Geometric Automated Theorem Proving

Pedro Quaresma
CISUC, Mathematics Department
Faculty of Science and Technology, University of Coimbra

University of Urbino, January 2019

Synthetic Methods

Synthetic methods attempt to automate traditional geometry proof methods, producing human-readable proofs.

Seminal paper of Gelernter et al. It was based on the human simulation approach and has been considered a landmark in the AI area [Gel59, GHL60].

▶ Geometric reasoning - small and easy to understand proofs.
▶ Use of predicates only allow reaching fix-points.
▶ numerical model;
▶ constructing auxiliary points;
▶ generating geometric lemmas.

In spite of the success and significant improvements with these methods, the results did not lead to the development of a powerful geometry theorem prover [BdC95, CP79, CP86, Gil70, KA90, Nev74, Qua89].

Gelernter’s GATP

A long-range program directed at the problem of “intelligent” behaviour and learning in machines has attained its first objective in the simulation on a high-speed digital computer of a machine capable of discovering proofs in elementary Euclidean plane geometry without resorting to exhaustive enumeration or to a decision procedure. The particular problem of a theorem proving in plane geometry was chosen as representative of a large class of difficult tasks that seem to require ingenuity and intelligence for their successful completion.

The theorem proving program relies upon heuristic methods to restrain if from generating proof sequences that do not have a high a priori probability of leading to a proof for the theorem in hand.

Gelernter’s GATP

Backward chaining approach.

∀ geometric elements\([H_1 \land \cdots \land H_r] \Rightarrow G]\)

To prove \(G\) we search the axiom rule set to find a rule of the following form

\([G_1 \land \cdots \land G_r] \Rightarrow G]\)

until the sub-goals are hypothesis.

The proof search will generate an and-or-proof-tree.

---

Example 1 - Gelernter

points\((A, B, C) \land AB \parallel CD \land AD \land BC \land \text{coll}(E, A, C) \land \text{coll}(E, B, D)\) \(\Rightarrow AE = EC\)

GEOM — A “Coelho” out of the hat

Two uses of the geometric diagram as a model [CP86]:
  ▶ the diagram as a filter (a counter-example);
  ▶ the diagram as a guide (an example suggesting eventual conclusions).

Top-down or bottom-up directions? A general prover should be able to mix both directions of execution [CP86].

The introduction of new points can be envisaged as a means to make explicit more information in the model [CP86].

Although various strategies and heuristics were subsequently adopted and implemented, the problem of search space explosion still remains and makes the methods of this type highly inefficient [CP79, CP86].
Example - GEOM

GEOM is a Prolog program that generates proofs for problems in high school plane geometry [CP86].

A user presents problems to GEOM by declaring the hypotheses, the optional diagram and the goal.

GEOM starts from the goal, top-down and with a depth-first strategy, outputing its deductions and reasons for each step of the proof.

The diagram works mostly as a source of counter-examples for pruning unprovable goals, and so proofs need not depend on it (...). However, the diagram may also be used in a positive guiding way.

The geometric knowledge of GEOM, i.e. some of the axioms and theorems of elementary plane geometry, is embodied in nine procedures.

They are: equal angles (EAI), right angles (RAI), equal magnitude (EM, EM1), equal segments (ESI), midpoints (MP), parallel segments (PRI), parallelogram (PG), congruence (DIRCON) and diagram routines.

Because each procedure may call itself through others, the search space can grow quite large, in particular when the clause for differences of segments is used.

Coordinate-free Methods

Instead of coordinates, some basic geometric quantities, e.g. the ratio of parallel line segments, the signed area, and the Pythagorean difference (vector methods).

▶ Area method [CGZ93, JNQ12, QJ06b];
▶ Full-angle method [CGZ94, CGZ96b];
▶ Solid geometry [CGZ95].

Pros: Geometric proofs, small and human-readable.

Cons:
▶ not the “normal” high-school geometric reasoning;
▶ for many conjectures these methods still deal with extremely complex expressions.
Area Method — Basic Geometric Quantities

**Definition (Ratio of directed parallel segments)**
For four collinear points $P$, $Q$, $A$, and $B$, such that $A \neq B$, the ratio of directed parallel segments, denoted $\frac{PQ}{AB}$, is a real number.

**Definition (Signed Area)**
The signed area of triangle $ABC$, denoted $S_{ABC}$, is the area of the triangle with a sign depending on its orientation in the plane.

**Definition (Pythagoras difference)**
For three points $A$, $B$, and $C$, the Pythagoras difference, is defined in the following way:

$$P_{ABC} = AB^2 + CB^2 - AC^2.$$
The Area Method — The Proof Algorithm

Express the hypothesis of a theorem using a set of constructive statements.

Each constructive statement introduces a new point.

The conclusion is expressed by a polynomial in some geometry quantities (defined above), without any relation to a given system of coordinates.

The proof is then developed by eliminating, in reverse order, the point introduced before, using for that purpose a set of lemmas.

Constructive Geometric Statements

ECS1 construction of an arbitrary point U; (…).

ECS2 construction of a point Y such that it is the intersection of two lines (\textsc{line} U V) and (\textsc{line} P Q);
ndg-condition: \textsc{line} U V \parallel \textsc{line} P Q; U \neq V; P \neq Q.
degree of freedom for Y: 0

ECS3 construction of a point Y such that it is a foot from a given point P to (\textsc{line} U V); (…).

Forms of Expressing the Conclusion

<table>
<thead>
<tr>
<th>property</th>
<th>in terms of geometric quantities</th>
</tr>
</thead>
<tbody>
<tr>
<td>points A and B are identical</td>
<td>( P_{ABA} = 0 )</td>
</tr>
<tr>
<td>points A, B, C are collinear</td>
<td>( S_{ABC} = 0 )</td>
</tr>
<tr>
<td>AB is perpendicular to CD</td>
<td>( P_{ABA} \neq 0 \land P_{CDB} \neq 0 \land P_{ACD} = P_{BCD} )</td>
</tr>
<tr>
<td>AB is parallel to CD</td>
<td>( P_{ABA} \neq 0 \land P_{CDB} \neq 0 \land S_{ACD} = S_{BCD} )</td>
</tr>
<tr>
<td>O is the midpoint of AB</td>
<td>( S_{ABO} = 0 \land P_{ABA} \neq 0 \lor S_{ABO} = \frac{1}{2} )</td>
</tr>
<tr>
<td>AB has the same length as CD</td>
<td>( P_{ABA} = P_{CDB} )</td>
</tr>
<tr>
<td>points A, B, C, D are harmonic</td>
<td>( S_{ABC} = 0 \lor S_{ABD} = 0 \land P_{BCB} = 0 \land P_{BDB} \neq 0 \lor S_{ACD} \neq S_{BCD} )</td>
</tr>
<tr>
<td>angle ABC has the same measure as DEF</td>
<td>( P_{ABA} \neq 0 \land P_{AEB} \neq 0 \land P_{ACB} = P_{BCD} \neq 0 \lor P_{OEF} \neq 0 \land S_{ABC} \cdot P_{DEF} = S_{DEF} \cdot P_{ABC} )</td>
</tr>
<tr>
<td>A and B belong to the same circle arc CD</td>
<td>( S_{ACD} \neq 0 \land S_{BCD} \neq 0 \land S_{CAD} \cdot P_{CBD} = S_{CBD} \cdot P_{CAD} )</td>
</tr>
</tbody>
</table>

Elimination Lemmas

EL2 Let \( G(Y) \) be a linear geometric quantity and point Y is introduced by the construction (\textsc{pratio} Y W (\textsc{line} U V) r).
Then it holds
\[
G(Y) = G(W) + r(G(V) - G(U)).
\]

EL3 Let \( G(Y) \) be a linear geometric quantity and point Y is introduced by the construction (\textsc{inter} Y (\textsc{line} U V) (\textsc{line} P Q)). Then it holds
\[
G(Y) = \frac{S_{UPQ} G(V) - S_{VPQ} G(U)}{S_{UPVQ}}.
\]
Constructive Steps & Elimination Lemmas

<table>
<thead>
<tr>
<th>Geometric Quantities</th>
<th>( P_{AYB} )</th>
<th>( P_{ABY} )</th>
<th>( P_{ABCY} )</th>
<th>( S_{ABY} )</th>
<th>( S_{ABCY} )</th>
<th>( AP )</th>
<th>( BY )</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECS2</td>
<td>EL5</td>
<td>EL3</td>
<td>EL11</td>
<td>EL1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ECS3</td>
<td>EL6</td>
<td>EL4</td>
<td>EL12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ECS4</td>
<td>EL7</td>
<td>EL2</td>
<td>EL13</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ECS5</td>
<td>EL10</td>
<td>EL9</td>
<td>EL8</td>
<td>EL14</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Elimination Lemmas

The Algorithm

\[ S = (C_1, C_2, \ldots, C_m, (E, F)) \] is a statement in \( C \).

The algorithm tells whether \( S \) is true, or not, and if it is true, produces a proof for \( S \).

for (i=m; i==1; i--) {
    if (the ndg conditions of \( C_i \) is satisfied) exit;
    // Let \( G_1, \ldots, G_n \) be the geometric quantities in \( E \) and \( F \)
    for (j=1; j<=n, j++) {
        \( H_j \) <-> eliminating the point introduced by construction \( C_i \) from \( G_j \)
        \( E \) <- \( E[G_j:=H_j] \)
        \( F \) <- \( F[G_j:=H_j] \)
    }
}

if (E==F) \( S \) <= true else \( S \) <= false

Adding to that it is needed to check the ndg condition of a construction (three possible forms).

An Example (Ceva’s Theorem)

Let \( \triangle ABC \) be a triangle and \( P \) be an arbitrary point in the plane. Let \( D \) be the intersection of \( AP \) and \( BC \), \( E \) be the intersection of \( BP \) and \( AC \), and \( F \) the intersection of \( CP \) and \( AB \). Then:

\[
\frac{AF}{FB} \frac{BD}{DC} \frac{CE}{EA} = 1
\]

Example — Proof

The proof of a conjecture is based on eliminating all the constructed points, in reverse order, until an equality in only the free points is reached.

\[
\begin{align*}
\frac{AF}{FB} \frac{BD}{DC} \frac{CE}{EA} &= \frac{S_{APC}}{S_{BCP}} \frac{BD}{DC} \frac{CE}{EA} & \text{the point } F \text{ is eliminated} \\
&= \frac{S_{APC}}{S_{BCP}} \frac{S_{BPA}}{S_{CAP}} \frac{CE}{EA} & \text{the point } D \text{ is eliminated} \\
&= \frac{S_{APC}}{S_{BCP}} \frac{S_{BPA}}{S_{CAP}} \frac{S_{EPC}}{S_{ABP}} \frac{CE}{EA} & \text{the point } E \text{ is eliminated} \\
&= 1
\end{align*}
\]

Elimination Steps: 3; Geometric Steps: 6; Algebraic Steps: 23; Total Steps: 32; CPU Time 0.004s. The GATP also provide the ndg conditions.
Intuitively, a full-angle $\angle [u, v]$ is the angle from line $u$ to line $v$. Two full-angles $\angle [l, m]$ and $\angle [u, v]$ are equal if there exists a rotation $K$ such that $K(l)\parallel u$ and $K(m)\parallel v$.

**Full-Angle is defined as an ordered pair of lines which satisfies the following rules** [CGZ96b]:

- **R1** For all parallel lines $AB \parallel PQ$, $\angle [0] = \angle [AB, PQ]$ is a constant.
- **R2** For all perpendicular lines $AB \perp PQ$, $\angle [1] = \angle [AB, PQ]$ is a constant.
- **R7** If $PX$ is parallel to $UV$, then $\angle [AB, PX] = \angle [AB, UV]$.
- **R8** If $PX$ is perpendicular to $UV$, then $\angle [AB, PX] = \angle [1] + \angle [AB, UV]$.

### Solid Geometry

Solid Geometry Method — For any points $A, B, C$ and $D$ in the space, the signed volume $V_{ABCD}$ of the tetrahedron $ABCD$ is a real number which satisfies the following properties [CGZ95]:

- **V.1** When two neighbor vertices of the tetrahedron are interchanged, the signed volume of the tetrahedron will change signs, e.g., $V_{ABCD} = -V_{ABDC}$.
- **V.2** Points $A, B, C$ and $D$ are coplanar iff $V_{ABCD} = 0$.
- **V.3** There exist at least four points $A, B, C$ and $D$ such that $V_{ABCD} \neq 0$.
- **V.4** For five points $A, B, C, D$ and $O$, we have $V_{ABCD} = V_{ABCO} + V_{ABOD} + V_{AOCD} + V_{OBCD}$.
- **V.5** If $A, B, C, D, E$ and $F$ are six coplanar points and $S_{ABC} = \lambda S_{DEF}$ then for any point $T$ we have $V_{TABC} = \lambda V_{TDEF}$.

### Algebraic Methods

**Algebraic Methods:** are based on reducing geometry properties to algebraic properties expressed in terms of Cartesian coordinates.

The biggest successes in automated theorem proving in geometry were achieved (i.e., the most complex theorems were proved) by algebraic provers based on:

- **Wu’s method** [Cho85, Cho88];
- **Gröbner bases method** [Buc98, Kap86].

Decision procedures.

No readable, traditional geometry proofs, only a yes/no answer (with a corresponding algebraic argument).

### Wu’s Method

An elementary version of Wu’s method is simple: Geometric theorem $T$ transcribed as polynomial equations and inequations of the form:

- **H:** $h_1 = O, \ldots, h_s = O, d_1 \neq 0, \ldots, d_t \neq 0$;
- **C:** $c = 0$.

Proving $T$ is equivalent to deciding whether the formula

$$\forall x_1, \ldots, x_n [h_1 = 0 \land \ldots \land h_s \land d_1 \neq 0 \land \ldots \land d_t \neq 0 \implies c = 0] \quad (1)$$

is valid.
Wu’s Method

Computes a characteristic set $C \{ h_1, \ldots, h_b \}$ and the pseudo-remainder $r$ of $c$ with respect to $C$.

If $r$ is identically equal to 0, then $T$ is proved to be true.

The subsidiary condition $J \neq 0$, where $J$ is the product of initials of the polynomials in $C$ are the ndg conditions [CG90, WT86, Wu00].

This is a decision procedure.

GCLC Implementation of Wu’s Method (cont)

Triangulation, step 1; step 2; step 3; step 4; step 5

Calculating final remainder of the conclusion:

$$g = 2u_1u_2 - 2u_1u_3 + u_2u_3x_1^2 + u_1u_3x_2^2 + 2u_1u_2x_1^2 - u_1u_2x_2^2$$

with respect to the triangular system.

Pseudo remainder with $p_r$ over variable $x_0$:

$$g = (2u_1u_2 - u_1u_3 - 2u_1u_3 + u_2u_3) x_0^2 x_1^3 + (2u_1u_2 - 2u_1u_3 + u_2u_3) x_0^2 x_1^2 +$$

$$+ (u_2u_3 - u_1u_3 - u_1u_3 + u_2u_3) x_0^2 x_1 + (u_1u_3 - u_2u_3 - u_1u_3 - u_2u_3) x_1^3 + (u_1u_3 - u_2u_3 - u_1u_3 - u_2u_3) x_1^2 x_1$$

(\ldots )

Pseudo remainder with $p_0$ over variable $x_1$: $g = 0$

Status: The conjecture has been proved.

\ldots but all the calculations made, are not translatable to geometric reasoning.

GCLC Implementation of Wu’s Method

Let $\triangle ABC$ be a triangle and $P$ be an arbitrary point in the plane. Let $D$ be the intersection of $AP$ and $BC$, $E$ be the intersection of $BP$ and $AC$, and $F$ the intersection of $CP$ and $AB$. Then: $\frac{AP}{DP} \cdot \frac{DP}{BP} \cdot \frac{BP}{CP} = 1$

$$p_1 = -u_3x_2 + (u_2 - u_1)x_1 + u_3u_1$$

$$p_2 = u_5x_0 - u_4x_1$$

$$p_3 = -u_3x_4 + u_2x_3$$

$$p_4 = u_5x_4 + (-u_4 + u_1)x_3 - u_5u_1$$

$$p_5 = (u_5 - u_3)x_5 + (-u_5u_2 + u_4u_3)$$

$$p_6 = 2u_5x_0x_2^2 x_2^2 + u_5x_0x_2^2 x_2 - u_3x_6x_3x_1 + u_2x_6x_3 x_2^2 -$$

$$- u_1x_1^2 x_1^2 + 2u_3 u_1 x_2^2 x_2 - u_2^2 u_3 x_3 x_1$$

Gröbner Basis

A Gröbner basis of an ideal is a special basis using which the membership problem of the ideal as well as the membership problem of the radical of the ideal can be easily decided.

(\ldots ) to decide whether a finite set of geometry hypotheses are not polynomial equations, in conjunction with a finite set of subsidiary conditions expressed as negations of polynomial equations which rule out degenerate cases, imply another geometry relation given as a conclusion.

Such a problem is shown to be equivalent to deciding whether a finite set of polynomials does not have a solution in an algebraically closed field. Using Hilbert’s Nullstellensatz, this problem can be decided by checking whether 1 is in the ideal generated by these polynomials.

This test can be done by computing a Gröbner basis of the ideal.
GCLC Implementation of Gröbner Basis Method

Let \( \triangle ABC \) be a triangle and \( P \) be an arbitrary point in the plane. Let \( D \) be the intersection of \( AP \) and \( BC \), \( E \) be the intersection of \( BP \) and \( AC \), and \( F \) the intersection of \( CP \) and \( AB \). Then:

\[
\frac{AP}{PB} = \frac{BC}{CD} = \frac{FA}{PA} = 1.
\]

Conjecture \( p_k = 2x_6x_1^2x_2^4 - 3u_1x_6x_2^2x_1^2 + u_2^2x_6x_2^2x_1 - u_2x_6x_3x_1^3 + u_2^2x_6x_1x_2^2 - u_1^2x_2^2x_1^2 + 2u_1x_2x_3x_1^2 - u_2^2x_1x_3x_1^2 \)

The used proving method is Buchberger’s method.

Input polynomial system is:

\[
\begin{align*}
p_0 &= -u_3x_2 + (u_2 - u_1)x_1 + u_3u_1 \\
p_1 &= u_5x_2 - u_4x_1 \\
p_2 &= -u_3x_4 + u_2x_3 \\
p_3 &= u_5x_4 + (-u_4 + u_1)x_3 - u_5u_1 \\
p_4 &= (u_5 - u_1)x_6 + (-u_5u_2 + u_4u_3)
\end{align*}
\]

Geometry Deductive Database

- In the general setting: structured deductive database and the data-based search strategy to improve the search efficiency.\(^1\)
- Selection of a good set of rules; adding auxiliary points and constructing numerical diagrams as models automatically.

The result program can be used to find fix-points for a geometric configuration, i.e. the program can find all the properties of the configuration that can be deduced using a fixed set of geometric rules.

Generate ndg conditions.

Structured deductive database (graphs) reduce the size of the database in some cases by one thousand times.

\(^1\)Semantic Graphs are an alternative?

“New” approaches

- An approach based on a deductive database and forward chaining works over a suitably selected set of higher-order lemmas and can prove complex geometry theorems, but still has a smaller scope than algebraic provers [CGZ94, CGZ00, YCG10b].

- Quaife used a resolution theorem prover to prove theorems in Tarski’s geometry [Qua89].

- A GATP based on coherent-logic capable of producing both readable and formal proofs of geometric conjectures of certain sort [SPJ11].

- Probabilistic verification of elementary geometry statements [CFGG97, RGK99].

- Visual Reasoning/Proofs [Kim89, YCG10a, YCG10b].
Deductive Databases

- Use canonical form for predicates;
- Use equivalent classes to represent some predicates;
- Use representative elements for equivalent classes;
- Breadth-first forward chaining search:
  where $D_0$ is the hypotheses of the geometry statement and $R$ is the rule set.

For each rule $r$ in $R$, apply it to $D_0$ to obtain new facts. Let $D_1$ be the union of $D_0$ and the set of new facts obtained.

Repeat the above process for $D_1$ to obtain $D_2$, and so on.
If at certain step $D_k = D_{k+1}$, we say that a fix-point for $D_0$ and $R$ is reached.

Geometry Deductive Database – The Orthocenter Theorem

```
points(A, B, C) \land coll(E, A, C) \land perp(B, E, A, C) \land coll(F, B, C) \land perp(A, F, B, C) \land coll(H, A, F) \land coll(H, B, E) \land coll(G, A, B) \land coll(G, C, H)
```

The fix-point contains two of the most often encountered properties of this configuration:

- $perp(C, G, A, B)$;
- $\angle FGC = \angle CGE$

Quaife’s GATP

Tarski axiomatic system: is, or rather its algebraic equivalent, complete and decidable.

Quaife developed a GATP for Euclidean plane geometry within the automated reasoning system OTTER (a resolution theorem prover) [Qua89].

(A1) Reflexivity axiom for equidistance.
\[ \rightarrow u \cdot v \equiv v \cdot u \]

(A2) Transitivity axiom for equidistance.
\[ u \cdot v \equiv w \cdot x , u \cdot v \equiv y \cdot z \rightarrow w \cdot x \equiv y \cdot z \]

(A4) Segment construction axiom, two clauses.
(A4.1) $\rightarrow B(u, v, \text{Ext}(u, v, w, x))$
(A4.2) $\rightarrow v \cdot \text{Ext}(u, v, w, x) \equiv w \cdot x$

(...)

Quaife’s GATP

Heuristics

- Maximum weight for retained clauses at 25,
- First attempt to obtain a proof in which no variables are allowed in any generated and retained clause.

The provers based upon Wu’s algorithm, are able to prove quite more difficult theorems in geometry then those by Quaife’s GATP.

However Wu’s method only works with hypotheses and theorems that can be expressed as equations, and not with inequalities as correspond to the relation $B$ in Quaife’s resolution prover.
Coherent Logic GATP

Coherent Logic is a fragment of first-order logic with formulae of the following form:

\[ A_1(x) \land \ldots \land A_n(x) \rightarrow \exists y_1 B(x,y_1) \lor \ldots \lor \exists y_m B(x,y_m) \]

with a breath-first proof procedure sound and complete [BC05].

ArgoCLP (Coherent Logic Prover of the Argo Group\(^2\))
- new proof procedures;
- proof trace exportable to:
  - a proof object in Isabelle/Isar;
  - human readable (English/\LaTeX).\(^3\)

not aimed at proving complex geometry theorems but rather at proving foundational theorems (close to the axiom level) [SPJ11].

\(^2\)http://argo.matf.bg.ac.rs/

---

Visual Reasoning/Representation

Visual Reasoning extend the use of diagrams with a method that allows the diagrams to be perceived and to be manipulated in a creative manner [Kim89].

Visually Dynamic Presentation of Proofs linking the proof done by a synthetic method (full-angle) with a visual presentation of the proof [QSGB19, SQ10, YCG10a, YCG10b].

---

Probabilistic Verification

Probabilistic verification of elementary geometry statements [CFG97, RGK99].

\textit{Cinderella (\ldots) use (\ldots) a technique called “Randomized Theorem Checking”}. First the conjecture (\ldots) is generated. Then the configuration is moved into many different [random] positions and for each of these it is checked whether the conjecture still holds. (\ldots) generating enough(!) random (!) examples where the theorem holds is at least as convincing as a computer-generated symbolic proof.

\textit{User Manual for the Interactive Geometry Software Cinderella, Jürgen Richter-Gebert, Ulrich H. Kortenkamp}

---

Visual Reasoning in Geometry Theorem Proving

We study the role of visual reasoning as a computationally feasible heuristic tool in geometry problem solving. We use an algebraic notation to represent geometric objects and to manipulate them.

We show that this representation captures powerful heuristics for proving geometry theorems, and that it allows a systematic manipulation of geometric features in a manner similar to what may occur in human visual reasoning.\(^4\)

Michelle Y. Kim,
An Example

Consider the problem in “Given a square $ABCD$, take the midpoints of the four sides, and prove that the two triangles $\triangle EEH$ and $\triangle GFH$ are congruent to each other.”

To solve this problem, backward-chaining methods used by most of previous geometry-theorem proving systems [Gel59, CP86] would first set up a goal to prove that the two triangles are congruent (…) A human mathematician, given the problem, may perceive an apparent symmetry in the diagram by observing a reflection across $FH$ or across $EG$. As a symmetry is observed, it can be shown with little effort that the two triangles are congruent, and thus repeated proofs can be avoided.

Formalisation

Full formal proofs mechanically verified by generic theorem proof assistants (e.g. Isabelle [Pau94, PN90]. Coq [Tea09]).

- Hilbert’s *Foundations of Geometry* [Hil77, MF03, DDS00];
- Jan von Plato’s constructive geometry [Kah95, vP95];
- French high school geometry [Gui04];
- Tarski’s geometry [Nar07b, BBN16];
- An axiom system for compass and ruler geometry [BNW18];
- Projective geometry [MNS11, FT11];
- Area Method [JNQ12, Nar06];
- Algebraic methods in geometry [MPPJ12].

Proof Assistants

Proof assistant (or interactive theorem prover) is a software tool to assist with the development of formal proofs by human-machine collaboration.

- Isabelle—https://isabelle.in.tum.de/—Isabelle is a generic proof assistant. It allows mathematical formulas to be expressed in a formal language and provides tools for proving those formulas in a logical calculus.
- Coq—https://coq.inria.fr/—Coq is a formal proof management system. It provides a formal language to write mathematical definitions, executable algorithms and theorems together with an environment for semi-interactive development of machine-checked proofs.

Others: HOL Light; Lean; Mizar; …
Area Method: Formalisation

Formalisation [JNQ12, Nar06, Nar09]:
1. $\overline{AB} = 0$ if and only if the points $A$ and $B$ are identical
2. $S_{ABC} = S_{CAB}$
3. $S_{ABC} = -S_{BAC}$
4. If $S_{ABC} = 0$ then $\overline{AB} + \overline{BC} = \overline{AC}$ (Chasles's axiom)
5. There are points $A$, $B$, $C$ such that $S_{ABC} \neq 0$ (dimension; not all points are collinear)
6. $S_{ABC} = S_{DBC} + S_{ADC} + S_{ABD}$ (dimension; all points are in the same plane)
7. For each element $r$ of $F$, there exists a point $P$, such that $S_{ABP} = 0$ and $\overline{AP} = \overline{PR}$ (construction of a point on the line)
8. If $A \neq B$, $S_{ABP} = 0$, $\overline{AP} = \overline{PR}$, $S_{ABP} = 0$ and $\overline{AP} = \overline{PR}$, then $P = P'$ (unicity)
9. If $PQ \parallel CD$ and $\overline{AB} = 1$ then $DIQ \parallel PC$ (parallelogram)
10. If $S_{PAC} = 0$ and $S_{ABC} = 0$ then $\overline{AP} = \overline{PC}$ (proportions)
11. If $C \neq D$ and $AB \perp CD$ and $EF \perp CD$ then $AB \parallel EF$
12. If $A \neq B$ and $AB \perp CD$ and $EF \parallel CD$ then $EF \perp CD$
13. If $FA \perp BC$ and $S_{ABC} = 0$ then $4S_{ABC} = \overline{AB}^2 \overline{BC}^2$ (area of a triangle)

Using this axiom system all the properties of the geometric quantities required by the area method were formally verified (within the Coq proof assistant), demonstrating the correctness of the system and eliminating all concerns about provability of the lemmas [Nar09].

Automated Discovery

▶ Locus Generation: to determine the implicit equation of a locus set [BAE07, BA12].

The set of points determined by the different positions of a point, the tracer, as a second point in which the tracer depends on, called the mover, runs along the one dimensional object to which it is restrained.

▶ Automated Finding of Theorems: the discovery of new facts about a given geometric configuration.

Finding locus equations

For most DGS a locus is basically a set of points in the screen with no algebraic information [BAE07, ABMR14].

▶ Numerical approach, based on interpolation (Cinderella, Cabri) [Bot02].

▶ Symbolic method, finding the equation of a locus [BL02, BA12, ABMR14].

Determine the equation of a locus set using remote computations on a server [EBA10].
Loci Finding: Algorithm

A statement is considered where the conclusion does not follow from the hypotheses.

Symbolic coordinates are assigned to the points of the construction (where every free point gets two new free variables $u_i$, $u_{i+1}$, and every bounded point gets up to two new dependent variables $x_j$, $x_{j+1}$) so the hypotheses and thesis are rewritten as polynomials $h_1, \ldots, h_n$ and $t$ in $\mathbb{Q}[u, x]$.

Eliminating the dependent variables in the ideal (hypotheses, thesis), the vanishing of every element in the elimination ideal (hypotheses, thesis) $\cap \mathbb{Q}[u]$ is a necessary condition for the statement to hold.

Implementation (cont.)

Two different tasks are performed over GeoGebra constructions:

- the computation of the equation of a geometric locus in the case of a locus construction;
  \[
  \text{LocusEquation}( \text{<Locus Point>}, \text{<Moving Point>} )
  \]

- the study of the truth of a general geometric statement included in the GeoGebra construction as a Boolean variable.

Both tasks are implemented using algebraic automatic deduction techniques based on Gröbner bases computations.

The algorithm, based on a recent work on the Gröbner cover of parametric systems, identifies degenerate components and extraneous adherence points in loci, both natural byproducts of general polynomial algebraic methods [BA12].

Automated Finding of Theorems

Deductive Database Approach. Forward chaining till reaching a fixed point.

An interesting application is to discover ‘new’ facts about a given geometric configuration.

Our experiments show that our program can discover most of the well-known results and often some unexpected ones.

Automated Geometer

The Automated Geometer, AG, (also meaning Amateur Geometer) intends to be a GeoGebra module where pure automatic discovery is performed.

It includes a generator of further geometric elements from those of a given construction, and a set of rules for producing conjectures on the whole set of elements.

But the ultimate AG aim is not just performing a systematic exploration of the space of possible conjectures, but mimicking human thought when doing elementary geometry.


Geometric Tools


- DGS: Cabri Geometry; C.a.R.; Cinderella; GCLC; GeoGebra; The Geometer’s Sketchpad;
  JGEX [Gro11, CGY04, Hoh02, Jac01, Jan06, LS90, RGK99]; . . .
- GATP; GCLC; OpenGeoProver; JGEX; GeoProof; . . .
  - verification of the soundness of a geometric construction [JQ07].
  - reason about a given DGS construction [CGZ96a, JQ06, Nar07a, QJ06b].
  - human-readable proofs [JNQ12, QJ06a].
- RGK [QJ07, Qua11].
- eLearning [ABY86, HLY86, QJ06b, SQ08, QSM18, SQMC18]
Dynamic Geometry Software

DGS are computer environments which allow one to create and then manipulate geometric constructions, primarily in plane geometry.

Geometry Automated Theorem Provers: GCLC

Proving geometrical theorems by computer programs.

```plaintext
*** Ceva's theorem
point A 80 10
point B 50 80
point C 100 80
point P 75 65
line a B C
line b A C
line c A B
line pa P A
line pb P B
line pc P C
*** constructed point
intersec D a pa
intersec E b pb
intersec F c pc
*** conjecture
prove {equal{mult{mult{sratio A F F B}{sratio B D D C}}{sratio C E E A}}}1
```
Integration: DGSs & GATPs

▶ GCLC/WinGCLC - A DGS tool that integrates three GATPs: Area Method, Wu’s Method and Gröbner Bases Method [JQ06, Jan06].
▶ JGEX - is a software which combines a DGS and some GATPs (full angle, Wu’s Method, Deductive Databases for the full angle) [YCG10a, YCG10b, CGY04].
▶ GeoProof - DGS tool that integrates three GATPs Area Method, Wu’s Method and Gröbner Bases Method [Nar07a].
▶ GeoGebra - DGS + CAS + GATPs [ABK16, BHJ15, Kov15].
▶ Theorema Project - Theorema is a project that aims at supporting the entire process of mathematical theory exploration within one coherent logic and software system [BCJ06]. Implementation of the Area Method [Rob02, Rob07].


Integration Issues

Integrate a mosaic of tools into a coherent system.

▶ Intergeo Project [SHK10];
▶ Deducation STREP Proposal [WSA12];
▶ Road to an Intelligent Geometry Book, COST Proposal, OC-2019-1-XXXX.
iGeometryBook
The “Road to an Intelligent Geometry Book” (COST) Action is dedicated to the study of how current developing methodologies and technologies of knowledge representation, management, and discovery in mathematics, can be incorporated effectively into the learning environments of the future.

Repositories of Geometric Problems

GeoThms: a Web-based framework for exploring geometrical knowledge that integrates Dynamic Geometry Software (DGS), Automatic Theorem Provers (ATP), and a repository of geometrical constructions, figures and proofs. [JQ06, QJ07].

TGTP: a Web-based library of problems in geometry to support the testing and evaluation of geometric automated theorem proving (GATP) systems [Qua11].

Sets of Examples and Communities: Integeo; GeoGebra; Geometriagon; examples in the DGSs/GATPs. 

Proofs/Readable Proofs/Visual Proofs

Readable Proofs
▷ What is a readable proofs [QSGB19]?
▷ Can GATPs produce readable proofs [JNQ12]?

Visual Reasoning
▷ Proofs with a visual counterpart [QS19].
▷ Proofs done by “visual means” [YCG10a, YCG10b]

TGTP
A comprehensive and easily accessible, library of GATP test problems [Qua11].
▷ Web-based, easily available to the research community. Easy to use.
▷ Tries to cover the different forms of automated proving in geometry, e.g. synthetic proofs and algebraic proofs.
▷ provides a mechanism for adding new problems.
▷ (...) 

It is independent of any particular GATP system → the i2GATP common format [QH12].
Readability of a Proof

According to [CGZ94, p.442] a formal proof, done using the area method, is considered readable if one of the following conditions holds:

- the maximal term in the proof is less than or equal to 5;
- the number of deduction steps of the proof is less than or equal to 10;
- the maximal term in the proof is less than or equal to 10 and the deduction step is less than or equal to 20.

The de Bruijn factor [deR94, We00], the quotient of the size of corresponding informal proof and the size of the formal proof, could also be used as a measure of readability. Using this quotient a proof can be considered readable if the value is less than or equal to 2 (the formal proof is at most twice as larger than a given informal proof).

GATP, Readable Proofs: GCLC Area Method

(1)
$$\left( \begin{array}{cc} 2 \alpha & 0 \\ 0 & 2 \alpha \\ \end{array} \right) + 1$$
by the statement

(2)
$$\left( \begin{array}{cc} 0 & 1 \\ 1 & 0 \\ \end{array} \right) + 1$$
by geometric simplifications

(3)
$$\left( \begin{array}{cc} 0 & 1 \\ 1 & 0 \\ \end{array} \right) + 1$$
by algebraic simplifications

(4)
$$\left( \begin{array}{cc} 0 & 1 \\ 1 & 0 \\ \end{array} \right) + 1$$
by geometric simplifications

(5)
$$\left( \begin{array}{cc} 0 & 1 \\ 1 & 0 \\ \end{array} \right) + 1$$
by algebraic simplifications

(6)
$$\left( \begin{array}{cc} 0 & 1 \\ 1 & 0 \\ \end{array} \right) + 1$$
by geometric simplifications

(7)
$$\left( \begin{array}{cc} 0 & 1 \\ 1 & 0 \\ \end{array} \right) + 1$$
by algebraic simplifications

(8)
$$\left( \begin{array}{cc} 0 & 1 \\ 1 & 0 \\ \end{array} \right) + 1$$
by geometric simplifications

(9)
$$\left( \begin{array}{cc} 0 & 1 \\ 1 & 0 \\ \end{array} \right) + 1$$
by algebraic simplifications

(10)
$$\left( \begin{array}{cc} 0 & 1 \\ 1 & 0 \\ \end{array} \right) + 1$$
by geometric simplifications

(11)
$$\left( \begin{array}{cc} 0 & 1 \\ 1 & 0 \\ \end{array} \right) + 1$$
by algebraic simplifications

GATP, Readable Proofs: Coherent Logic

Example: Proof Generated by ArgoCLP

Let us prove that $p = r$ by reductio ad absurdum.

1. Assume that $p \neq r$.
2. It holds that the point $A$ is incident to the line $q$ or the point $A$ is not incident to the line $q$ (by axiom of excluded middle).
3. Assume that the point $A$ is incident to the line $q$.
4. From the facts that $p \neq q$ and the point $A$ is incident to the line $q$, it holds that the lines $p$ and $q$ intersect (by axiom ax21).
5. From the facts that the lines $p$ and $q$ intersect, and the lines $p$ and $q$ do not intersect we get a contradiction.
6. Assume that the point $A$ is not incident to the line $q$.
7. From the facts that the lines $p$ and $q$ do not intersect, it holds that the lines $p$ and $q$ do not intersect (by axiom ax8, line 21).
8. From the facts that the point $A$ is not incident to the line $q$, and the point $A$ is incident to the plane $\alpha$, and the line $q$ is incident to the plane $\alpha$, and the point $A$ is incident to the line $r$, and the line $r$ is incident to the plane $\alpha$, and the line $q$ and $r$ do not intersect, it holds that $p \neq r$ (by axiom ax22).
9. From the facts that $p \neq r$ and $p \neq r$ we get a contradiction.

Therefore, it holds that $p = r$.
This proves the conjecture.
Geometrography

Geometrography, “alias the art of geometric constructions” was proposed by Émile Lemoine between the late 1800s and the early 1900s [SBQ19, Mac93, Lem02, QSGB19].

Measure the complexity of ruler-and-compass geometric constructions.

Coefficient Simplicity: denoting the number of times any particular operation is performed.

Coefficient Exactitude: each time a drawing instrument is used, two types of error can be introduced in the image, systematic error and accidental errors due to personal operator’s actions.

Considering the modifications proposed by Mackay [Mac93], the following ruler-and-compass constructions and the corresponding coefficients can be considered.

To place the edge of the ruler in coincidence with one point . . . . \( R_1 \)
To place the edge of the ruler in coincidence with two points . . . \( 2R_1 \)
To draw a straight line . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . \( R_2 \)
To put one point of the compasses on a determinate point . . . \( C_1 \)
To put one point of the compasses on two determinate points . \( 2C_1 \)
To describe a circle . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . \( C_2 \)

For a given construction with \( l_1R_1 + l_2R_2 + m_1C_1 + m_2C_2 \) steps.

\[ \text{cs} = l_1 + l_2 + m_1 + m_2, \] is called the coefficient of simplicity.

\[ \text{ce} = l_1 + m_1 \] is called the coefficient of exactitude.

Extrapolating (modernising) geometrography to DGS.

Coefficient of simplicity – must be adapted to new tools.

Coefficient of exactitude – loose its meaning (error free manipulations).

Coefficient of freedom – counts the degrees of freedom, gives a value for the dynamism of the construction.

Geometrography in GCLC (commands in the GCL language): a point in the plane \( (D) \), two degrees of freedom; a line defined by two points \( (2C) \); a point in a line \( D \), one degree of freedom; etc.

Geometrography in GeoGebra: similar to GCLC, but using GeoGebra tools.

Geometrography as a way to measure the complexity and dynamism of a given construction, being able to compare between different solutions to a same goal

... and how about complexity of a proofs?
Geometric Search

When accessing RGK it should be possible to do geometric searches, i.e. we should be able to provide a geometric construction and look for similar constructions [QH12, HQ18].

Given (in the RGK) a triangle with three equal sides, the query about a triangle with three equal angles (which is geometrically equivalent) should be successful.

---

Conference & Journals

CADE (IJCAR/FLoC) International Conference on Automated Deduction http://www.cadeinc.org/conferences, every year.


---

What to Do Next?

Integration of Methods integrate the study of logical, combinatorial, algebraic, numeric and graphical algorithms with heuristics, knowledge bases and reasoning mechanisms.

Applications design and implement integrated systems for computer geometry, integrating, in a modular fashion, DGSs, ITPs, GATPs, RGPs, etc. in research and/or educational environments.

Higher Geometry The existing algorithms should be extended and improved, new and advanced algorithms be developed to deal with reasoning in different geometric theories.

Axiom Systems Development of new axiom systems, motivated by machine formalisation. [ADM09]

Formalisation formalising geometric theories and methods.

Discovery Automated discovery of new results.

---

Taxonomies for Geometry

The usefulness of repositories of geometric knowledge is directly related with the possibility of an easy retrieval of the information a given user is looking for [QSGB19, Qua18].

MSC—Mathematics Subject Classification (http://msc2010.org/)

CCS—Common Core Standard (http://www.corestandards.org/Math/)

---

Synthetic Methods Algebraic Methods Formalisation & Discovery GKM & Tools Bibliography
Bibliography I

Miguel Abánades, Francisco Botana, Zoltán Kovács, Tomás Recio, and Cailla Sólyom-Gecse.
Development of automatic reasoning tools in geogebra.

Miguel Á. Abánades, Francisco Botana, Antonio Montes, and Tomás Recio.
An algebraic taxonomy for locus computation in dynamic geometry.

The geometry tutor.

Jeremy Avigad, Edward Dean, and John Mumma.
A formal system for Euclid’s elements.

Francisco Botana and Miguel A. Abánades.
Automatic deduction in dynamic geometry using sage.
in THeU+21, CTP Components for Educational Software (postproceedings), volume 79 of EPTCS, 2012.

Francisco Botana, Miguel A. Abánades, and Jesús Escribano.
Computing locus equations for standard dynamic geometry environments.

Bibliography II

Pieter Bouwry, Gabriël Braun, and Julien Nabourx.
From Tanki to Descartes: Formalization of the Arithmetization of Euclidean Geometry.

Marc Bezem and Thierry Coquand.
Automating coherent logic.

Theorema: Towards computer-aided mathematical theory exploration.

Philippe Balbiani and Luis del Cerro.
Affine geometry of collinearity and conditional term rewriting.

Bibliography III

Automated Theorem Proving in GeoGebra: Current Achievements.

Francisco Botana and José L. Vallsance.
A dynamic-symbolic interface for geometric theorem discovery.

Michael Besson, Julien Narboux, and Freck Wiedijk.
Proof-checking siscal.
2018.

Francisco Botana.
Interactive versus symbolic approaches to plane loci generation in dynamic geometry environments.

B. Buchberger.
Introduction to Gröbner Bases.

Bibliography IV

Giuseppe Carrà Ferro, Giovanni Gallo, and Rosario Gennaro.
Probabilistic verification of elementary geometry statements.

Pedro Cobo, Josep Fortuny, Etel Puertas, and Philippe Richard.
AgentGeoM: a multiagent system for pedagogical support in geometric proof problems.

Shang-Ching Chou and Xiao-Shan Gao.
Ritt-woo’s decomposition algorithm and geometry theorem proving.

Shang-Ching Chou and Xiao-Shan Gao.
Automated reasoning in geometry.

Shang-Ching Chou, Xiao-Shan Gao, and Zheng Ye.
Java geometry expert.
Bibliography

Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang.
Automated production of traditional proofs for constructive geometry theorems.

Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang.
Machine Proofs in Geometry.

Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang.
Automated production of traditional proofs in solid geometry.

Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang.
Automated generation of readable proofs with geometric invariants. I. multiple and shortest proof generation.

Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang.
Automated generation of readable proofs with geometric invariants. II. theorem proving with full-angles.

Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang.
A deductive database approach to automated geometry theorem proving and discovering.

David Hilbert.
*Foundations of Geometry.*
Open Court Publishing, 1902.

H. Coelho and L. M. Pereira.
Geom: A Prolog geometry theorem prover.
Memórias 525, Laboratório Nacional de Engenharia Civil, Ministério de Habitação e Obras Públicas, Portugal, 1979.

H. Coelho and L. M. Pereira.
Automated reasoning in geometry theorem proving with Prolog.

Christophe Delhinger, Jean-fracçois Dufourd, and Pascal Schreck.
Higher-order intuitionistic formalization and proofs in Hilbert's elementary geometry.

N. G. de Bruijn.

Bibliography VI

S. C. Chou.
Proving and discovering geometry theorems using Wu's method.

Shang-Ching Chou.

H. Coelho and L. M. Pereira.
Geometric search in TGTP.

H. Coelho and L. M. Pereira.
A formalization of Grassmann-Cayley algebra in Coq and its application to theorem proving in projective geometry.

Bibliography VII

Jesús Escribano, Francisco Botana, and Miguel A. Álvarado.
Adding remote computational capabilities to dynamic geometry systems.

Laurent Fuchs and Laurent Thiéry.
A formalization of Grassmann-Cayley algebra in Coq and its application to theorem proving in projective geometry.

H. Galenter.
Realization of a geometry-theorem proving machine.

H. Galenter, J. R. Hansen, and D. W. Loveland.
Empirical explorations of the geometry theorem machine.

Paul C. Gilmore.
An examination of the geometry theorem machine.

Ron Gottsch.
About C.a.R.

Bibliography VIII

Frédérique Guilhot.
Formalisation en Coq d'un cours de géométrie pour le lycée.

David Hilbert.
*Foundations of Geometry.*
Open Court Publishing, 1902.

M. Haddad, F. Lichtenberger, and D. Y. Y. Yun.
An application of knowledge-base technology in education: a geometry theorem prover.
In *Proceedings of the 9th ACM symposium on Symbolic and algebraic computation, SYMSAC '86*, pages 141–147, New York, NY, USA, 1986. ACM.

M. Hohenwarter.
GEOGebra - a software system for dynamic geometry and algebra in the plane.

Yannis Haralambous and Pedro Quaresma.
Querying geometric figures using a controlled language, ontological graphs and dependency lattices.

Yannis Haralambous and Pedro Quaresma.
Geometric search in TCTP.
Bibliography IX

N. Jackie.
The Geometer's Sketchpad v4.0.

Predrag Janičić.
GCLC — A tool for constructive euclidean geometry and more than that.

Predrag Janičić, Julien Narboux, and Pedro Quaresma.
The Area Method: a recapitulation.

Predrag Janičić and Pedro Quaresma.
System description: GCLCprover + GeoThms.

Predrag Janičić and Pedro Quaresma.
Automatic verification of regular constructions in dynamic geometry systems.

Kenneth R. Koedinger and John R. Anderson.
Abstract planning and perceptual chunks: Elements of expertise in geometry.

J. S. Mackay.
The geometrography of euclid's problems.

Laura Meikle and Jacques Fleuriot.
The Area Method: a recapitulation.
Formalizing Hilbert's Grundlagen in Isabelle/Isar.
Laura Meikle and Jacques Fleuriot.

Noboru Matsuda and Kurt VanLehn.
Advanced geometry tutor: An intelligent tutor that teaches proof-writing with construction.

Julien Narboux.
Formalisation de la méthode des plans projectifs en Coq.

Julien Narboux.
A graphical user interface for formal proofs in geometry.

Julien Narboux.
Mechanical theorem proving in Tarski's geometry.

Julien Narboux.
Formalization of the area method.
Coq user contribution, 2009.

Gilles Kahn.
Constructive geometry according to Jan von Plato.

Deepak Kapur.
Using Gröbner bases to reason about geometry problems.

Michelle Y. Kim.
Visual reasoning in geometry theorem proving.

Zoltán Kovács.
The relation tool in geogebra 5.

Émile Lemoine.
Géométrie ou Art des constructions géométriques, volume 18 of Phys-Mathématique.
Scentia, 1902.

J. M. Labords and R. Strässer.
Cabri-géomètre: A microworld of geometry guided discovery learning.
Bibliography XIII

Arthur J. Nevins.
Plane geometry theorem proving using forward chaining.
AI Lab memo 363, MIT, Jan 1974.

Lawrence C. Paulson.
Isabelle: A Generic Theorem Prover, volume 828 of LNCS.

Lawrence C. Paulson and Tobias Nipkow.
Isabelle tutorial and user's manual.

Pedro Quaresma and Yannis Haralambous.
Geometry Constructions Recognition by the Use of Semantic Graphs.

Pedro Quaresma and Predrag Janičić.
Framework for constructive geometry (based on the area method).
technical report 2006/101, Centre for Informatics and Systems of the University of Coimbra, 2006.

Bibliography XIV

Pedro Quaresma and Predrag Janičić.
Integrating dynamic geometry software, deduction systems, and theorem repositories.

Pedro Quaresma and Predrag Janičić.
GeoThms – a Web System for euclidean constructive geometry.

Pedro Quaresma and Vanda Santos.
Proof Technology in Mathematics Research and Teaching, chapter Computer-generated geometry proofs in a learning context.

Pedro Quaresma, Vanda Santos, Pierluigi Graziani, and Nuno Baeta.
Taxonomies of geometric problems.

Pedro Quaresma, Vanda Santos, and Milena Matić.
WGL, a web laboratory for geometry.

Art Quaife.
Automated development of Tarski’s geometry.

Pedro Quaresma.
Thousands of geometric problems for geometric theorem provers (TGTP).

Pedro Quaresma.
Automatic deduction in an AI geometry book.

Philippe R. Richard, José M Fortuny, Markus Hohewarter, and Michel Gagnon.
geogebraTUTOR: une nouvelle approche pour la recherche sur l'apprentissage compétentiel et instrumenté de la géométrie à l'école secondaire.

Jürgen Richter-Gebert and Ulrich Kortenkamp.
The Interactive Geometry Software Cinderella.

Judit Robu.
Geometry Theorem proving in the Frame of the Theorema Project.

Bibliography XV

Pedro Quaresma.
Thousands of Geometric problems for geometric Theorem Provers (TGTP).

Pedro Quaresma.
Automatic deduction in an AI geometry book.

Philipp R. Richard, José M Fortuny, Markus Hohewarter, and Michel Gagnon.
geogebraTUTOR: une nouvelle approche pour la recherche sur l'apprentissage compétentiel et instrumenté de la géométrie à l'école secondaire.

Jürgen Richter-Gebert and Ulrich Kortenkamp.
The Interactive Geometry Software Cinderella.

Judit Robu.
Geometry Theorem Proving in the Frame of the Theorema Project.

Bibliography XVI

Judit Robu.
Automated Proof of Geometry Theorems Involving Order Relation in the Frame of the Theorema Project.

Vanda Santos, Nuno Baeta, and Pedro Quaresma.
Geometrography in dynamic geometry.

E. Santiago, Maxim Hendriks, Yves Kreis, Ulrich Kortenkamp, and Daniel Marqués.
I2G, a web laboratory for geometry.

Judit Robu.
Automated Proof of Geometry Theorems Involving Order Relation in the Frame of the Theorema Project.

Vanda Santos, Nuno Baeta, and Pedro Quaresma.
Geometrography in dynamic geometry.

E. Santiago, Maxim Hendriks, Yves Kreis, Ulrich Kortenkamp, and Daniel Marqués.
I2G, a web laboratory for geometry.

Pedro Quaresma.
Thousands of Geometric problems for geometric Theorem Provers (TGTP).

Pedro Quaresma.
Automatic deduction in an AI geometry book.

Philipp R. Richard, José M Fortuny, Markus Hohewarter, and Michel Gagnon.
geogebraTUTOR: une nouvelle approche pour la recherche sur l'apprentissage compétentiel et instrumenté de la géométrie à l'école secondaire.

Jürgen Richter-Gebert and Ulrich Kortenkamp.
The Interactive Geometry Software Cinderella.

Judit Robu.
Geometry Theorem Proving in the Frame of the Theorema Project.
Vanda Santos and Pedro Quaresma.

Vanda Santos, Pedro Quaresma, Milena Marić, and Helena Campos.
Web geometry laboratory: case studies in portugal and serbia.

The Coq Development Team.
TypiCal Project, Lyon, France, 2009.

Jan von Plato.
The axioms of constructive geometry.

Dongming Wang.
Geometry machines: From ai to smc.
10.1007/3-540-61732-9_60.

Freek Wiedijk.
The de Bruijn factor.
Portland, Oregon, USA, 14-18 August 2000.

Deductive framework for math-oriented collaborative teaching environments.
Small or Medium-scale focused Research Project (STREP) ICT Call 8: FP7-ICT-2011-8 proposal, January 2012.

W.-T. Wu.
Basic principles of mechanical theorem proving in elementary geometries.
10.1007/BF02328447.

Zheng Ye, Shang-Ching Chou, and Xiao-Shan Gao.
Visually dynamic presentation of proofs in plane geometry, part 1.

Zheng Ye, Shang-Ching Chou, and Xiao-Shan Gao.
Visually dynamic presentation of proofs in plane geometry, part 2.