Sigma(g : B \to A).(f \circ g \sim \mathrm{id})$$

is a proposition.

A first plan is to show that $\Sigma(g : B \to A).(g \circ f \sim \mathrm{id})$ is a proposition (and similarly for the other component) and conclude because propositions are closed under product.

Being an equivalence is a proposition

Theorem ([Uni13, Lemma 4.2.9])
For a map $f : A \to B$, being an equivalence is a proposition.

Proof (first draft).
Our goal is to show that

$$\mathrm{isEquiv}(f) \quad \hat{=} \quad \Sigma(g : B \to A).(g \circ f \sim \mathrm{id}) \times \Sigma(g : B \to A).(f \circ g \sim \mathrm{id})$$

is a proposition.

A first plan is to show that $\Sigma(g : B \to A).(g \circ f \sim \mathrm{id})$ is a proposition (and similarly for the other component) and conclude because propositions are closed under product.

But we cannot do this!

Being an equivalence is a proposition

Theorem ([Uni13, Lemma 4.2.9])
For a map $f : A \to B$, being an equivalence is a proposition.

Proof (first draft).
Our goal is to show that

$$\mathsf{isEquiv}(f) \quad \hat{=} \quad \Sigma(g : B \to A).(g \circ f \sim \mathrm{id}) \times \Sigma(g : B \to A).(f \circ g \sim \mathrm{id})$$

is a proposition.

A first plan is to show that $\Sigma(g : B \to A).(g \circ f \sim \mathrm{id})$ is a proposition (and similarly for the other component) and conclude because propositions are closed under product.

But we cannot do this! For instance

$$f : \begin{tikzcd}[row sep=small] 0 \arrow[r, mapsto] & 0 \\ 1 \arrow[r, mapsto] & 1 \\ & 2 \end{tikzcd}$$

Proof (continued).
However, what we can show is that *for an equivalence* having a left inverse is contractible (and thus a proposition).

Proof (continued).
However, what we can show is that *for an equivalence* having a left inverse is contractible (and thus a proposition).

As center of contraction, we choose

$$(g : B \to A, \eta : g \circ f \sim \mathrm{id})$$

given by the fact that we have an equivalence.

Proof (continued).
However, what we can show is that *for an equivalence* having a left inverse is contractible (and thus a proposition).

As center of contraction, we choose

$$(g : B \to A, \eta : g \circ f \sim \mathrm{id})$$

given by the fact that we have an equivalence.

Given $(g' : B \to A, \eta, \eta' : g' \circ f \sim \mathrm{id})$, we need to show $p : g = g'$ and $\eta =_{\text{"}p\text{"}} \eta'$.

Proof (continued).
However, what we can show is that *for an equivalence* having a left inverse is contractible (and thus a proposition).

As center of contraction, we choose

$$(g : B \to A, \eta : g \circ f \sim \mathrm{id})$$

given by the fact that we have an equivalence.

Given $(g' : B \to A, \eta, \eta' : g' \circ f \sim \mathrm{id})$, we need to show $p : g = g'$ and $\eta =_{``p"} \eta'$.
For the first one, we have

$$g\,y = g'(f(g\,y)) = g'\,y$$

so that $g = g'$ (if we assume funext!).
The second one can be done, but is less clear for now... (TBC) $\square$

[Uni13] The Univalent Foundations Program.
Homotopy Type Theory: Univalent Foundations of Mathematics.
Institute for Advanced Study, 2013.
`https://homotopytypetheory.org/book`, `arXiv:1308.0729`.

[Whi49a] J. H. C. Whitehead.
Combinatorial homotopy. I.
Bulletin of the American Mathematical Society, 55(3):213 – 245, 1949.

[Whi49b] J. H. C. Whitehead.
Combinatorial homotopy. II.
Bulletin of the American Mathematical Society, 55(5):453 – 496, 1949.