

Gauge Theory in Homotopy Type Theory

Greg Langmead

April 9, 2023

Abstract

This thesis will show how to formalize parts of differential geometry and gauge theory in homotopy type theory.[1] [2] [3] [4] [5] [6]

Contents

Introduction	2
Groups	2
Bundles over BG	3
Mapping into BG	4
BG classifies bundles, not just equivalence classes	5
Eilenberg-Mac Lane spaces	5
Cohomology	6
Universes versus bundles	6
Towards connections	6

Introduction

Differential geometry and gauge theory lie at the intersection of mathematics and physics. The theory of general relativity is expressed as formulas relating the curvature of spacetime to the distribution of energy and matter. The standard model of particle physics is expressed as a collection of principal bundles and associated vector bundles, sections of which represent the fields of matter and forces. And in the latter decades of the 20th century the tools of the standard model were applied back into mathematics to make discoveries about the geometry of 3- and 4-dimensional manifolds.

Homotopy type theory seems at first like it's from a different universe entirely. It grew out of questions of a foundational nature inside mathematics and computer science. It brings a synthetic stance, where we are invited to stop thinking of mathematical objects as sets with structure, and instead to assemble the objects into higher categories with collective properties like being locally cartesian closed, or having homotopical model structures. Such categories have internal languages, which are syntactic structures much like programming languages, which allow us to reason about the objects in a generic way.

Groups

Group theory including the theory of higher groups is built directly into HoTT, via the identity types. A thorough introductory treatment of group theory that adopts this philosophy can be found in the book *Symmetry* [7]. We will build on this synthetic framework to explore gauge theory. But it's important to be clear that this is a choice, and there is an alternate approach. Namely we could define a group to be a type together with some extra structure: an identity, a multiplication operation, and an inverse operation, satisfying some properties. We would do the same for higher groups, and we'd prove an equivalence between the type of higher groups and the type of pointed, connected types. Such a project has been undertaken in the context of higher topos theory, but at the time of this writing it hasn't been done inside HoTT. We certainly can define the type of groups which are sets, and prove that this type of 0-groups is equivalent to connected 1-types, and this is done in *Symmetry*. But in the case of higher groups we only have one definition: pointed connected types. This happens to be a lovely theory, and so we'll be glad to run with that.

Definition: a group is a pointed 1-type BG with basepoint $*$, which is connected,

i.e. that we have a term of the type $(a : BG) \rightarrow ||* = a||_{-1}$. We denote the type $* = *$ by G , and so if we have a particular group in mind then we think of that group as being installed as the identity type of the basepoint in a type we call BG .

Given any pointed type X with basepoint x we can form the type $\mathbf{BAut}_x(X)$ by taking the connected component of x . If we consider the terms of X to be at type-theoretic level 0, then \mathbf{Aut}_x brings one of the identity types from level 1 down to level 0 to look at it on its own. Then the \mathbf{B} can be thought of as reinstalling the identity type at level 1 of the new type, which contains only the connected component of x . When we do this, though, we don't forget the identity types of all the other terms in the connected component. That's why there really isn't a true \mathbf{B} operator, just a notation that is meant to evoke the classical idea of a classifying space, since as we'll see we really have captured that idea.

A homomorphism of groups is a pointed map $f : BG \rightarrow X$. The rules of type theory will ensure that this map contains a map between the identity types inside the types, forming a homomorphism in the usual sense. We can think of G acting on x since the homomorphism is $* = * \rightarrow x = x$. If we take $X = \mathcal{U}$ to be the universe, and if $f(*) = H$, then we can think of G acting on the type H .

Bundles over BG

Pointed connected types BG come with three built-in type families by choosing identity types with different endpoints. There's $LG := (a : BG) \times a = a$ which is at each point the group of self-identifications of that point. There's $EG := (a : BG) \times * = a$ which aren't groups, but which are similar to the self-identifications, and in fact are torsors over G ! At each point $a : BG$ we can concatenate a term of $EG(a)$ with a term of $LG(a)$ by concatenating the latter to the a endpoint of $EG(a)$. And lastly there's the non-dependent type $BG \times G$ which I'm mentioning because we can also concatenate a term of $EG(a)$ with a term of G , by attaching it to the $*$ endpoint.

EG is contractible (Lemma 3.11.8 of The HoTT Book[8]). It's worth reflecting for a moment on the fact that EG is contractible but BG is not. After all, BG is connected and so all its terms are equal to the base point via some path. But we don't select any specific paths as part of the data. Forming EG is simply to install all the solutions to this missing data as the fibers of a bundle over BG .

Example: for $G = \mathbb{Z}, \mathbb{Z}^n, \mathbb{Z}_2$ provide BG, EG , and LG .

Mapping into BG

Pulling back EG via a map $X \rightarrow BG$ deserves to be called a principal G -bundle over X , and pulling back LG via the same map deserves to be called a gauge transformation.

In HoTT given a map $f : A \rightarrow B$ and a type family $P : B \rightarrow \mathcal{U}$ we can form the following square and it will commute and be a pullback:

$$\begin{array}{ccc} (a : A) \times P(f(a)) & \xrightarrow{\lambda(a,p).(f(a),p)} & (b : B) \times P(b) \\ \text{pr}_1 \downarrow & & \downarrow \text{pr}_1 \\ A & \xrightarrow{f} & B \end{array}$$

If we have an arbitrary type X and $f : X \rightarrow BG$ then we can pull back EG this way:

$$\begin{array}{ccc} (x : X) \times (* = f(x)) & \xrightarrow{\lambda(x,p).(f(x),p)} & (b : BG) \times (* = b) \\ \text{pr}_1 \downarrow & & \downarrow \text{pr}_1 \\ X & \xrightarrow{f} & BG \end{array}$$

or we can pull back LG :

$$\begin{array}{ccc} (x : X) \times (f(x) = f(x)) & \xrightarrow{\lambda(x,p).(f(x),p)} & (b : BG) \times (b = b) \\ \text{pr}_1 \downarrow & & \downarrow \text{pr}_1 \\ X & \xrightarrow{f} & BG \end{array}$$

Let's name these pullbacks $E_f X$ and $L_f X$ respectively. We include the classifying map f in the notation but maybe we'll leave it out sometimes. Terms of $E_f X$ are a pair (x, p) : a term of X and a term of the torsor $* = f(x)$. Terms of $L_f X$ are also a pair (x, l) : a term of X and a term of the loop space $f(x) = f(x)$. We can form $(x, p \cdot l)$ and in fact we can act on all of $E_f X$ with a section of $L_f X$:

$$\mathcal{G}X := ((x, l) : L_f X) \rightarrow E_f X \rightarrow E_f X$$

which acts fiberwise by concatenating l to any torsor.

The bundle $E_f X$ would classically be called something like P , a principal bundle with gauge group G over X . The bundle $L_f X$ would then be referred to as $\text{Ad}P$. Why is that?

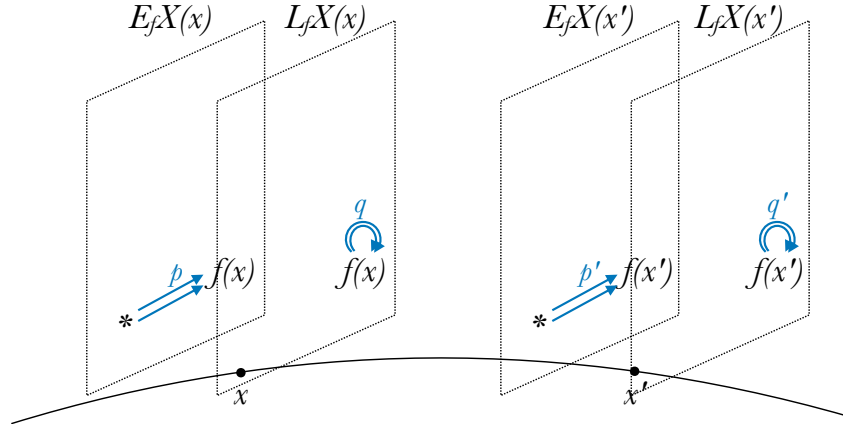


Figure 1: Torsors and gauge transformations over X . p and q are terms in the fibers of the total spaces $E_f X$ and $L_f X$ respectively. Similarly for p' and q' . We can then form $p \cdot q$ and $p' \cdot q'$. If we had a whole section of such q then we could use that to act on the whole of $E_f X$.

BG classifies bundles, not just equivalence classes

The classical theorems take the form: taking the pullback of $EG \rightarrow BG$ forms a bijection between homotopy classes of maps $f : X \rightarrow BG$ and equivalence classes of principal G -bundles over X .

Eilenberg-Mac Lane spaces

Inductive types are collections of data and rules for mapping out of instances of the collection. For example here is an inductive type defining the natural numbers in Agda:

```
data nat : Type where
  zero : nat
  succ : nat → nat
```

This theory can be extended to support data having values in the identity types of data that was specified earlier in the definition. Here is the circle defined as a HIT in Cubical Agda:

```
data circle : Type where
  point : circle
  loop : point ≡ point
```

Given a group G (meaning, a 0-type) with identity $G.1g$ and multiplication $G.mul$, we can form the Eilenberg-Mac Lane type $K(G, 1)$ as the following HIT:

```
data eil-mac (G : Group) : Type where
  embase : eil-mac
  emloop : G → (embase ≡ embase)
  emloop-id : emloop G.1g ≡ refl embase
  emloop-comp : (x y : G) → emloop (x G.mult y) ≡ (emloop x) · (emloop y)
  emsquash : ∀ (x y : eil-mac) (p q : x ≡ y) (r s : p ≡ q) → r ≡ s
```

Cohomology

Universes versus bundles

Towards connections

References

- [1] D. S. Freed and M. J. Hopkins, “Chern-weil forms and abstract homotopy theory,” 2013. [Online]. Available: <https://arxiv.org/abs/1301.5959>
- [2] S. Kobayashi and K. Nomizu, *Foundations of Differential Geometry*, ser. Foundations of Differential Geometry. Interscience Publishers, 1963, no. v. 1. [Online]. Available: <https://books.google.com/books?id=wn4pAQAAMAAJ>
- [3] M. Hamilton, *Mathematical Gauge Theory: With Applications to the Standard Model of Particle Physics*, ser. Universitext. Springer International Publishing, 2017. [Online]. Available: <https://books.google.com/books?id=8zVCDwAAQBAJ>
- [4] A. Scorpan, *The Wild World of 4-Manifolds*. Providence, R.I: American Mathematical Society, May 2005.

- [5] J. Baez and J. Muniain, *Gauge Fields, Knots And Gravity*, ser. Series On Knots And Everything. World Scientific Publishing Company, 1994. [Online]. Available: <https://books.google.com/books?id=qvw7DQAAQBAJ>
- [6] M. F. Atiyah and R. Bott, “The yang-mills equations over riemann surfaces,” *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, vol. 308, no. 1505, pp. 523–615, 1983.
- [7] M. Bezem, U. Buchholtz, P. Cagne, B. I. Dundas, and D. R. Grayson, “Symmetry,” <https://github.com/UniMath/SymmetryBook>.
- [8] T. Univalent Foundations Program, *Homotopy Type Theory: Univalent Foundations of Mathematics*. Institute for Advanced Study: <https://homotopytypetheory.org/book>, 2013.