Geometry Theorem Proving in the Frame of *Theorema* Project

Judit Robu
Babes-Bolyai University Cluj, Romania
e-mail *robu@cs.ubbcluj.ro*
RISC-Linz, Austria
e-mail *jrobu@risc.uni-linz.ac.at*

Abstract

During the last few years many algorithms for automated geometry theorem proving appeared and there exist some very good implementations of these algorithms. We present an implementation of some of these algorithms in the frame of the *Theorema* system. *Theorema* isa system that provides a uniform frame for theorem proving in all areas of mathematics, which is programmed in *Mathematica*. *Theorema* is being developed at the RISC Institute by the *Theorema* Group under the direction of Bruno Buchberger. Several examples of geometry proof generated by our implementation are provided. Besides the implementation of known proving methods we also present two new approaches: systematic exploration of geometric configurations and a new method for proving nontrivial geometry theorems involving order relations.

1 Introduction

In the course of an ongoing PhD thesis we are implementing various geometric theorem proving methods in the frame of the *Theorema* system [3], [2] which is programmed in *Mathematica* and, hence, can be installed on all machines on which *Mathematica* is available. The functionalities described in this paper are available as *Theorema* constructs and, typically, take formulae in *Theorema* notation as input.

Theory exploration is an important new paradigm in computer-supported formal mathematics, which was introduced in [1]. In this paper, we consider theory exploration for the special case of geometric theories. In this case, theory exploration has a well-defined meaning and a certain degree of "completeness" of explorations can be achieved. Given a set of geometric constructions, there are many properties of the resulting geometric elements (points, lines, segments, triangles, etc), that can be proved. There are some properties that are well-known but there might also be some hidden properties that are not known so far. A good way for discovering new theorems is generating all the properties of a given configuration and then analyzing which of these properties might be unknown so far. By *systematic exploration* of a geometric configuration we mean finding, step by step, all the properties that result from the given constructions. For the moment we are concerned only of some *basic properties*: collinear points, parallel and perpendicular lines, segments with proportional length and triangles with proportional area.

Using traditional geometry theorem proving methods no theorems that involve order relation or contain inequalities in the conclusion can be proved. Collins' Cylindrical Algebraic Decomposition method [8] can deal with such problems but it is too slow for the majority of the geometry theorems. Combining the area method with Collins' CAD we developed a proof method with which we proved many nontrivial theorems whose algebraic transcription involves, in addition to polynomial equalities, also polynomial inequalities.

2 Problem Input

The geometric configuration can be given using two kinds of *Theorema* constructs:

If a geometric configuration is given that should be explored or used as hypothesis in computing an expression we use

`KnowledgeBase[`*`label`*`, any[`*`variable-list`*`], `*`constr-conj`*`]`

where

– *label* is the identifier of the knowledge base,

– *variable-list* enumerates all the points and other variables that appear in the constructions

– *constr-conj* is a formal conjunction of geometric constructions admitted by the provers and other methods. If a geometry proposition is given that should be proved we use

`Proposition[`*`label`*`, any[`*`variable-list`*`], `*`constr-conj`*`⇒`*`concl-conj`*`]`

where

– *label* is the identifier of the proposition,

– *variable-list* enumerates all the points and other variables that appear in the constructions

– *constr-conj* is a formal conjunction of geometric constructions admitted by the provers, which represent the hypothesis of the proposition

– *concl-conj* is a conclusion or a formal conjunction of conclusions admitted by the provers, which represents the conclusion of the proposition. Examples:

$$
\begin{aligned}
&KnowledgeBase[example, any[A, B, C, D, A1, A2, B1, r],\\
&\quad point[A, B, C] \wedge inter[D, pline[A, B, C], pline[C, A, B]] \wedge\\
&\qquad divpoint[A1, C, D, r/(r-1)] \wedge divpoint[B1, D, A, r/(r-1)] \wedge\\
&\qquad inter[A2, line[A, A1], line[B, B1]]]\\
&Proposition[Pappus,\ any[A, B, A1, B1, C, C1, P, Q, S],\\
&\quad point[A, B, A1, B1] \wedge pon[C, line[A, B]] \wedge pon[C1, line[A1, B1]] \wedge\\
&\qquad\quad inter[P, line[A, B1], line[A1, B]] \wedge inter[Q, line[A, C1], line[A1, C]] \wedge\\
&\qquad\quad inter[S, line[B, C1], line[B1, C]] \Rightarrow\\
&\qquad collinear[P, Q, S]]\\
&KnowledgeBase[Pappus\_g, any[A, B, A1, B1, C, C1],\\
&\quad \{\{A, \{10, 120\}\}, \{B, \{100, 150\}\}, \{A1, \{25, 60\}\}, \{B1, \{120, 50\}\},\\
&\qquad \{C, \{260, 200\}\}, \{C1, \{250, 45\}\}\}]
\end{aligned}
$$

3 Graphic Representation

As a first step we visualize the geometry statement to be proved using the graphical possibilities offered by *Mathematica*. Unfortunately the actual frontend does not admit interactive graphics. The graphic representation can use random coordinates for the free points or the user may specify some coordinates. A numerical check of the statement is also performed for the actual coordinates of the points.

Example

$$
\begin{aligned}
&Simplify[Proposition[Pappus], by->GraphicSimplifier,\\
&\quad using \rightarrow KnowledgeBase[Pappus\_g]]
\end{aligned}
$$

4 Algebraic Methods

4.1 Algebraic Form

To be able to use the algebraic methods, the problem has to be transformed from its external form into polynomial form. We have to separate the coordinates into independent and dependent variables and to find an appropriate coordinate system. Our implementation tries, by using an appropriate coordinate system, to simplify these polynomials as much as possible. The internal algebraic form can be visualized.

Example

$$Simplify[Proposition[Pappus], by \to AlgebraicSimplifier,$$
$$SimplifierOptions- > \{MyAxis \to False\}]$$

4.2 Wu's Method

Wu's method, its theoretical background and some improvements are presented in detail in [14], [7], [13], and the literature cited there. Since in *Mathematica* there is no built-in implementation of the characteristic sets, as a first step this implementation had to be done. For the case of more than one expressions in the conclusion the characteristic set is computed only once. We also use an iterated algorithm to compute the characteristic set. This way for many problems the time consumed for this operation can be reduced drastically.

Example

$$Prove[Proposition[Pappus], by \to GeometryProver,$$
$$ProverOptions \to \{Method- > Wu, MyAxis \to False, Iterate- > True\}]$$

4.3 Gröbner Bases Method

Another algebraic method used in geometry theorem proving is the Gröbner bases method, as presented in [10] and [9] (for a theoretical presentation of Gröbner bases see [4]). As *Mathematica* offers a very good implementation of Gröbner bases, we use this function in our implementation.

Example

$$Prove[Proposition[Pappus], by \to GeometryProver,$$
$$ProverOptions \to \{Method- > GroebnerProver, MyAxis \to True,$$
$$ReverseVars \to False, Refutation \to True\}]$$

5 Area Method

5.1 Proving

The area method is presented in detail in [5] and [6]. In the same way as for the other methods the prover can be called with the *Theorema* Prove function.

Example

$$Prove[Proposition[Pappus], by \to GeometryProver,$$
$$ProverOptions \to \{Method \to Area\}]$$

5.2 Computing

The area method is very convenient for computing expressions involving geometric quantities relative to a specified construction. The construction has to be given using the *Theorema* construct `KnowledgeBase`.

Example

$$
\begin{aligned}
&Compute[trarea[A,B,A2]/qarea[A,B,C,D]\,,\\
&\quad by- > AreaCompute, using- > KnowledgeBase[example],\\
&\quad EvaluatorOptions \to \{VerifyNdg \to False, ShowSteps \to True\}]
\end{aligned}
$$

5.3 Exploring

Using the area method we can systematically explore given geometric configurations [11]. Starting from a knowledge base that specifies some constructions many theorems concerning parallel and perpendicular lines, segments with proportional lengths and triangles with proportional areas are obtained. As a next step further constructions can be specified in a new knowledge base and the exploration may continue without recomputing any of the results already obtained.

Example

$$
\begin{aligned}
&KnowledgeBase[Trapezoid1, any[A,B,C,D,M,N],\\
&\quad triangle[A,B,C] \wedge pon[D, pline[C,A,B]] \wedge line[A,D] \wedge\\
&\qquad midpoint[M,A,D] \wedge midpoint[N,B,C] \wedge line[M,N]]
\end{aligned}
$$

$$kb1 = Explore[KnowledgeBase[Trapezoid1], by \to AreaExplore, producing - > Tr1];$$

$$
\begin{aligned}
&KnowledgeBase[Trapezoid2, any[A,B,C,D,M,N,X,P,Q],\\
&\quad inter[X, line[M,N], line[B,D]] \wedge inter[P, line[A,C], line[B,D]] \wedge\\
&\qquad inter[Q, line[B,C], pline[P,A,B]] \wedge inter[S, line[P,Q], line[A,D]]]
\end{aligned}
$$

$$
\begin{aligned}
&kb2 = Explore[KnowledgeBase[Trapezoid2], by \to AreaExplore,\\
&\qquad using \to kb1, producing - > Tr2];
\end{aligned}
$$

By the help of the `ShowKnowledgeBase` function the results of all the intermediate steps can be visualized.

Example

$$ShowKnowledgeBase[kb2, AfterConstruction \to 5]$$

6 Theorems Involving Order Relations

6.1 Cylindrical Algebraic Decomposition

The methods presented are very efficient for problems whose algebraic formulation involves only polynomial equalities but neither of them can deal with polynomial inequalities. Thus, no geometrical theorem that involves the notion of betweenness can be treated.

The CAD method introduced in [8] , improving earlier work of [12] , can be used to prove geometry theorems that involve order relations (that is, the algebraic form consists of polynomial equalities and inequalities) as has been observed in ks86 and other papers. In our implementation we use the

CylindricalAlgebraicDecomposition function of the Experimental package of *Mathematica*. Unfortunately only very simple geometric statements can be proved this way, as the method is very time consuming.

6.2 AreaCad Method

To overcome the slowness of Collins' CAD method for complicated polynomials, we developed a method that combines the area and the CAD method. First we compute the expressions involved in inequalities using the area method. This way we obtain a new problem equivalent to the original one that is expressed only in terms of the independent points of the original constructions. Applying the CAD method to this new problem we can obtain the result in reasonable time even for quite complicated problems.

Example

$$\begin{aligned}
&Proposition[Cad_44_6,\ any[A, B, O, C, P, M, N, O1, A1, A2, B1, B2, C1, C2],\\
&\quad incircle[O, A, B, C, P] \wedge midpoint[M, A, B] \wedge midpoint[N, A, C] \wedge\\
&\qquad inter[O1, tline[M, A, B], tline[N, A, C]] \wedge\\
&\qquad inter[A1, line[A, O], line[B, C]] \wedge inter[A2, line[A, O], circle[O1, A]] \wedge\\
&\qquad inter[B1, line[B, O], line[A, C]] \wedge inter[B2, line[B, O], circle[O1, B]] \wedge\\
&\qquad inter[C1, line[C, O], line[A, B]] \wedge inter[C2, line[C, O], circle[O1, C]]\\
&\quad \Rightarrow segratio[A, A1, A, A2] + segratio[B, B1, B, B2] + segratio[C, C1, C, C2] \leq 9/4]
\end{aligned}$$

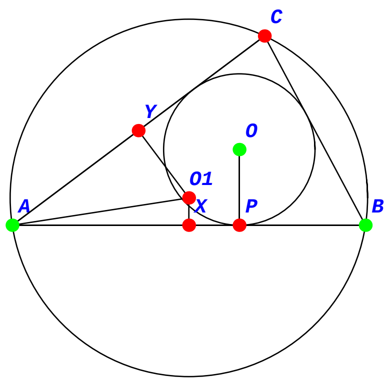
$$\begin{aligned}
&Prove[Proposition[Cad_44_6], by \to GeometryProver,\\
&\quad ProverOptions \to \{Method \to AreaCAD\}]
\end{aligned}$$

7 Example

Let r the radius of the incircle and R the radius of the circumcircle of a triangle. Then $r \leq 1/2R$

$$\begin{aligned}
&Proposition[Cad_141_13,\ any[O, A, B, C, P, X, Y, O1],\\
&\quad incircle[O, A, B, C, P] \wedge midpoint[X, A, B] \wedge\\
&\qquad midpoint[Y, A, C] \wedge inter[O1, tline[X, A, B], tline[Y, A, C]]\\
&\quad \Rightarrow 4 * seglengthsq[O, P] <= seglengthsq[O1, A]]
\end{aligned}$$

$$Simplify[Proposition[Cad\_141\_13], by -> GraphicSimplifier]$$



$Prove[Proposition[Cad\_141\_13], by \to GeometryProver,$
$ProverOptions \to \{Method \to AreaCAD\}]$

============ *Theorema* Notebook ============

Prove:

$(Proposition\,(Cad\_141\_13))$

$$\underset{O,A,B,C,X,Y,O1}{\forall} (incircle[O,A,B,C,P] \wedge midpoint[X,A,B] \wedge midpoint[Y,A,C] \wedge inter[O1, tline[X,A,B], tline[Y,A,C]] \wedge circle[O1,A] \Rightarrow 4\,seglengthsq[O,P] \leq seglengthsq[O1,A])$$

with no assumptions.

As the proposition contains inequalities, we have to use the CAD method. We shall use the area method first to obtain a simpler input for CAD.

First we have to transform the problem in internal form for the AreaCAD. We have to prove that constructions

{A,B,O} free points
$\alpha_A B \perp AO$ and $\alpha_A \in AO$ (ndg. cond. $A \neq O$)
$\alpha 1_A B \parallel B\alpha_A$ and $\frac{\overline{B\alpha 1_A}}{\overline{B\alpha_A}} = 2$ (ndg. cond. $B \neq \alpha_A$)
$\alpha_B A \perp BO$ and $\alpha_B \in BO$ (ndg. cond. $B \neq O$)
$\alpha 1_B A \parallel A\alpha_B$ and $\frac{\overline{A\alpha 1_B}}{\overline{A\alpha_B}} = 2$ (ndg. cond. $A \neq \alpha_B$)
$C = A\alpha 1_A \cap B\alpha 1_B$ (ndg. cond. $A \neq \alpha 1_A, B \neq \alpha 1_B, A\alpha 1_A \nparallel B\alpha 1_B$)
$PO \perp AB$ and $P \in AB$ (ndg. cond. $A \neq B$)
$\alpha_C O \perp AC$ and $\alpha_C \in AC$ (ndg. cond. $A \neq C$)
$XA \parallel AB$ and $\frac{\overline{AX}}{\overline{AB}} = \frac{1}{2}$ (ndg. cond. $A \neq B$)
$YA \parallel AC$ and $\frac{\overline{AY}}{\overline{AC}} = \frac{1}{2}$ (ndg. cond. $A \neq C$)
$Y\gamma_{O1} \perp YA$ and $\frac{\overline{Y\gamma_{O1}}}{\overline{YA}} = r_2 3$ (ndg. cond. $Y \neq A$)
$X\beta_{O1} \perp XA$ and $\frac{\overline{X\beta_{O1}}}{\overline{XA}} = r_2 2$ (ndg. cond. $X \neq A$)
$O1 = X\beta_{O1} \cap Y\gamma_{O1}$ (ndg. cond. $X \neq \alpha 1_A \beta_{O1}, Y \neq \gamma_{O1}, \beta_{O1} X \nparallel \gamma_{O1} Y$)

with additional constraints

$\overline{AB}^2 - \overline{AP}^2 > 0$
$\overline{AB}^2 - \overline{BP}^2 > 0$
$\overline{AC}^2 - \overline{A\alpha_C}^2 > 0$
$\overline{AC}^2 - \overline{C\alpha_C}^2 > 0$

imply that $-\overline{AO1}^2 + 4\overline{PO}^2 \leq 0$

Step 1

Now we try to obtain a simpler form of the constraints and of the conclusion using the area method.

As the area method supports just a few basic constructions, we had to rewrite the initial problem using only these constructions.

We used β_P (and $\gamma_P\ \alpha_P\ \alpha 1_P$) to denote the auxiliary points needed to express the construction for a point P using only the constructions admitted by the area method

The area method consists of eliminating constructed points from the lefthand-side of the conclusion expression, in the reverse order of their construction. Elimination of a point means using certain rewrite rules depending on the geometric quantity the point appears in, and the construction by which the point was introduced.

Expression 1

Using constructions

// the same constructions are enumerated as above //

we have to compute $\overline{AB}^2 - \overline{AP}^2$

First we have to verify the non-degenerate for each construction, then to prove our statement.
// check of the non-degenerate conditions using the area method //

All the non-degenerate conditions were verified. Now we compute expression $\overline{AB}^2 - \overline{AP}^2$

Eliminating point O1:

Eliminating point P:

All the constructed points have been eliminated, so we have to rewrite the expression in area coordinates of the free points:

We try to simplify further this expression using the possible relations between areas and lengths of segments

we obtain

$$\bullet P_{\{A,B,C\}} - \bullet P_{\{A,P,A\}} = \frac{1}{2 \bullet L_{\{A,B\}}{}^2}(3 \bullet L_{\{A,B\}}{}^4 - 2 \bullet L_{\{A,B\}}{}^2(\bullet L_{\{A,O\}}{}^2 - \bullet L_{\{B,O\}}{}^2) - (\bullet L_{\{A,O\}}{}^2 - \bullet L_{\{B,O\}}{}^2)^2$$

Expression 2

// Expression 2 is computed in the same way //

Expression 3

// Expression 3 is computed in the same way //

Expression 4

// Expression 4 is computed in the same way //

Expression 5

// Expression 5 is computed in the same way //

Step 2

Our initial problem becomes

$$\underset{O,A,B}{\forall}\left(\frac{1}{2\bullet L_{\{A,B\}}{}^2}\,(3\bullet L_{\{A,B\}}{}^4 - 2\bullet L_{\{A,B\}}{}^2(\bullet L_{\{A,O\}}{}^2 - \bullet L_{\{B,O\}}{}^2) - (\bullet L_{\{A,O\}}{}^2 - \bullet L_{\{B,O\}}{}^2)^2 > 0 \wedge \right.$$
$$\left. \ldots\ldots \wedge \ldots\ldots \wedge \ldots\ldots \Rightarrow \ldots\ldots\right)$$

We rewrite the lengths of segments and areas of triangles using the coordinates of the points. Choosing a cartesian coordinate system with the x-axes {A,O} and performing substitution:

$\{x_A \to 0,\ y_A \to 0,\ x_O \to 1,\ y_O \to 0\}$

the expression becomes:

// expression in x_B and y_B //

Applying the CAD algorithm to this expression we obtain that the proposition is true.

============ End of *Theorema* Notebook ============

References

[1] Buchberger B. Theory Exploration with Theorema. In *Proc. SYNASC 2000 (2nd International Workshop on Symbolic and Numeric Algorithms for Scientific Computing, Timisoara, Romania, Oct. 4-6, 2000)*, pages 1-16, 2000.

[2] Buchberger, B., Dupre, C., Jebelean, T., Kriftner, F., Nakagawa, K., Vasaru, D., Windsteiger, W. The Theorema Project: A Progress Report. *Symbolic Computation and Automated Reasoning*, M. Kerber and M. Kohlhase (eds.), 98-113, 2001.

[3] Buchberger, B., T. Jebelean, T.,Kriftner, F., Marin, M., Tomuta E., Vasaru, D. An Overview on the Theorema project, In *Proc. ISSAC'97 (International Symposium on Symbolic and Algebraic Computation, Maui, Hawaii, July 21-23, 1997)*,W. Kuechlin (ed.), pages 384-391, ACM Press, 1997.

[4] Buchberger, B.. Gröbner bases: An algorithmic method in polynomial ideal theory. In *Multidimensional Systems Theory*, 6, (N.K. Bose ed.) D. Reidel Publ. Comp., 1985.

[5] Chou, S.-C., Gao, X.-S., Zhang, J.-Z.. Automated production of traditional proofs for theorems in Euclidean geometry. In *Proc. of Eigth IEEE Symp. on Logic in Comp. Sci.*, pages 48-56, IEEE Computer Society Press, 1993.

[6] Chou, S.-C., Gao, X.-S., Zhang, J.-Z. *Machine Proofs in Geometry: Automated Production of Readable Proofs for Geometry Theorems*, volume 6, series Applied Mathematics. World Scientific, Singapore, 1994.

[7] Chou, S.-C. *Mechanical geometry theorem proving*, Reidel, Dordrecht Boston, 1988.

[8] Collins, G. E. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. *Springer's LNCS*, 33:134-165, 1975.

[9] Kapur, D.. Using Gröbner bases to reason about geometry problems. J. Symb. Comput., 2:399-408, 1986.

[10] Kutzler, B., Stifter, S. On the application of Buchberger's algorithm to automated geometry theorem proving. *J. Symb. Comput.*, 2:389-397, 1986.

[11] Robu, J. Systematic exploration of geometric configurations using Theorema based on Mathematica. In *Proc. SYNASC '01 (3rd International Workshop on Symbolic and Numeric Algorithms for Scientific Computing, Timisoara, Romania, Oct. 2-5, 2001)*, to appear.

[12] Tarski, A. *A decision method for elementary algebra and geometry*, Univ. of California Press, Berkeley Los Angeles, 2nd edition, 1951.

[13] Wang, D. Algebraic factoring and geometry theorem proving. *Springer's LNCS*, 814:386-400, 1994. 2.p

[14] Wu, W.-t. Basic principles of mechanical theorem proving in elementary geometries. *J. Automat. Reason.*, 2:221-252, 1986