Isomorphism Revisited

David McAllester

December 6, 2019

Abstract

Isomorphism is central to the structure of mathematics and has been formalized in various ways within dependent type theory. All previous treatments have done this by replacing quantification over sets with quantification over groupoids of some form — categories in which every morphism is an isomorphism. Quantification over sets is replaced by quantification over standard groupoids in the groupoid model, by quantification over infinity groupoid in homotopy type theory, and by quantification over morphoids in the morphoid model. Our treatment in [6] is based on the intuitive notion of sets as collections without internal structure. Quantification over sets remains as quantification over sets. Isomorphism and groupoid structure then emerge from simple but subtle syntactic restrictions on set-theoretic language. This approach more fully unifies the classical ZFC foundations with a rigorous treatments of isomorphism, symmetry, canonicality, functors, and natural transformations. This is all done without reference to category theory.

1 Introduction

Isomorphism is central to the structure of mathematics. Mathematics is organized around concepts such as graphs, groups, topological spaces and manifolds each of which is associated with a notion of isomorphism. Each concept is associated with a classification problem — can we enumerate the instances of a given concept *up to isomorphism*. We also have the related notions of symmetry and canonicality. There is no canonical point on a geometric circle — any point can be mapped to any other point by rotating the circle. A rotation of the circle is an isomorphism of the circle with itself — a symmetry or automorphism. Similarly, there is no canonical basis for a finite dimensional vector space. For any basis there is a symmetry (automorphism) of the vector space which moves the basis to a different basis — a situation precisely analogous to a point on a circle. Isomorphism is also central to understanding representation. A group can be represented by a family of permutations. Different (non-isomorphic) families of permutation can represent the same group (up to isomorphism).

At first isomorphism seems simple. The notion of isomorphic graphs, and the intuition that two isomorphic graphs are "the same", seems intuitively clear to essentially anyone who encounters the concept. Indeed, for a broad class of concepts the notion of isomorphism is easily defined. More specifically we can consider concepts defined by a carrier set of "points" plus predicates, relations and functions providing structure on those points. A graph consists of a set of nodes (points) plus an edge relation on the nodes. For concepts defined by a carrier set plus structure, two instances are isomorphic if there exists a bijection between their points which identifies their structure — which "carries" the structure of one to the structure of the other. This notion of isomorphism is easily formalized for concepts defined as the models of a given (higher order) signature where a signature is a set of predicate symbols, relation symbols and function symbols operating over a carrier set of points.

But general mathematics is carried out in a language richer than that defined by a single higher order signature. Mathematical statement typically involve several different instances of several different concepts. For example, we can abstract a document — a sequence of words — to a multiset (bag) of words. When we do this we understand that structure has been lost. It is more subtle than simply noting that different sequences can map to the same bag. A bag fundamentally has less structure than a sequence. The grammatical (well typed) statements about a bag are more restricted than the grammatical statements about a sequence. We cannot talk about the first element of a bag.

Dependent type theory [1] is a formal system for specifying interfaces to objects and can be used as a formal foundation for mathematics [2]. Unlike set theory, type theory handles concepts (types) with a specified interface to the instances of each concept. Type theory allows for statements relating concepts and their instances in a way that mirrors natural mathematical language.

Isomorphism has been formalized in dependent type theory using the groupoid model [3]. The groupoid model replaces quantification over sets with quantification over groupoids — categories in which every morphism is an isomorphism. Homotopy type theory replaces quantification over sets with quantification over a form of infinity groupoid related to algebraic topology. The Morphoid model achieves compositionality by replacing quantification over sets with quantification over "morphoids" [5].

In [6] we achieve a treatment of isomorphism which preserves the intuitive concept of a set as a collection without internal structure — quantification over sets remains as quantification over sets. Isomorphism and groupoid structure then emerge naturally from simple but subtle syntactic restrictions on set-theoretic language. Functors and natural transformations also emerge naturally without any explicit introduction of groupoids or category theory.

Kevin Buzzard, in his talk at AITP 2018, described the understanding of canonical isomorphisms as a human superpower. Hopefully the approach to isomorphism given in [6] will facilitate the automation of this superpower.

References

[1] Per Martin Löf. An intuitionistic theory of types: predicative part. In *Logic Colloquium '73 (Bristol, 1973)*, volume 80 of *Studies in Logic and the Foundations of Mathematics*. North-Holland, 1975.

[2] B. Barras, S. Boutin, C. Cornes, J. Courant, J.C. Filliatre, E. Gimenez, H. Herbelin, G. Huet, C. Munoz, C. Murthy, et al. The coq proof assistant reference manual: Version 6.1. INRIA Research Report, 1997.

[3] Martin Hofmann and Thomas Streicher. The groupoid interpretation of type theory. In *Twenty-five years of constructive type theory (Venice, 1995)*. Oxford Univ. Press, New York, 1998.

[4] HoTT-Authors. Homotopy type theory, univalent foundations of mathematics. http://hottheory.files.wordpress.com/2013/03/hott-online-611-ga1a258c.pdf, 2013.

[5] David McAllester. Set-theoretic type theory. *CoRR*, abs/1407.7274, 2018.

[6] David McAllester. Isomorphism Revisited. Scheduled for announcement on ArXiv 1:00 GMT, Monday December 9, 2019.