EPiC Series in Computing

Volume 68, 2020, Pages 26–36

ANDREI-60. Automated New-era Deductive Reasoning Event in Iberia

Who needs category theory?

Andreas Blass and Yuri Gurevich

University of Michigan, Ann Arbor, Michigan, U.S.A.
ablass@umich.edu, gurevich@umich.edu

Abstract

In mathematical applications, category theory remains a contentious issue, with enthusiastic fans and a skeptical majority. In a muted form this split applies to the authors of this note. When we learned that the only mathematically sound foundation of topological quantum computing in the literature is based on category theory, the skeptical author suggested to "decategorize" the foundation. But we discovered, to our surprise, that category theory (or something like it) is necessary for the purpose, for computational reasons. The goal of this note is to give a high-level explanation of that necessity, which avoids details and which suggests that the case of topological quantum computing is far from unique.

1 Introduction

Category theory is indispensable in some parts of mathematics, e.g. in algebraic geometry, homological algebra, algebraic topology. Yet, even among mathematicians the attitude toward category theory varies greatly, as witnessed by the following joke of John Baez [1].

> I hope most mathematicians continue to fear and despise category theory,
> so I can continue to maintain a certain advantage over them.

The chasm between the fans of category theory and the silent majority is even more pronounced in computer science where the fans tend to be super-enthusiastic while the majority is indifferent.

In a muted form this split applies to the authors of this note. As we mentioned in [2, §1], "The first author of this paper has long been a fan of category theory; even as a graduate student, he was described by one of his professors as 'functorized'. The second author has been far more skeptical about the value of category theory in computer science."

It turns out, however, that the only mathematically sound foundation of topological quantum computing in the literature is based on category theory; see [2, 16, 17] for example. Why?

Is this just an accident of history or there is more to it? We have been debating this question for a while, and now we agree that something like category theory is necessary for the purpose.

Categories were introduced by Samuel Eilenberg and Saunders Mac Lane as an auxiliary notion in their theory of natural equivalences [5], and category theory is famous for its high abstraction level. Here we posit that there is a more basic need for something like categories. Ironically, the basic need is related to the problem of over-abstraction rather than under-abstraction.

It is common in mathematics and its applications to deal with algebraic operations on structures; think for example about direct sums and tensor products of vector spaces. Various properties of these algebraic operations come into play: associativity, commutativity, distributivity, etc. But traditional algebra may be insufficient. For various reasons, in particular for computational reasons, knowing that two structures are isomorphic (or equivalent in some sense or another) may not be good enough. We need to have a particular isomorphism (or appropriate equivalence) to witness the isomorphism. And we may have to coherently manipulate such witnesses. Hence the *coherent witness-manipulation problem*.

We came across the coherent witness-manipulation problem in topological quantum computing. But the problem seems to be much more general. We believe that pure and applied mathematicians, computer scientists, and theoretical physicists are well advised to know that category theory or something like it provides an appropriate framework to address the coherent witness-manipulation problem.

2 What's category theory?

For those who have only a vague idea of category theory, let us say a few words about it. The experts can safely skip this section.

Categories were introduced by Samuel Eilenberg and Saunders Mac Lane as an auxiliary notion in their general theory of natural equivalences [5]. "It is not too misleading, at least historically, to say that categories are what one must define in order to define functors, and that functors are what one must define in order to define natural transformations," writes Peter Freyd in the introduction to his book [7].

In the rest of this section, we quickly explain the notion of natural equivalence.

2.1 Categories and functors

Definition 2.1. A *category* comprises

1. a collection of *objects*,
2. for any objects x, y, a collection of *arrows* $\alpha : x \to y$
   including, if $x = y$, an *arrow* $1_x : x \to x$ called *identity*,
3. a *composition* $\beta\alpha : x \to z$ of arrows $x \xrightarrow{\alpha} y \xrightarrow{\beta} z$ which is associative and treats the identity arrows as expected.

An arrow with a two-sided inverse is an *isomorphism*. ◁

Examples 2.2.

1. Sets, total functions, and function composition.
2. Groups, group homomorphisms, and function composition.

We will be using below a restricted version of the first of the two examples where objects are finite sets and arrows are isomorphisms, that is, one-to-one correspondences. For brevity, this category of finite sets with isomorphisms will be denoted `FinSet+Iso`.

Definition 2.3. Let C, D be categories. A *functor* $F : C \to D$ is a mapping from C-objects and C-arrows to D-objects and D-arrows respectively such that

- if $\alpha : x \to y$ then $F\alpha : Fx \to Fy$,
- $F1_x = 1_{Fx}$,
- $F(\beta\alpha) = (F\beta)(F\alpha)$.

Examples 2.4. We describe two functors F and G from `FinSet+Iso` to `FinSet+Iso`. Let S and S' be arbitrary finite sets.

1. $F(S)$ is the set of all permutations of S, that is, bijections from S to S. For any isomorphism $\alpha : S \to S'$, $F\alpha$ transforms every permutation π of S into a permutation $\alpha\pi\alpha^{-1}$ of S'. That is, if π maps x to y then $(F\alpha)(\pi)$ maps αx to αy. It is easy to check that $F(\beta\alpha) = (F\beta)(F\alpha)$.
2. $G(S)$ is the set of linear orderings of S. For any isomorphism $\alpha : S \to S'$, $G\alpha$ transforms every linear ordering $<$ of S into the ordering

$$s <' t \iff \alpha^{-1}s < \alpha^{-1}t$$

of S'. It is easy to check that $G(\beta\alpha) = (G\beta)(G\alpha)$.

2.2 Natural equivalences

We start with a motivating example. For any finite set S, there are as many permutations of S as linear orderings. If n is the cardinality of S then there are $n!$ permutations and $n!$ linear orderings. It follows that there is a bijection between the permutations and linear orderings of S.

If $n \geq 2$, there are multiple such bijections. Yet, on the level of abstraction where you don't distinguish between elements of S, you cannot single out any such bijection. The reason is that linear orderings are all automorphic while permutations are not. For example, the identity permutation is preserved by all automorphisms.

On the other hand, suppose that S comes furnished with a particular linear order

$$s_1, s_2, \ldots, s_n$$

where n is the cardinality of S. Then we have a standard bijection, let us call it τ_0, between the permutations and the linear orders: $\tau_0(\pi)$ is the linear order

$$\pi(s_1), \pi(s_2), \ldots, \pi(s_n).$$

This bijection is natural in that it works uniformly for any finite set S furnished with a linear order.

Definition 2.5. Given two functors $F, G : C \to D$, a *natural transformation* τ of F to G assigns to each object x of C an arrow $\tau x : Fx \to Gx$ of D in such a way that every arrow $\alpha : x \to y$ in C yields a commutative diagram

$$\begin{array}{ccc} Fx & \xrightarrow{\tau x} & Gx \\ {\scriptstyle F\alpha}\downarrow & & \downarrow{\scriptstyle G\alpha} \\ Fy & \xrightarrow{\tau y} & Gy \end{array}$$

Further, τ is a *natural equivalence* if every τx is an isomorphism. ◁

Coming back to the motivating example, let C and D be the category `FinSet+Iso`. The functors F, G of Examples 2.4 are not naturally equivalent, for the reason mentioned above. Indeed, suppose toward contradiction that τ is a natural equivalence from F to G. Let x and y be the same set $S = \{a, b\}$, so that there are two permutations of S, $\pi_1 = \left(\begin{smallmatrix} a & b \\ a & b \end{smallmatrix}\right)$ $\pi_2 = \left(\begin{smallmatrix} a & b \\ b & a \end{smallmatrix}\right)$, and two linear orderings, $a <_1 b$ and $b <_2 a$. Let α transpose a and b, so that $\alpha^{-1} = \alpha$, $(F\alpha)(\pi_i) = \alpha\pi\alpha = \pi_i$ and $(G\alpha)(<_i) = <_{3-i}$. If $\tau\pi_i = <_i$ then

$$(G\alpha)(\tau x)(\pi_1) = (G\alpha)(<_1) = <_2$$
$$(\tau y)(F\alpha)(\pi_1) = (\tau y)(\pi_1) = <_1$$

If $\tau\pi_i = <_{3-i}$ then

$$(G\alpha)(\tau x)(\pi_1) = (G\alpha)(<_2) = <_1$$
$$(\tau y)(F\alpha)(\pi_1) = (\tau y)(\pi_1) = <_2$$

In either case, the diagram above does not commute.

On the other hand, suppose that the domains of F and G are modified so that, for any object x, the finite set x is furnished with a fixed linear order and, for any two objects, x and y, the arrows from x to y respect the fixed orders, so that, in fact, there is a unique arrow from x to y. Then the functors F, G become naturally equivalent. The standard bijection τ_0, described above, makes the diagram commute. (In the case where x and y are the set $\{a, b\}$ with the same linear order, the transposition is not a legitimate arrow; the only legitimate arrow is the identity.)

3 A dialog on category theory and its applications

Q[1]**:** The formalization of the intuitive notion of natural equivalence is impressive. Can you explain other advantages of category theory to me? Unfortunately I don't know more category theory than you have just taught me here.

A[2]**:** Let us illustrate one advantage of category theory. It is related to the notion of isomorphism. You worked with isomorphisms.

Q: Sure. I know the notion from universal algebra in general and group theory in particular. To me, an isomorphism is a homomorphism that happens to be bijective. Judging by Examples 2.2, the notion of isomorphism of Definition 2.1 is rather similar. I am thinking of arrows as homomorphisms. The two notions of isomorphism may be equivalent.

A: Actually, the categorical notion of isomorphism is more general. Consider the category of partially ordered sets (in short, posets) with monotone maps. Let A, B be two-element posets where the two elements are incomparable in A but ordered in B. Any bijection from A to B is a monotone map but categorically — and intuitively! — the two posets are not isomorphic.

Another category where not all bijective arrows are isomorphisms is the category of topological spaces with continuous functions. Here, isomorphisms are exactly homeomorphisms, but continuous bijections form a larger class. For example, let A, B be two-element topological spaces where every subset of A is open but, in B, only $\emptyset$ and B itself are open. Any bijection from A to B is continuous but not a homeomorphism. For a more interesting example, let A be the half-open interval $[0, 1)$ of the real line and B be the unit circle in the complex plane. The bijection $r \mapsto e^{2\pi ir}$ is continuous but its inverse is discontinuous at 1.

Q: Fascinating. Anything else?

A: What do these mathematical constructions

- free groups,
- tensor algebra,
- universal enveloping algebras,
- abelianizations of groups, and
- Stone-Čech compactifications

have in common?

Q: I don't know. They come from different parts of mathematics and look disparate to me. Certainly, the free group construction and group abelianization are quite different. There is something universal about each of the constructions, but I don't see more than that. Oh, wait. I guess that every one of these constructions is a functor.

[1] Quisani, a former student of the second author
[2] The authors, speaking one at a time

A: You are right, all these constructions can be viewed as functors. But the amazing part is that these are examples of a single categorical construction. All these functors are left adjoints of forgetful functors. Unfortunately, we haven't covered forgetful functors or adjoints here.

Q: The unifying power of category theory seems awesome. That's got to be useful in many areas of mathematics, I guess. What about computing?

A: The usefulness of category theory in computing is less obvious. Until recently when we started to work on topological quantum computing (TQC for short), one of us had been skeptical.

Q: Why?

A: Because of the distance of this very abstract theory from computing and because of the peril of potential (and in some cases actual) over-abstraction. There is also the hammer-and-nail phenomenon: "For a person with a hammer, everything looks like a nail."

Q: You don't mean that category theory itself is an over-abstraction.

A: No, we don't. As Seneca the Younger said in the first century, "gladius neminem occidit: occidentis telum est," that is "a sword kills nobody; it is a tool of the killer."

Q: Give me a relevant example of that hammer-and-nail phenomenon.

A: Here is a true life example, but allow us to omit the reference. A computation can be seen as a category where objects are states and morphisms are state transitions. If you take this point of view, then you might want computation transformers to be functorial, which narrows unreasonably your library of computation transformers. For example, you lose compilers.

Q: Why isn't a compiler functorial?

A: Typically, the target language is at a lower abstraction level and uses different data structures. Some higher-level steps may have no meaning at the lower level. Besides, think of compiler optimization.

Q: How did topological quantum computing influence the skepticism? And what is topological quantum computing?

A: Topological quantum computation employs two-dimensional quasiparticles called anyons [6, 11]. What is relevant for our purposes here is that the generally accepted mathematical basis for the theory of anyons is the framework of modular tensor categories. That framework, as presented in [17] or [16] or [2] involves a substantial amount of category theory and is, as a result, considered rather difficult to understand.

Why is the only mathematically sound theory of anyons in the literature based on category theory? The skeptic among us suspected that this is just an accident of history, another nail for the categorical hammer. Hence the idea to "decategorize" the theory of anyons.

As we worked on the decategorization project, we realized that, surprisingly, category theory — or something like category theory — is necessary for the theory of anyon computations.

Q: Can you explain to me, who knows nothing about anyons, why category theory is necessary for the purpose? Is the reason specific to the anyon theory?

A: The reason seems to us more generic and not at all specific to the anyon theory, but at this point we do not have other natural examples where category theory is necessary for the same reason.

In the next section, we will try to illustrate the reason behind the necessity of category theory (or something like it) for the theory of anyons.

4 Coherent witness manipulation

We illustrate why (something like) category theory is needed in topological quantum computing.

4.1 Algebra of structures

Consider an algebra $\mathcal{A}$ of structures (that is, elements of $\mathcal{A}$ are structures) together with operations of addition $+$ and multiplication $*$ where, up to isomorphism,

- both operations are commutative and associative,
- both operations have their respective neutral elements $\mathbf{0}$ and $\mathbf{1}$, and
- multiplication distributes over addition.

The following example is a simplified version of the algebra used in topological quantum computing.

- The structures in $\mathcal{A}$ are finite-dimensional vector spaces, over the field of complex numbers, each furnished with a fixed basis.
- The vector space $A + B$ is the direct sum, also known as the direct product, of vector spaces A and B furnished with the disjoint union (of a particular form[3]) of the fixed bases of A and B.
- The product $A * B$ is the tensor product of the vector spaces A and B furnished with the cartesian product of the fixed bases of A and B.

[3] In the standard construction of $A + B$, the base set is the set of ordered pairs (a, b) where $a \in A$ and $b \in B$. With that convention, the standard basis for $A + B$ consists of vectors $(a, 0)$ for a in the standard basis of A and $(0, b)$ for b in the standard basis of B.

In the case of topological quantum computing, the structures are more sophisticated — involving e.g. tuples of Hilbert spaces, duality, ribbon structures — but this is not important. For the purposes of this note, we may and will pretend that the example above is the one used in topological quantum computing.

So far, we are within the realm of universal algebra. But we need to go beyond universal algebra.

4.2 Witness isomorphisms

Suppose that algebra $\mathcal{A}$ is equipped with standard isomorphisms

`associative +`	$\alpha^+_{A,B,C}:$	$(A+B)+C$	$\to$	$A+(B+C)$
`commutative +`	$\gamma^+_{A,B}:$	$A+B$	$\to$	$B+A$
`associative` $*$	$\alpha^*_{A,B,C}:$	$(A*B)*C$	$\to$	$A*(B*C)$
`commutative` $*$	$\gamma^*_{A,B}:$	$A*B$	$\to$	$B*A$
`distributive`	$\delta_{A,B,C}:$	$A*(B+C)$	$\to$	$(A*B)+(A*C)$

working properly with $\mathbf{0}$ and $\mathbf{1}$.

It is convenient to think of an isomorphism $\xi : A \to B$ as a *witness* that A, B are isomorphic. Accordingly, the standard isomorphisms above are *standard witnesses*.

There are numerous requirements that we have to impose on the standard isomorphisms. In particular, it is required that $\gamma^+_{A,B} = (\gamma^+_{B,A})^{-1}$. This property is called *symmetry*. Thus the additive structure of $\mathcal{A}$ is symmetric.

A relevant peculiarity of topological quantum computing is that the multiplicative structure is not symmetric. It is not required that $\gamma^*_{A,B}$ coincides with $(\gamma^*_{B,A})^{-1}$. It is convenient to think about this topologically: as $A * B$ is transformed into $B * A$, it matters whether A passes in front of or behind B. The isomorphisms $\gamma^*_{A,B}$ are $(\gamma^*_{B,A})^{-1}$, known as braiding isomorphisms, are in general different. The multiplicative structure of $\mathcal{A}$ is *braided*.

The most important aspect is related to computing. It is not enough for us to know that there are two braiding isomorphisms from $A * B$ to $B * A$ or that there is an associativity isomorphism from $(A*B)*C$ to $A*(B*C)$. We need these isomorphisms, in matrix form with respect to the fixed bases, for computational purposes.

4.3 Witness requirements

This subsection is more specialized. We mentioned above that the standard witnesses are subject to various requirements. One may wonder what are those requirements. We give the appropriate references.

For the additive structure, the appropriate requirements were found by Mac Lane [13] and subsequently simplified by Kelly [10]. For the braided multiplicative structure, the requirements were supplied by Joyal and Street [8, 9].

Multiplication interacts with addition via the distributivity laws. For the case where both the additive and multiplicative structures of $\mathcal{A}$ are symmetric, the requirements for distributivity have been identified by Miguel Laplaza [14, 15] who was a postdoc of Mac Lane. In [3], we identified appropriate requirements for distributivity in the case where the additive structure is symmetric but the multiplicative structure is braided.

4.4 The additive structure

This subsection is devoted to a technical issue of independent interest.

We hoped that the addition operation + on $\mathcal{A}$ can be taken to be literally (not only up to isomorphism) commutative and associative, that is, that we can get by with the identity witnesses for the commutativity and associativity of addition. Unfortunately this is impossible.

For illustration of what goes wrong with the identity witnesses, we simplify the example described above. Let's abstract from vector spaces and concentrate on their fixed bases: finite sets with disjoint union as addition. We recall the standard definition of disjoint union of sets.

`Definition 1.` The disjoint union of sets A, B is the set

$$A + B = \{(a, 0) : a \in A\} \cup \{(b, 1) : b \in B\}. \qquad \triangleleft$$

Q: Definition 1 does not look standard to me. If fact, it looks rather arbitrary. Instead of 0 and 1, I can use different tags, say, 1 and 2.

A: It takes a bit of category theory to explain the standard character of the definition. For any choice of the two tags, there is a natural enhancement of the definition with canonical embeddings of A and B into $A + B$; the resulting operation has the universal property of the coproduct. That is what makes the definition, in any of these variations, standard.

This disjoint union of Definition 1 is neither commutative nor associative. One may think that there is no definition that is better in the sense that it makes disjoint union literally, not just up to isomorphism, commutative and associative. But such a "better" definition does exist. Let $\mathbb{N}$ be the set of natural numbers, i.e., nonnegative integers.

`Definition 2.` The disjoint union of finite sets A, B is the set

$$A \dot{+} B = \{n \in \mathbb{N} : n < |A| + |B|\}. \qquad \triangleleft$$

Let's adopt the set-theoretic convention that a natural number is the set of smaller natural numbers. Then Definition 2 says that the disjoint union $A \dot{+} B$ is the number $|A| + |B|$.

It is easy to see that $A \dot{+} B$ is indeed commutative and associative, so that the standard witnesses for the commutativity and associativity can be taken to be identities. Unfortunately,

we cannot push our luck too far. For example, if a finite set A is not a number then the equality $A = A\dot{+}\emptyset$ cannot be witnessed by the identity. For, if A is identical to $A\dot{+}\emptyset$, then A is a number. Furthermore, there is no canonical embedding of A into $A\dot{+}B$.

Q: How about generalizing sets to multisets? There is, I think, a natural disjoint union of multisets which is commutative and associative.

A: Recall that our finite sets are fixed bases of vector spaces.

Q: I see the problem. A multiset basis of a vector space does not make much sense.

5 Summary

As we mentioned above, categories were introduced by Samuel Eilenberg and Saunders Mac Lane as an auxiliary notion in their general theory of natural equivalences [5]. Here we argue that something like categories is needed on a more basic level.

As you work with operations on structures, it may be necessary to coherently manipulate witnesses for various properties of these operations. We mentioned associativity, commutativity and distributivity, but many additional properties are in play in topological quantum computing and elsewhere. The coherent witness-manipulation problem may be hard.

This necessity of coherent witness-manipulation cannot be proven mathematically, and in some cases one can get around the coherent witness-manipulation problem. For example, for limited purposes, the narrow problem of a reasonable definition of commutative and associative disjoint union of sets can be solved by generalizing sets to multisets. Unfortunately this solution is of little help if the sets in question are vector-space bases.

In general, a working mathematician, to use Mac Lane's term [12], is well advised to be aware of the coherent witness-manipulation problem and to know that category theory or something similar provides an appropriate framework to address the problem. Of course, the working mathematician in question may be a computer scientist or physicist.

Q: What do you mean by something similar to category theory?

A: We didn't want to rule out possible alternatives. In some situations, it suffices to consider groupoids, i.e., to restrict attention to isomorphisms. This setting can be presented in a way closer to traditional algebra [4].

Q: Is there an objective need to deal with more general homomorphisms?

A: Yes, isomorphisms are sometimes insufficient. Consider, for example, Definition 2 of disjoint union. Why does it feel so lousy? One reason is that it does not say where A and B are in the disjoint union. To have a useful disjoint union, one needs even more, namely where individual

elements of A and B lie in the disjoint union. That information amounts to embeddings of A and B into the disjoint union, and those are not isomorphisms.

Acknowledgement

We thank Samson Abramsky, John Baez, Bob Coecke and Prakash Panangaden for their comments.

References

[1] John Baez. Opinions of category theory. https://www.arsmathematica.net/2006/06/24/opinions-of-category-theory/, 2006.

[2] Andreas Blass and Yuri Gurevich. On quantum computation, anyons, and categories. In "Martin Davis on Computability, Computational Logic, and Mathematical Foundations", 209–241, Springer 2016. Also in arXiv:1502.00669.

[3] Andreas Blass and Yuri Gurevich. Braided distributivity. *Theoretical Computer Science* 807 73–94 2020. Also in arXiv:1807.11403.

[4] Andreas Blass and Yuri Gurevich. Witness algebra and anyon braiding. *Mathematical Structures in Computer Science* 2020, to appear. Also in arXiv:1807.10414.

[5] Samuel Eilenberg and Saunders Maclane. General theory of natural equivalences. *Transactions of the American Mathematical Society* 58:2 231–294 1945.

[6] Michael H. Freedman, Alexei Kitaev, Michael J. Larsen, and Zhenghan Wang. Topological quantum computation. *Bulletin of the American Mathematical Society* 40:1 31–38 2003.

[7] Peter J. Freyd. Abelian categories. *Reprints in Theory and Applications of Categories* 3 2003, http://www.tac.mta.ca/tac/reprints/, originally published by Harper & Row in 1964.

[8] André Joyal and Ross Street. Braided tensor categories. *Macquarie Mathematics Reports* 860081 1986.

[9] André Joyal and Ross Street. Braided tensor categories. *Advances in Mathematics* 102 20–78 1993.

[10] G. Max Kelly. On MacLane's conditions for coherence of natural associativities, commutativities, etc. *Journal of Algebra* 1 397–402 1964.

[11] Alexei Kitaev. Fault-tolerant quantum computation by anyons. arXiv:quant-ph/9707021 1997.

[12] Saunders Mac Lane. Categories for the working mathematician. Springer 1971.

[13] Saunders Mac Lane. Natural associativity and commutativity. *Rice University Studies* 49:4 28–46 1963.

[14] Miguel Laplaza. Coherence for categories with associativity, commutativity and distributivity. *Bulletin of the American Mathematical Society* 78 220–222 1972.

[15] Miguel Laplaza. Coherence for distributivity. In "Coherence in Categories," eds. G.M. Kelly et al.,*Springer Lecture Notes in Mathematics* 281 29–65 1972.

[16] Prakash Panangaden and Éric O. Paquette. A categorical presentation of quantum computation with anyons. In "New Structures for Physics," ed. Bob Coecke, *Springer Lecture Notes in Physics* 813 983–1025 2011.

[17] Zhenghan Wang. Topological quantum computation. *CBMS Regional Conference Series in Mathematics* 112, American Mathematical Society 2010.