Formalization and Automation of Euclidean Geometry

Sana Stojanović, Vesna Pavlović, Predrag Janičić
Faculty of Mathematics, University of Belgrade, Serbia
URL: www.matf.bg.ac.yu/ ~sana, ~vesnap, ~janicic

Workshop on Formal and Automated Theorem Proving and Applications
Belgrade, Serbia, January 31, 2009.

Agenda

- Motivation and our goals
- State of the Art in Formalization of Geometry
- State of the Art in Automation of Geometry
- State of the Art in Formalization and Automation of Geometry
- Our Approach
- Future Work and Conclusions

Motivation

- Formalization of geometry have always been vital for mathematics

- Formal, verifiable geometrical knowledge lead to reliable proofs and have applications in

  - mathematical education

  - applied algorithms of computational geometry

Our Goals

- To build an algorithm that can automatically produce formal and readable proofs within different geometrical axiomatizations

- It should be capable of proving at least simple, foundational theorems that are the current subject of formal proving by mathematicians

State of the Art in Formalization of Geometry

- Formalization of Hilbert's *Grundlagen der Geometrie*:
  - Dehlinger/Dufourd/Schreck within Coq (2000)
  - Fleuriot/Meikle within Isabelle/Isar (2003)
- Formalization of Tarski's geometry:
  - Narboux within Coq (2006)
- Formalization of projective plane geometry:
  - Narboux/Magaud/Schreck within Coq (2008)

State of the Art in Formalization of Geometry (part 2)

- A formal system for Euclids Elements by Jeremy Avigad et.al (2008) — provides a faithful model of the proofs in Euclids Elements (still not formalized)

- All of the above works reveal numerous flaws in proofs in classical geometrical books

- All of the above formalizations were made by hand

State of the Art in Automation of Geometry

- Several efficient algorithms:
  - Gröbner bases method, adapted for geometry (Buchberger, 1965)
  - Wu's method (Wu, 1977)
  - The area and the full angle methods (Chou et.al.1992)
- All of them are algebraic or semi-algebraic, and reduce geometrical reasoning to computations

State of the Art in Formalization and Automation of Geometry

- Several methods have been formalized or are under development:
  - Gröbner bases method (Harrison, Pottier, Chaieb,...)
  - Wu's method (Narboux, Mahboubi, Chaieb,...)
  - The area methods (Narboux)
- They produce formal proofs, but still not traditional, readable proofs.

Our Approach

- We want to build a general system that can:
  - automatically produce traditional, readable proofs
  - export formal proofs to Isabelle/Isar (and later Coq)
  - use different sets of axioms (not just geometrical)
- Such system should automate what others did by hand

Our Approach — Scope

All axioms and conjectures have one of the forms:

- $\forall x_1 \forall x_2 \ldots \forall x_n \exists Y_1 \exists Y_2 \ldots, \exists Y_m$

  $(\phi(x_1, x_2, \ldots, x_n) \Rightarrow \psi(x_1, x_2, \ldots, x_n, Y_1, Y_2, \ldots, Y_m))$

- $\forall x_1 \forall x_2 \ldots \forall x_n \exists Y_1 \exists Y_2 \ldots \exists Y_m (\phi(x_1, x_2, \ldots, x_n \Rightarrow$

  $\psi_1(x_1, x_2, \ldots, x_n, Y_1, Y_2, \ldots, Y_m) \vee \ldots \vee \psi_k(x_1, x_2, \ldots, x_n, Y_1, Y_2, \ldots, Y_m))$

- $\forall x_1 \forall x_2 \ldots \forall x_n (\phi(x_1, x_2, \ldots x_n) \Rightarrow \psi(x_1, x_2, \ldots, x_n))$

- $\exists Y_1 \exists Y_2 \ldots \exists Y_m (\psi(Y_1, Y_2, \ldots Y_m))$

Our Approach — Scope (part 2)

- Fortunately, geometrical axioms typically meet the above constraint
- Expected to cover introductory level theorems, usually simple, but numerous
- Expected to be useful as an assistant for numerous different geometries, conjectures, and theorem provers
- Can be used for other theories, not only geometries

Our Approach — Algorithm

- Uses simple forward chaining
- Derives new conclusions in iterations (by putting restrictions on introduced symbols of constants)
- Sound, and complete and terminating for theorems that admit proofs involving only formulae of the above form

Our Approach — Implementation

- Currently under development, in C++
- Generic, so can be used for different theories
- Currently, it has over 2000 lines of code
- First tests made for Hilbert-style axioms

Our Approach — Simple Example

- Axiom III3: If $AB \cong CD$ and $AB \cong PQ$ then $CD \cong PQ$.

- Added: congruence(E, F, C, D)
  from congruence(C, D, E, F) and congruence(C, D, C, D)
  using III3

- from ‘congruence C D E F‘ and ‘congruence C D C D‘
  have ”congruence E F C D”
  using III3 [of ”C” ”D” ”E” ”F” ”C” ”D”]
  by auto

Our Approach — Simple Example

- Assume: between(B, C, A)
  Added: between(A, C, B)
  from between(B, C, A)
  using II2
  Contradiction
  from between(A, C, B) and not_between(A, C, B)

- { assume "between B C A"
  from 'between B C A'
  have "between A C B"
  using II2 by auto
  from 'between A C B' and '~between A C B'
  have False by auto }

Future Work

- Finish and polish the implementation
- Improve efficiency by some guiding
- Add export to different theorem provers
- Explore different geometries and also other theories

Conclusions

- In current approaches to dealing with geometry, either formal proofs are not produces, or proofs are not readable, or they are not generated automatically

- Our system should be able to automatically produce formal and readable, traditional geometrical proofs

- It is still subject of development

- It should be useful for other theories too