THE BLUEPRINT FOR FORMALIZING GEOMETRIC ALGEBRA IN LEAN

ERIC WIESER, UTENSIL SONG

Introduction. The goal of this document is to provide a detailed account of the formalization of Geometric Algebra (GA) a.k.a. Clifford Algebra [HS84] in the Lean formal proof verification system [dMKA+15] and using its Mathematical Library [The20].

1. Foundations

1.1. **Preliminaries.** This section introduces the algebraic environment of Clifford Algebra, covering vector spaces, groups, algebras, representations, modules, multilinear algebras, quadratic forms, filtrations and graded algebras.

The material in this section should be familiar to the reader, but it is worth reading through it to become familiar with the notation and terminology that is used, as well as their counterparts in Lean, which usually require some additional treatment, both mathematically and technically (probably applicable to other formal proof verification systems).

No details will be given as these are given in standard textbooks, such as TODO.

1.2. **Clifford algebras - definition.**

1.2.1. *Involutions.*

1.3. **Structure of Clifford algebras.**

1.4. **Classifying Clifford algebras.**

1.5. **Representing Clifford algebras.**

1.6. **Spin.**

2. Geometric Algebra

2.1. **Axioms.**

2.2. **Operations and properties.**

3. Concrete algebras - definition

3.1. **CGA.**

3.2. **PGA.**

3.3. **STA.**

Date: August 21, 2022.

4. Applications

4.1. **Geometry.**

References

[dMKA+15] Leonardo de Moura, Soonho Kong, Jeremy Avigad, Floris van Doorn, and Jakob von Raumer, *The Lean Theorem Prover (System Description)*, Automated Deduction - CADE-25 (Amy P. Felty and Aart Middeldorp, eds.), Lecture Notes in Computer Science, vol. 9195, Springer International Publishing, Cham, 2015, pp. 378–388.

[HS84] David Hestenes and Garret Sobczyk, *Clifford algebra to geometric calculus: A unified language for mathematics and physics*, vol. 5, Springer Science & Business Media, 1984.

[The20] The mathlib Community, *The lean mathematical library*, Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs, CPP 2020, Association for Computing Machinery, 2020, pp. 367–381.