A Library of Monoidal Categories for Display and Univalence

Benedikt Ahrens[1,2], Ralph Matthes[3], and Kobe Wullaert[1]

[1] Delft University of Technology, The Netherlands
[2] University of Birmingham, United Kingdom
[3] IRIT, Université de Toulouse, CNRS, Toulouse INP, UT3, Toulouse, France

In this work in progress, we aim at developing a usable library of univalent monoidal categories. In particular, we develop the notion of displayed monoidal category for easier construction of complicated monoidal categories and proof of their univalence.

A new formalization of monoidal categories A monoidal structure on a category C is given by a tensor product $\otimes$, that is, a functorial binary operation on the objects and morphisms of C. Furthermore there is a unit object I, that is neutral for the tensor operation modulo (natural) isomorphism. The tensor is associative up to isomorphism, and a pentagon law holds for that isomorphism, as well as a triangle law connecting all three isomorphisms.

Functors between categories are accordingly extended to monoidal functors, so as to ensure preservation of the extra structure. The operations doing this for tensor and unit have to interact properly with the aforementioned isomorphisms. *Strong* monoidal functors need these operations to be isomorphisms, while *strict* monoidal functors satisfy the laws "on the nose".

Typically, the tensor product is seen as a bifunctor on C, i. e., a functor from $\mathsf{C} \times \mathsf{C}$ to C. Our previous attempts at defining displayed monoidal categories (see next paragraph) on that basis suffered from major difficulties with transport along components of pairs arising with the use of this product category $\mathsf{C} \times \mathsf{C}$. Instead of working with two-place functions (encoded by pairing), one can move to a curried view that first takes the left argument and then is a function that expects the right-hand side argument—which is good for the object mapping. For the two-place morphism mapping, we employ a symmetric approach, by considering the one-place mappings where the left resp. right argument is fixed to the identity, which we call the left resp. right *whiskering*, respectively. This notion is not confined to the tensor of a monoidal category but is an alternative view for any bifunctor $\mathsf{A} \times \mathsf{B} \to \mathsf{C}$. However, calling it whiskering comes from the analogous treatment of horizontal composition in bicategories in the UniMath library.

As a benefit, the formal development of monoidal categories in this format is in close correspondence with bicategories (as they are formalized in UniMath [1, Definition 2.1])—mathematically, monoidal categories are just one-object bicategories.

A formalization of displayed monoidal categories A displayed category D over some base category C ([2] involving the first author) allows one to add structure (resp. properties) to objects (resp. morphisms) of C without having to reconstruct all the information which is already known about C. More technically, D provides the data of a category and a functor into C by specifying the fibers of that functor; the category, called $\int \mathsf{D}$ (the *total* category), and the functor $\pi_1 : \int \mathsf{D} \to \mathsf{C}$ can then be constructed from those fibers. An advantage of the displayed presentation is that certain equalities satisfied by π_1 can be made to hold definitionally.

The notion of displayed category has led to "displayed versions" of numerous categorical concepts and is seen to have potential for much more [5]. In general, "displaying" means constructing on top of the ingredients of the underlying category C in a concise way, avoiding copying as much as possible.

Of special interest to our application is the identification of the class of functors F from C to $\int \mathsf{D}$ that have $\pi_1 \circ F = 1_{\mathsf{C}}$. This has been formalized in the UniMath library as "sections",[1] for which such a functor F can be generically constructed. Of course, this is not formulated in terms of the total category but gives adaptations of the functor laws to hold "over" C. By working with sections, we efficiently replace the very cumbersome use of $\pi_1 \circ F = 1_{\mathsf{C}}$ as an equation between functors for rewriting (functor equality is very bad for rewrites from the point of view of intensional type theory) by definitional equality.

Given a monoidal category C in the whiskered format and a displayed category D over (the base category of) C, we add a tensor "over" the tensor of C (as a displayed bifunctor, similarly defined in whiskered format) and add the other ingredients and laws to have a definition of what is a displayed monoidal category. Let D^+ be any such displayed monoidal category. A corresponding total (curried) monoidal category $\int \mathsf{D}^+$ is then obtained, with a forgetful functor π_1 back to C that we show to be strict monoidal.

Then, we identify the class of strong monoidal functors F from C to $\int \mathsf{D}^+$ s. t. $\pi_1 \circ F = 1_{\mathsf{C}}$ with a concise description in terms of D^+, which gives the notion of strong monoidal sections.

Application scenario Recent work [3, §4.3] involving the first and second authors attempted to establish a bijection between the class of parameterized distributivities (given actions in the sense of Janelidze and Kelly [4] as strong monoidal functors into some functor category) and the monoidal sections for a specially crafted displayed monoidal category, but that work only obtained a function in one direction. Using our new definition of monoidal category and the resulting workable definition of displayed monoidal category, we have been able to construct the full bijection: we have constructed the missing function, using the aforementioned monoidal sections of that displayed monoidal category as its domain, and shown that the two functions are inverse to each other (for a bicategorical generalization). We have thus solved the open question of [3, §4.3]; furthermore, the compilation time of the respective file[2] is 53% less than for the original one (43s versus 92s wall clock time measured on some current Intel processor).

More examples We constructed the displayed monoidal categories of pointed sets and of sets with binary operations, but also generically a displayed (cartesian) monoidal category from any cartesian monoidal category and *displayed* finite products. The also constructed displayed monoidal category of pointed endofunctors is based on a tensor that is not the product, and this construction is concise and elegant (to our eyes), through the display mechanism.[3]

WIP: univalent bicategory of univalent monoidal categories We are working on a proof of univalence of the bicategory $\mathsf{UMonCat}$ of univalent monoidal categories. The machinery of displayed bicategories allows us to construct a bicategory $\mathsf{UMonCat}_{TU}$ whose objects are univalent categories together with the data of a tensor product (together with the functoriality axioms) and a fixed object. The morphisms (resp. 2-cells) are functors (resp. natural transformations) which preserve the tensor product and the unit in a weak sense. This is done by constructing two displayed bicategories on the bicategory of univalent categories. The associator and both unitors, as well as the pentagon and triangle identities, each require an extra displayed bicategory on $\mathsf{UMonCat}_{TU}$, in total five. Since $\mathsf{UMonCat}$ is constructed using these different *layers*, the proof of univalence then reduces to showing that each displayed layer is univalent, reusing that the bicategory of univalent categories is univalent. Univalence of the

[1] https://github.com/UniMath/UniMath/blob/1580dab0/UniMath/CategoryTheory/DisplayedCats/Constructions.v#L399

[2] https://github.com/UniMath/UniMath/blob/1580dab0/UniMath/Bicategories/MonoidalCategories/ActionBasedStrongFunctorsWhiskeredMonoidal.v

[3] https://github.com/UniMath/UniMath/blob/fa48ad1/UniMath/Bicategories/MonoidalCategories/PointedFunctorsWhiskeredMonoidal.v

bicategory of *strong* monoidal categories then also follows, through extra displayed bicategories on UMonCat for the *strongness*-conditions.

Conclusion We have presented work toward a library on univalent monoidal categories, implemented in the UniMath library. The released code consists of around 6kloc (of Coq vernacular and proofs measured with the tool `coqwc`), among them 2.1kloc for the application scenario (code mostly not written from scratch but adapted from the earlier approach), the rest about one third vernacular and two thirds proof, and additionally, 1.1kloc for the WIP.

References

[1] Benedikt Ahrens, Dan Frumin, Marco Maggesi, Niccolò Veltri, and Niels van der Weide. Bicategories in univalent foundations. *Mathematical Structures in Computer Science*, 2022. 38 pages, online publication at https://doi.org/10.1017/S0960129522000032.

[2] Benedikt Ahrens and Peter LeFanu Lumsdaine. Displayed categories. *Log. Methods Comput. Sci.*, 15(1), 2019.

[3] Benedikt Ahrens, Ralph Matthes, and Anders Mörtberg. Implementing a category-theoretic framework for typed abstract syntax. In Andrei Popescu and Steve Zdancewic, editors, *CPP '22: 11th ACM SIGPLAN International Conference on Certified Programs and Proofs, Philadelphia, PA, USA, January 17 - 18, 2022*, pages 307–323. ACM, 2022.

[4] George Janelidze and Gregory Maxwell Kelly. A note on actions of a monoidal category. *Theory and Applications of Categories*, 9(4):61–91, 2001. Online available at http://www.tac.mta.ca/tac/volumes/9/n4/9-04abs.html.

[5] Jonathan Sterling and Carlo Angiuli. Foundations of relative category theory. Online lecture course, available at https://www.jonmsterling.com/math/lectures/categorical-foundations.html, 2022.