Formalization of Ceva

Jun Akita

Waseda University, School of Fundamental Science and Engineering, Japan

HP: `https://c46b3c12.myhomepage-1xu.pages.dev/`

GitHub: `https://github.com/Aj1905`

Abstract

In this paper, we present a formalization of Ceva's theorem (and its converse) in the Lean 4 theorem prover. Working in the general setting of affine spaces over an arbitrary field, we state and prove Ceva's theorem for triangles embedded in an n-dimensional ambient space, without appealing to coordinates or metric notions such as distances or angles.

Our development introduces reusable infrastructure for affine geometry in `mathlib`, including definitions and lemmas about cevians, affine ratios, and concurrency of lines. Along the way, we refactor and extend existing material on barycentric coordinates so that classical synthetic arguments can be expressed in an implementation-friendly form.

We compare several standard textbook proofs (via area ratios, vector methods, and barycentric coordinates) and explain how their key ideas are reflected in the final Lean proof. This case study illustrates how a classical theorem of elementary geometry can be stated in appropriately general form, integrated into a large mathematical library, and used as a stepping stone toward further formalization of plane and affine geometry in interactive theorem provers.

2012 ACM Subject Classification Replace ccsdesc macro with valid one

Keywords and phrases interactive theorem proving, Lean4, Ceva's theorem, Euclidean geometry, formalization

Digital Object Identifier 10.4230/LIPIcs.CVIT.2016.23

Acknowledgements I want to thank ...

1 Introduction

Ceva's theorem is one of the central results of classical Euclidean triangle geometry. Given a triangle ABC and points D, E, and F on the sides BC, CA, and AB respectively, the theorem states that the cevians AD, BE, and CF are concurrent if and only if

$$\frac{BD}{DC} \cdot \frac{CE}{EA} \cdot \frac{AF}{FB} = 1,$$

where the ratios are interpreted as signed segment ratios in the oriented setting. This simple multiplicative condition captures a wide range of concurrency phenomena in triangle geometry and is closely related to other fundamental results such as Menelaus' theorem. Ceva's theorem appears routinely in geometry textbooks and mathematical olympiad problems, and it serves as a gateway to more sophisticated tools such as barycentric coordinates, projective arguments, and trigonometric forms of geometric theorems.

From the perspective of interactive theorem proving, Ceva's theorem is an attractive case study. On the one hand, its statement is elementary and well known, so it provides a concrete target against which one can validate the expressiveness and usability of a geometry library. On the other hand, its proofs come in many different flavors: purely synthetic Euclidean proofs, coordinate-based proofs, barycentric proofs, and trigonometric proofs. Formalizing Ceva's theorem in an interactive theorem prover therefore tests not only the underlying geometric infrastructure, but also the system's ability to support and compare different proof styles within a single coherent framework.

42nd Conference on Very Important Topics (CVIT 2016).
Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:5
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

In this paper we formalize Ceva's theorem in Lean 4, building on the mathematical library `mathlib` and its affine and Euclidean geometry components. Our formalization is not merely a direct encoding of a single textbook proof. Instead, we use Ceva's theorem as a benchmark to design and organize reusable geometry infrastructure that can support a variety of concurrent and collinear configurations in the Euclidean plane and beyond.

Contributions.

The main contributions of this work are as follows.

- We give a flexible formal statement of Ceva's theorem in Lean 4, phrased in terms of abstract real affine (or inner product) spaces rather than a fixed coordinate model of the Euclidean plane. This formulation cleanly separates the geometric content of the theorem from any particular representation of points and lines.
- We develop reusable infrastructure for triangle geometry in `mathlib`: definitions of cevians and concurrency, oriented segment ratios compatible with the existing affine/Euclidean APIs, and auxiliary lemmas about collinearity, parallelism, and ratio manipulations that are useful beyond Ceva's theorem itself.
- We formalize and compare multiple proof styles for Ceva's theorem. In particular, we implement a synthetic proof that closely follows classical Euclidean arguments, and a more algebraic proof that proceeds via coordinates (or barycentric-like representations). We discuss how these styles differ in terms of readability, automation, and reusability, and what this reveals about proof engineering for geometry in Lean.
- We position our development within the broader landscape of formal Euclidean geometry, and we identify parts of our infrastructure that can serve as building blocks for future formalizations, such as Menelaus' theorem, various triangle center constructions, and more advanced projective arguments.

Outline.

The rest of the paper is organized as follows. In Section 2 we recall Ceva's theorem in its classical forms and summarize the relevant parts of Lean 4 and `mathlib`'s geometry library. Section **??** introduces our formal setting for Euclidean geometry, together with the key definitions and helper lemmas for cevians, oriented segment ratios, and concurrency. In Section **??** we present our synthetic Lean proof of Ceva's theorem, mirroring a standard Euclidean argument. Section **??** describes an alternative, more algebraic proof and explains how it is connected to coordinate and barycentric approaches. Section **??** compares the different proof styles from a proof-engineering standpoint and discusses their implications for the design of geometry libraries in interactive theorem provers. Finally, Section 8 concludes and outlines directions for extending our infrastructure to a broader class of geometric theorems.

2 Background

2.1 Classical Ceva's Theorem

Ceva's theorem is a concurrency criterion for cevians in a triangle. Let ABC be a triangle in the Euclidean plane and let D, E, and F be points on the sides BC, CA, and AB respectively, with the understanding that each point may lie on the line determined by the corresponding side (so that external cevians are allowed). The cevians are the lines AD, BE, and CF.

In its most familiar (unsigned) form, Ceva's theorem states that the cevians AD, BE, and CF are concurrent if and only if

$$\frac{BD}{DC} \cdot \frac{CE}{EA} \cdot \frac{AF}{FB} = 1,$$

where the ratios are taken as positive real numbers and one typically assumes that D, E, and F lie on the segments BC, CA, and AB respectively. In this formulation, the theorem is restricted to internal cevians.

A more flexible formulation uses *oriented lengths* (or signed ratios). Each side of the triangle is given an orientation (for instance, $B \to C$, $C \to A$, $A \to B$), and the oriented length $\overrightarrow{XY}$ of a segment from X to Y is taken to be positive if Y lies in the positive direction from X and negative otherwise. The ratios

$$\frac{\overrightarrow{BD}}{\overrightarrow{DC}}, \quad \frac{\overrightarrow{CE}}{\overrightarrow{EA}}, \quad \frac{\overrightarrow{AF}}{\overrightarrow{FB}}$$

are then real numbers, possibly negative. In this oriented setting, Ceva's theorem asserts that AD, BE, and CF are concurrent if and only if

$$\frac{\overrightarrow{BD}}{\overrightarrow{DC}} \cdot \frac{\overrightarrow{CE}}{\overrightarrow{EA}} \cdot \frac{\overrightarrow{AF}}{\overrightarrow{FB}} = 1.$$

This version simultaneously covers internal and external configurations and is more natural from an affine or projective viewpoint.

For the purposes of formalization, it is convenient to phrase the theorem in an abstract affine setting rather than in a concrete coordinate model of the Euclidean plane. We therefore fix a real affine space P modeled on a real inner product space V, which plays the role of Euclidean space in `mathlib`. Given distinct, non-collinear points $A, B, C : P$ and points $D, E, F : P$ lying on the lines BC, CA, and AB respectively, we define the oriented ratios $\frac{BD}{DC}$, $\frac{CE}{EA}$, and $\frac{AF}{FB}$ using the underlying vector space structure and a consistent choice of orientation on each side.

The precise statement that we formalize can be summarized as follows:

> **Ceva's theorem (oriented affine version).** Let P be a real affine space modeled on an inner product space. Let $A, B, C : P$ be non-collinear points, and let $D, E, F : P$ lie on the lines BC, CA, and AB respectively. Then the cevians AD, BE, and CF are concurrent (i.e. there exists a point $X : P$ lying on each of the three cevians) if and only if
>
> $$\frac{BD}{DC} \cdot \frac{CE}{EA} \cdot \frac{AF}{FB} = 1,$$
>
> where the fractions are interpreted as oriented ratios along the corresponding lines.

In Lean, this statement is expressed using the existing notions of affine spaces and lines in `mathlib`, together with a definition of (signed) ratios along a line that is compatible with the affine structure. The rest of the development is devoted to specifying these notions in a reusable way and proving the equivalence between concurrency and the product condition.

2.2 Lean4 and `mathlib` Geometry

Lean 4 is an interactive theorem prover based on dependent type theory. Its core logic is a version of the calculus of inductive constructions: propositions live in the universe `Prop`, types of mathematical objects live in universes `Type u`, and proofs are terms inhabiting these types. The kernel checks all type-correctness and definitional equalities, ensuring that every accepted proof is mechanically verified. On top of this kernel, Lean 4 offers an expressive language for defining structures, inductive types, and typeclasses, together with a tactic framework and an extensible elaboration and automation layer.

The `mathlib` library is a large community-driven collection of formalized mathematics for Lean. It provides a substantial hierarchy of algebraic, analytic, and geometric structures, as well as many classical theorems. Our work builds on the existing `mathlib` infrastructure for affine and Euclidean geometry, which includes:

- *Inner product spaces* and *Euclidean spaces*, formalizing finite-dimensional real inner product spaces with the usual notions of distance and angle.
- *Affine spaces* modeled on normed or inner product spaces, including the general framework for points, vectors, and translations. In particular, Euclidean geometry is developed in terms of real affine spaces modeled on inner product spaces.
- *Affine subspaces*, which represent lines, planes, and higher-dimensional affine subsets. Lines through two points are special cases of affine subspaces.
- *Segments and rays*, defined using affine combinations of points; these support the usual operations on line segments and provide a bridge between synthetic and analytic viewpoints.
- Tools related to *barycentric coordinates* and *simplices*, which allow points to be expressed as affine combinations of vertices of a simplex and are particularly well suited to reasoning about triangles and their cevians.

Our formalization of Ceva's theorem is organized to fit naturally into this existing ecosystem. At a high level, the development is split into two parts:

- A collection of general-purpose geometry lemmas and definitions: oriented ratios along a line, cevians defined in terms of affine subspaces, and notions of concurrency for triples of lines through a triangle.
- The statement and proofs of Ceva's theorem itself, expressed in the abstract affine setting and instantiated in Euclidean spaces as needed.

The code is written in Lean 4 and depends on a recent version of `mathlib`. The project is structured so that the reusable geometric infrastructure (definitions of cevians, concurrency, and oriented ratios) lives in standalone files that do not depend specifically on Ceva's theorem, while the files containing the synthetic and coordinate-style proofs import this infrastructure. This separation is intended to make it easy to reuse our setup in subsequent developments of triangle geometry, such as formalizations of Menelaus' theorem, properties of triangle centers, or projective configurations.

3 Formal Statement in Lean4

4 Proof Strategy

5 Implementation in Lean4

6 Evaluation and Reusability

7 Related Work

8 Conclusion and Future Work

References

A Additional Lean4 Code and Proof Details