# Geometric Theorem Proving

Pedro Quaresma

CISUC, Mathematics Department
University of Coimbra

Days in Logic 2012, University of Évora, 6-8 February 2012

# Geometric Automated Theorem Proving

An history more than sixty years long [CG01, Wan96].

Two major lines of research in GATP:

- Synthetic methods;
- Algebraic methods.

Mechanical Geometric Formula Derivation:

- Finding locus equation;
- Deriving geometry formulas.

News fields

- Geometric Tools: DGS/GATP/CAS/RGP/eLearning;
- Formalisation.

# Geometric Automated Theorem Proving

An history more than sixty years long [CG01, Wan96].

Two major lines of research in GATP:

- Synthetic methods;
- Algebraic methods.

Mechanical Geometric Formula Derivation:

- Finding locus equation;
- Deriving geometry formulas.

News fields

- Geometric Tools: DGS/GATP/CAS/RGP/eLearning;
- Formalisation.

# Geometric Automated Theorem Proving

An history more than sixty years long [CG01, Wan96].

Two major lines of research in GATP:

- Synthetic methods;
- Algebraic methods.

Mechanical Geometric Formula Derivation:

- Finding locus equation;
- Deriving geometry formulas.

News fields

- Geometric Tools: DGS/GATP/CAS/RGP/eLearning;
- Formalisation.

# AI (synthetic) Methods

Synthetic methods attempt to automate traditional geometry proof methods that produce human-readable proofs.

In 1950s Gelernter created a theorem prover that could find solutions to a number of problems taken from high-school textbooks in plane geometry [Gel59].

It was based on the human simulation approach and has been considered a landmark in the AI area for this time.

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

# AI (synthetic) Methods

Synthetic methods attempt to automate traditional geometry proof methods that produce human-readable proofs.

In 1950s Gelernter created a theorem prover that could find solutions to a number of problems taken from high-school textbooks in plane geometry [Gel59].

It was based on the human simulation approach and has been considered a landmark in the AI area for this time.

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

# 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 [Cho87];
- Gröbner bases method [Buc06, BCJ$^+$06, Kap86].

Decision procedures.

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

# 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 [Cho87];
- Gröbner bases method [Buc06, BCJ+06, Kap86].

Decision procedures.

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

# 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 [Cho87];
- Gröbner bases method [Buc06, BCJ$^{+}$06, Kap86].

Decision procedures.

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

# Coordinate-free Methods

Instead of coordinates, three basic geometric quantities: the ratio of parallel line segments, the signed area, and the Pythagorean difference.

- Area method [CGZ93, JNQ11, QJ06a];
- Full angle method [CGZ94, CGZ96a];
- Solid geometry [CGZ95].

Geometric proofs, small and human-readable.

But:

- not the "normal" high-school geometric reasoning;
- for many conjectures these methods still deal with extremely complex expressions involving certain geometric quantities.

# Coordinate-free Methods

Instead of coordinates, three basic geometric quantities: the ratio of parallel line segments, the signed area, and the Pythagorean difference.

- Area method [CGZ93, JNQ11, QJ06a];
- Full angle method [CGZ94, CGZ96a];
- Solid geometry [CGZ95].

Geometric proofs, small and human-readable.

But:

- not the "normal" high-school geometric reasoning;
- for many conjectures these methods still deal with extremely complex expressions involving certain geometric quantities.

# Coordinate-free Methods

Instead of coordinates, three basic geometric quantities: the ratio of parallel line segments, the signed area, and the Pythagorean difference.

- Area method [CGZ93, JNQ11, QJ06a];
- Full angle method [CGZ94, CGZ96a];
- Solid geometry [CGZ95].

Geometric proofs, small and human-readable.

But:

- not the “normal” high-school geometric reasoning;
- for many conjectures these methods still deal with extremely complex expressions involving certain geometric quantities.

# Other 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].
- 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 [SPJ10].
- Probabilistic verification of elementary geometry statements [CFGG97, RGK99].
- Visual Reasoning/Proofs [Kim89, YCG10a, YCG10b].

# Other 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].
- 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 [SPJ10].
- Probabilistic verification of elementary geometry statements [CFGG97, RGK99].
- Visual Reasoning/Proofs [Kim89, YCG10a, YCG10b].

# Other 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].
- 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 [SPJ10].
- Probabilistic verification of elementary geometry statements [CFGG97, RGK99].
- Visual Reasoning/Proofs [Kim89, YCG10a, YCG10b].

# Other 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].
- 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 [SPJ10].
- Probabilistic verification of elementary geometry statements [CFGG97, RGK99].
- Visual Reasoning/Proofs [Kim89, YCG10a, YCG10b].

# Other 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].
- 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 [SPJ10].
- Probabilistic verification of elementary geometry statements [CFGG97, RGK99].
- Visual Reasoning/Proofs [Kim89, YCG10a, YCG10b].

ALAN TURING YEAR 2012

# Mechanical Geometric Formula Derivation

- 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.

- Deriving Geometry Formulas: automatic derivation of geometry formulas [cCsG90, KSY94, RV99].

  Example: find the formula for the area of a triangle $ABC$ in terms of its three sides.

# Mechanical Geometric Formula Derivation

- 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.

- Deriving Geometry Formulas: automatic derivation of geometry formulas [cCsG90, KSY94, RV99].

  Example: find the formula for the area of a triangle $ABC$ in terms of its three sides.

# Geometric Tools & Integration Issues

Geometric tools: DGS & GATP & CAS & RGP.

- DGS - Dynamic Geometry Software [Gro11, Hoh02, Jac01, Jan06, RGK99] - "visual proofs" [CGY04];
- GATP - Geometry Automated Theorem Provers
  - verification of the soundness of a geometric construction [JQ07].
  - reason about a given DGS construction [CGZ96b, JQ06, Nar07a, QP06, QJ06b].
  - human-readable proofs [JNQ11, QJ06a, QJ09].
- RGP - Repositories of Geometric Problems [QJ07, Qua11].
- eLearning [ABY86, HLY86, QJ06b, SQ08, SQ10, SQ12]

Integration: Intergeo Project [SHK+10] — Deducation STREP Proposal [WSA+12].

# Geometric Tools & Integration Issues

Geometric tools: DGS & GATP & CAS & RGP.

- DGS - Dynamic Geometry Software [Gro11, Hoh02, Jac01, Jan06, RGK99] - "visual proofs" [CGY04];
- GATP - Geometry Automated Theorem Provers
  - verification of the soundness of a geometric construction [JQ07].
  - reason about a given DGS construction [CGZ96b, JQ06, Nar07a, QP06, QJ06b].
  - human-readable proofs [JNQ11, QJ06a, QJ09].
- RGP - Repositories of Geometric Problems [QJ07, Qua11].
- eLearning [ABY86, HLY86, QJ06b, SQ08, SQ10, SQ12]

Integration: Intergeo Project [SHK+10] — Deducation STREP Proposal [WSA+12].

# Geometric Tools & Integration Issues

Geometric tools: DGS & GATP & CAS & RGP.

- DGS - Dynamic Geometry Software [Gro11, Hoh02, Jac01, Jan06, RGK99] - "visual proofs" [CGY04];
- GATP - Geometry Automated Theorem Provers
  - verification of the soundness of a geometric construction [JQ07].
  - reason about a given DGS construction [CGZ96b, JQ06, Nar07a, QP06, QJ06b].
  - human-readable proofs [JNQ11, QJ06a, QJ09].
- RGP - Repositories of Geometric Problems [QJ07, Qua11].
- eLearning [ABY86, HLY86, QJ06b, SQ08, SQ10, SQ12]

Integration: Intergeo Project [SHK+10] — Deducation STREP Proposal [WSA+12].

# Geometric Tools & Integration Issues

Geometric tools: DGS & GATP & CAS & RGP.

- DGS - Dynamic Geometry Software [Gro11, Hoh02, Jac01, Jan06, RGK99] - "visual proofs" [CGY04];
- GATP - Geometry Automated Theorem Provers
  - verification of the soundness of a geometric construction [JQ07].
  - reason about a given DGS construction [CGZ96b, JQ06, Nar07a, QP06, QJ06b].
  - human-readable proofs [JNQ11, QJ06a, QJ09].
- RGP - Repositories of Geometric Problems [QJ07, Qua11].
- eLearning [ABY86, HLY86, QJ06b, SQ08, SQ10, SQ12]

Integration: Intergeo Project [SHK+10] — Deducation STREP Proposal [WSA+12].

# Geometric Tools & Integration Issues

Geometric tools: DGS & GATP & CAS & RGP.

- DGS - Dynamic Geometry Software [Gro11, Hoh02, Jac01, Jan06, RGK99] - "visual proofs" [CGY04];
- GATP - Geometry Automated Theorem Provers
  - verification of the soundness of a geometric construction [JQ07].
  - reason about a given DGS construction [CGZ96b, JQ06, Nar07a, QP06, QJ06b].
  - human-readable proofs [JNQ11, QJ06a, QJ09].
- RGP - Repositories of Geometric Problems [QJ07, Qua11].
- eLearning [ABY86, HLY86, QJ06b, SQ08, SQ10, SQ12]

Integration: Intergeo Project [SHK$^{+}$10] — Deducation STREP Proposal [WSA$^{+}$12].

# Formalisation

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

- Hilbert's *Grundlagen* [Hil77, MF03, DDS00];
- Jan von Plato's constructive geometry [Kah95, vP95];
- French high school geometry [Gui04];
- Tarski's geometry [Nar07b];
- An axiom system for compass and ruler geometry [Dup10];
- Projective geometry [MNS09, MNS10];
- Area Method [JNQ11, Nar06];
- Algebraic methods in geometry [MPPJ12].

# Synthetic Methods

AI, intelligence simulated by a machine.
AI in Geometry, development of human-readable proofs.

- Geometric reasoning - small and easy to understand proofs.
- Use of predicates only allow reaching fix-points.
- Some successes over the algebraic provers.

Seminal paper of Gelernter et al [Gel59].

- numerical model;
- constructing auxiliary points;
- generating geometric lemmas.

The AI approaches are not decision procedures and are less powerful then the algebraic approaches.

# Gelernter's GATP

Backward chaining approach.

$$\forall_{\text{geometric elements}}[(H_1 \wedge \cdots \wedge H_r) \Rightarrow G]$$

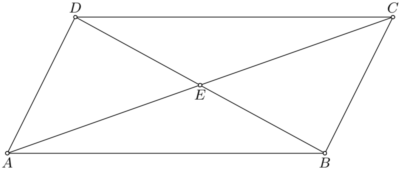
To prove $G$ we search the *axiom rule set* to find a rule of the following form

$$[(G_1 \wedge \cdots \wedge G_r) \Rightarrow G]$$

until the sub-goals is one of the hypothesis.

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

# Example 1 - Gelernter




$$points(A,B,C) \wedge AB \| CD \wedge AD \| BC \wedge coll(E,A,C) \wedge coll(E,B,D) \Rightarrow AE = EC$$

$\top$ ⟶ $AC = CA$

$\top$ ⟶ ($AB \| CD$) $\angle ACD = \angle CAB$

$\top$ ⟶ ($AD \| BC$) $\angle CAD = \angle ACB$

$AC = CA$, $\angle ACD = \angle CAB$, $\angle CAD = \angle ACB$ ⟶ $\Delta ABC \cong \Delta CDA$ ⟶ $AB = CD$

$\top$ ⟶ $\angle AEB = \angle CED$

$\top$ ⟶ ($AB \| CD$) $\angle ACD = \angle CAB$ ⟶ ($coll(E,A,C)$) $\angle ECD = \angle EAB$

$AB = CD$, $\angle AEB = \angle CED$, $\angle ECD = \angle EAB$ ⟶ $\Delta ECD \cong \Delta EAB$ ⟶ $AE = EC$

# GATPs - Synthetic methods

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 implement, the problem of search space explosion still remains and makes the methods of this type highly inefficient [CP79, CP86].

# Geometry Deductive Database

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

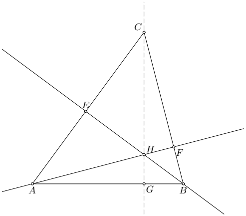
The result program can be used to find fix-points for a geometric configuration, i.e. the program can find all the properties of the configuration that can be deduced using a fixed set of geometric rules.

Generate ndg conditions.

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

# Geometry Deductive Database - The Orthocenter Theorem

$points(A, B, C) \wedge coll(E, A, C) \wedge perp(B, E, A, C) \wedge coll(F, B, C) \wedge perp(A, F, B, C) \wedge coll(H, A, F) \wedge coll(H, B, E) \wedge coll(G, A, B) \wedge coll(G, C, H)$




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) $\rightarrow B(u, v, Ext(u, v, w, x))$
(A4.2) $\rightarrow v \cdot Ext(u, v, w, x) \equiv w \cdot x$

(...)

# 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 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.

# 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 [YCG10a, YCG10b].

# 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: $h_l = O, \ldots, h_s = O, d_1 \neq 0, \ldots, d_t \neq 0;$
- C: c=0.

Proving $T$ is equivalent to deciding whether the formula

$$\forall_{x_l,\ldots x_n}[h_1 = 0 \wedge \cdots \wedge h_s \wedge d_1 \neq 0 \wedge \ldots \wedge d_t \neq 0 \Rightarrow c = 0] \quad (1)$$

is valid.

# Wu's Method

Computes a characteristic set $C$ of $\{h_1, \ldots, h_s\}$ 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 [WT86, Wu00].

This is a decision procedure.

# Area Method — Basic Geometric Quantities[1]

## 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 $\frac{\overline{PQ}}{\overline{AB}}$ 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: $\mathcal{P}_{ABC} = \overline{AB}^2 + \overline{CB}^2 - \overline{AC}^2$.

[1] [JNQ11, QJ06a, QJ09]

# Area Method — Basic Geometric Quantities[1]

## 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 $\frac{\overline{PQ}}{\overline{AB}}$ 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: $\mathcal{P}_{ABC} = \overline{AB}^2 + \overline{CB}^2 - \overline{AC}^2$.

[1] [JNQ11, QJ06a, QJ09]

# Area Method — Basic Geometric Quantities[1]

## 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 $\frac{\overline{PQ}}{\overline{AB}}$ 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: $\mathcal{P}_{ABC} = \overline{AB}^2 + \overline{CB}^2 - \overline{AC}^2$.

[1] [JNQ11, QJ06a, QJ09]

# Area Method — Basic Geometric Quantities[1]

## 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 $\frac{\overline{PQ}}{\overline{AB}}$ 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: $\mathcal{P}_{ABC} = \overline{AB}^2 + \overline{CB}^2 - \overline{AC}^2$.

[1] [JNQ11, QJ06a, QJ09]

# Properties of the Ratio of Directed Parallel Segments

- $\frac{\overline{PQ}}{\overline{AB}} = -\frac{\overline{QP}}{\overline{AB}} = \frac{\overline{QP}}{\overline{BA}} = -\frac{\overline{PQ}}{\overline{BA}}$;
- $\frac{\overline{PQ}}{\overline{AB}} = 0$ iff $P = Q$;
- $(\ldots)$

EL1 (The Co-side Theorem) Let $M$ be the intersection of two non-parallel lines $AB$ and $PQ$ and $Q \neq M$. Then it holds that $\frac{\overline{PM}}{\overline{QM}} = \frac{\mathcal{S}_{PAB}}{\mathcal{S}_{QAB}}$; $\frac{\overline{PM}}{\overline{PQ}} = \frac{\mathcal{S}_{PAB}}{\mathcal{S}_{PAQB}}$; $\frac{\overline{QM}}{\overline{PQ}} = \frac{\mathcal{S}_{QAB}}{\mathcal{S}_{PAQB}}$.

# Properties of the Signed Area

- $\mathcal{S}_{ABC} = \mathcal{S}_{CAB} = \mathcal{S}_{BCA} = -\mathcal{S}_{ACB} = -\mathcal{S}_{BAC} = -\mathcal{S}_{CBA}$.
- $\mathcal{S}_{ABC} = 0$ iff $A$, $B$, and $C$ are collinear.
- $PQ \parallel AB$ iff $\mathcal{S}_{PAB} = \mathcal{S}_{QAB}$, i.e., iff $\mathcal{S}_{PAQB} = 0$.
- Let $ABCD$ be a parallelogram, $P$ and $Q$ be two arbitrary points. Then it holds that $\mathcal{S}_{APQ} + \mathcal{S}_{CPQ} = \mathcal{S}_{BPQ} + \mathcal{S}_{DPQ}$ or $\mathcal{S}_{PAQB} = \mathcal{S}_{PDQC}$.
- Let $R$ be a point on the line $PQ$. Then for any two points $A$ and $B$ it holds that $\mathcal{S}_{RAB} = \frac{\overline{PR}}{\overline{PQ}}\mathcal{S}_{QAB} + \frac{\overline{RQ}}{\overline{PQ}}\mathcal{S}_{PAB}$.
- $(\ldots)$

# Properties of the Pythagoras Difference

- $\mathcal{P}_{AAB} = 0$.
- $\mathcal{P}_{ABC} = \mathcal{P}_{CBA}$.
- If $A$, $B$, and $C$ are collinear then, $\mathcal{P}_{ABC} = 2\overline{BA}\ \overline{BC}$.
- $AB \perp BC$ iff $\mathcal{P}_{ABC} = 0$.
- Let $AB$ and $PQ$ be two non-perpendicular lines, and $Y$ be the intersection of line $PQ$ and the line passing through $A$ and perpendicular to $AB$. Then, it holds that

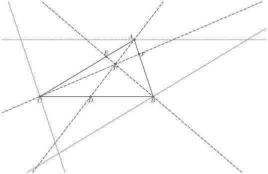
$$\frac{\overline{PY}}{\overline{QY}} = \frac{\mathcal{P}_{PAB}}{\mathcal{P}_{QAB}}, \quad \frac{\overline{PY}}{\overline{PQ}} = \frac{\mathcal{P}_{PAB}}{\mathcal{P}_{PAQB}}, \quad \frac{\overline{QY}}{\overline{PQ}} = \frac{\mathcal{P}_{QAB}}{\mathcal{P}_{PAQB}}.$$

- $(\ldots)$

# 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.



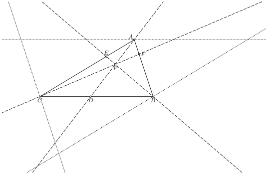

The conclusion is expressed by a polynomial in some geometry quantities (defined above), without any relation to a given system of coordinates.

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

# 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.



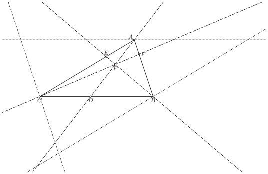

The conclusion is expressed by a polynomial in some geometry quantities (defined above), without any relation to a given system of coordinates.

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

# 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.



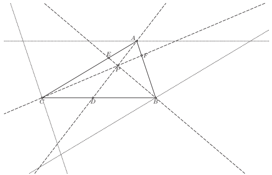

The conclusion is expressed by a polynomial in some geometry quantities (defined above), without any relation to a given system of coordinates.

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

# 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 (LINE U V) and (LINE P Q);
ndg-condition: $UV \nparallel PQ$; $U \neq V$; $P \neq Q$.
degree of freedom for Y: 0

**ECS3** construction of a point Y such that it is a foot from a given point P to (LINE U V); (. . . ).

**ECS4** construction of a point $Y$ on the line passing through point $W$ and parallel to (LINE U V), such that $\overline{WY} = r\overline{UV}$, (. . . ).

**ECS5** construction of a point $Y$ on the line passing through point $U$ and perpendicular to (LINE U V), such that $r = \frac{4\mathcal{S}_{UVY}}{\mathcal{P}_{UVU}}$, (. . . ).

# Forms of Expressing the Conclusion

| property | in terms of geometric quantities |
|---|---|
| points $A$ and $B$ are identical | $\mathcal{P}_{ABA} = 0$ |
| points $A$, $B$, $C$ are collinear | $\mathcal{S}_{ABC} = 0$ |
| $AB$ is perpendicular to $CD$ | $\mathcal{P}_{ABA} \neq 0 \wedge \mathcal{P}_{CDC} \neq 0 \wedge \mathcal{P}_{ACD} = \mathcal{P}_{BCD}$ |
| $AB$ is parallel to $CD$ | $\mathcal{P}_{ABA} \neq 0 \wedge \mathcal{P}_{CDC} \neq 0 \wedge \mathcal{S}_{ACD} = \mathcal{S}_{BCD}$ |
| $O$ is the midpoint of $AB$ | $\mathcal{S}_{ABO} = 0 \wedge \mathcal{P}_{ABA} \neq 0 \wedge \frac{\overline{AO}}{\overline{AB}} = \frac{1}{2}$ |
| $AB$ has the same length as $CD$ | $\mathcal{P}_{ABA} = \mathcal{P}_{CDC}$ |
| points $A$, $B$, $C$, $D$ are harmonic | $\mathcal{S}_{ABC} = 0 \wedge \mathcal{S}_{ABD} = 0 \wedge \mathcal{P}_{BCB} \neq 0 \wedge \mathcal{P}_{BDB} \neq 0 \wedge \frac{\overline{AC}}{\overline{CB}} = \frac{\overline{DA}}{\overline{DB}}$ |
| angle $ABC$ has the same measure as $DEF$ | $\mathcal{P}_{ABA} \neq 0 \wedge \mathcal{P}_{ACA} \neq 0 \wedge \mathcal{P}_{BCB} \neq 0 \wedge \mathcal{P}_{DED} \neq 0 \wedge \mathcal{P}_{DFD} \neq 0 \wedge \mathcal{P}_{EFE} \neq 0 \wedge \mathcal{S}_{ABC} \cdot \mathcal{P}_{DEF} = \mathcal{S}_{DEF} \cdot \mathcal{P}_{ABC}$ |
| $A$ and $B$ belong to the same circle arc $CD$ | $\mathcal{S}_{ACD} \neq 0 \wedge \mathcal{S}_{BCD} \neq 0 \wedge \mathcal{S}_{CAD} \cdot \mathcal{P}_{CBD} = \mathcal{S}_{CBD} \cdot \mathcal{P}_{CAD}$ |

# Elimination Lemmas

EL2 Let $G(Y)$ be a linear geometric quantity and point $Y$ is introduced by the construction (PRATIO Y W (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 (INTER Y (LINE U V) (LINE P Q). Then it holds

$$G(Y) = \frac{\mathcal{S}_{UPQ} G(V) - \mathcal{S}_{VPQ} G(U)}{\mathcal{S}_{UPVQ}}.$$

- (...)

# Constructive Steps & Elimination Lemmas

| | | Geometric Quantities | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | $\mathcal{P}_{AYB}$ | $\mathcal{P}_{ABY}$ | $\mathcal{P}_{ABCY}$ | $\mathcal{S}_{ABY}$ | $\mathcal{S}_{ABCY}$ | $\frac{\overline{AY}}{\overline{CD}}$ | $\frac{\overline{AY}}{\overline{BY}}$ |
| Constructive Steps | ECS2 | EL5 | EL3 | | | | EL11 | EL1 |
| | ECS3 | EL6 | EL4 | | | | EL12 | |
| | ECS4 | EL7 | EL2 | | | | EL13 | |
| | ECS5 | EL10 | EL9 | | EL8 | | EL14 | |
| | | 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 $S$.

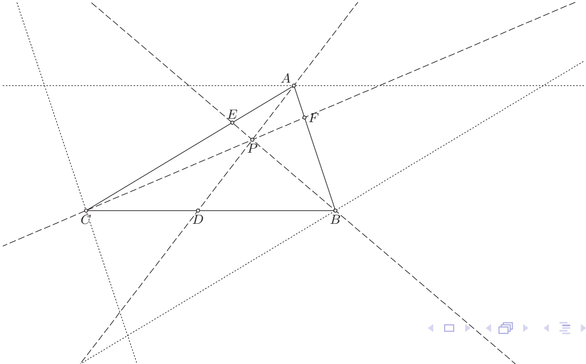
```
for (i=m;i==1;i--) {
   if (the ndg conditions of Ci is satisfied) exit;
   // Let G1,\ldots,Gn be the geometric quantities in E and F
   for (j=1;j<=n,j++) {
      Hj <- eliminating the point introduced
                      by construction Ci from Gj
      E <- E[Gj:=Hj]
      F <- F[Gj:=Hj]
   }
}
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:

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



# 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.

$$
\begin{aligned}
\frac{\overline{AF}}{\overline{FB}}\frac{\overline{BD}}{\overline{DC}}\frac{\overline{CE}}{\overline{EA}} &= \frac{S_{APC}}{S_{BCP}}\frac{\overline{BD}}{\overline{DC}}\frac{\overline{CE}}{\overline{EA}} && \text{the point } F \text{ is eliminated}\\
&= \frac{S_{APC}}{S_{BCP}}\frac{S_{BPA}}{S_{CAP}}\frac{\overline{CE}}{\overline{EA}} && \text{the point } D \text{ is eliminated}\\
&= \frac{S_{APC}}{S_{BCP}}\frac{S_{BPA}}{S_{CAP}}\frac{S_{CPB}}{S_{ABP}} && \text{the point } E \text{ is eliminated}\\
&= 1
\end{aligned}
$$

Elimination Steps: 3; Geometric Steps: 6; Algebraic Steps: 23; Total Steps: 32; CPU Time 0.004s. The GATP also provide the ndg conditions.

# 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.

$$
\begin{aligned}
\frac{\overline{AF}}{\overline{FB}}\frac{\overline{BD}}{\overline{DC}}\frac{\overline{CE}}{\overline{EA}} &= \frac{S_{APC}}{S_{BCP}}\frac{\overline{BD}}{\overline{DC}}\frac{\overline{CE}}{\overline{EA}} && \text{the point } F \text{ is eliminated}\\
&= \frac{S_{APC}}{S_{BCP}}\frac{S_{BPA}}{S_{CAP}}\frac{\overline{CE}}{\overline{EA}} && \text{the point } D \text{ is eliminated}\\
&= \frac{S_{APC}}{S_{BCP}}\frac{S_{BPA}}{S_{CAP}}\frac{S_{CPB}}{S_{ABP}} && \text{the point } E \text{ is eliminated}\\
&= 1
\end{aligned}
$$

Elimination Steps: 3; Geometric Steps: 6; Algebraic Steps: 23; Total Steps: 32; CPU Time 0.004s. The GATP also provide the ndg conditions.

# 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.

$$
\begin{aligned}
\frac{\overline{AF}}{\overline{FB}}\frac{\overline{BD}}{\overline{DC}}\frac{\overline{CE}}{\overline{EA}} &= \frac{\mathcal{S}_{APC}}{\mathcal{S}_{BCP}}\frac{\overline{BD}}{\overline{DC}}\frac{\overline{CE}}{\overline{EA}} && \text{the point } F \text{ is eliminated}\\
&= \frac{\mathcal{S}_{APC}}{\mathcal{S}_{BCP}}\frac{\mathcal{S}_{BPA}}{\mathcal{S}_{CAP}}\frac{\overline{CE}}{\overline{EA}} && \text{the point } D \text{ is eliminated}\\
&= \frac{\mathcal{S}_{APC}}{\mathcal{S}_{BCP}}\frac{\mathcal{S}_{BPA}}{\mathcal{S}_{CAP}}\frac{\mathcal{S}_{CPB}}{\mathcal{S}_{ABP}} && \text{the point } E \text{ is eliminated}\\
&= 1
\end{aligned}
$$

Elimination Steps: 3; Geometric Steps: 6; Algebraic Steps: 23; Total Steps: 32; CPU Time 0.004s. The GATP also provide the ndg conditions.

# 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.

$$
\begin{aligned}
\frac{\overline{AF}}{\overline{FB}}\frac{\overline{BD}}{\overline{DC}}\frac{\overline{CE}}{\overline{EA}} &= \frac{\mathcal{S}_{APC}}{\mathcal{S}_{BCP}}\frac{\overline{BD}}{\overline{DC}}\frac{\overline{CE}}{\overline{EA}} && \text{the point } F \text{ is eliminated}\\
&= \frac{\mathcal{S}_{APC}}{\mathcal{S}_{BCP}}\frac{\mathcal{S}_{BPA}}{\mathcal{S}_{CAP}}\frac{\overline{CE}}{\overline{EA}} && \text{the point } D \text{ is eliminated}\\
&= \frac{\mathcal{S}_{APC}}{\mathcal{S}_{BCP}}\frac{\mathcal{S}_{BPA}}{\mathcal{S}_{CAP}}\frac{\mathcal{S}_{CPB}}{\mathcal{S}_{ABP}} && \text{the point } E \text{ is eliminated}\\
&= 1
\end{aligned}
$$

Elimination Steps: 3; Geometric Steps: 6; Algebraic Steps: 23; Total Steps: 32; CPU Time 0.004s. The GATP also provide the ndg conditions.

# 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.

$$
\begin{aligned}
\frac{\overline{AF}}{\overline{FB}}\frac{\overline{BD}}{\overline{DC}}\frac{\overline{CE}}{\overline{EA}} &= \frac{\mathcal{S}_{APC}}{\mathcal{S}_{BCP}}\frac{\overline{BD}}{\overline{DC}}\frac{\overline{CE}}{\overline{EA}} && \text{the point } F \text{ is eliminated}\\
&= \frac{\mathcal{S}_{APC}}{\mathcal{S}_{BCP}}\frac{\mathcal{S}_{BPA}}{\mathcal{S}_{CAP}}\frac{\overline{CE}}{\overline{EA}} && \text{the point } D \text{ is eliminated}\\
&= \frac{\mathcal{S}_{APC}}{\mathcal{S}_{BCP}}\frac{\mathcal{S}_{BPA}}{\mathcal{S}_{CAP}}\frac{\mathcal{S}_{CPB}}{\mathcal{S}_{ABP}} && \text{the point } E \text{ is eliminated}\\
&= 1
\end{aligned}
$$

Elimination Steps: 3; Geometric Steps: 6; Algebraic Steps: 23; Total Steps: 32; CPU Time 0.004s. The GATP also provide the ndg conditions.

# 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.

$$\begin{aligned}
\frac{\overline{AF}}{\overline{FB}}\frac{\overline{BD}}{\overline{DC}}\frac{\overline{CE}}{\overline{EA}} &= \frac{\mathcal{S}_{APC}}{\mathcal{S}_{BCP}}\frac{\overline{BD}}{\overline{DC}}\frac{\overline{CE}}{\overline{EA}} && \text{the point } F \text{ is eliminated}\\
&= \frac{\mathcal{S}_{APC}}{\mathcal{S}_{BCP}}\frac{\mathcal{S}_{BPA}}{\mathcal{S}_{CAP}}\frac{\overline{CE}}{\overline{EA}} && \text{the point } D \text{ is eliminated}\\
&= \frac{\mathcal{S}_{APC}}{\mathcal{S}_{BCP}}\frac{\mathcal{S}_{BPA}}{\mathcal{S}_{CAP}}\frac{\mathcal{S}_{CPB}}{\mathcal{S}_{ABP}} && \text{the point } E \text{ is eliminated}\\
&= 1
\end{aligned}$$

Elimination Steps: 3; Geometric Steps: 6; Algebraic Steps: 23; Total Steps: 32; CPU Time 0.004s. The GATP also provide the ndg conditions.

# Area Method - Formalisation

Formalisation [JNQ11, Nar06, Nar09];

1. $\overline{AB} = 0$ if and only if the points $A$ and $B$ are identical
2. $\mathcal{S}_{ABC} = \mathcal{S}_{CAB}$
3. $\mathcal{S}_{ABC} = -\mathcal{S}_{BAC}$
4. If $\mathcal{S}_{ABC} = 0$ then $\overline{AB} + \overline{BC} = \overline{AC}$ (Chasles's axiom)
5. There are points $A$, $B$, $C$ such that $\mathcal{S}_{ABC} \neq 0$ (dimension; not all points are collinear)
6. $\mathcal{S}_{ABC} = \mathcal{S}_{DBC} + \mathcal{S}_{ADC} + \mathcal{S}_{ABD}$ (dimension; all points are in the same plane)
7. For each element $r$ of $F$, there exists a point $P$, such that $\mathcal{S}_{ABP} = 0$ and $\overline{AP} = r\overline{AB}$ (construction of a point on the line)
8. If $A \neq B, \mathcal{S}_{ABP} = 0, \overline{AP} = r\overline{AB}, \mathcal{S}_{ABP'} = 0$ and $\overline{AP'} = r\overline{AB}$, then $P = P'$ (unicity)
9. If $PQ \parallel CD$ and $\frac{\overline{PQ}}{\overline{CD}} = 1$ then $DQ \parallel PC$ (parallelogram)
10. If $\mathcal{S}_{PAC} \neq 0$ and $\mathcal{S}_{ABC} = 0$ then $\frac{\overline{AB}}{\overline{AC}} = \frac{\mathcal{S}_{PAB}}{\mathcal{S}_{PAC}}$ (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 $AB \parallel EF$ then $EF \perp CD$
13. If $FA \perp BC$ and $\mathcal{S}_{FBC} = 0$ then $4\mathcal{S}_{ABC}^2 = \overline{AF}^2\overline{BC}^2$ (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 [Tea09]), demonstrating the correctness of the system and eliminating all concerns about provability of the lemmas [Nar09].

# Full Angle Method/Solid Geometry

**Full Angle Method** Full Angle is defined as an ordered pair of line which satisfies the following rules (...) [CGZ96a].

**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].

# Coherent Logic GATP

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

$$A_1(x) \wedge \ldots \wedge A_n(x) \rightarrow \exists_{y_1} B(x, y_1) \vee \ldots \vee \exists_{y_m} B(x, y_m)$$

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

ArgoCLP (**C**oherent **L**ogic **P**rover of the Argo Group[2])

- new proof procedures;
- proof trace exportable to:
  - a proof object in Isabelle/Isar;
  - human readable (English/LaTeX).

not aimed at proving complex geometry theorems but rather at proving foundational theorems (close to the axiom level) [SPJ10].

[2] `http://argo.matf.bg.ac.rs/`

# Finding locus equations[3]

For most DGS a locus is basically a set of points in the screen with no algebraic information.

- Numerical approach, based on interpolation (Cinderella, Cabri) [Bot02].
- Symbolic method, finding the equation of a locus [BL02, BA12].

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

[3] [BAE07, BA12]

# Loci Finding — Algorithm

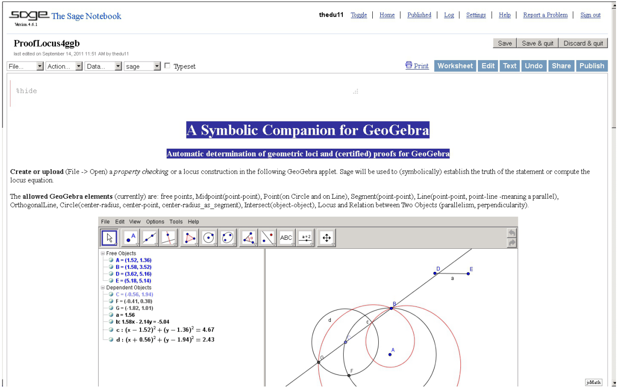
A statement is considered where the conclusion does not follow from the hypotheses.

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.

# Locus Finding — Implementation

A *Sage* worksheet integrating *GeoGebra*



# 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;
- 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].

# Integration — DGSs & GATPs

- GCLC/WinGCLC - A DGS tool that integrates three GATPS: Area Method, Wu's Method and Gröbner Bases Method [Pre08, JQ06, Jan06].
- Theorema Project - Theorema is a project that aims at supporting the entire process of mathematical theory exploration within one coherent logic and software system [BCJ+06]. Implementation of the Area Method Method[Rob02, Rob07].
- 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].

Others: The Geometry Tutor, Mentoniezh, Defi, Chypre, Cabri-Euclide, Geometrix, Baghera, MMP-Geometer, Geometry Explorer, Cinderella.

# Integration — DGSs & GATPs

- GCLC/WinGCLC - A DGS tool that integrates three GATPS: Area Method, Wu's Method and Gröbner Bases Method [Pre08, JQ06, Jan06].
- Theorema Project - Theorema is a project that aims at supporting the entire process of mathematical theory exploration within one coherent logic and software system [BCJ$^{+}$06]. Implementation of the Area Method Method[Rob02, Rob07].
- 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].

Others: The Geometry Tutor, Mentoniezh, Defi, Chypre, Cabri-Euclide, Geometrix, Baghera, MMP-Geometer, Geometry Explorer, Cinderella.

# Integration — DGSs & GATPs

- GCLC/WinGCLC - A DGS tool that integrates three GATPS: Area Method, Wu's Method and Gröbner Bases Method [Pre08, JQ06, Jan06].
- Theorema Project - Theorema is a project that aims at supporting the entire process of mathematical theory exploration within one coherent logic and software system [BCJ+06]. Implementation of the Area Method Method[Rob02, Rob07].
- 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].

Others: The Geometry Tutor, Mentoniezh, Defi, Chypre, Cabri-Euclide, Geometrix, Baghera, MMP-Geometer, Geometry Explorer, Cinderella.

# Integration — DGSs & GATPs

- GCLC/WinGCLC - A DGS tool that integrates three GATPS: Area Method, Wu's Method and Gröbner Bases Method [Pre08, JQ06, Jan06].
- Theorema Project - Theorema is a project that aims at supporting the entire process of mathematical theory exploration within one coherent logic and software system [BCJ+06]. Implementation of the Area Method Method[Rob02, Rob07].
- 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].

Others: The Geometry Tutor, Mentoniezh, Defi, Chypre, Cabri-Euclide, Geometrix, Baghera, MMP-Geometer, Geometry Explorer, Cinderella.

# Integration — DGSs & GATPs

- GCLC/WinGCLC - A DGS tool that integrates three GATPS: Area Method, Wu's Method and Gröbner Bases Method [Pre08, JQ06, Jan06].
- Theorema Project - Theorema is a project that aims at supporting the entire process of mathematical theory exploration within one coherent logic and software system [BCJ+06]. Implementation of the Area Method Method[Rob02, Rob07].
- 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].

Others: The Geometry Tutor, Mentoniezh, Defi, Chypre, Cabri-Euclide, Geometrix, Baghera, MMP-Geometer, Geometry Explorer, Cinderella.

# Integration/eLearning (DGSs & GATPs & RGPs)

Modular approach: I2G & I2GATP formats.

WebGeometryLab: a Web environment incorporating a DGS (GATPs) and a repository of geometric problems, that can be used in a synchronous and asynchronous fashion and with some adaptative and collaborative features [QJ06b, SQ08, SQ10, SQ12].

Deducation: Deductive Framework for Math-oriented Collaborative Teaching Environments — STREP (ICT Call 8: FP7-ICT-2011-8), January 17, 2012 — This research project will extend an existing interface for modern theorem proving systems to an implementation platform for domain-specific personal research and teaching environments [WSA+12].

# Integration/eLearning (DGSs & GATPs & RGPs)

Modular approach: I2G & I2GATP formats.

WebGeometryLab: a Web environment incorporating a DGS (GATPs) and a repository of geometric problems, that can be used in a synchronous and asynchronous fashion and with some adaptative and collaborative features [QJ06b, SQ08, SQ10, SQ12].

Deducation: Deductive Framework for Math-oriented Collaborative Teaching Environments — STREP (ICT Call 8: FP7-ICT-2011-8), January 17, 2012 — This research project will extend an existing interface for modern theorem proving systems to an implementation platform for domain-specific personal research and teaching environments [WSA+12].

# Integration/eLearning (DGSs & GATPs & RGPs)

Modular approach: I2G & I2GATP formats.

WebGeometryLab: a Web environment incorporating a DGS (GATPs) and a repository of geometric problems, that can be used in a synchronous and asynchronous fashion and with some adaptative and collaborative features [QJ06b, SQ08, SQ10, SQ12].

Deducation: Deductive Framework for Math-oriented Collaborative Teaching Environments — STREP (ICT Call 8: FP7-ICT-2011-8), January 17, 2012 — This research project will extend an existing interface for modern theorem proving systems to an implementation platform for domain-specific personal research and teaching environments [WSA+12].

# 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].

# 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].

# 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].

# TGTP[4]

A comprehensive and easily accessible, library of GATP test problems.

- 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 $\mapsto$ the I2GATP common format, an extension of the I2G format [SHK+10] to accommodate the geometric conjectures [Qua12, WSA+12].

[4] [Qua11]

# What to Do?

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

**System Construction** 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. [ADMAv]

**Formalisation** formalising geometric theories and methods.

# What to Do?

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

**System Construction** 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. [ADMAv]

**Formalisation** formalising geometric theories and methods.

# What to Do?

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

System Construction 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. [ADMAv]

Formalisation formalising geometric theories and methods.

# What to Do?

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

**System Construction** 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. [ADMAv]

**Formalisation** formalising geometric theories and methods.

# What to Do?

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

**System Construction** 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. [ADMAv]

**Formalisation** formalising geometric theories and methods.

# What to Do?

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

**System Construction** 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. [ADMAv]

**Formalisation** formalising geometric theories and methods.

# Bibliography I

J. R. Anderson, C. F. Boyle, and G. Yost.
The geometry tutor.
*The Journal of Mathematical Behavior*, pages 5–20, 1986.

Jeremy Avigad, Edward Dean, and John Mumma.
A formal system for Euclid's elements.
*The Review of Symbolic Logic,*, 2:700–768, 2009Av.

Francisco Botana and Miguel A. Abánades.
Automatic deduction in dynamic geometry using sage.
In *THedu'11, CTP Components for Educational Software (postproceedings)*, 2012.

Francisco Botana, Miguel A. Abánades, and Jesús Escribano.
Computing locus equations for standard dynamic geometry environments.
In Yong Shi, G. Dick van Albada, Jack Dongarra, and Peter M. A. Sloot, editors, *International Conference on Computational Science (2)*, volume 4488 of *Lecture Notes in Computer Science*, pages 227–234. Springer, 2007.

Marc Bezem and Thierry Coquand.
Automating coherent logic.
In Geoff Sutcliffe and Andrei Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning*, volume 3835 of *Lecture Notes in Computer Science*, pages 246–260. Springer Berlin / Heidelberg, 2005.

# Bibliography II

B. Buchberger, A. Craciun, T. Jebelean, L. Kovacs, T. Kutsia, K. Nakagawa, F. Piroi, N. Popov, J. Robu, M. Rosenkranz, and W. Windsteiger.
Theorema: Towards computer-aided mathematical theory exploration.
*Journal of Applied Logic*, 4(4):470–504, 2006.

Philippe Balbiani and Luis del Cerro.
Affine geometry of collinearity and conditional term rewriting.
In Hubert Comon and Jean-Pierre Jounnaud, editors, *Term Rewriting*, volume 909 of *Lecture Notes in Computer Science*, pages 196–213. Springer Berlin / Heidelberg, 1995.
10.1007/3-540-59340-3_14.

Francisco Botana and José L. Valcarce.
A dynamic-symbolic interface for geometric theorem discovery.
*Computers and Education*, 38:21–35, 2002.

Francisco Botana.
Interactive versus symbolic approaches to plane loci generation in dynamic geometry environments.
In Peter Sloot, Alfons Hoekstra, C. Tan, and Jack Dongarra, editors, *Computational Science — ICCS 2002*, volume 2330 of *Lecture Notes in Computer Science*, pages 211–218. Springer Berlin / Heidelberg, 2002.
10.1007/3-540-46080-2_22.

B. Buchberger.
*An Algorithm for Finding the Basis Elements in the Residue Class Ring Modulo a Zero Dimensional Polynomial Ideal.*
PhD thesis, 3 2006.

# Bibliography III

Shang ching Chou and Xiao shan Gao.
Mechanical formula derivation of elementary geometries.
In *PROC. INTL. SYMP. ON SYMBOLIC AND ALGEBRAIC COMPUTATION (ISSAC*, pages 265–270, 1990.

Giuseppa Carrá Ferro, Giovanni Gallo, and Rosario Gennaro.
Probabilistic verification of elementary geometry statements.
In Dongming Wang, editor, *Automated Deduction in Geometry*, volume 1360 of *Lecture Notes in Computer Science*, pages 87–101. Springer Berlin / Heidelberg, 1997.
10.1007/BFb0022721.

Shang-Ching Chou and Xiao-Shan Gao.
Automated reasoning in geometry.
In John Alan Robinson and Andrei Voronkov, editors, *Handbook of Automated Reasoning*, pages 707–749. Elsevier and MIT Press, 2001.

Shang-Ching Chou, Xiao-Shan Gao, and Zheng Ye.
Java geometry expert.
http://www.cs.wichita.edu/~ye/, 2004.

Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang.
Automated production of traditional proofs for constructive geometry theorems.
In Moshe Vardi, editor, *Proceedings of the Eighth Annual IEEE Symposium on Logic in Computer Science LICS*, pages 48–56. IEEE Computer Society Press, June 1993.

# Bibliography IV


Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang.
*Machine Proofs in Geometry*.
World Scientific, 1994.

Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang.
Automated production of traditional proofs in solid geometry.
*Journal of Automated Reasoning*, 14:257–291, 1995.

Shang-Ching Chou, Xiao-Shan Gao, and Ji Zhang.
Automated generation of readable proofs with geometric invariants, II. theorem proving with full-angles.
*Journal of Automated Reasoning*, 17:349–370, 1996.

Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang.
Automated generation of readable proofs with geometric invariants, I. multiple and shortest proof generation.
*Journal of Automated Reasoning*, 17:325–347, 1996.

Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang.
A deductive database approach to automated geometry theorem proving and discovering.
*Journal of Automated Reasoning*, 25:219–246, 2000.

Shang-Ching Chou.
*Mechanical Geometry Theorem Proving*.
D. Reidel Publishing Company, Dordrecht, 1987.

# Bibliography V

H. Coelho and L. M. Pereira.
Geom: A prolog egeometry 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.
*Journal of Automated Reasoning*, 2(4):329–390, 1986.

Christophe Dehlinger, Jean-François Dufourd, and Pascal Schreck.
Higher-order intuitionistic formalization and proofs in Hilbert's elementary geometry.
In Dongming Wang Jürgen Richter-Gebert, editor, *Proceedings of Automated Deduction in Geometry (ADG00)*, volume 2061 of *Lecture Notes in Computer Science*, pages 306–324, 2000.

Jean Duprat.
The Euclid's Plane : Formalization and Implementation in Coq.
In *Proceedings of ADG'10*, 2010.

Jesús Escribano, Francisco Botana, and Miguel A. Abánades.
Adding remote computational capabilities to dynamic geometry systems.
*Mathematics and Computers in Simulation*, 80(6):1177 – 1184, 2010.
¡ce:title¿Fifth IMACS Seminar on Monte Carlo Methods¡/ce:title¿ ¡ce:title¿Applications of Computer Algebra 2007 (AC A 2007) special session on Nonstandard Applications of Computer Algebra¡/ce:title¿ ¡ce:title¿Computational Biomechanics and Biology, a collection of papers presented at the 1st IMACS International Conference on the Computational Biomechanics and Biology ICCBB 2007¡/ce:title¿.

# Bibliography VI


H. Gelernter.
Realization of a geometry theorem proving machine.
In *Proceedings of the International Conference Information Processing*, pages 273–282, Paris, June 15-20 1959.

Paul C. Gilmore.
An examination of the geometry theorem machine.
*Artif. Intell.*, 1(3):171–187, 1970.

René Grothmann.
About c.a.r.
`http://compute.ku-eichstaett.de/MGF/wikis/caruser/doku.php?id=history`, 2011.

Frédérique Guilhot.
Formalisation en Coq d'un cours de géométrie pour le lycée.
In *Journées Francophones des Langages Applicatifs*, Janvier 2004.

David Hilbert.
*Foundations of Geometry*.
Open Court Publishing, 1977.
10th Revised edition. Editor: Paul Barnays.

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

# Bibliography VII


M Hohenwarter.
Geogebra - a software system for dynamic geometry and algebra in the plane.
Master's thesis, University of Salzburg, Austria, 2002.

N Jackiw.
*The Geometer's Sketchpad v4.0.*
Key Curriculum Press, 2001.

Predrag Janičić.
GCLC — a tool for constructive euclidean geometry and more than that.
*Lecture Notes in Computer Science*, 4151:58–73, 2006.

P. Janičić, J. Narboux, and P. Quaresma.
The Area Method: a recapitulation.
*Journal of Automated Reasoning*, (to appear), 2011.

Predrag Janičić and Pedro Quaresma.
System Description: GCLCprover + GeoThms.
In Ulrich Furbach and Natarajan Shankar, editors, *Automated Reasoning*, volume 4130 of *LNAI*, pages 145–150. Springer, 2006.

Predrag Janičić and Pedro Quaresma.
Automatic verification of regular constructions in dynamic geometry systems.
In *Automated Deduction in Geometry*, number 4869 in LNAI, pages 39–51, Berlin / Heidelberg, 2007. Springer.
6th International Workshop, ADG 2006, Pontevedra, Spain, August 31-September 2, 2006. Revised Papers.

# Bibliography VIII


Kenneth R. Koedinger and John R. Anderson.
Abstract planning and perceptual chunks: Elements of expertise in geometry.
*Cognitive Science*, 14(4):511–550, 1990.

Gilles Kahn.
Constructive geometry according to Jan von Plato.
Coq contribution, 1995.
Coq V5.10.

Deepak Kapur.
Using Gröbner bases to reason about geometry problems.
*Journal of Symbolic Computation*, 2(4):399–408, 1986.

Michelle Y. Kim.
Visual reasoning in geometry theorem proving.
In *IJCAI*, pages 1617–1622, 1989.

Deepak Kapur, Tushar Saxena, and Lu Yang.
Algebraic and geometric reasoning using dixon resultants.
In *IN ACM ISSAC 94*, pages 99–107. Press, 1994.

Laura Meikle and Jacques Fleuriot.
Formalizing Hilbert's Grundlagen in Isabelle/Isar.
In David A. Basin and Burkhart Wolff, editors, *Theorem Proving in Higher Order Logics*, volume 2758 of *Lecture Notes in Computer Science*, pages 319–334. Springer-Verlag, 2003.

# Bibliography IX


Nicolas Magaud, Julien Narboux, and Pascal Schreck.
Formalizing Desargues' Theorem in Coq using Ranks.
In Sung Y. Shin and Sascha Ossowski, editors, *SAC*, pages 1110–1115. ACM, 2009.

Nicolas Magaud, Julien Narboux, and Pascal Schreck.
Formalizing Projective Plane Geometry in Coq.
In *Post-Proceedings of ADG'08*, Lecture Notes in Artifical Intelligence, 2010.
to appear.

Filip Marić, Ivan Petrović, Danijela Petrović, and Predrag Janičić.
Formalization and implementation of algebraic methods in geometry.
In *Post-Proceedings THedu'11, CTP Components for Educational Software*, 2012.

Julien Narboux.
*Formalisation et Automatisation du Raisonnement Géométrique en Coq*.
PhD thesis, Université de Paris Sud, 2006.

Julien Narboux.
A graphical user interface for formal proofs in geometry.
*Journal of Automated Reasoning*, 39:161–180, 2007.

Julien Narboux.
Mechanical theorem proving in Tarski's geometry.
In *Proceedings of Automatic Deduction in Geometry 06*, volume 4869 of *Lecture Notes in Artificial Intelligence*, pages 139–156. Springer-Verlag, 2007.

# Bibliography X


Julien Narboux.
Formalization of the area method.
Coq user contribution, 2009.
http://dpt-info.u-strasbg.fr/~narboux/area_method.html.

Arthur J. Nevins.
Plane geometry theorem proving using forward chaining.
AI Lab memo 303, MIT, Jan 1974.

Lawrence C. Paulson.
*Isabelle: A Generic Theorem Prover*, volume 828 of *LNCS*.
Springer-Verlag, 1994.

Lawrence C. Paulson and Tobias Nipkow.
Isabelle tutorial and user's manual.
Technical Report 189, University of Cambridge, Computer Laboratory, January 1990.

Goran Predović.
Automatsko dokazivanje geomtrijskih teorema primenom vuove i buhbergerove metode.
Magistarska teza, Faculty of Mathemetics, University of Belgrade, 2008.

Pedro Quaresma and Predrag Janičić.
Framework for constructive geometry (based on the area method).
Technical Report 2006/001, Centre for Informatics and Systems of the University of Coimbra, 2006.

# Bibliography XI


Pedro Quaresma and Predrag Janičić.
Integrating dynamic geometry software, deduction systems, and theorem repositories.
In Jonathan M. Borwein and William M. Farmer, editors, *Mathematical Knowledge Management*, volume 4108 of *Lecture Notes in Artificial Intelligence*, pages 280–294. Springer, 2006.

Pedro Quaresma and Predrag Janičić.
Geothms – a Web System for Euclidean Constructive Geometry.
*Electronic Notes in Theoretical Computer Science*, 174(2):35 – 48, 2007.

Pedro Quaresma and Predrag Janičić.
The area method, rigorous proofs of lemmas in hilbert's style axiom system.
Technical Report 2009/006, Centre for Informatics and Systems of the University of Coimbra, 2009.

Pedro Quaresma and Ana Pereira.
Visualização de construções geométricas.
*Gazeta de Matemática*, 151:38–41, Junho 2006.

Art Quaife.
Automated development of tarski's geometry.
*Journal of Automated Reasoning*, 5:97–118, 1989.
10.1007/BF00245024.

Pedro Quaresma.
Thousands of geometric problems for geometric theorem provers (TGTP).
In P. Schreck, J. Narboux, and J. Richter-Gebert, editors, *ADG 2010*, number 6877 in LNAI, pages 168–180, Heidelberg, 2011. Springer.

# Bibliography XII


Pedro Quaresma.
A format for proofs in geometry.
In *Proceedings of the Mathematical Knowledge Management 2012*. Springer, 2012. to be submitted.

Jürgen Richter-Gebert and Ulrich Kortenkamp.
*The Interactive Geometry Software Cinderella*.
Springer, 1999.

Judit Robu.
*Geometry Theorem Proving in the Frame of the Theorema Project.*
PhD thesis, Johannes Kepler Universität, Linz, September 2002.

Judit Robu.
Automated Proof of Geometry Theorems Involving Order Relation in the Frame of the Theorema Project.
In Horia F. Pop, editor, *Knowledge Engineering: Principles and Techniques*, number Special Issue in Studia Universitatis "Babes-Bolyai", Series Informatica, pages 307–315, 2007.

T. Recio and M. P. Vélez.
Automatic discovery of theorems in elementary geometry.
*J. Autom. Reason.*, 23:63–82, July 1999.

E. Santiago, Maxim Hendriks, Yves Kreis, Ulrich Kortenkamp, and Daniel Marquès.
I2G Common File Format Final Version.
Technical Report D3.10, The Intergeo Consortium, 2010.

# Bibliography XIII


Sana Stojanovic, Vesna Pavlovic, and Predrag Janicic.
A coherent logic based geometry theorem prover capable of producing formal and readable proofs.
In *Automated Deduction in Geometry*, pages 201–220, 2010.

Vanda Santos and Pedro Quaresma.
eLearning course for Euclidean Geometry.
In *Proceedings of the 8th IEEE International Conference on Advanced Learning Technologies, July 1st- July 5th, 2008, Santander, Cantabria, Spain*, pages 387–388, 2008.

Vanda Santos and Pedro Quaresma.
*Adaptative Learning Environment for Geometry*, volume Advances in Learning Processes, chapter 5, pages 71–92.
I-Tech Education and Publishing KG, Vienna, Austria, 2010.

Vanda Santos and Pedro Quaresma.
Integrating dgss and gatps in an adaptative and collaborative blended-learning web-environment.
In Pedro Quaresma and Ralph-Johan Back, editors, *First Workshop on CTP Components for Educational Software (THedu'11)*, Electronic Proceedings in Theoretical Computer Science, 2012.

The Coq Development Team.
*The Coq Proof Assistant, Reference Manual, Version 8.2*.
TypiCal Project, Lyon, France, 2009.

Jan von Plato.
The axioms of constructive geometry.
In *Annals of Pure and Applied Logic*, volume 76, pages 169–200, 1995.

# Bibliography XIV


Dongming Wang.
Geometry machines: From ai to smc.
In Jacques Calmet, John Campbell, and Jochen Pfalzgraf, editors, *Artificial Intelligence and Symbolic Mathematical Computation*, volume 1138 of *Lecture Notes in Computer Science*, pages 213–239. Springer Berlin / Heidelberg, 1996.
10.1007/3-540-61732-9_60.

Burkhart Wolff, Pascal Schreck, Serge Autexier, Achim D. Brucker, Wolfgang Schreiner, Pedro Quaresma, Ralph-Johan Back, Predrag Janicic, Christian Gütl, and Markus Hohenwarter.
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.

Wu Wen-Tsun.
Basic principles of mechanical theorem proving in elementary geometries.
*Journal of Automated Reasoning*, 2:221–252, 1986.
10.1007/BF02328447.

W.-T. Wu.
The characteristic set method and its application.
In X.-S. Gao and D. Wang, editors, *Mathematics Mechanization and Applications*, pages 3–41, San Diego, CA, 2000. Academic Press.

Zheng Ye, Shang-Ching Chou, and Xiao-Shan Gao.
Visually dynamic presentation of proofs in plane geometry, part 1.
*J. Autom. Reason.*, 45:213–241, October 2010.

# Bibliography XV

Zheng Ye, Shang-Ching Chou, and Xiao-Shan Gao.
Visually dynamic presentation of proofs in plane geometry, part2.
*J. Autom. Reason.*, 45:243–266, October 2010.