On proving and discovering theorems by computer

Pavel Pech[*]
pech@pf.jcu.cz
Faculty of Education
University of South Bohemia České Budějovice
371 15
Czech Republic

Abstract

Proofs of mathematics theorems belong to the most difficult part of mathematics. For this reason proofs are often omitted at schools. But without proofs there is no mathematics. Despite this, proving or at least verification of statements should be done in teaching mathematics of all school categories. It seems that new technologies such as CAS and DGS could help remedy this state. In the last four decades new methods of proving, deriving and discovering theorems by computers were invented. At the same time various dynamic geometry software was developed.

In this paper, basic methods of computer supported discovery and proving are shown. Both DGS and CAS will be used. With DGS we describe a problem and verify some related conjectures. With CAS we do rigorous proofs. The theory of automated geometry theorem proving is demonstrated with examples.

1 Introduction

Problem solving belongs to one of main goals in teaching mathematics, to which computers yield ideal possibilities. Let us look at Descartes' view of proving.
R. Descartes' general principle of problem solving [18], [25]:

- Reduce any kind of the problem to a mathematical problem,

- Reduce any kind of a mathematical problem to a problem of algebra,

- Reduce any problem of algebra to the solution of a single equation.

Descartes' general principle is still valid. Most problems can really be translated into the system of algebraic equations (usually non-linear) and then this system is solved by ingenious mathematical algorithms with the help of computers.

[*]This work was partially supported by EU grant ECP-2006-EDU-410016.

In the paper we will prove mathematical theorems using the theory of automated theorem proving which is based on results of commutative algebra developed in the last forty years [7], [19]. These proving methods would not be possible without powerful computers and appropriate mathematical software. We will use both Dynamic Geometry Systems (DGS) and Computer Algebra Systems (CAS). First we give a brief description about the role of DGS and CAS regarding proving theorems.

Dynamic geometry systems can be used in proving geometry theorems mainly due to following features:

- Dynamic description of problems,

- Verification of statements,

- Stating conjectures,

- Visualization of proofs without words [13], [14].

Since DGS are based on numerical computations, in DGS mostly we are not able to prove theorems. That is why we need CAS which are based on symbolic computations. In CAS we can use particularly the following properties:

- Elimination of variables,

- Solving algebraic equations,

- Proving theorems,

- Discovering theorems.

Elimination of variables is a basic technique which enables both solving algebraic equations and proving and discovering theorems.
In teaching mathematics at various types of schools, we need both CAS and DGS. Whereas in DGS we can demonstrate and verify theorems, in CAS, we are able to do exact proofs.

2 Proving theorems

We will be concerned with two proof categories which can be done by computer:

- Verification in DGS,

- Computer (automated) proofs.

Let us briefly characterize them.

Verification in DGS: In the past students verified a given statement in several concrete situations using a ruler and circle. This is what we can call a classical verification.
Nowadays DGS enable to verify a statement in *infinitely* many situations. We call it a verification in DGS. Since, the dragging function in DGS could be considered a continuous movement, if a statement is valid by dragging all the possible free parameters, then it can be proved that the statement is actually true with very high probability. This gives students confidence that the fact is indeed true and what we need is a logical proof. We should realize that verification in DGS is *not* a proof! Despite of it

verification is an important tool even for experts since we can state conjectures. In elementary schools verification in DGS can replace the exact mathematical proof and motivate students.

Computer proof: By computer we can prove most of the problems which can be proved classically. We can also prove problems which are difficult or even impossible to prove by a classical approach (the first was Four colours problem which was solved in 1976).
New questions arise — is a computer proof a real proof? Are we able to check it?
By computer we can solve even such problems which we can not construct by ruler and circle (non–Euclidean constructions).

Proving theorems does not belong to favorite activities at schools. To attract students we should keep the following rules:

- Persuade students that proofs are necessary,
- Prove such statements we are doubting about,
- Show statements which seem to be true but in fact are not valid,
- Visualize a proof if possible,
- Show nice proofs,
- Use proofs without words [13], [14] — the best.

To show that we should not believe any statements which are not exactly proven let us look at the following example.

Ancient Chinese prime number test:

Natural number $n > 2$ *is prime* $\Leftrightarrow n \mid (2^{n-1} - 1)$.

Let us verify it!

$$\begin{array}{cccc} 3 & \text{is prime} & \Leftrightarrow & 3 \mid (2^{3-1}-1) & true \\ 4 & \text{is not prime} & \Leftrightarrow & 4 \nmid (2^{4-1}-1) & true \\ 5 & \text{is prime} & \Leftrightarrow & 5 \mid (2^{5-1}-1) & true \\ 6 & \text{is not prime} & \Leftrightarrow & 6 \nmid (2^{6-1}-1) & true \\ 7 & \text{is prime} & \Leftrightarrow & 7 \mid (2^{7-1}-1) & true \\ 8 & \text{is not prime} & \Leftrightarrow & 8 \nmid (2^{8-1}-1) & true \\ 9 & \text{is not prime} & \Leftrightarrow & 9 \nmid (2^{9-1}-1) & true \\ & & \cdots & & \end{array}$$

but the statement *does not hold!!!*
Namely, for $n = 341$ which is *a compound* number since $341 = 11 \cdot 31$, we get $341 \mid (2^{340} - 1)$ and the statement is *not true.*
There are another such numbers $561, 645, 1105, 1387, 1729, \ldots$ which are called 2-pseudoprimes.

3 Automated theorem proving

In this section we describe some computer proving methods which belong to the theory of automated geometry theorem proving [7]. By this theory we can *prove* many theorems from geometry. This theory also enables to *discover* new theorems. Under discovering we mean searching for additional conditions which are necessary to add to the given assumptions so that the statement becomes true. Searching for geometric loci of points we put among the simplest form of discovering. Hundreds of unknown theorems have been discovered in the last thirty years by this method.

There are three basic methods of automated geometry theorem proving:

- Gröbner basis (GB) method [2], [19],

- Wu–Ritt (WR) method [22],

- Quantifier elimination (QE) [21], [5].

In automated theorem proving we suppose that a statement is of the form

$$\forall x \in \mathbb{C}: \ \mathbf{H} \ \Rightarrow \ \mathbf{C},$$

where $\mathbf{H}$ is a set of hypotheses

$$h_1(x) = 0, h_2(x) = 0, \dots, h_r(x) = 0,$$

and $\mathbf{C}$ is a conclusion

$$c(x) = 0,$$

where $\mathbb{C}$ is the field of complex numbers and $h_1(x), h_2(x), \dots, h_r(x), c(x)$ are polynomials with coefficients from the field of rational numbers $\mathbb{Q}$.
To prove the statement above, we are to show that

$$c^k(x) = c_1(x)h_1(x) + c_2(x)h_2(x) + \cdots + c_r(x)h_r(x) \tag{1}$$

— *Gröbner basis* approach [2], or

$$d(x)c(x) = c_1(x)h_1(x) + c_2(x)h_2(x) + .. + c_r(x)h_r(x) \tag{2}$$

— *Wu–Ritt* approach ($h_i(x)$ are in a triangular form) [22],

where k is a non-negative integer and $c_1(x), \dots, c_r(x)$, $d(x)$ are polynomials.
If a conclusion polynomial c can be expressed by (1) as

$$c^k = c_1h_1 + c_2h_2 + \cdots + c_rh_r$$

then, since $h_1 = h_2 = \cdots = h_r = 0$, we get $c = 0$.
Similarly, if by (2)

$$dc = c_1h_1 + c_2h_2 + ... + c_rh_r$$

and $d \neq 0$, then from $h_1 = h_2 = \cdots = h_r = 0$ the conclusion $c = 0$ follows.

WR method was developed by Chinese mathematician Wu W.-t. before GB method which was developed by B. Buchberger. GB and WR methods are related exactly to the same class of geometric theorems and they give equivalent results. The strength of WR method is that it is quicker by proving a statement. The reason is that computation of a triangular set of given polynomials requires less

effort than computing a Gröbner basis of the ideal generated by these polynomials. However Gröbner bases contain more information about the given ideal.
WR packages are not publicly available in such an extent as GB packages. We can find them for instance in Epsilon Library [23] and freely download at
`http://www-calfor.lip6.fr/~wang/epsilon/`. On the other hand GB packages are implemented in almost well known computer algebra systems including Maple, Mathematica, CoCoA, Singular, Reduce, MuPAD, Axiom, Macsyma. Perhaps that is why GB method is used more frequently than WR method.
The disadvantage of both GB and WR methods is that in the real case we cannot in general disprove statements. The reason is that the theory of automated theorem proving which is behind GB and WR methods is by Hilbert Nullstellensatz related to algebraic closed fields, for instance to the field of complex numbers. But by proving geometric statements we usually work with real numbers. If we prove such a statement, it is valid in the field of complex numbers, although we are working with reals. But it could happen that a statement which is not valid in complex numbers is valid in real numbers.

The reasoning is usually not so simple. We often need to rule out *degeneracy conditions,* like e.g. two vertices of a triangle coincide, the radius of a circle equals zero, etc.
Their algebraic expression is in the form of *inequations*

$$d_1(x) \neq 0, d_2(x) \neq 0, \ldots, d_s(x) \neq 0.$$

Then algebraic form of a statement has the form

$$\forall x \in \mathbb{C} : [(h_1 = 0, \ldots, h_r = 0, d_1 \neq 0, \ldots, d_s \neq 0) \Rightarrow (c = 0)].$$

Searching for degeneracy conditions and their *geometric interpretation* is a difficult problem which has not been completely solved to date.

Quantifier Elimination by *Cylindrical Algebraic decomposition* (CAD) [5] is, unlike GB and WR methods, working in real space. At the beginning there was a discovery of a Polish mathematician and logician A. Tarski that the theory $(R, +, \ldots, 0, 1, <)$ is complete. It implies that in a so called elementary theory of real closed fields it is possible to carry out the elimination of quantifiers. Collins CAD approach is based on a decomposition of a parametric space into cells. Due to the fact that we are working with real numbers we can solve even inequalities by this method. There are several programs using cell-decomposition, for instance QEPCAD [6], REDLOG [8], Bottema [26]. CAD method is also implemented in the program Mathematica. The weakness of CAD method is that the computational complexity increases very quickly with the number of parameters. Its use is limited to date.

4 Examples

In this part we demonstrate various computer methods of proving and discovering with examples. For more examples see [16].

4.1 Simson–Wallace theorem

For demonstrating computer supported theory proving, we consider the well-known Simson–Wallace theorem as our first example.

Let ABC be a triangle and P a point of the circumcircle of ABC. Then the feet of perpendiculars K, L, M from P onto the sides of ABC lie on a straight line.

Verification in DGS: Working with students, first we verify the statement in DGS, where the verifi-

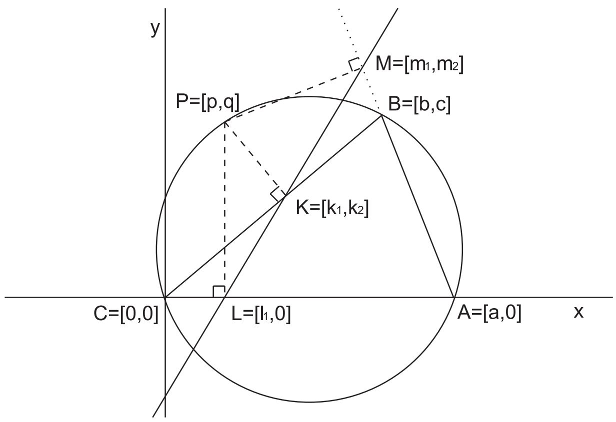

Figure 1: Simson–Wallace theorem

cation is done in Cabri II Plus or in Geogebra.
Consider a straight line KL and ask whether the point M is a member of the line KL, Fig. 1. The answer is *This point lies on the object* even if we interactively change the form of a triangle ABC. Hence the statement is confirmed in infinitely many cases. But we did not show that the statement is true in *all* cases (perhaps with some exceptions). We should realize that verification is *not* a proof.

After verification we usually prove the theorem classically since the classical proof enables a deeper insight into the problem.[1] We will omit it, see [16], so that we could concentrate on computer proof.

Computer proof (GB approach): Let us choose a Cartesian system of coordinates so that $A = [a, 0]$, $B = [b, c]$, $C = [0, 0]$, $P = [p, q]$, $K = [k_1, k_2]$, $L = [l_1, 0]$, $M = [m_1, m_2]$, Fig. 1.
The hypotheses are as follows:

$PL \perp AC \Leftrightarrow h_1 : p - l_1 = 0,$
$K \in BC \Leftrightarrow h_2 : ck_1 - bk_2 = 0,$
$PK \perp BC \Leftrightarrow h_3 : (p - k_1)b + (q - k_2)c = 0,$
$M \in AB \Leftrightarrow h_4 : ac + bm_2 - cm_1 - am_2 = 0,$
$PM \perp AB \Leftrightarrow h_5 : (p - m_1)(b - a) + (q - m_2)c = 0,$

[1]Classical proof of the Simson–Wallace theorem can be generated automatically as well, see e.g. [4].

P lies on the circumcircle of $ABC \Leftrightarrow$

$h_6 : -acp + cp^2 + abq - b^2q - c^2q + cq^2 = 0,$

see [16]. The conclusion c has the form:

K, L, M are collinear $\Leftrightarrow c : l_1m_2 + k_2m_1 - k_1m_2 - k_2l_1 = 0.$

We need to find out whether the conclusion polynomial c can be expressed in the form (1) which is equivalent to the fact that c belongs to the radical of the ideal $(h_1, h_2, \ldots, h_6)$, or equivalently, whether 1 is an element of the ideal $I = (h_1, h_2, \ldots, h_6, ct - 1)$, where t is a slack variable [10]. Program CoCoA [2] returns

```
Use R::=Q[abcpqk[1..2]l[1..2]m[1..2]t];
I:=Ideal(p-l[1],ck[1]-bk[2],(p-k[1])b+(q-k[2])c,ac+bm[2]-cm[1]
-am[2],(p-m[1])(b-a)+(q-m[2])c,-acp+cp^2+abq-b^2q-c^2q+cq^2,
(l[1]m[2]+k[2]m[1]-k[1]m[2]-k[2]l[1])t-1); NF(1,I);
```

the answer 1 and the statement is not generally true.
Let us look for non-degeneracy conditions. Elimination of dependent variables $p, q, k_1, k_2, l_1, m_1, m_2$ and t in the ideal I

```
Use R::=Q[abcpqk[1..2]l[1..2]m[1..2]t];
I:=Ideal(p-l[1],ck[1]-bk[2],(p-k[1])b+(q-k[2])c,ac+bm[2]-cm[1]
-am[2],(p-m[1])(b-a)+(q-m[2])c,-acp+cp^2+abq-b^2q-c^2q+cq^2,
(l[1]m[2]+k[2]m[1]-k[1]m[2]-k[2]l[1])t-1); Elim(p..t,I);
```

gives the condition $(b^2+c^2)((a-b)^2+c^2) = 0$, which means that for the vertices of a triangle $B = C$ or $A = B$. We rule out these cases assuming that $B \neq C$ and $B \neq A$. We will add the polynomial $(b^2 + c^2)((a - b)^2 + c^2)v - 1$, where v is another slack variable, to the ideal I and the procedure now repeats. Denoting $J = I \cup \{(b^2 + c^2)((a - b)^2 + c^2)v - 1\}$ we get

```
Use R::=Q[abcpqk[1..2]l[1..2]m[1..2]vt];
J:=Ideal(p-l[1],ck[1]-bk[2],(p-k[1])b+(q-k[2])c,ac+bm[2]-cm[1]
-am[2],(p-m[1])(b-a)+(q-m[2])c,-acp+cp^2+abq-b^2q-c^2q+cq^2,
(b^2+c^2)((a-b)^2+c^2)v-1,(l[1]m[2]+k[2]m[1]-k[1]m[2]-k[2]l[1])
t-1); NF(1,J);
```

the answer `NF=0` which means that the conclusion polynomial c is in the form (1). The Simson–Wallace theorem is proved.

Now let us show Wu–Ritt approach on the same example.

Computer proof (WR approach): With the same notation as above we enter in Epsilon [3] (which is working under Maple)

```
with(epsilon);
> Simson:=Theorem({p-l[1],c*k[1]-b*k[2],(p-k[1])*b+(q-k[2])*c,
```

[2]program CoCoA is freely distributed at `http://cocoa.dima.unige.it`

[3]program Epsilon is freely distributed at `http://www-calfor.lip6.fr/~wang/epsilon/`

```
a*c+b*m[2]-c*m[1]-a*m[2],(p-m[1])*(b-a)+(q-m[2])*c,-a*c*p+c*p^2
+a*b*q-b^2*q-c^2*q+c*q^2},{l[1]*m[2]+k[2]*m[1]-k[1]*m[2]-k[2]*l[1]},
[a,b,c,p,q,k[1],k[2],l[1],l[2],m[1],m[2]]): Prove(Simson);
```

with the answer *The theorem is true under the following subsidiary conditions:*

$$\begin{aligned} b &\neq 0, && (3)\\ b^2 - 2ba + a^2 + c^2 &\neq 0, && (4)\\ c &\neq 0, && (5)\\ -b + a &\neq 0, && (6)\\ b^2 + c^2 &\neq 0. && (7) \end{aligned}$$

Comparison with GB approach shows that now we have three more conditions (3), (5) and (6), whereas conditions (4) and (7) are the same. When the theorem is true in degenerate cases we can verify using the same method. We find out that instead of five conditions it suffices to have two conditions (4), (7) to confirm the GB result.

4.1.1 Generalization of Gergonne

In this part we will show a generalization of Simson–Wallace theorem which is ascribed to J. D. Gergonne [3]. To formulate it, we will use discovery approach by computer. We will solve the following problem:

Let K, L, M be the feet of perpendiculars dropped from a point P to the sides BC, CA, AB of a triangle ABC respectively. We look for points P such that a triangle KLM has the fixed area s.

This problem is a generalization of the previous one since for zero area s of KLM, that is, for the points K, L, M being collinear, the locus of points P is the circumcircle of ABC.

Solution (discovery): To solve the problem we use the same notation as in the last problem. Adopt a Cartesian coordinate system so that $A = [a, 0]$, $B = [b, c]$, $C = [0, 0]$, $P = [p, q]$, $K = [k_1, k_2]$, $L = [l_1, 0]$, $M = [m_1, m_2]$, Fig. 1. Suppose that the hypotheses $h_1, h_2, \dots, h_5$, which are the same as in the previous case, hold.

For the area s of a triangle KLM we have

area of $KLM = s \Leftrightarrow h_7 : l_1 m_2 + k_2 m_1 - k_1 m_2 - k_2 l_1 - 2s = 0$,

since

$$s = \frac{1}{2} \begin{vmatrix} k_1 & k_2 & 1 \\ l_1 & 0 & 1 \\ m_1 & m_2 & 1 \end{vmatrix} . \tag{8}$$

Now the problem is more complex. Unlike the previous task we do not know the locus of points P — we have to *discover* it. Consider the ideal I which contains polynomials $h_1, h_2, \dots, h_5$ and the condition h_7 of fixed area. In this ideal we eliminate all variables besides a, b, c, p, q, s. We get

```
Use R::=Q[abcpqk[1..2]l[1..2]m[1..2]s];
I:=Ideal(p-l[1],ck[1]-bk[2],(p-k[1])b+(q-k[2])c,ac+bm[2]-cm[1]
```

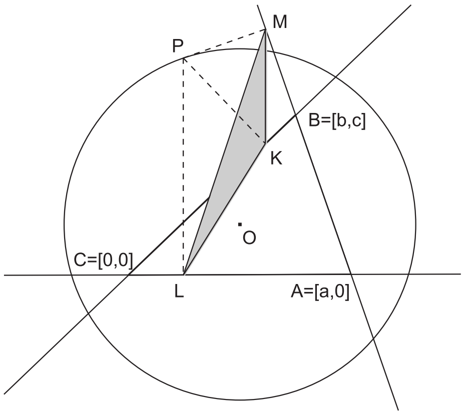

Figure 2: Generalization of Gergonne — triangle KLM has the fixed area

```
-am[2],(p-m[1])(b-a)+(q-m[2])c,l[1]m[2]+k[2]m[1]-k[1]m[2]-k[2]
l[1]-2s); Elim(k[1]..m[2],I);
```

the equation of the circle centered at $O = [q/2, (b^2 - ab + c^2)/(2c)]$ and radius

$$r = \sqrt{(b^2 + c^2)((a - b)^2 + c^2)(ac + 8s)/(4ac^3)} \tag{9}$$

which is concentric with the circumcircle of ABC, Fig. 2.
We found that the condition for the triangle KLM having fixed area s is, that a point P lies on the circle. Similarly we prove a converse statement. We can state the following Gergonne's generalization of Simson–Wallace theorem:

The feet of perpendiculars from a point P onto the sides of a triangle ABC form a triangle of the constant area iff P lies on a circle which is concentric with the circumcircle of ABC.

4.1.2 Simson–Wallace generalization on a tetrahedron

In the following part we will show a generalization of Simson–Wallace theorem into space which has been done by computer.

Let K, L, M, N be the feet of perpendiculars dropped from a point P onto the faces BCD, ACD, ABD, ABC of a tetrahedron $ABCD$. What is a locus of points P such that the volume of $KLMN$ equals the constant s?

Solution (discovery): Choose a Cartesian system of coordinates so that $A = [0, 0, 0]$, $B = [a, 0, 0]$, $C = [b, c, 0]$, $D = [d, e, f]$, $K = [k_1, k_2, k_3]$, $L = [l_1, l_2, l_3]$, $M = [m_1, m_2, m_3]$, $N = [n_1, n_2, n_3]$, $P = [p, q, r]$. Using elimination, in a similar way as in the previous Gergonne's generalization, we get the cubic equation

$$F(s) := ac^2 f^3 G + s \cdot Q = 0,$$

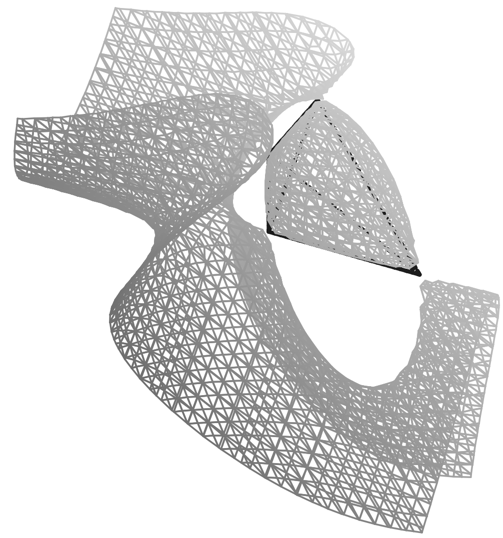

Figure 3: Cubic surface which is associated with a tetrahedron with $s = 0$.

where

$G = bf^2q^3(b-a)+fr^3(abe-acd+cd^2-b^2e-c^2e+ce^2)+c^2f^2p^2q+cfp^2r(e^2-ce+f^2)+cf^2q^2p(a-2b)+fq^2r(abe-acd+cd^2-b^2e+cf^2)+cf^2r^2p(a-2d)+f^2r^2q(b^2-ab+c^2-2ce)+2cefpqr(b-d)+abcf^2q^2+r^2(abce^2-ac^2de+c^2d^2e+acde^2-2bcde^2-abe^3+b^2e^3+acdf^2-abef^2+b^2ef^2+c^2ef^2)-ac^2f^2pq+acfpr(ce-e^2-f^2)+fqr(ac^2d-2abce-c^2d^2+2bcde-b^2e^2+abe^2+abf^2-b^2f^2-c^2f^2)$

and

$Q = -6(e^2+f^2)((cd-be)^2+b^2f^2+c^2f^2)(a^2c^2-2ac^2d+c^2d^2-2a^2ce+2abce+2acde-2bcde+a^2e^2-2abe^2+b^2e^2+a^2f^2-2abf^2+b^2f^2+c^2f^2)$.

We can state a generalization of Simson–Wallace theorem in space [16]:

Let $KLMN$ be orthogonal projections of an arbitrary point P consecutively on the faces BCD, ACD, ABD, ABC of a tetrahedron $ABCD$. Then the points P such that the tetrahedron $KLMN$ has constant volume s belong to the surface $F(s) = 0$.

For $s = 0$, i.e., if K, L, M, N are complanar, and $a = 1$, $b = 0$, $c = 1$, $d = 0$, $e = 0$, $f = 1$, we get a cubic surface, Fig. 3

$$p^2q + pq^2 + p^2r + q^2r + pr^2 + qr^2 - pq - pr - qr = 0.$$

This surface has many interesting properties, see [16].

Next figure shows a cubic surface associated with a regular tetrahedron for $s = 10\sqrt{2}/729$, Fig. 4. The surface has the equation

$$6\sqrt{6}x^2y + 6\sqrt{3}x^2z - 2\sqrt{6}y^3 + 6\sqrt{3}y^2z - 4\sqrt{3}z^3 + 9\sqrt{2}x^2 + 9\sqrt{2}y^2 + 9\sqrt{2}z^2 - 7\sqrt{2} = 0 .$$

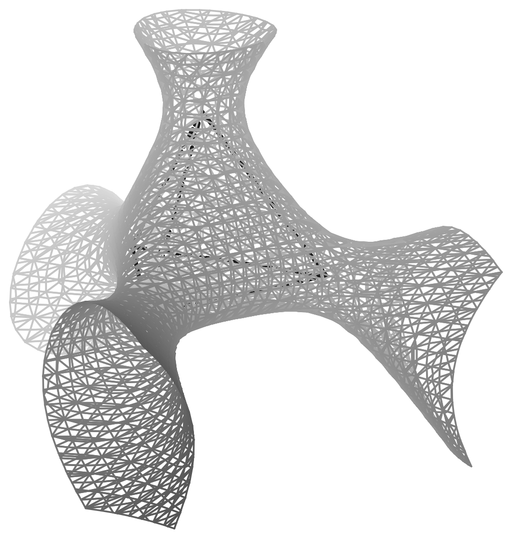

Figure 4: Cubic surface associated with a regular tetrahedron with $s \neq 0$.

4.2 Pascal theorem

WR method is quicker than GB method by proving theorems from elementary geometry. Let us demonstrate it on the following example which is known as the theorem of Pascal:

Let $ABCDEF$ be a cyclic hexagon and let $X = AB \cap DE, Y = BC \cap EF, Z = CD \cap FA$ be the intersections of opposite sides of a hexagon. Then X, Y, Z are collinear.

Verification in DGS: First we draw the Fig. 5 in DGS and ask whether the point Z lies on the line XY. The answer is *This point lies on the object*.

Computer proof (WR method): Denote the coordinates of the vertices of a hexagon as $A = [0, 0]$, $B = [b_1, b_2]$, $C = [c_1, c_2]$, $D = [d_1, d_2]$, $E = [e_1, e_2]$, $F = [f_1, f_2]$ and let the circumcenter $S = [r, 0]$, where r is the radius, Fig. 5. First we express conditions for vertices B, C, D, E, F being on the circle:

$|BS| = r \Leftrightarrow h_1 : (b_1 - r)^2 + b_2^2 - r^2 = 0,$

$|CS| = r \Leftrightarrow h_2 : (c_1 - r)^2 + c_2^2 - r^2 = 0,$

$|DS| = r \Leftrightarrow h_3 : (d_1 - r)^2 + d_2^2 - r^2 = 0,$

$|ES| = r \Leftrightarrow h_4 : (e_1 - r)^2 + e_2^2 - r^2 = 0,$

$|FS| = r \Leftrightarrow h_5 : (f_1 - r)^2 + f_2^2 - r^2 = 0.$

Further we describe points X, Y, Z:

$X \in AB \Leftrightarrow h_6 : x_1 b_2 - x_2 b_1 = 0,$

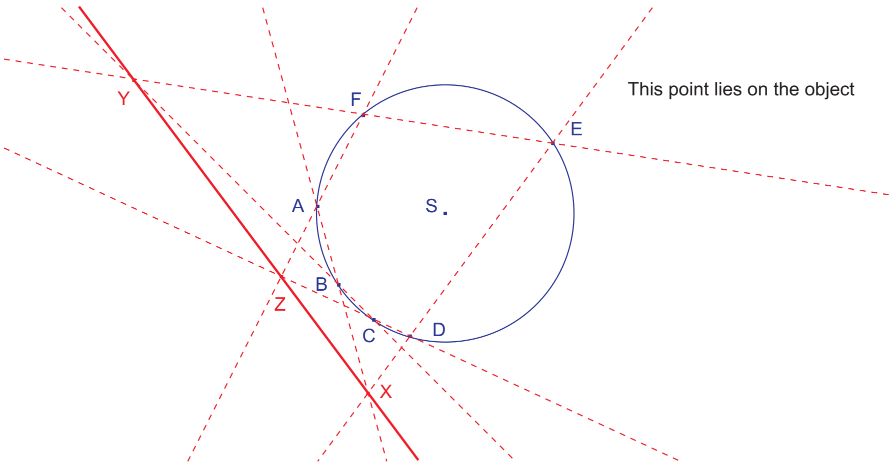

Figure 5: Pascal theorem—points X, X, Z are collinear

$X \in DE \Leftrightarrow h_7 : x_1d_1 + d_1e_2 + x_2e_1 - d_2e_1 - x_1e_2 - x_2d_1 = 0,$

$Y \in BC \Leftrightarrow h_8 : y_1e_2 + e_1f_2 + y_2f_1 - e_2f_1 - y_1f_2 - y_2e_1 = 0,$

$Y \in EF \Leftrightarrow h_9 : y_1b_2 + b_1c_2 + y_2c_1 - b_2c_1 - y_1c_2 - y_2b_1 = 0,$

$Z \in CD \Leftrightarrow h_{10} : z_2f_1 - z_1f_2 = 0,$

$Z \in FA \Leftrightarrow h_{11} : z_1c_2 + c_1d_2 + z_2d_1 - c_2d_1 - z_1d_2 - z_2c_1 = 0.$

The conclusion polynomial c has the form

X, Y, Z are collinear $\Leftrightarrow c : x_1y_2 + y_1z_2 + x_2z_1 - y_2z_1 - x_1z_2 - x_2y_1 = 0.$

In Epsilon we enter

```
with(epsilon);
Pascal:=Theorem({(b[1]-r)^2+b[2]^2-r^2,(c[1]-r)^2+c[2]^2-r^2,
(d[1]-r)^2+d[2]^2-r^2,(e[1]-r)^2+e[2]^2-r^2,(f[1]-r)^2+f[2]^2-r^2,
x[1]*b[2]-x[2]*b[1],x[1]*d[2]+d[1]*e[2]+x[2]*e[1]-d[2]*e[1]-x[1]*
e[2]-x[2]*d[1],y[1]*e[2]+e[1]*f[2]+y[2]*f[1]-e[2]*f[1]-y[1]*f[2]-
y[2]*e[1],y[1]*b[2]+b[1]*c[2]+y[2]*c[1]-b[2]*c[1]-y[1]*c[2]-y[2]*
b[1],z[2]*f[1]-z[1]*f[2],z[1]*c[2]+c[1]*d[2]+z[2]*d[1]-c[2]*d[1]-
z[1]*d[2]-z[2]*c[1]},{x[1]*y[2]+y[1]*z[2]+x[2]*z[1]-y[2]*z[1]-x[1]
*z[2]-x[2]*y[1]},
[b[1],b[2],c[1],c[2],d[1],d[2],e[1],e[2],f[1],f[2],x[1],x[2],
y[1],y[2],z[1],z[2],r]): Prove(Pascal);
```

and in 0.1 second get *The theorem is true under the following subsidiary conditions:*

$b_1d_2 - b_1e_2 + b_2e_1 - b_2d_1 \neq 0,$

$-c_1e_2 + c_1f_2 + b_1e_2 - b_1f_2 + b_2f_1 - b_2e_1 - c_2f_1 + c_2e_1 \neq 0.$

$-f_2d_1 + c_1f_2 - c_2f_1 + f_1d_2 \neq 0,$

$b_1 \neq 0, -c_1 + b_1 \neq 0, -d_1 + c_1 \neq 0.$

The first condition is equivalent to

$$\begin{vmatrix} b_1, & b_2 \\ d_1 - e_1, & d_2 - e_2 \end{vmatrix} \neq 0 \tag{10}$$

which means that $AB \nparallel DE$. Similarly, next two conditions give $BC \nparallel EF$ and $CD \nparallel FA$. The condition $b_1 \neq 0$ follows from $h_1 = 0$ and (10). Remaining two conditions $-c_1 + b_1 \neq 0, -d_1 + c_1 \neq 0$ are also redundant as we can directly verify by the same method.

The use of GB approach on the Pascal theorem fails. The major problem is searching for subsidiary (non-degeneracy) conditions. If we add the first three non-degeneracy conditions above to the hypotheses ideal, then we obtain `NF=0` in 4.2 seconds.

4.3 Neuberg–Pedoe inequality

Although both GB and WR methods are working with equality-type statements we are able to use them to prove statements containing inequalities as well. Let us see the following inequality (11) which is known as the Neuberg–Pedoe inequality [12].

Given a triangle ABC with side lengths a, b, c and the area P and a triangle KLM with side lengths k, l, m and the area Q. Prove that then

$$k^2(-a^2 + b^2 + c^2) + l^2(a^2 - b^2 + c^2) + m^2(a^2 + b^2 - c^2) \ \geq \ 16\, PQ. \tag{11}$$

When the equality is attained?

Computer proof (GB approach): Let $A = [x, y]$, $B = [0, 0]$, $C = [a, 0]$, $K = [u, v]$, $L = [0, 0]$, $M = [k, 0]$, Fig 6. We express the side lengths a, b, c, k, l, m and areas P, Q in algebraic equations:

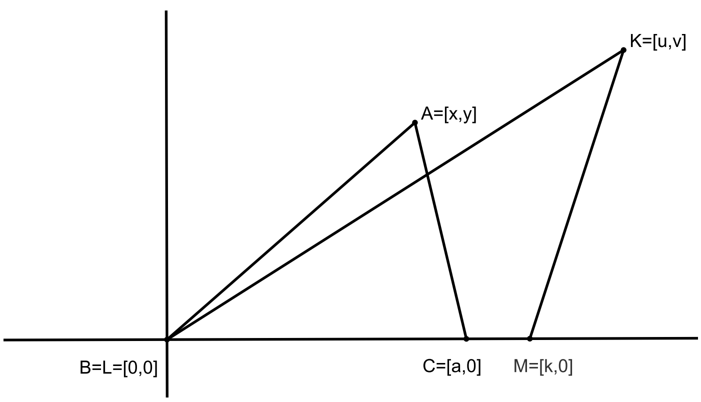

Figure 6: Neuberg-Pedoe inequality - computer proof

$b = |CA| \Leftrightarrow h_1 : (x-a)^2 + y^2 - b^2 = 0,$

$c = |AB| \Leftrightarrow h_2 : x^2 + y^2 - c^2 = 0,$

$l = |MK| \Leftrightarrow h_3 : (u-k)^2 + v^2 - l^2 = 0,$

$m = |KL| \Leftrightarrow h_4 : u^2 + v^2 - m^2 = 0,$

$P =$ area $ABC \Leftrightarrow h_5 : 2P - ay = 0,$

$Q =$ area $KLM \Leftrightarrow h_6 : 2Q - kv = 0.$

Denote the difference of the left side minus the right side of (11) by t. Then

$h_7 : k^2(-a^2 + b^2 + c^2) + l^2(a^2 - b^2 + c^2) + m^2(a^2 + b^2 - c^2) - 16PQ - t = 0.$

We are to show that $t \geq 0$. We will execute two basic steps:

1) We express the variable t in terms of *independent* variables x, y, a, u, v, k in the ideal $I = (h_1, h_2, \ldots, h_7)$.

2) We write t in such a form from which its non-negativity follows.

In the ideal $I = (h_1, h_2, \ldots, h_7)$ we eliminate *dependent* variables b, c, l, m, p, q. In CoCoA we get

```
Use R::=Q[xyuvakbclmpqt]; I:=Ideal((x-a)^2+y^2-b^2,x^2+y^2-c^2,
(u-k)^2+v^2-l^2,u^2+v^2-m^2,2p-ay,2q-kv,
k^2(-a^2+b^2+c^2)+l^2(a^2-b^2+c^2)+ m^2(a^2+b^2-c^2)-16pq-t);
Elim(b..q,I);
```

the polynomial which leads to the equation

$$t = 2u^2a^2 + 2v^2a^2 - 4xuak - 4yvak + 2x^2k^2 + 2y^2k^2$$

which is equivalent to

$$t = 2(xk - ua)^2 + 2(yk - va)^2.$$

We expressed the left side t of Neuberg–Pedoe inequality as the sum of squares, hence $t \geq 0$. The inequality (11) is proved.
The equality is attained iff $xk - ua = 0$ and $yk - va = 0$, which means that triangles ABC and KLM are similar.

Remark 1:
1) We expressed t as the sum of squares of polynomials by hand — without computer.

2) Expression of a non-negative polynomial as the sum of squares is difficult. In addition in some cases a non-negative polynomial *cannot* be expressed as the sum of polynomials [20].

3) This issue is connected with the 17th Hilbert problem which was presented at the International Congress of Mathematicians in Paris in 1900 [20].

Now we will prove the Neuberg–Pedoe inequality (11) using quantifier elimination by cell-decomposition method — the method which is based on the Collins CAD. We will use the program Bottema, which was developed by Chinese mathematician Lu Yang [26]. By the program Bottema we are able to prove inequality-type theorems whose hypotheses and thesis are inequalities in rational functions or radicals. The program is especially efficient for geometric inequalities in a triangle. After a translation of a geometric inequality into a required algebraic form the program proves the inequality on the basis of quantifier elimination based on decomposition of the parametric space into finite number of cells. Choosing a test point in every cell, we only need to check the inequality in these test points. If the inequality holds we obtain the answer *inequality holds*; otherwise we get *The inequality does not hold* with a counter-example. The program Bottema is working under Maple.

Computer proof (QE approach): To prove (11) by the program Bottema we translate the inequality using the relations $h_1, h_2, \ldots, h_6$. Typing

```
read`bottema`;
yprove(k^2*(-a^2+(x-a)^2+y^2+x^2+y^2)+((u-k)^2+v^2)*(a^2-(x-a)^2
-y^2+x^2+y^2)+(u^2+v^2)*(a^2+(x-a)^2+y^2-x^2-y^2)-4*a*y*k*v>=0);
```

we obtain the answer *The inequality holds.*
We do not get any information when the equality is attained.

Remark 2:
If KLM is equilateral then $k^2 = 4Q/\sqrt{3}$ and (11) transforms into the form (Weitzenböck inequality [24])

$$a^2 + b^2 + c^2 \ \geq \ 4\sqrt{3}\, P,$$

where equality occurs iff the triangle is equilateral.

It is equivalent to

$$\frac{a^2\sqrt{3}}{4} + \frac{b^2\sqrt{3}}{4} + \frac{c^2\sqrt{3}}{4} \ \geq \ 3\, P.$$

In the Fig. 7 we can see a graphical demonstration of Weitzenböck inequality, in the style of proofs without words [1], [13], [14].

4.4 Non-elementary constructions

The following example represents a non-elementary construction which is solved both in a computer way and classically.

Given four lines a, b, c, d in the plane, construct a square $KLMN$ such that $K \in a$, $L \in b$, $M \in c$, $N \in d$.

Solution by computer: Let us choose a Cartesian coordinate system so that the vertices K, L, M, N of a square have coordinates $K = [k_1, k_2]$, $L = [l_1, l_2]$, $M = [m_1, m_2]$, $N = [n_1, n_2]$, Fig 8, and

$a : a_1x + a_2y + a_3 = 0$, $b : b_1x + b_2y + b_3 = 0$, $c : c_1x + c_2y + c_3 = 0$, $d : d_1x + d_2y + d_3 = 0$.

Then

$K \in a \Leftrightarrow h_1 : a_1k_1 + a_2k_2 + a_3 = 0,$

$L \in b \Leftrightarrow h_2 : b_1l_1 + b_2l_2 + b_3 = 0,$

$M \in c \Leftrightarrow h_3 : c_1m_1 + c_2m_2 + c_3 = 0,$

$N \in d \Leftrightarrow h_4 : d_1n_1 + d_2n_2 + d_3 = 0.$

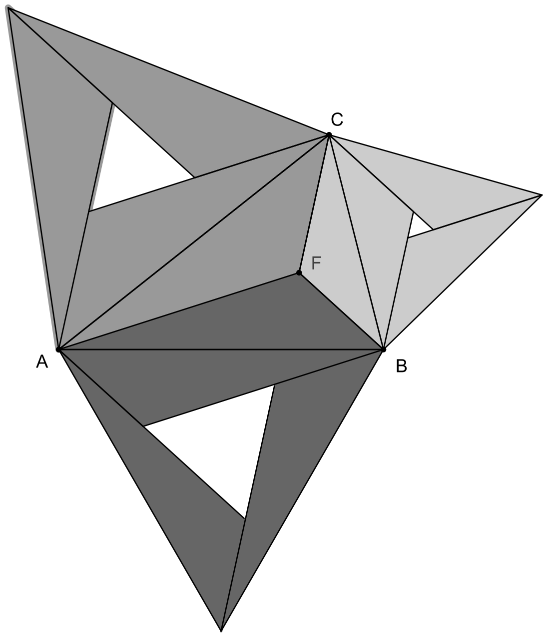

Figure 7: Graphical proof of Weitzenböck inequality

To ensure that $KLMN$ is a square, first we rotate vectors $L-K$, $K-N$ by 90° in a positive sense to get vectors $N-K$, $M-N$ respectively. Then

$h_5 : -(l_2-k_2)-(n_1-k_1)=0,$

$h_6 : l_1-k_1-(n_2-k_2)=0,$

$h_7 : -(k_2-n_2)-(m_1-n_1)=0,$

$h_8 : k_1-n_1-(m_2-n_2)=0,$

We get the system of 8 linear equations $h_1=0, h_2=0, \ldots, h_8=0$ with 8 unknowns k_1, k_2, l_1, l_2, m_1, m_2, n_1, n_2. There is no loss of generality if we put $a_1=0$, $a_2=1$, $a_3=0$, $c_3=0$.

In the ideal $I=(h_1,h_2,\ldots,h_8)$ we eliminate dependent variables $k_2,\ldots,n_2$ and get

$k_1=(-b_3c_1d_1-b_3c_2d_1+b_3c_1d_2-b_3c_2d_2-b_1c_1d_3-b_2c_1d_3+b_1c_2d_3-b_2c_2d_3)/(b_1c_1d_1+b_2c_1d_1+b_2c_2d_1-b_1c_1d_2-b_2c_1d_2+b_1c_2d_2)$.

Similarly we find the remaining unknowns.[4] Now we can draw the resulting square in DGS. Notice that the square $KLMN$ is positively oriented.
Rotation of vectors $L-K$, $K-N$ by 90° in a negative sense leads to the second solution.

Classical solution: The solution is based on one theorem from equiform kinematics [11]. It says that if three points have straight trajectories in an equiform motion, then all points have straight

[4]We could also solve the system $h_1=0, h_2=0, \ldots, h_8=0$ by the Cramer's rule. Then k_1 is expressed as the quotient of two determinants which is in accordance with above result.

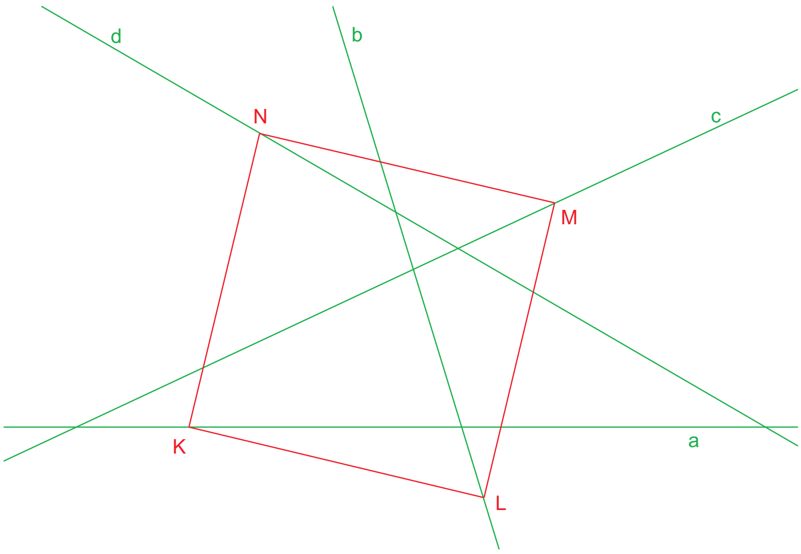

Figure 8: Square $KLMN$ with vertices on lines a, b, c, d - computer proof

trajectories.
By this theorem it suffices to construct two arbitrary squares X', Y', U', V' and X'', Y'', U'', V'' with only *three* vertices X', Y', U' and X'', Y'', U'' on given lines a, b, c. Then the remaining vertices V', V'' determine the line p which is a trajectory of a vertex N, Fig. 9.

5 Conclusion

Proving techniques mentioned above are taught at the University of South Bohemia at initial teacher training in the subject Geometric seminar. Some parts are also taught at in-service teacher training. This seminar is obligatory, offered at the 4th year of study, two hours a week, 3 credits, in English. Seminar work is required.
After a discussion students solve a given problem (mostly from http://www.cut-the-knot.org/geometry.shtml). Seminar work consists of the following items:

- Introduction into the problem,

- Description of a problem in DGS (Cabri, Geogebra,...),

- Verification in DGS,

- Classical proof,

- Automated (computer) proof.

Similar computer techniques could be also used in another areas of mathematics especially in analysis. There are powerful methods for searching limits of rational functions "just from the definition of a limit" [9] or indefinite integrals. We have efficient methods for summing series' [17], we can factor

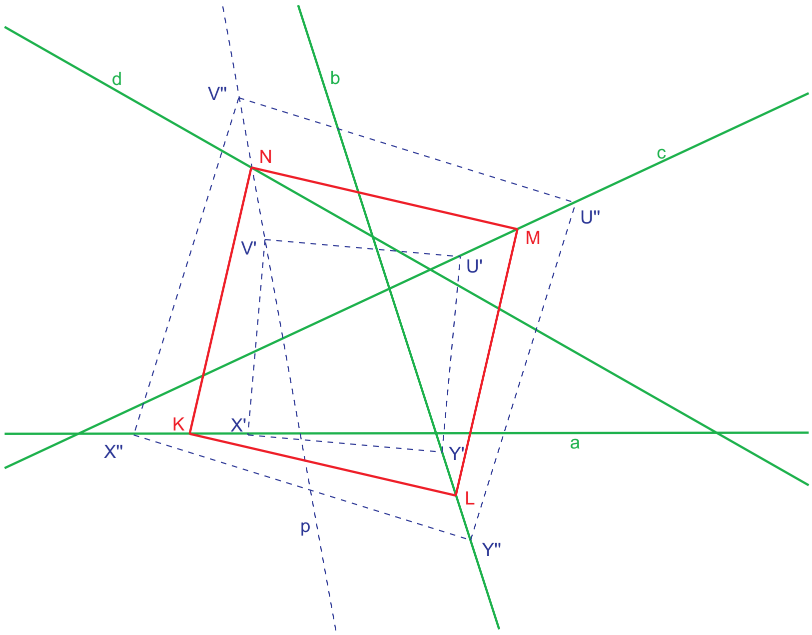

Figure 9: Square $KLMN$ with vertices on lines a, b, c, d - classical solution

polynomials, there is a sos method for decomposition of polynomials into the sum of squares, though its use is limited by the number of parameters [15], etc.
We should realize that behind these methods efficient algorithms of computer algebra are hidden. It is a question, how these methods could be introduced into initial teacher training including understanding of main principles of given algorithms.

Acknowledgements: The author wishes to thank the referees for their valuable suggestions.

References

[1] http://www.cut-the-knot.org/geometry.shtml

[2] Buchberger, B.: Gröbner bases: an algorithmic method in polynomial ideal theory. In: *Multidimensional Systems Theory* (Bose, N.– K., ed.), pp. 184–232. Reidel, Dordrecht (1985).

[3] Chou, S. C.: *Mechanical Geometry Theorem Proving*. D. Reidel Publishing Company, Dordrecht 1987.

[4] Chou, S. C., Gao, X. S., Zhang, J. Z.: A Deductive Database Approach to Automated Geometry Theorem Proving and Discovering. *J. Automated Reasoning* **25**/(3) (2000), 219–246

[5] Collins, G. E.: Quantifier elimination for the elementary theory of real closed fields by cylindrical algebraic decomposition. *Lecture Notes In Computer Science*, vol. 33, pp. 134–183. Springer-Verlag, Berlin (1975).

[6] Collins, G. E., Hong, H.: Partial cylindrical algebraic decomposition for quantifier elimination. *J. Symbolic Computation* **12** (1991), 299–328.

[7] Cox, D., Little, J., O'Shea, D.: *Ideals, Varieties, and Algorithms*. Second Edition, Springer 1997.

[8] Dolzman, A., Sturm, T.: REDLOG: Computer algebra meets computer logic. *ACM SIGSAM Bulletin*, **31**/(2) (1997), 2–9.

[9] Hora, J., Pech, P.: On One Unusual Method of Computation of Limits of Rational Functions in the Program Mathematica. *Int. Journal of Technology in Mathematics Education* **12** (2005), 161–164 .

[10] Kapur, D.: A Refutational Approach to Geometry Theorem Proving. *Artificial Intelligence Journal* **37** (1988), 61–93.

[11] Karger, A.: Classical Geometry and Computers. *Journal for Geometry and Graphics* **2** (1998) 7–15.

[12] Mitrinovic, D. S., Pecaric, J. E., Volenec, V.: *Recent Advances in Geometric Inequalities*. Kluwer Acad. Publ., Dordrecht, Boston, London 1989.

[13] Nelsen, R.: *Proofs Without Words*. MAA 1993.

[14] Nelsen, R.: *Proofs Without Words II*. MAA 2000.

[15] Parrilo, P. A.: Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization. *PhD. thesis.* California Institute of Technology, Pasadena, California (2000).

[16] Pech, P.: *Selected Topics in Geometry with Classical vs. Computer Proving*. World Scientific, Singapore 2007.

[17] Petrovšek, M., Wilf, H. S., Zeilberger, D.: *A=B*. A. K. Peters, Ltd., Wellesley, MA 1996.

[18] Polya, G.: *Mathematical Discovery*. Willey & Sons 1962.

[19] Recio, T., Sterk, H., Vélez, M. P.: *Project 1. Automatic Geometry Theorem Proving*. In: Some Tapas of Computer Algebra, A. Cohen, H. Cuipers, H. Sterk (eds), Algorithms and Computations in Mathematics, Vol. 4, Springer, 1998, 276–296.

[20] Reznick, B.: Some concrete aspects of Hilbert's 17th Problem. *Contemp. Math.* **253** (2000), 257–272.

[21] Tarski, A.: *A Decision Method for Elementary Algebra and Geometry*. University of California Press, Berkeley 2000.

[22] Wang, D.: *Elimination Methods*. Springer-Verlag, Wien New York 2001.

[23] Wang, D.: *Elimination Practice. Software Tools and Applications*. Imperial College Press, London, (2004).

[24] Weitzenböck, R.: *Math. Zeitschrift* **5** (1919), 137–146.

[25] Wu, W.-t., Gao, X.: Mathematics Mechanization and applications after thirty years. *Front. Comput. Sci. China* **1** (2007), 1–8.

[26] Yang, L., Zhang, J.: A Practical Program of Automated Proving for a Class of Geometric Inequalities. In: Automated Deduction in Geometry (Richter-Gebert, J., Wang, D. eds.), *Lecture Notes in Computer Science* **2061**, pp. 41–57. Springer-Verlag, Berlin Heidelberg (2001).