

Visually Dynamic Presentation of Proofs in Plane Geometry

Part 1. Basic Features and the Manual Input Method

Zheng Ye · Shang-Ching Chou · Xiao-Shan Gao

Received: 30 November 2009 / Accepted: 6 December 2009 / Published online: 18 December 2009
© Springer Science + Business Media B.V. 2009

Abstract With dynamic mediums such as computer displays, we propose a new kind of visually dynamic presentation of proofs in plane geometry. In a **single** diagram for the proof, when the proof text goes on step by step with mouse clicks, the related geometry elements in the diagram are added, animated, or deleted dynamically with various visually dynamic effects. It solves not only the problem of identifying geometry elements in the proof text with those in the diagram, but also makes the proof more vividly visualized and intuitive. Our ongoing developing system “Java Geometry Expert” (JGEX) uses two methods to create such visually dynamic presentations: the manual input method and the automatic method. In this first part of the series of our work, we propose the main features of our visually dynamic presentation of proofs and present the manual input method to create such presentations. The manual input method mainly uses mouse clicks to create the dynamic geometry diagram and the proof text.

The work reported here was supported by NSF Grant CCR-0201253.

Zheng Ye is on leave from ZJU and working at Wichita State University.

Z. Ye (✉)

College of Computer Science and Technology, Zhejiang University,
Hangzhou, China
e-mail: yezheng@gmail.com

S.-C. Chou

Department of Computer Science, Wichita State University,
Wichita, KS 67220, USA

X.-S. Gao

KLMM, Institute of Systems Science, The Chinese Academy of Sciences,
Beijing, China

Keywords Geometry theorem proving · Visually dynamic presentation of proof · Dynamic geometry · Unordered geometry · Gelernter · Java Geometry Expert · Morley's theorem · Feuerbach's theorem · Pythagorean theorem · Pedal triangle

1 Introduction

1.1 The Problem with the Presentation of Proofs in Traditional Geometry Books

When we read a proof in a geometry book, we often need to spend time and energy on identifying a geometry element (a point, a line, a segment, an angle, a triangle, a circle, etc.) in the proof text with that in the corresponding diagram. When the same element is mentioned later in the proof text we might spend equal amounts of time and energy on identifying it again in the diagram. When the diagram becomes complicated, e.g., there are over a dozen of points involved in the diagram, the problem becomes serious. Any approach to alleviating the problem is worth exploring.

Geometry textbooks generally alleviate this problem by using two or more diagrams with different marks for angles and segments, and possibly with shadowed areas, e.g., a shadowed triangle, in the diagrams. But this solution is static.

1.2 Our Approach to Solving the Problem

With dynamic mediums such as computer displays, we propose a new kind of presentation of proofs in plane geometry.

In a **single** diagram for the proof, when the proof text goes on step by step with mouse clicks, the related geometry elements in the diagram are added, highlighted, animated, or deleted dynamically with various visually dynamic effects. It solves not only the problem of identifying geometry elements in the proof text with those in the diagram, but also makes the proof more vividly visualized and intuitive. This kind of the representation is impossible 30 years ago because of the low resolution of computer displays and slow CPU speeds. Before going into the details, we shall mention that there exist many excellent materials on visual presentations of proofs in geometry or in mathematics in general in the form of books and webpages. In particular, there are three excellent books on visual proofs in mathematics [1–3] and many webpages at the WWW [4, 5] (<http://www.kangmath.org/maths/animations.html>, <http://mathforum.org/mathtools/>).

However, the most distinctive features of our approach are: visually *dynamic* diagram for the proof (instead of a static diagram or a series of static diagrams) and the *interrelationship* between geometry elements in the proof text and those in the related diagram. These presentations can be easily created with our system JGEX (Java Geometry Expert) and can be saved in the form of either animated GIF files or Java applets.

1.3 Java Geometry Expert

JGEX is based on our previous versions [6–8] of Geometry Expert (GEX) which was originally developed around 1995 under the X-Window Openwin graphics

environment [7]. However, since the later Linux distributions do not support Openwin (SunView), we cannot satisfy the requests for GEX from students, educators, and researchers over the World after 2000. This is why there are other versions of GEX developed and implemented mainly by the third author [6, 9].

JGEX has been rewritten completely in Java with emphasis on its ease of use by high school students, teachers, and researchers in geometric drawing and reasoning. Since Java is platform-independent, users can install JGEX on their PCs, laptops, and workstations. Or they can also use a browser to connect to our server [10] to use the most of its features.

JGEX consists of three parts: the drawing part, the reasoning and proving part, and the part of the visually dynamic presentation of proofs (VDPP). The drawing part is a typical dynamic geometry system. Wu's method, the Gröbner basis method, the full-angle method, and the deductive database method are implemented in the reasoning and proving part.

The part of visually dynamic presentation of proofs makes JGEX most distinctive from other dynamic geometry systems on one side, and from other geometry reasoning systems, including our previous versions of GEX, on the other side. It is based on our work on automated generation of readable proofs accomplished in the past, on the work currently being carried out, and on the work to be done in the future (e.g., automated addition of auxiliary geometry elements, etc).

As the first step, instead of automated generation of visually dynamic presentations of proofs, we implement the manual input method for creating these proofs. On our server [10], there is a collection of over 100 examples created manually with JGEX; we refer it as *Collection* throughout the papers of this series. We collect mainly those examples that do not mix algebraic expressions or computations with the geometry diagrams, except for very simple expressions such as putting a^2 in a square in the diagram of a Pythagoras theorem proof with the area dissection method.

There are three reasons for not mixing algebraic expressions with diagrams in current implementation. First, unlike application-driven examples, our system JGEX provides a general-purpose tool for creating the visually dynamic presentation of a proof (manually or automatically). The manual input method of JGEX can create a proof with various visually dynamic effects. However, it is almost impossible for JGEX to put a free-hand expression such as $\sqrt{1+m^2}$ in any place and in any orientation on computer displays. Second, the visual nature of algebraic expressions or computations is generally not so obvious as geometry diagrams themselves. Finally, and more importantly, visualizations of algebraic expressions and equations should be also dynamic as an integral part of the diagram.

In our opinion, visualizations of simple algebraic expressions or computations are another important topic. There are many elegant webpages with algebraic expression animations, e.g., the webpage [11] for the presentation of the Thales theorem is very elegant. Visualizations of (simple) algebraic computations will be in our future work on automated generation of visual presentations of proofs with our area method [12].

In part 1 of this series, we first introduce the drawing part of JGEX. Then we propose the main features of our visually dynamic presentation of proofs. After that we will introduce the visual presentation part of JGEX with the manual input method. At the end of this paper, we will give some more examples as well as the comparison with various works related to this paper.

2 The Drawing Part of JGEX

There have been very popular and excellent commercial geometry theorem drawing systems such as the *Geometer's Sketchpad* in the US [13], *Cabri* in France [14], and *Cinderella* in Germany [15]. All of them are capable of doing dynamic geometry. Each of them has its own advantages and extends to other areas such as drawing in 3D geometry, etc.

For dynamic geometry, the three commercial geometry drawing systems are more professional than JGEX, however JGEX also has some distinctive features.

2.1 The Input Methods for Geometry Statements

The input methods of JGEX are the further enhancement of the input methods used by GEX. It is similar to that used in the *Geometer's Sketchpad*, but emphasizes on its ease of use by the user.

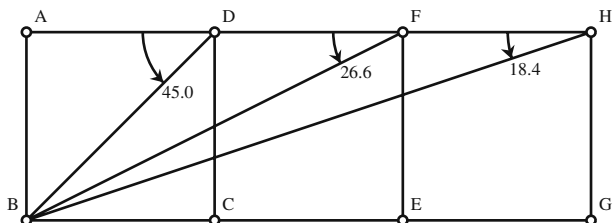
The first method is the graphic input method which allows the user (not asked by the program) to use the mouse to choose points in the diagram pane to draw a geometry diagram. After the diagram is drawn, the text of the hypotheses is generated automatically in the text pane and it can be saved in plain-text or saved in JGEX own format with the “.gex” extension. Then the user can try to prove an assertion on the diagram.

In addition to dynamic geometry nature, JGEX provides other features to speedup or to facilitate the drawing process for users. For example, if a square is wanted, the user only needs to drag a segment to have a square. This process can be cascaded so that it is easy to draw three congruent squares in a row and to specify the following problem that $\angle BDA + \angle BFA + \angle BHA = 90^\circ$ (Fig. 1). The only additional adjustment or modification we did here manually is to move label *B* to left so that it does not overlap with three lines. The whole process took us about 30 s. The angle marks are easy to use. The user only needs to click the two sides of the angle, then the default style is shown with angle measures in degrees.

If a perpendicular line from a point to an existing line with the foot on the line is needed, the user only needs to click the point and drag a segment from it to touch the existing line, then the foot point with a right angle mark are created.

The second input method is the text-based method which was essentially the same as the second author's geometry theorem prover on Lisp Machine. It was developed during 1984–1985 and was extensively used in 1987 to generate over 540 diagrams, including then recently proved Thébault's conjecture, in the book [16].

Fig. 1 A three square problem



With text-based method, the user prepares a geometry statement in plain-text in the predicate form:

$$F : (H_1 \wedge H_2 \wedge \dots \wedge H_r) \Rightarrow C$$

where the formula F is understood universal-quantified, i.e., F is valid for all points involved in the formula, H_1, H_2, \dots, H_r are the hypotheses of equality type, and C is the conclusion. However, formulated literally as this, F is most likely false because of missing nondegenerate conditions which will be discussed later.

2.2 Use of Algebraic Equations

For a diagram generated by JGEX, we keep a set of algebraic equations of the geometry constraints, so that we can prove or disprove properties of it with our algebraic methods efficiently.

Let the set of the polynomial equations corresponding to the hypotheses of the above formula F be

$$(TH) \quad \begin{aligned} h_1(u_1, \dots, u_d, x_1, \dots, x_r) &= 0 \\ h_2(u_1, \dots, u_d, x_1, \dots, x_r) &= 0 \\ &\dots \\ h_r(u_1, \dots, u_d, x_1, \dots, x_r) &= 0 \end{aligned}$$

Then we triangulate¹ the polynomial set so that each polynomial introduces only one new (dependent) variable x_i . Thus the polynomial set is transformed to a triangular form:

$$(TH) \quad \begin{aligned} h'_1(u_1, \dots, u_d, x_1) &= 0 \\ h'_2(u_1, \dots, u_d, x_1, x_2) &= 0 \\ &\dots \\ h'_r(u_1, \dots, u_d, x_1, \dots, x_r) &= 0 \end{aligned}$$

The variables u_1, \dots, u_d can be arbitrarily chosen and once they are fixed, the variables x_1, \dots, x_r can be successively solved. When one or two of the variables u_1, \dots, u_d are changed with the mouse, then the variables of x_1, \dots, x_r are updated to achieve the dynamic geometry effects.

One of advantages of this approach to drawing is that the reasoning with algebraic methods can be immediately applied to the diagram. Currently, we have implemented Wu’s method and the Gröbner basis methods for various formulations (see [16]).

Another important advantage of use of polynomials is that we can construct *all* diagrams or cases with the degrees of leading variables less than or equal to four, because JGEX implements the formulas for quadratic, cubic, and quartic equations.

Example 1 (The Thébault “Conjecture” (Theorem))² Through the vertex A of a triangle ABC a straight line AM is drawn, cutting the side BC in M . Let O and

¹The concept of triangularizing is covered in [16] on page 14.

²For the credit and discussion of the proofs by human or by machine see Page 200 of the book [18] by Roger Penrose.

I be the centers of the circumscribed circle (O) and the *inscribed circle* (I) of ABC . The circle (w_1) and (w_2) with centers w_1 and w_2 are each tangent to (O) and the first is tangent also to two sides of angle AMB , while the second is tangent to the two sides of angle AMC . Prove that the straight line joining w_1 and w_2 passes through I (Fig. 2).

This conjecture was proposed by V. Thébault in 1938 and was solved by K. B. Taylor in 1983. The Taylor solution involves 24 page trigonometrical computations which was omitted in the posted solution by the AMM editor [17]. The machine proof with Wu’s method was given at the end of 1985 by the second author [16]. The machine proof discovers more—our solution indicates that there are 4 lines of collinearity instead of one as Thébault originally proposed. This is one of the most distinctive advantages of Wu’s method as it deals with unordered geometries which will be discussed in the second paper of this series.

In the triangular set of hypotheses (TH) of the Thébault theorem, there are quadratic as well as quartic equations. JGEX can find all solutions of the equations. Figure 2 shows two of the four cases of the Thébault lines:

For each of the four centers I (the incenter and the three excenters) there exist a w_1 and a w_2 such that w_1, I and w_2 are collinear.

2.2.1 Elimination of Duplicated Points

As a direct application of the JGEX algebraic proof engine, we can eliminate a newly constructed point if it is identical to one of the previously constructed points. The three commercial systems can only eliminate this point numerically. Since drawing a geometry diagram is so easy with a few mouse clicks, we repeatedly observed this phenomenon at the early stage of developing JGEX.

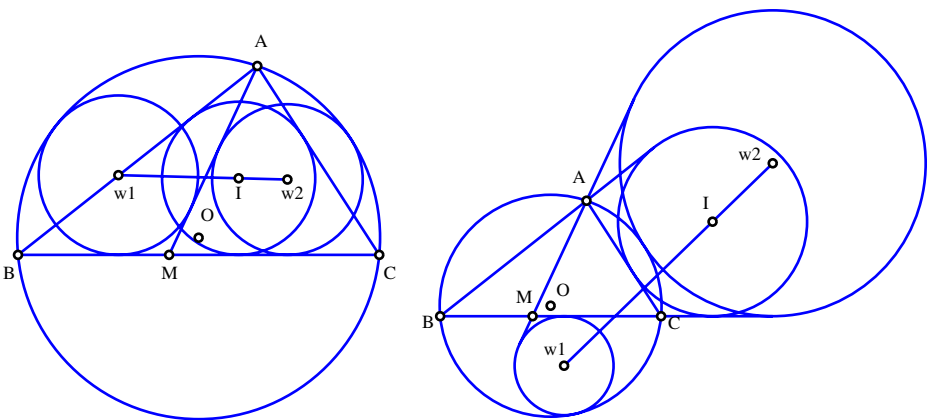
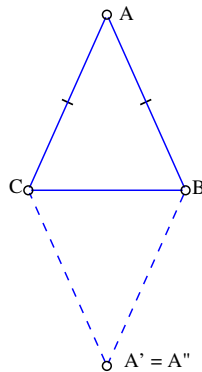


Fig. 2 The Thébault conjecture (Theorem)

Example 2 Let ABC be an isosceles triangle with $AB = AC$. Reflecting AB with respect to (wrpt) BC , we get segment BA' ; then reflecting AC wrpt BC , we get segment CA'' . Points A' and A'' are actually identical.

This is a simple example, but we have also encountered cases in which the identity itself is a relatively deep theorem. Here is our approach. Whenever a new point is constructed, we first check whether this point is identical to a previous one numerically. If so, we use Wu’s method to check the identity. If the identity is valid, we eliminate this point [19]. Without the powerful algebraic proof engine, it could eliminate the redundant points with only approximate numerical computations. But approximate numerical computations does not prove points A and X to be identical.

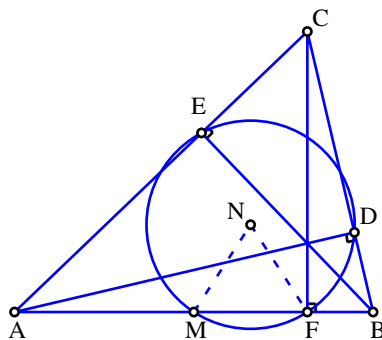


2.2.2 Automated Generation of Nondegenerate Conditions

A geometry theorem is true generally under some additional conditions called nondegenerate conditions. The identification of nondegenerate conditions is subtle.

Example 3 (One form of the nine-point circle theorem) Let points D , E and F be the three feet of the altitudes of triangle ABC . Let N be such that $NF = NE$ and $NF = ND$, and M be the midpoint of AB . Then $NM = NF$ (Fig. 3).

Fig. 3 One form of the nine-point circle theorem



For this example, beside the condition that A , B and C are not collinear, we need at least the nondegenerate condition that triangle ABC is not a right triangle, a subtle condition which is not easy to detect. With the reasoning ability of JGEX, we can generate the nondegenerate conditions for a class of geometry statements of the constructive type [20].

3 Features of Visually Dynamic Presentations of Proofs

The goal of our work is to help the user (reader) in reading a proof text of a geometry theorem and in identifying geometry elements in the proof text with their counterparts in the diagram(s). To achieve this goal, we propose features of visually dynamic presentation of proofs. They include, but are not limited to.

3.1 The Organization of the Proof Text

- V1.** The proof text created by the program (either manually or automatically) is readable, similar to proofs in geometry textbooks or books.³ The reader can understand the proof by reading it step by step.
- V2.** The proof is hierarchically organized, e.g., under a main step there might be substeps. Under each substep, there might be further substeps, and so on. In some cases, organizing a proof hierarchically is closely related to applications of lemmas in mathematical proofs. Generally the arrangement as substeps or lemmas are subjective and it does not have a formal semantics (See Miquel's theorem for $n = 5$ in Example 12 in Part 2 of this series [49].)
- V3.** By clicking a main step, its substeps (if any) shrink to just this main step and when it is clicked again, it will expand back to the original form. This allows the user to concentrate on top-level (main) steps, and when needed, to look at their substeps.
- V4.** By clicking a terminal substep (i.e., a substep without further substeps), it might pop another window possibly with a geometry diagram in animation to further explain this substep. This process can be nested with the hypertext technique (See Simson's theorem in Example 10 in [49].)
- V5.** If a step uses an assertion proved in a previous step, clicking this step causes the previous step to be highlighted. In this way, it is much easier for the user to find the related step. This is especially useful for complicate theorems.

3.2 The Dynamic Visual Effects Related to the Proof Text

- V6.** The displays of the proof text and the geometry elements in the diagram are separated, but are internally related. By clicking a step or a part of the step,

³Here we mean textbooks or books such as [21–23], etc. Proofs in the current US high school geometry textbooks are generally too simple.

the corresponding geometric elements in the diagram respond with various dynamic visual effects.

V7. The visual effect varies depending on the contents of the corresponding proof text. Although these effects do not have exact semantics, they intend to reflect the meaning of the proof text. Here we will list some visual effects commonly used by JGEX.

- *Triangle Congruency* If the proof text is the triangle congruence, then the program moves a copy of one color-filled triangle on the fly by a rotation followed by a translation, possibly with a flip (reflection) if the orientations of the two triangles are opposite, and drops it to the position of the other triangle, at last blinks the corresponding sides of the two triangles alternately.
- *Triangle Similarity* If there are two similar triangles, then a copy of one triangle is first translate to the other, then scaled to fit the other one. The other visual effects are the same as those for two congruent triangles mentioned above.
- *Sliding and Shearing* If the two triangles have equal area, which often appear, for example, in proofs of the Pythagoras theorem with the area dissection method [48], the animation is an area preserving transformation such as a sliding, a shearing, or a sliding following by a rotation.

V8. While **V1–V7** are some what *ad libitum* and naive, we could add additional dynamic visual effects in later version of JGEX other than listed above. Proofs in geometry can have more imaginative dynamic visual effects than proofs in

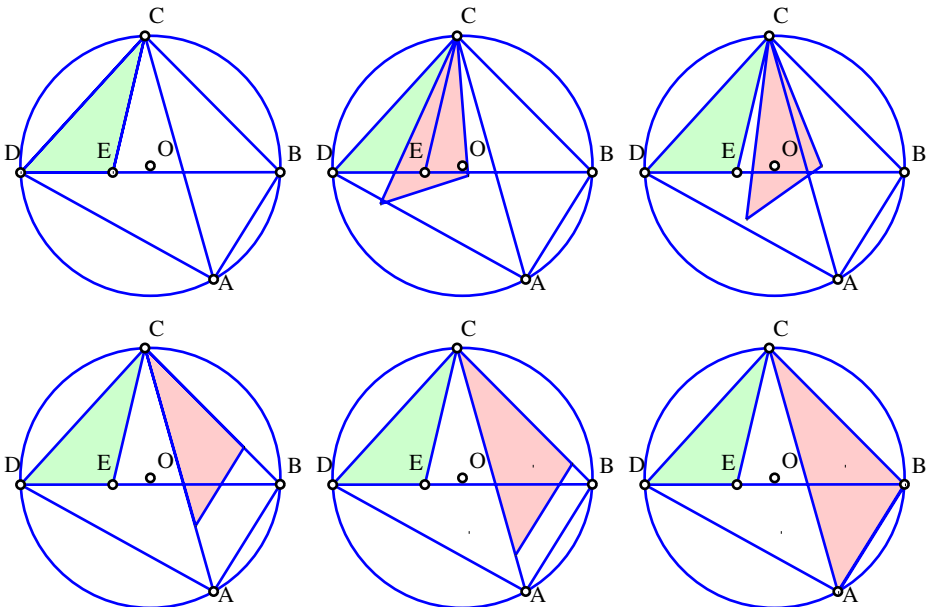


Fig. 4 A proof of ptolemy theorem

other branches of mathematics. For a complete list of visual effects and their corresponding text in the proofs, see <http://www.jgex.net/help/effects.html>.

Example 4 (Ptolemy Theorem) Let circle O be the circumscribed circle of triangle ABC and D a point on the arc AC . Prove that $AD * BC + AB * CD = AC * BD$.

Figure 4 shows the visual effect of the assertion $\triangle DCE \sim \triangle ACB$: first $\triangle DCE$ and $\triangle ACB$ are filled with colors and a copy of $\triangle DCE$ is rotated around point C and drops to $\triangle ACB$, then $\triangle DCE$ is scaled to the size of $\triangle ACB$. For the detail VDPP of this theorem, please see Example 117 in *Collection*.

4 The Manual Input Method for Creating VDPP

4.1 General Discussions

There are two ways to prove a theorem: a machine proof or a human proof. For the machine proofs, the highly successful algebraic methods, such as Wu's method [16, 24, 25] and the Gröbner basis method [26–28], are not applicable to visual presentations for two reasons: (1) the basic quantities of these methods, the Cartesian coordinates of points, generally do not have geometric meanings; (2) the proof itself generally involves intensive polynomial computations, thus is not readable.

Machine proofs similar to those generated by the methods of H. Gelernter and subsequent authors [29–32] are generally appropriate for visual presentations. These methods address theorems of the equality type, or theorems in **unordered** geometries.⁴ These theorems can be proved without using the order relation (betweenness) or the axioms of order in the Hilbert's axiom system [33]. However, the methods used by Gelernter and other authors do use betweenness and its implicit assumptions such as *two sides of a line* and the *interior of an angle*. These proofs are generally diagram-dependent. We will call these methods *traditional methods* in which betweenness is used even for theorems that do not involve the order relation and the proofs generated by these methods are more similar to proofs in geometry books.

So far the successes in automated proving with traditional methods are limited. Not only are there relatively few theorems proved by computer programs, but also the theorems proved are relatively simple. Moderately difficult theorems, such as the nine point circle theorem, Simson's theorem, etc., are beyond the abilities of the traditional methods proposed and implemented so far.

Machine proofs generated by the area method [12], the full-angle method [34, 35], and the deductive database method based on full-angles [36] are also appropriate for the visual presentation. These methods are for unordered geometries. We will discuss automated generation of visual presentations based on these methods, and the experiments in the parts subsequent to the present part of the series.

In this section, we will present our manual input method for creating visually dynamic presentations of proofs.

⁴See books [16, 25]. Also we shall discuss unordered geometries in Part 2 of this series [49] in detail.

4.2 The Manual Input Method

In JGEX, a manually created proof text generally consists of three parts: The “Given” part (the hypotheses), the “To Prove” part (the conclusion), and the “Proof” part. The Given part is the hypotheses in construction form. To input the hypotheses, the user can just draw in the diagram pane and the hypotheses will be generated accordingly. Then the user chooses the keywords “To Prove” with mouse to indicate the inputting of the conclusion. After the conclusion the user can write the Proof part with any proof methods.

The manual input method mainly uses mouse clicks and uses keyboard strokes only when necessary or convenient, e.g., some annotations. With mouse clicks, it is not only easy and intuitive to use, but also less error-prone. For example, instead of typing “ $\triangle DAG$ ”, we can use the mouse to click points D , A , and G in the diagram to generate the text “ $\triangle DAG$ ”. If the mouse clicks are correct, then there is no error: the order of the points in the text and the orientation of the triangle in the diagram are preserved. Furthermore, with the equilateral triangle icon, we can click any two points to drag an equilateral triangle in the diagram, and at the same time the text, say, “equilateral $\triangle ABC$ ”, is automatically generated.

4.3 Four Modes for Visual Presentation with the Manual Input Method

JGEX allows to manually create proofs in four different modes.

4.3.1 Mode 1: Animated Diagrams Only

The approach in this mode is very similar to the approach of Proof Without Words (PWW). However, we add another dimension to the PWW approach, i.e., instead of a static diagram or a series of static diagrams, the diagram here is visually dynamic.

There are many proofs, e.g., various proofs of the Pythagoras theorem with the area dissection method, for which the proofs themselves are apparent by watching the animated diagrams. In our experience with generation of the gif file from a JGEX example, an animated diagram generally consists of several dozens static diagrams. However, in a static medium, even with a series of key diagrams, the presentation is not always apparent to some readers. Or in some other cases, it causes ambiguities (different readers come up with different interpretations).

Example 5 The Pythagorean Theorem: Proof 1.

A visual proof with the two diagrams above is in the book [1] as the Pythagorean Theorem I. What do these two diagrams suggest? It is clear that the two outer squares, say, SQ_1 and SQ_2 , are congruent. Then an average reader, such as the authors of this paper, would make one of the following two interpretations:

1. At the beginning, we have square SQ_1 on the left which contains two squares (a^2 and b^2) and four congruent right triangles. Then we draw a right triangle T_1' , the copy of the right triangle T_1 at left corner of SQ_1 , in the right diagram. Then we erect c^2 on the hypotenuse of T_1' . After that, we draw copies of the remaining three right triangles in SQ_1 on the other three sides of c^2 to make SQ_2 . See Example 101 in [48] for an animated version of this interpretation.

2. Alternatively, at the beginning we have SQ1 and SQ2 as shown in Fig. 5. Then we move each of the four right triangles in SQ1 to SQ2 to take out the four right triangles in SQ2. Thus on the left diagram there are only two squares, a^2 and b^2 left; on the right diagram there is only square c^2 left. Hence $c^2 = a^2 + b^2$. Also see Example 1 in *Collection* for an animated version of this interpretation.

In book [3], the authors cited the same example in book [1] and gave another interpretation.

Example 6 The Pythagorean Theorem: Proof 2.

The first and the last diagrams are the same as Example 1. However, the authors of the book [3] suggest a completely different proof with a series of five static diagrams and with the text “It uses only translations of triangles within a square.” With this important text clue, and thinking for a while, we finally came up with the elegant solution that the authors of book [3] suggest.

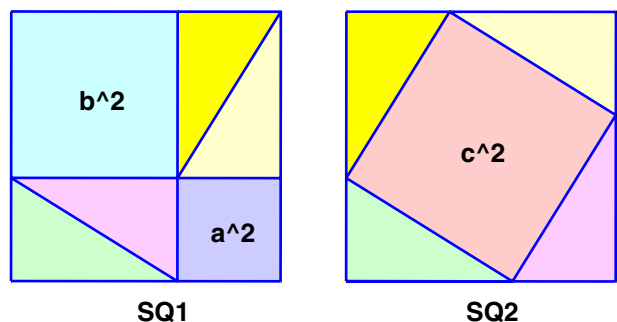
With our dynamic diagram, this elegant proof is very clear without any text explanation. Because of the page limitation, in Fig. 6 we choose only eight static diagrams from the animated gif diagram of Example 2 in *Collection*. The animated gif file consists of 39 static diagrams, thus giving the reader an impression that the diagram is in continuous motion. However, we believe that the eight diagrams here already make it clear that the proof needs only three translations of the (congruent) right triangles.

Another proof with only three translations is Proof #24 of 72 Pythagoras’ Theorem Proofs in the webpage [5]. The proof is listed as ibn Qurra’s (826-901 A.D.) proof. We have collected over 30 proofs of the Pythagoras theorem from the three books [1–3], the 72 proofs in the webpage [5], and from other sources. These proofs use only the area dissection method without any algebraic computations. The reader can look at our examples in animation in *Collection*.

4.3.2 Mode 2: Animated Diagrams Together with Text

In this mode, the user can create the proof text and link the visual effects to the proof text. When stepping through the proof text, the key geometry elements in the diagram are animated. The proof text is in a free format, e.g., the format that a high school students use papers and pencils to write proof exercises freely and the

Fig. 5 A proof of the pythagorean theorem



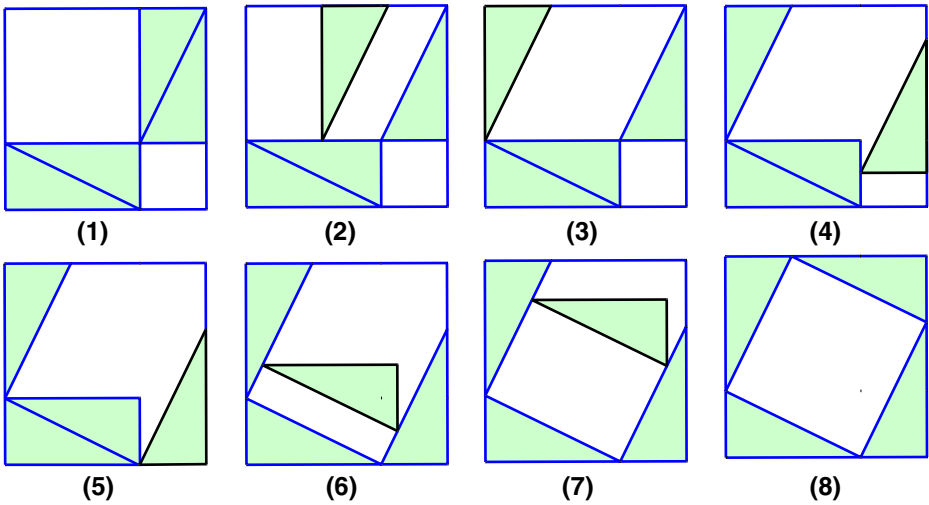


Fig. 6 A proof of pythagorean theorem with three translations

proof can use any proof methods. It is user’s responsibility for the correctness of the interrelationship between the textual part of the proof text and its corresponding geometry elements in the diagram pane, as JGEX does not check anything to be correct.

This method is especially useful for high school teachers or students to write the proofs and present them to the students, as the proofs in high school are not rigorous and complete.

Example 7 The Pythagorean Theorem: Proof 3 (Version 1). This is another proof of the Pythagorean theorem with the area dissection method. Here we use the diagram mixed with its proof text in the diagram pane (Fig. 7).

In Fig. 7, we want to prove $a^2 + b^2 = c^2$. The square a^2 consists of two pieces: one is common with c^2 ; the other one, $\triangle LGB$, is congruent to one of the pieces of c^2 , i.e., $\triangle KHJ$. This takes care of a^2 . Likewise b^2 consists of three pieces: one is common with c^2 ; the other two triangles are congruent to two triangles in c^2 . That takes care of b^2 . When we use mouse clicks to step through the proof text from the beginning (the ‘Given’ line), the proof can be vividly seen. For example, during stepping Step 7 (highlighted in Fig. 7), the two congruent triangles IDK and AFL are filled with blue color and are blinking.

Here the floating icon bar appears generally at the bottom of the window. The different arrow icons in the bar are for stepping through the construction (hypothesis) process and the proof text in various ways.

By repeatedly clicking one of the icons, we step through the whole process, from the construction to the proof, beginning with the first text line “Given: Right triangle ABC” highlighted and the only one blinking right triangle ABC in the diagram. Then highlighted text lines and the corresponding dynamic effects in the diagram proceed in sequence.

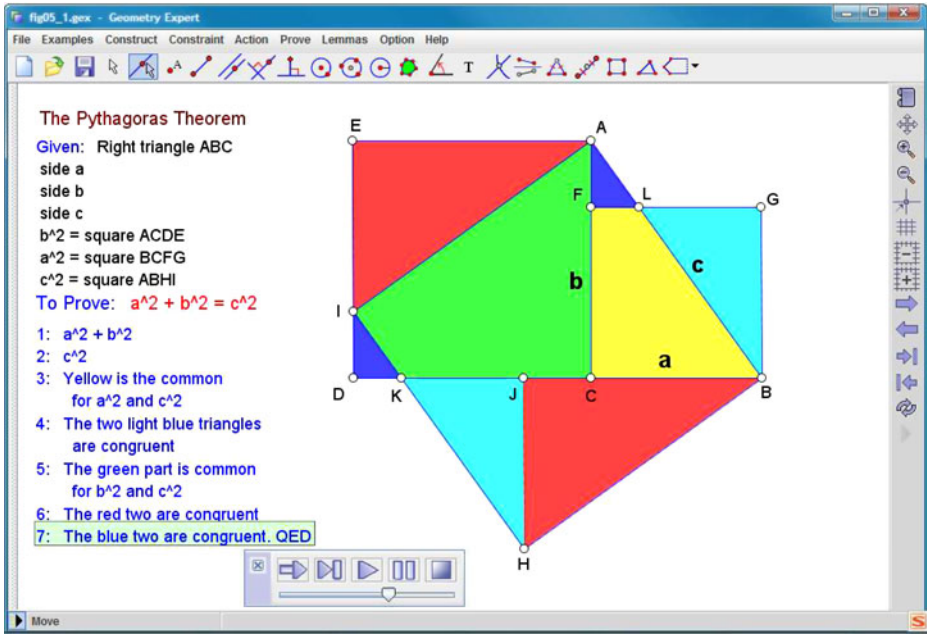


Fig. 7 A proof of the pythagorean theorem—version 2: the proof text in the diagram pane

Example 8 The Pythagorean Theorem: Proof 3 (Version 2). Just for a comparison with Example 3, we can use Mode 1 to present the proof as a dynamic diagram and we choose eight static diagrams in Fig. 8 to illustrate the proof. This example was made with only the mouse clicks. For the real animated diagram, see Example 22 in *Collection*.

Example 9 Identity Method 1 (Example 101 in *Collection*) Let $ABCD$ be a square and E is a point inside the square such that $\angle EDC = \angle ECD = 15^\circ$. Show that $\triangle ABE$ is an equilateral triangle (Fig. 9).

The identity method is another kind of indirect proofs which are closely related to proofs by contradiction. This proof is presented in a geometry book with only one diagram and point E' being drawn in a slightly different position from point E . This theorem is notoriously hard with a direct proof method using triangle congruence only for high school students. For visual effects of the proof see Example 101 in *Collection*, where there is another proof with the identity method for Morley’s trisector theorem, i.e., Example 150.

4.3.3 Mode 3: Dynamic Diagrams with Numerically Verified Proof Text

Dynamic diagrams together with their corresponding proof text parts verified with floating number computations. In this mode, the user can construct the proof mainly with mouse clicks. Each step can be verified to be approximately true in floating number computations. In [Appendix](#), we list all numerically verifiable assertions or the

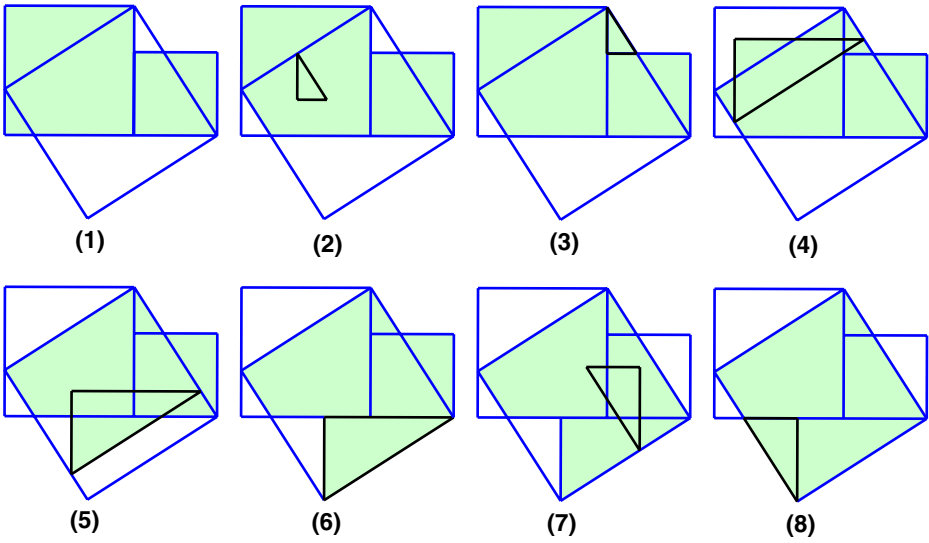


Fig. 8 A proof of the pythagorean theorem—version 2: an animated version

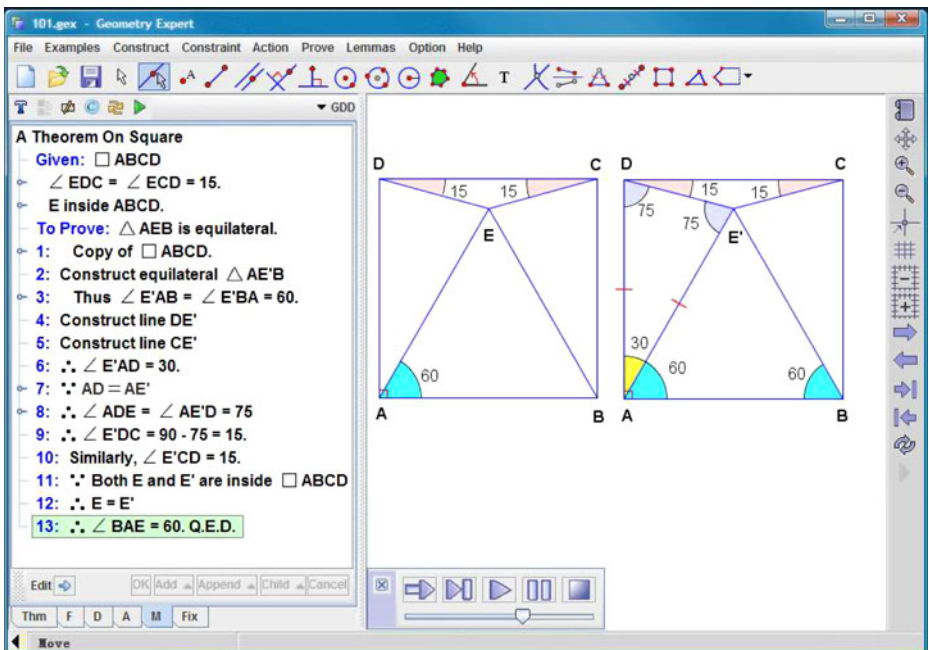


Fig. 9 A theorem of square with the identity method

language JGEX recognizes.⁵ On the webpage <http://www.jgex.net/help/effects.html> one can see all corresponding visual effects.

This would be a good tool for students to learn how to write a proof without incorrect assertions because each input step, i.e., an assertion line in the proof, can be verified approximately to be true with about a dozen randomly generated numerical instances of the diagram and JGEX only accepts these kinds of assertions. Thus when JGEX accepts an assertion that students input, they feel more confident in the correctness of the input assertion and an assertion which is most likely to be false will be rejected.

However, whether an input assertion in a proof is a logical consequence of the previous input assertions, the hypotheses, and the basic rules or axioms is not checked. To check the logical relationship among these input assertions is in our next mode—the proof-checker mode.

Example 10 Viviani's Theorem: Proof 1. From a point D inside an equilateral triangle, three lines are drawn perpendicular to the three sides with three feet E , F , and G . Show that $DE + DF + DG = CH$, where CH is the length of the altitude of the triangle (Fig. 10).

When stepping the given part, the triangle and the four perpendicular segments appear and blink in sequence. At the “To Prove:” step, the four segments blink sequentially for visual identification of the segments involved in the equality. After drawing the three auxiliary lines in Steps 1–3, it confirms two segment equalities with segments on the fly. Then it comes to the first important fact in Step 5. Steps 7 and 8 carry a series of the equation to be numerically verified. At last, segment DE translates to KH on the fly, then IL , carrying the sum of DG and DF , moves on the fly (shown in the diagram) and drops to the position of CK .

The proof was created by mouse clicks with the only exception that the name of the theorem (Viviani's) was typed (and with a typo).

Remark If we remove the hypothesis that point D is inside the equilateral triangle, then there are 6 different cases when point D is outside the triangle. The proofs for 6 cases are similar but different with the traditional method.

4.3.4 Mode 4: The Proof-Checker Mode

We plan to implement two proof checkers at two different levels: an educational proof checker that checks proofs written by students (or teachers) and an academic proof checker of axioms of geometry based on formal logic.

The Proof Checker for Educational Purpose The work on proof-checker of this type is currently in progress together with the work on automated generation of readable proofs with the traditional method (for ordered geometry) [42].

⁵There are two kinds of numerical computations: (1) in the linear case, the computation is exact with a pair of two arbitrarily large integers to represent a rational number; (2) approximate computations using floating numbers. We shall call them exact numerical computation and numerical computation, respectively.

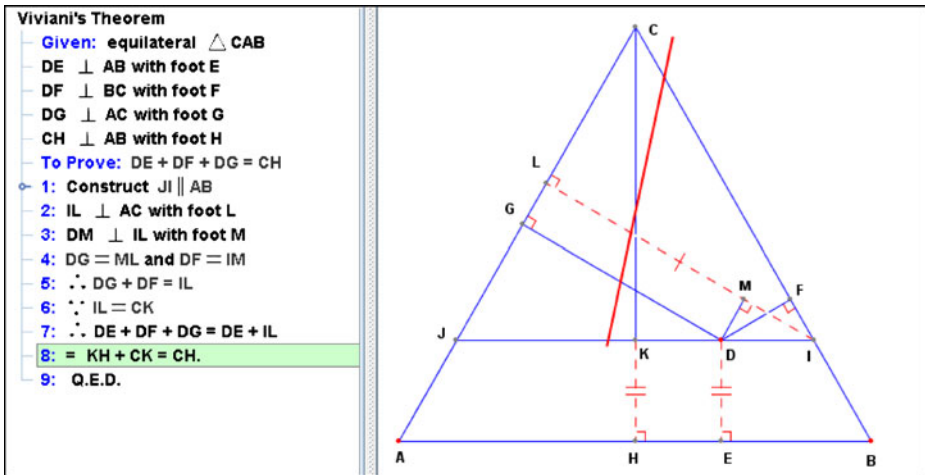


Fig. 10 A proof of Viviani's theorem

Suppose a student write a proof with 10 steps. The proof-checker will verify whether the 10 steps in sequence are logically in step by step, e.g., whether Step 8 is the consequence of the previous steps, the hypotheses, and the basic geometric rules. If not, the program refuses Step 8. However, if it needs to insert more steps to reach Step 8, the program can act in two ways: it either insert all necessary steps to reach Step 8, or it insert only first step(s) for reaching Step 8, giving students a hint to write a correct and complete proof. We can do so because JGEX is powerful enough for automated generation of full proofs of theorems that are within its scope.

Most high school students are dealing with relatively easy theorems. Simson's theorem or the Nine Point Circle theorem should be considered relatively hard to them. Our program implemented for unordered geometry [49] and the the program developed so far [42] for ordered geometry are powerful enough for fast (generally within a second) automated generation of proofs of this kind of relatively easy theorems. Thus right after the user input of specification of a theorem, JGEX has already automatically generate the full proof(s). This is crucial for our proof-checker to be developed, because unlike proof-checkers based on formal logic, our proof-checker, in knowing the proof(s) in advance, can actively guide students to come up with a proof; whereas for other formal logic proof-checkers, a user should know the proof in advance and assists the proof-checker to check correctness of the proof he or she knows in advance.

In the PWW diagrams, many facts are assumed to be true. However, the proofs of some of these facts are sometimes by no means trivial. Thus the diagrams are very good for students to come up with the solution(s) that the diagrams suggest. The manual input method provides a unique visually dynamic environment for our interactive proof-checker.

The Proof Checker using Formal Proofs based on Standard Axiom System of Geometry There have been excellent works on formalization of axiom systems and proofs within those systems in elementary geometry with general proof checking

systems such as Coq and Isabelle [38–40]. Since these systems are of high order logic, it can also reason on geometry statements involving natural numbers n , e.g., n points. These works are for ordered geometries. Currently, the most successful area in geometry theorem proving is in unordered geometry (see Part 2 of this series). The proof check of unordered geometries would be very interesting [41]. We have very few experience in proof checking based on formal logic. Developing a proof checker based on formal logic would be our long term plan. Since JGEX is capable of dealing with diagrams, developing a proof checker with the help of diagrams would be a very interesting topic.

5 More Examples

Example 11 (The Pedal Triangle) Let AF , BE , and CD be the three altitudes of triangle ABC . Show that $\angle[BFD] = \angle[EFC]$, where $\angle[BFD]$, etc., are understood as full angles (see Part 2 of this series [49]).

Figure 11 shows the proof of this theorem. The hypotheses of the theorem consists of only two lines. When expanding the second line “Three Altitudes”, there appear three lines “ $CD \perp AB$ with foot D ”, etc. To input the hypotheses, we just need to draw the diagram and the text for the hypotheses will be generated accordingly. Then we can select the three hypotheses for the three altitudes and combine them. Thus

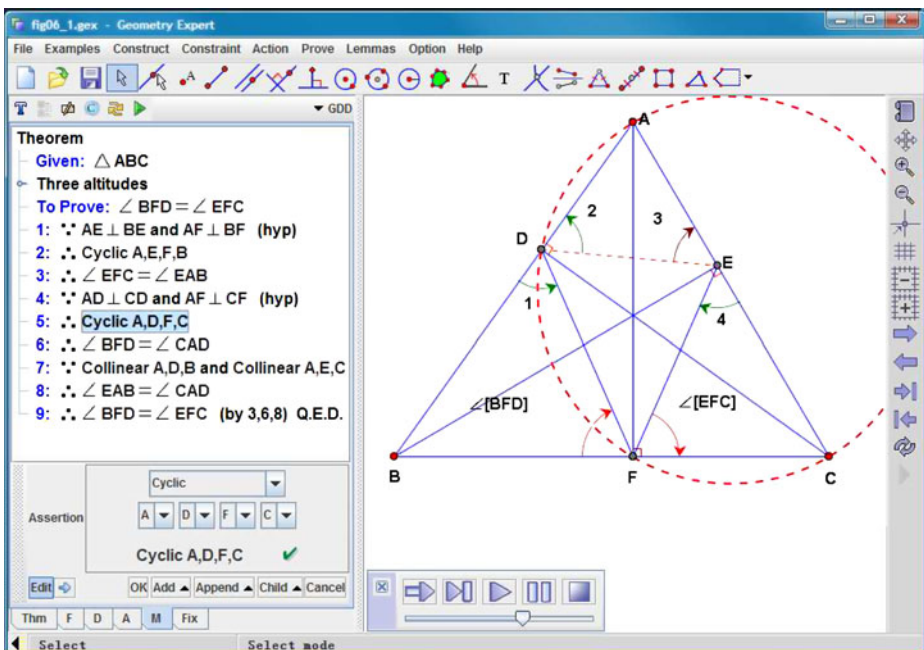


Fig. 11 Pedal triangle

we get a new hypotheses with three sub hypotheses. Note that we use the keyboard to type ‘Three Altitudes’. Without using the keyboard, the generic text ‘Construction 1’ is used.

To input the conclusion assertion $\angle[BFD] = \angle[EFC]$, we can use the pull-down menu to select the keyword ‘assertion’. Then there is another pull-down menu listing all assertions (predicates). By clicking the assertion ‘eqangle’ the user is prompted to select six points. Then JGEX will check the assertion to be approximately true. If the assertion is numerically verified to be true, the input is completed and the text $\angle BFD = \angle EFC$ is generated automatically.

In the proof text, Step 5 is highlighted, thus the auxiliary circle appears. To input this step, we first select the keyword \therefore and then select the assertion ‘cyclic’ with its four points on the diagram. Generally, we can use the mouse to choose items on the menu and click points in the diagram to complete the rest of the proof.

Similarly, we can prove $\angle 1 = \angle 2$ and $\angle 3 = \angle 4$ and $\triangle DEF$ is called the *pedal triangle* of $\triangle ABC$.

Example 12 (The Pedal Triangle Problem) Find points D , E , and F on segments BC , AC , and AB of an acute triangle ABC such that the perimeter of triangle DEF is minimal.

From a minimal problem of two points on one side of a line we can infer that if the triangle DEF with the minimal perimeter exists, it must be the pedal triangle. Also from “a continuous function reaches its minimal value on a compact set” in calculus, the pedal triangle is the solution. However, H. A. Schwarz gave an elegant and elementary solution to the pedal triangle problem.

Example 13 (The Schwarz Solution to the Pedal Triangle Problem) In Fig. 12, reflect triangle ABC five times. Then the **straight** line DT is twice of the perimeter of the pedal triangle (Fig. 12).

This example can be created with JGEX as following. First we draw $\triangle ABC$ with three altitudes and a pedal triangle $\triangle DEF$ as we have done in Example 11. Then we select $\triangle ABC$ and $\triangle DEF$ and then reflect them with respect to line AC . In this way we get $\triangle ACD$ with $\triangle HIF$ as its pedal triangle. This reflection can be cascaded and after five times of reflection we can get the final diagram shown in Fig. 12. This is the Schwarz solution to the pedal triangle problem. According to the angle equality in the preceding example, *straight* segment DT contains points F , H , K , N and R , and is twice the perimeter of the pedal triangle. The twice of the perimeter of triangles other than the pedal triangle is a broken line. Thus DT is the minimal, so is the perimeter of the pedal triangle.

According to the angle equality in the preceding example, straight segment DT contains points F , H , K , N and R , and is twice the perimeter of the pedal triangle. The twice of the perimeter of triangles other than the pedal triangle is a broken line. Thus DT is the minimal, so is the perimeter of the pedal triangle. The requirement that triangle ABC be acute is essential. One can see that by dragging point B with the mouse, then the shapes of the six triangles are changing correspondingly. When $\angle ABC > 90^\circ$, there are overlaps among the six triangles.

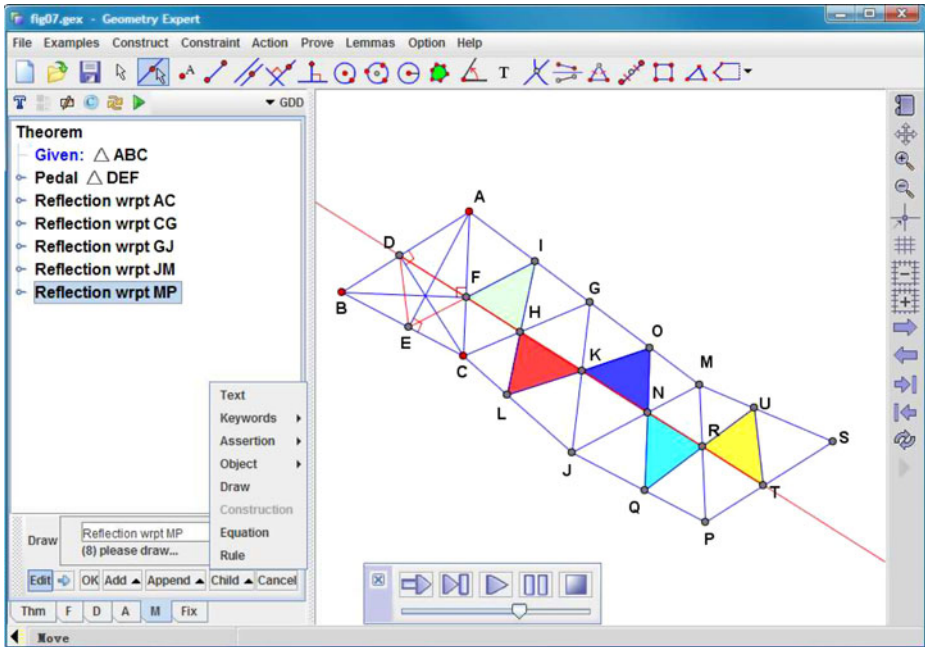


Fig. 12 The Schwarz solution to the pedal triangle theorem

The last example is an inequality theorem. The traditional proof is elegant. We have not seen any kind of machine proofs of this theorem, although we expect that the method of elimination by real root counting introduced by V. Weispfenning can solve them [43, 45].

Example 14 (Inequality Theorem) Let E be a point on the internal bisector AD of triangle ABC with $AC < AB$. Show that $EB - EC < AB - AC$ (Fig. 13).

In Step 5 we use the theorem “the difference of two sides of a triangle is less than the third side” (THM 20). When clicking Step 5, triangle BEF is filled with color and then blinks, followed by segments EB , EF , and FB blinking successively.

The visual effects of the current highlighted step, Step 3, are as following: first, triangles AEF and AEC are filled with different colors, then a copy of $\triangle AEF$ is reflected with the axis AE on the fly and drops into the position of $\triangle AEC$. Then the two triangles blink alternately.

Here we have another implicit assumption of the order relation, i.e., E is between A and D . Otherwise the conclusion might not be valid. However, this assumption is not explicit in the statement of the theorem. This assumption is implicit in Fig. 13. Thus the proof is valid for the diagram in Fig. 13.

The appearance of the green check mark \checkmark to the right of text $ADB \cong EDC$ at the bottom pane indicates that the two triangles are numerically verified to be congruent.

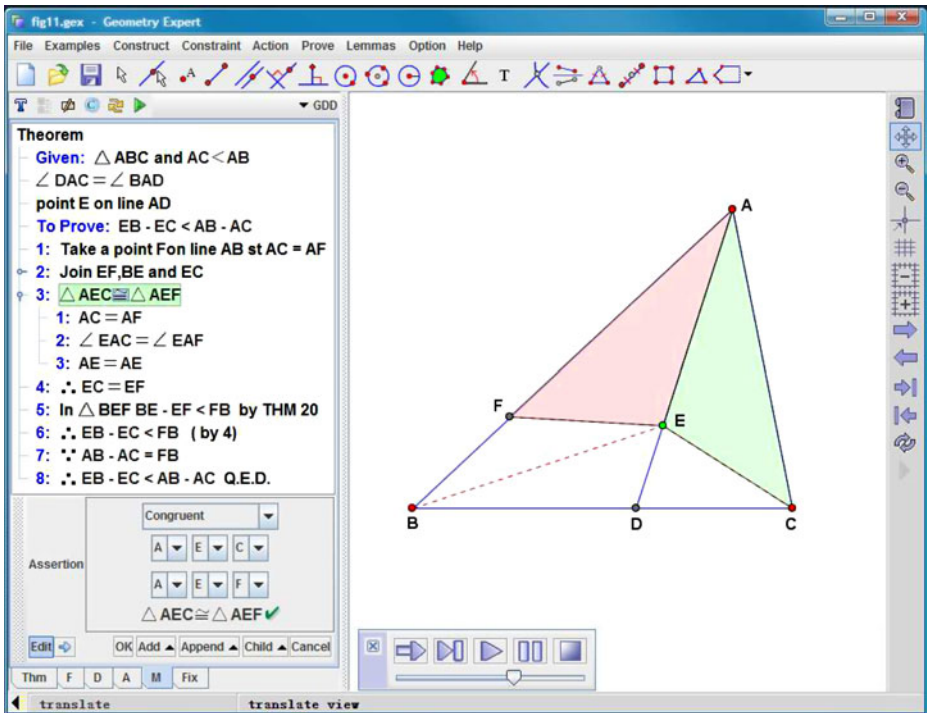


Fig. 13 The proof of an inequality theorem

6 Related Work

6.1 The Proof without Words Approach

6.1.1 The Three Books on the PWV

The work most related to the work reported here is the three excellent books [1–3] on the PWV that cover not only geometry, but also algebra, trigonometry, analytic geometry, calculus, etc.

Our initial motivation is to help students to read a proof in plane geometry and to identify geometry elements in the text with those in the diagram. Our visual presentation with the manual input method adds another important feature: visually dynamic diagrams. The implemented manual input method in JGEX provides a general tool to create these diagrams easily.

We can express many theorems in the three books dynamically, e.g., many proofs of the Pythagoras theorem. Also in the proof of the formula $2a^2 + 2b^2 = c^2 + d^2$, where a , b , c , and d are the lengths of the sides and the two diagonals of a parallelogram respectively, the book [2] uses a series of six diagrams to convey the proof that the creator of the diagrams suggests. However, our animated diagram makes the proof suggested by the creator in a much clearer way (see Example 12 in [48]).

The proof of the Viviani's Theorem with two rotations suggested by Kawasaki in the book [3] is clearer. The diagram uses three circular arrows to indicate a rotation. With our animation the two rotations can be clearly seen; see Example 16 in the *Collection*.

6.1.2 The Webpages Related to PWW

We have searched for examples at the WWW extensively. Perhaps the webpage [4] is the most comprehensive collection of examples. In particular, [5] is a collection of 78 proofs of the Pythagoras theorem, among which eight are animated proofs (animated gif or applet). With the manual input method of JGEX, we have created over 20 animated proofs in addition to the 8 animated proofs in [5].

6.2 The Work or Systems based on the Diagrammatic Proof Approach

The research on the diagrammatic proofs has been very popular since the 1990s (see [37, 44, 46]) and the classic Gelernter's Geometry Machine [30] was considered as an early diagrammatic reasoning system in geometry [37].

Although related, the visually dynamic presentation of proofs proposed by us are different from the general diagrammatic proofs in the related work mentioned above. According to [37], "A diagrammatic representation does not necessarily have to be presented visually so that the user can see it, e.g., visualize it on a computer screen." However, our visual presentation relies heavily on high resolution computer displays and the mouse interaction. The displayed proof text and the diagram of the theorem are separated, but are closely related **internally**. By clicking a part of the proof text, the corresponding geometric elements in the diagram respond with various effects.

People could consider our work as a new branch in diagrammatic proofs. The ease of reading a proof of a geometry theorem is the motivation of our work. And we are focusing on a very specific domain, i.e., plane geometry. Plane geometry provides much more imaginative visually dynamic effects than proofs in other branches of mathematics. Furthermore, we have achieved the full automation of visually dynamic presentations for the full-angle method and the deductive database method as we will show in Part 2 of this series [49].

6.2.1 The System DIAMOND and Related Work

The semi-automatic system DIAMOND is a realization of a formalization of diagrammatic reasoning in the domain of natural number arithmetic, whereas our system focuses on the domain of plane geometry with a fixed number of points.

6.2.2 The System ARCHIMEDES-STUDENT

According to R.K Lindsay, "diagrammatic representations are in another important sense more general than visual images." The system ARCHIMEDES-STUDENT "understand" a demonstration in the sense that it can construct an appropriate diagram from propositional instructions and can verify that the conclusion of the demonstration is true because it follows from knowledge the program has about certain types of objects and their spatial relations. The work is more cognitive science rather than geometric reasoning.

6.2.3 The Dr.Doodle System

The Dr.Doodle system [46] is an interactive theorem prover for diagrammatic representations of metric-space and real-line analysis concepts (involving objects such as functions, sets and lengths, and properties such as open, closed and continuous).

Unlike the approach used in JGEX, in Dr.Doodle the diagrams are heterogeneous representations, i.e., the diagrams combine graphical and textual elements. Each diagram is an assertion in the proof. The inference rules from one diagram to the next diagram are specified with redraw rules based on its dynamic diagram logic [47].

The proof can be presented as a sequence of animated diagrams with the hypertext technique. The proofs, presented in this way, are visually intuitive. The proof of “the function $f(x) = 1/x$ is continuous” is very impressive.

6.3 The Work or Systems Related to Coq

6.3.1 GeoView

GeoView provides the drawing facility to visualize geometry statements with input in Coq, a formalized proof assistant. It is the basis for formalization and visualization of theorem proofs in French high school geometry.

Then GeoView transforms the input in Coq to the formula:

$$F : (H_1 \wedge H_2 \wedge \cdots \wedge H_n) \Rightarrow C$$

Nondegenerate conditions in the original statement in Coq are not in the above formula. Then GeoView uses an algorithm to generate the construction sequence from F . The algorithm is very similar to the one used by Chou in [16]. However, unlike the algorithm in GeoView, we need first to specify the point order in which the points are constructed one by one (see the input of Simson’s theorem). There are two reasons for this requirement. First, the diagrams of most geometry statements are expressed in a constructive way and the readers need to be able to construct the diagram before the proof. Second, without the point order, both algorithms might generate “constructions” that change the meaning of the original statement.

After the construction sequence is fixed, the user is prompted to choose free points and semi-free points on geometry objects with the mouse. Once these points are chosen in the computer screen, the remaining geometry elements of the diagram are automatically constructed [16]. With different point orders in the Simson’s theorem input, our method generate “eight essentially different construction sequences that preserve the original meaning of the theorem.” See Page 31 in [20].

6.3.2 Formalization and Visualization of a French High School Textbook of Geometry

F. Guillot has developed a library in Coq for a French high school textbook of geometry and formalized a geometry system [38]. The system uses the graphical interface of Pcoq and the drawing tool GeoView. It can let the user to visualize geometry statements with dynamic geometry.

The system can prove many relatively hard theorems, such as the Nine-Point Circle theorem, Simson’s theorem etc., with the Coq proof assistant system. But the

system seems not to provide visual effects in diagrams when generating the proof text. Also it does not provide the internal relationship between geometry elements in the proof text and those in the diagram.

6.3.3 *GeoProof*

The GeoProof is a system for plane geometry that combines the following three parts.

- (1) The automatic theorem proving part. It implements the Gröbner basis and the Wu methods. Also it exports “our own implementation of Chou” decision procedure for affine geometry⁶ in the Coq proof assistant.”
- (2) The interactive theorem proving part. This part uses a formal proving system, i.e., the Coq proving assistant. This allows to prove geometry statements with n points by using induction on n . This is impossible for all of our methods developed since 1982.
- (3) The diagrammatic visualization part. The dynamic geometry in this part allows the user to see the visual effects of a geometry statement when a point is dragged with the mouse. The graphic interface is well-designed and impressive. This allows to export a construction built with the graphic input into a statement in the language of the Coq development system.

The proofs automatically generated by GeoProof are readable but there is no visual connection between the geometry elements in the proof text and those in the diagram. Also the system takes the precaution that the set of the hypotheses of a geometry statement is not inconsistent.

GeoProof has a method for automated adding nondegenerate conditions which is a further extension of Chou–Gao’s work [20]. For the constructions to which Chou–Gao’s method does not apply, GeoProof adds a condition that this new point is well constructed. This approach is not totally satisfactory because the point to be constructed is already used in the nondegenerate condition. Also it has not been proved that these conditions are sufficient. GeoProof actually put this burden on the Coq system: if Coq verifies the statement with nondegenerate conditions to be true, the nondegenerate conditions for this particular theorem are sufficient. However, if Coq fails to verify it, the nature of the original statement is unclear: whether it is due to missing the equality hypotheses or due to missing some nondegenerate condition(s)? In addition, people prefer nondegenerate conditions in terms of original given points. For example, the geometric nondegenerate condition of Morley’s trisector Theorem specified by Wu on Page 121 of [16] is still unclear. Is the condition that the three vertices of the triangle are not collinear sufficient? In Coq when we assume the domain is real geometry (Euclidean geometry), it might be. But problem remains open unless the provers based on real closed fields can verify the statement. For complex geometry, for which Wu’s method and the Gröbner basis method are complete, we might need additional conditions, e.g., some segments are non-isotropic.

⁶This is the area method introduced by S.C. Chou, X.S. Gao, J.Z. Zhang, which is capable of proving theorems of congruences of segments and of (full) angles, etc. See book [12].

6.4 The Approach in *Geometer's Sketchpad*

The *Geometer's Sketchpad* (version 4.06) provides a general tool for exploring geometry, algebra, calculus and other areas in mathematics. It can also be used to create animated diagrams for proofs, e.g., proofs of the Pythagoras theorem. As far as we have encountered from the software distribution and the WWW, there are about a dozen animated proofs, which need the whole system itself, whereas once animated diagrams are created with JGEX can be easily saved in GIF format or Java Applet, which can be posted in the web without JGEX.

7 Conclusions and Further Discussions

In this paper we have proposed a new presentation of proofs of theorems in plane geometry, and implemented the manual input method in JGEX for creating these visually dynamic presentations.

JGEX has potential applications in geometry education. It would be a valuable tool for students or teachers to write good, correct proofs. Modes 3 and 4 are designed for this purpose. During the development of JGEX, we have already received a quite few email inquiries for JGEX. Most of the email senders are high school math teachers. Two particular cases are worth mentioning here.

- The Chinese Department of Education invited the third author to give a talk on the influence of information technology on education, in particular the technology of dynamic geometry and automated reasoning software. The talk is based on JGEX and the show will be used by Chinese Department of Education to train middle school teachers throughout the country.
- An Iranian high school math teacher has used JGEX in his geometry class. Furthermore, he and his colleagues organized a workshop on JGEX in 10th Iranian mathematics education conference. They even have created a website in Iran specifically for JGEX: <http://jgex.ir/>.

The system JGEX is still an ongoing developing system: the current version is Beta 0.80 which is available on our server [10]. In addition, JGEX can be run online with over one hundred examples and the counterparts of these examples in gif files can be seen in *Collection* without the Java run-time environment.

Part 2 of this series is automated generation of visual presentation of proofs for the full-angle method and for the deductive database method [49].

Acknowledgements Professor Robert S. Boyer have been giving us valuable advice and suggestions from the early work on JGEX to the final touch of this paper. The authors also wish to thank the referees for their helpful criticism and suggestions. The NSF grant provides Zheng Ye with an unique opportunity to work at WSU. X.S. Gao is supported by NSFC.

Appendix: The List of All Assertions (Predicates)

We first list all 34 predicates in the given part.

The Given Part (or the Hypotheses)

Atomic Predicates:

P1. collinear A B C	A, B, and C are collinear
P2. parallel A B C D	AB and CD are parallel
P3. perpendicular A B C D	AB and CD are perpendicular
P4. eq-distance A B C D	$ AB = CD $
P5. cyclic A B C D	A, B, C and D are cyclic
P6. eq-angle [A B C D] [A ₁ B ₁ C ₁ D ₁]	full-angles $\angle[ABCD] = \angle[A_1B_1C_1D_1]$
P7. eq-angle A B C A ₁ B ₁ C ₁	traditional angles $\angle ABC = \angle A_1B_1C_1$
P8. congruent A B C D E F	$\triangle ABC \cong \triangle DEF$
P9. similiar A B C D E F	$\triangle ABC \sim \triangle DEF$
P10. eq-ratio [A B C D] [A ₁ B ₁ C ₁ D ₁]	$AB / CD = A_1B_1 / C_1D_1$
P11. midpoint M A B	M is the midpoint of AB, i.e., A, M, B are collinear and $AM = MB$
P12. lratio C A B n m	A, B, C are collinear and $\overline{AC} / n = \overline{CB} / m$
P13. pratio C D A B n m	AB and CD are parallel and $\overline{CD} / n = \overline{AB} / m$
P14. tratio C D A B n m	AB and CD are perpendicular and $CD / AB = n / m$ where n and m are two positive integers.

Compounded Predicates

P15. triangle A B C	A, B, and C are not collinear
P16. R-triangle C A B	$AC \perp BC$ and A, B, and C are not collinear
P17. iso-triangle C A B	$CA = CB$ and A, B, C are not collinear
P18. iso-right triangle C A B	$BC \perp AC$, $BC = AC$, A, B, and C are not collinear
P19. foot D C A B	$CD \perp AB$, A, C and D are collinear, and AB is non-isotropic
P20. equi-triangle A B C	$AB = BC = CA$, $\angle ABC = \angle CAB = \angle BCA = 60^\circ$. A, B, C are not collinear and are in the counter-clockwise orientation
P21. trapezoid A B C D	AB and CD are parallel & AB and CD have the same direction, which is equivalent to \overline{AB} and \overline{CD} are parallel, where \overline{AB} denotes the oriented segment of AB. Also we need a nondegenerate condition: A, B, and C are not collinear.
P22. parallelogram A B C D	$AB \parallel CD$, $BC \parallel AD$, and A, B, C are not collinear
P23. rectangle A B C D	Parallelogram A B C D and $DA \perp BA$
P24. square A B C D	Rectangle A B C D and $AB = AD$

Inequality Predicates:

P25. between A B C	A, B, and C are collinear, $B \neq C$, and A is between B and C
P26. angle-inside D A B C	D is inside (traditional) angle $\angle ABC$
P27. angle-outside D A B C	D is outside $\angle ABC$
P28. triangle-inside D A B C	D is inside triangle ABC

- P29.** paral-inside E A B C D E is inside parallelogram ABCD
- P30.** opposite-side C D A B C and D are on the opposite sides of line AB, and $A \neq B$
- P31.** same-side C D A B points C and D are on the same side of line AB, and $A \neq B$
- P32.** $AB < DC$ segment AB is less than segment DC
- P33.** $\angle ABC < \angle CDE$
- P34.** The $(n+1)$ -polygon $A_0A_1A_2 \dots A_n$ is convex ($n \geq 3$). points A_i and A_{i+3} are on the same side of line $A_{i+1}A_{i+2}$, where indexes are understood mod $n + 1$.

Notation Convention of Angles A full-angle is defined to be an ordered pair of lines u and v : $\angle[u, v]$. We use $\angle[ABC]$ to denote the full-angle $\angle[AB, BC]$ and use $\angle ABC$ to denote the traditional angle ABC.

The To Prove Part (or the Conclusion)

The conclusion is generally a predicate from **P1** to **P12**.

The Proof Part

Each line in the proof part can be of the following type:

- L1.** *Assertions or assertions in conjunction* The assertion can be either atomic or compound, just as those defined in the hypotheses. If the assertion is construction related, the proof line is taken as a construction. If there are more than one assertion in a line, they are interpreted as in conjunction.
- L2.** *Equations and expressions* An equation is of the form $E_1 = E_2$ where E_1 and E_2 are two expressions of the same type of quantity. Equations can be cascaded: $E_1 = E_2 = E_3$.
The expression can be of following type: oriented segments, the area of a convex polygon, the product of two segments, and the ratio of two quantities. With the same type of quantities, we can do the operation “+” and “-” to get compounded expressions. For examples, for a $\square ABCD$, we can write an expression as follow: $\triangle ABC + \triangle ACD = \square ABCD$.
- L3.** *Keywords* Keywords and assertions are always together to form a meaningful sentence. The most frequently used keywords are: “and”, “because”, “hence”.
- L4.** *Rulers* JGEX provides a list of build-in rulers to generating proofs.
- L5.** *Remarks and Comments* Remarks and comments in the proof text can give the reader a better understanding of the proof. The user needs to type the text for it and JGEX just treats it as plain text.

References

1. Nelsen, R.B.: Proofs without Words: Exercises in Visual Thinking (Classroom Resource Material) (1993)
2. Nelsen, R.B.: Proofs without Words II: more Exercises in Visual Thinking (Classroom Resource Material), by Roger B. Nelsen (2001)

3. Alsina, C., Nelsen, R.B.: [C] Math Made Visual: Creating Images For Understanding Mathematics. The Mathematical Association of America, New York (2006)
4. Cut-The-Knot: Geometry articles, theorems, problems. <http://www.cut-the-knot.org/geometry.shtml> (2009)
5. Cut-The-Knot: Pythagorean theorem. <http://www.cut-the-knot.org/pythagoras/index.shtml> (2009)
6. Gao, X.S., Zhang, J.Z., Chou, S.C.: Geometry Expert. Nine Chapters Pub. (in Chinese) (1998)
7. Chou, S.C., Gao, X.S., Zhang, J.Z.: An introduction to geometry expert. In: McRobbie, M.A., Slaney, J.K. (eds.) Proc. CADE-13, pp. 235–239. Springer, New York (1996)
8. Gao, X.S.: Building dynamic mathematical models with geometry expert, III. A geometry deductive database. In: Yang, W., Wang, D. (eds.) Proc. ASCM99. ATCM, Chadstone Centre (1999)
9. Gao, X.S., Lin, Q.: MMP/Geometer—a software package for automated geometry reasoning. In: Winkler, F. (ed.) Automated Deduction in Geometry, pp. 44–66 (2004)
10. Chou, S.C., Gao, X.S., Ye, Z.: Java geometry expert server: <http://woody.cs.wichita.edu> (2009) (One can run the most part of JGEX with a browser.)
11. Les Editions du Kangourou: Le théorème de Thales. <http://www.mathkang.org/swf/thales2.html> (2009)
12. Chou, S.C., Gao, X.S., Zhang, J.Z.: Machine Proofs in Geometry. World Scientific, Singapore (1994)
13. Sktektee, S., Jackiw, N., Chanan, S.: *Geometer's Sketchpad* Verison 4.0, Reference Manual. Key Curriculum, Emeryville (2001)
14. Laborde, J.M., et al.: Cabri II Plus (2000)
15. Kortenkamp, U., Richter-Gebert, J.: User Manual for the Interactive Geometry Software Cinderella. Springer, New York (2000)
16. Chou, S.C.: Mechanical Geometry Theorem Proving. D. Reidel, Dordrecht (1988)
17. Taylor, K.B.: Three circles with collinear centres, solution of advanced problem 3887. *Am. Math. Mon.* **90**, 486–487 (1983)
18. Penrose, R.: *Shadows of the Mind*. Oxford University Press, Oxford (1994)
19. Chou, S.C., Gao, X.S., Ye, Z.: Java geometry expert. In: Proceedings of 10th Asian Technology Conference in Mathematics, pp. 78–84 (2005)
20. Chou, S.C., Gao, X.S.: A Class of Geometry Statments of Constructive Type and Geometry Theorem Proving, TR-89-37. Department of Computer Sciences, University of Texas at Austin, 32 p., November (1989)
21. Adler, C.F.: *Modern Geometry*. McGraw-Hill Book, Sydney (1958)
22. Coxeter, H.S.M., Greitzer, S.L.: *Geometry Revisited*, (New Mathematical Library). The Mathematical Association of America, New York (1975)
23. Hadamard, J.: *Lecons de Geometrie Elementaire*, I. Paris (1931)
24. Wu, W.T.: On the decision problem and the mechanization of theorem in elementary geometry. *Scientia Sinica* **21**, 159–172 (1978); Automated theorem proving: after 25 years. *A.M.S., Contemp. Math.* **29**, 213–234 (1984)
25. Wu, W.T.: *Basic Principles of Mechanical Theorem Proving in Geometries*. Press, Beijing (in Chinese) (1984). (English Version, Springer-Verlag (1993))
26. Kapur, D.: Geometry theorem proving using Hilbert's Nullstellensatz. In: Proc. of SYMSAC'86, pp. 202–208. Waterloo (1986)
27. Chou, S.C., Schelter, W.F.: Proving geometry theorem with rewrite rules. *J. Autom. Reason.* **4**, 253–273 (1986)
28. Kutzler, B., Stifter, S.: Automated geometry theorem proving using Buchberger's algorithm. In: Proc. of SYMSAC'86, pp. 209–214. Waterloo (1986)
29. Gelernter, H., Hanson, J.R., Loveland, D.W.: Empirical explorations of the geometry-theorem proving machine. In: Proc. West. Joint Computer Conf., pp. 143–147 (1960)
30. Gelernter, H.: Realization of a geometry-theorem proving machine, In: Feigenbaum, E.A., Feldman, J. (eds.) *Computers and Thought*, pp. 134–152. Mcgraw Hill, London (1963)
31. Nevins, A.J.: Plane geometry theorem proving using forward chaining. *Artif. Intell.* **6**(1), 1–23 (1975)
32. Coelho, H., Pereira, L.M.: Automated reasoning in geometry theorem proving with prolog. *J. Autom. Reason.* **2**(4), 329–390 (1986)
33. Hilbert, D.: *Foundations of Geometry*, the first edition (in Germany) was published in 1899. Open Court, La Salla (1971)
34. Chou, S.C., Gao, X.S., Zhang, J.Z.: Automated generation of of readable proofs with geometric invariants, II. Proving theorems with full-angles. *J. Autom. Reason.* **17**, 349–370 (1996)

35. Chou, S.C., Gao, X.S., Zhang, J.Z.: A collection of 110 geometry theorems and their machine produced proofs using full-angles. WSU technical report (1995)
36. Chou, S.C., Gao, X.S., Zhang, J.Z.: A deductive database approach to automated geometry theorem proving and discovering. *J. Autom. Reas.* **25**(3), 219–246 (2000)
37. Jamnik, M.: *Mathematical Reasoning with Diagrams: From Intuition to Automation*. CSLI, Stanford University, Stanford (2001)
38. Guilhot, F.: Premiers pas vers un Cours de Géométrie on Coq pour le Lycée. Research Report N° 4893, 44 pages, INRIA (2003)
39. Dehlinger, C., Dufourd, J.F., Schreck, P.: Higher-order intuitionistic formalization and proofs in Hilbert's elementary geometry. In: 3rd Int. Workshop on Automated Deduction in Geometry, Zurich, Switzerland, 2000. LNAI, vol. 2061, pp. 306–323. Springer, New York (2001)
40. Narboux, J.: Mechanical theorem proving in Tarski's geometry. In: Botana, F., Recio, T. (eds.) *Workshop on Automated Deduction in Geometry*, Pontevedra, Spain, 2006. LNAI, vol. 4869, pp. 139–156. Springer, New York (2008)
41. Narboux, J.: A decision procedure for geometry in Coq. In: TPHOL'04 Proceedings. LNCS, vol. 3223. Springer, New York (2004)
42. Chou, S.C., Ye, Z., Gao, X.S.: Visually Dynamic Presentation of Proofs in Plane. Part 3. Automated Generation of Visually Dynamic Presentations with the Traditional Method and Automated Addition of Auxiliary Geometry Elements (2009, in preparation)
43. Dolzmann, A., Gilch, L.A.: Generic Hermitian quantifier elimination. In: Campbell, J.A., Buchberger, B. (eds.) *Artificial Intelligence and Symbolic Computation: 7th International Conference, AISC 2004, Linz, Austria*. Lecture Notes in Computer Science, vol. 3249, pp. 80–93. Springer, Berlin (2004)
44. Jamnik, M., Bundy, A., Green, I.: On automating diagrammatic proofs of arithmetic arguments. In: *Proceedings of International Joint Conference on AI* (1997)
45. Weispfenning, V.: A new approach to quantifier elimination for real algebra. In: Caviness, B.F., Johnson, J.R. (eds.) *Quantifier Elimination and Cylindrical Algebraic Decomposition*, Texts and Monographs in Symbolic Computation, pp. 376–392. Springer, Wien (1998)
46. Winterstein, D., Bundy, A., Gurr, C.: Dr. Doodle: a diagrammatic theorem prover In: *Proceedings of 2nd International Joint Conference*, Cork, Ireland, pp. 331–335, July 4–8, (2004)
47. Winterstein, D.: Using diagrammatic reasoning for theorem proving in a continuous domain. Ph.D. thesis, 263 p., The University of Edinburgh (2004)
48. Ye, Z., Chou, S.C., Gao, X.S.: A Collection of Examples Created with JGEX. In: <http://woody.cs.wichita.edu/collection/index.html> (2009)
49. Ye, Z., Chou, S.C., Gao, X.S.: Visually dynamic presentation of proofs in plane, part 2. Automated generation of visually dynamic presentations with the full-angle method and the deductive database method. *J. Autom. Reason.* (2009, in press)