Coherence for symmetric monoidal groupoids in HoTT/UF

Stefano Piceghello

University of Bergen, Norway
stefano.piceghello@uib.no

We aim to produce a formalized proof of coherence for symmetric monoidal groupoids in HoTT, using equivalent constructions of a free symmetric monoidal groupoid that highlight different properties: commutativity of coherence diagrams, normal forms of symmetric monoidal expressions, and invariance under symmetries.

Our analysis starts from monoidal structures without symmetries. The proof of a statement for coherence for monoidal categories – namely, "in a free monoidal category, every diagram commutes" – in intensional MLTT was formalized in ALF in [1]; in there, a category consists of a set of objects and a family of Hom-setoids. A similar result can be obtained in HoTT, where we choose instead to employ the built-in higher groupoid structure of types to represent categories with all arrows invertible (i.e., groupoids), using the correspondence between objects and terms, invertible arrows and paths, and commutative diagrams and 2-paths. This set-up is sufficient to express the same statement for coherence, since (a.) all arrows in a free monoidal category are products of instances of the natural isomorphisms defining the monoidal structure, and hence they are invertible; and (b.) coherence is achieved by means of strong monoidal functors. A monoidal groupoid is then a 1-type endowed with a monoidal structure, and coherence can be formulated as follows: a free monoidal 1-type is monoidally equivalent to a monoidal 0-type.

In order to explicitly describe the free monoidal 1-type on a type X of generators, we use higher inductive types (HITs) as in [2]. We define the recursive HIT $F(X)$ with 0-constructors for the objects of the groupoid (a unit, the inclusion of the generators, and a monoidal product), 1-constructors for associativity of the monoidal product and the unit laws, 2-constructors for the coherence diagrams, and a 1-truncation. A normalization of the monoidal expressions in $F(X)$ is then used to show that this type is monoidally equivalent to the type $\mathrm{list}(X)$ of lists over X, with list concatenation as the monoidal product. This is then easily proven to be a 0-type whenever X is.

$$
\begin{aligned}
F(X) ::= {} & e : F(X) \quad | \; g : X \to F(X) \quad | \; \otimes : F(X) \to F(X) \to F(X) \\
& | \; \alpha : (a,\, b,\, c : F(X)) \to (a \otimes b) \otimes c = a \otimes (b \otimes c) \quad | \; \lambda : (b : F(X)) \to e \otimes b = b \\
& | \; \rho : (a : F(X)) \to a \otimes e = a \quad | \; \tau : (a,\, b : F(X)) \to (\rho_a \otimes 1_b) \cdot \alpha_{a,e,b} = (1_a \otimes \lambda_b) \\
& | \; \pi : (a,\, b,\, c,\, d : F(X)) \to (\alpha_{a,b,c} \otimes 1_d) \cdot \alpha_{a,b\otimes c,d} \cdot (1_a \otimes \alpha_{b,c,d}) = \alpha_{a\otimes b,c,d} \cdot \alpha_{a,b,c\otimes d} \\
& | \; \mathrm{trunc} : \mathrm{IsTrunc}\ 1\ F(X)
\end{aligned}
$$

Though conceptually similar to [1], our HoTT-based implementation presents important features of their own. First of all, the elimination principle of $F(X)$ guarantees that this type really represents a free monoidal groupoid, in the precise sense that the construction is left-adjoint to the forgetful functor to types (this one realized by the first projection out of a Σ-type). Secondly, in [1] the normalizing functor from monoidal expressions to lists factors through endomorphisms of $\mathrm{list}(X)$, where associativity and the unit laws hold definitionally. While appropriate for the task, this method does not generalize to other coherence theorems (e.g. when symmetry is part of the structure), so we choose to adopt a more straightforward approach, by mapping the monoidal product directly to list concatenation. Finally, as the coherence morphisms and diagrams rest on identity types, the resulting proof of a monoidal equivalence $F(X) \simeq \mathrm{list}(X)$ is much shorter than the one in [1] (performed by induction on the arrows of the category): all the cases with an analogue in the groupoid structure of identity types – inverses, composition, product of arrows – are redundant in our proof. The trade-off for this approach is its intrinsic specificity to structures with invertible morphisms only.

We then investigate symmetric monoidality. A free symmetric monoidal 1-type $FS(X)$ defined similarly to $F(X)$ can be proven to be equivalent, via a strong symmetric monoidal functor, to another HIT $\mathrm{slist}(X)$ defined with the 0-constructors of lists, 1-constructors for transpositions of two adjacent elements in a list, 2-constructors for the relations between the transpositions, matching those in the presentation of the symmetric groups Σ_n,

$$\Sigma_n := \frac{(a_1, \dots, a_{n-1})}{\begin{array}{c} a_i^2 = 1, \qquad (a_i a_{i+1})^3 = 1, \\ a_i a_j = a_j a_i \;\text{ for } |i-j| \geq 2 \end{array}} \tag{1}$$

and a 1-truncation. In contrast to $\mathrm{list}(X)$, the type $\mathrm{slist}(X)$ (and hence $FS(X)$) is, in general, not a 0-type, as indeed not every diagram in a free symmetric monoidal groupoid commutes.

While $\mathrm{slist}(X)$ is essentially meant to represent a "free permutative 1-type", i.e. a free symmetric monoidal 1-type in which associativity and the unitors are strict, our framework does not actually allow to express strictness of a monoidal structure, so coherence cannot be concluded by the established equivalence alone. A proof of coherence would instead entail showing that the connected components of $\mathrm{slist}(X)$ corresponding to lists with no repetitions are contractible. We believe that this can be attained in the following way: first of all, by exhibiting a symmetric monoidal equivalence

$$\mathrm{slist}(X) \simeq \sum_{n:\mathrm{nat}} \sum_{A:B\Sigma_n} (A \to X), \tag{2}$$

where $B\Sigma_n$ is the classifying space of the symmetric group Σ_n, and then by observing that

$$\sum_{n:\mathrm{nat}} \sum_{A:B\Sigma_n} (A \hookrightarrow X),$$

i.e. the subtype selecting symmetric monoidal expressions with no repetitions, is a 0-type. The equivalence in (2) should follow from the equivalence between the description of $B\Sigma_n$ as

$$B\Sigma :\equiv (n : \mathrm{nat}) \to \sum_{Y:\mathcal{U}} \|Y \simeq \mathrm{Fin}(n)\|_{-1}$$

and the family, indexed by nat, of deloopings of Σ_n. This can be defined as a family of 1-truncated HITs, with a basepoint in the first type in the family, an inclusion of each type into the next one, and a new loop at each type, satisfying the relations defining Σ_n as in (1).

The last equivalence and the one in (2) are, at the present time, yet to be formalized. All other claims have been formalized in Coq using the HoTT library[1].

References

[1] Ilya Beylin and Peter Dybjer. Extracting a proof of coherence for monoidal categories from a proof of normalization for monoids. In *Types for Proofs and Programs*, pages 47–61. Springer Berlin Heidelberg, 1996.

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

[1] https://github.com/HoTT/HoTT