



Measuring the Readability of Geometric Proofs: The Area Method Case

Pedro Quaresma¹ · Pierluigi Graziani²

Received: 20 January 2021 / Accepted: 26 September 2022
© The Author(s) 2023

Abstract

Using an approach, inspired by our modernisation of Lemoine’s *Geometrography*, this paper proposes a new readability criterion for formal proofs produced by automated theorem provers for geometry. We analyse two criteria to measure the readability of a proof: the criterion given by Chou et al. and the one given by Wiedijk. After discussing the limitations of these two criteria, we introduce a novel approach, which provides a new criterion. We conclude discussing some future work.

Keywords Readability · Formal proof · Automated deduction in geometry · Geometrography

Mathematics Subject Classification 51A05 · 68T15

1 Introduction

An important feature of a text is its *readability*. Readability is the ease with which a reader can understand a written text. The readability of a text is determined by many factors and plays an important role in many areas of interest. For example, it might depend both on the content of the text, i.e., the complexity of the vocabulary and syntax, and on the layout of the

The first author was partially support by FCT - Foundation for Science and Technology, I.P., within the scope of the project CISUC - UID/CEC/00326/2020 and by European Social Fund, through the Regional Operational Program Centro 2020. The second author was partially support by Italian Ministry of Education, University and Research through the PRIN 2017 project “The Manifest Image and the Scientific Image” prot. 2017ZNNW7F_004.

✉ Pedro Quaresma
pedro@mat.uc.pt
Pierluigi Graziani
pierluigi.graziani@uniurb.it

¹ Departamento de Matemática da FCTUC, Largo D. Dinis, 3000–143 Coimbra, Portugal

² Department of Pure and Applied Sciences, University of Urbino, Via Sant’ Andrea, 34, Urbino, PU 61029, Italy

text, e.g. its typographical aspects. Readability has to be distinguished from *legibility* that is the ease with which a reader can recognise individual characters in a text.

In order to quantify the readability of a text, various formulas have been defined [5]. In this paper we will deal with the readability of geometric proofs produced by automatic theorem provers. A potential approach that has been followed in the past is to take formulas that were developed for applications to non-scientific texts and apply them to mathematical texts [13]. However, this sort of approach has not been extended in a proper way to measure the readability of proofs produced by automatic theorem provers and much more work needs to be done¹ [6, 12, 19, 22, 26].

A mathematical text is composed of many elements: descriptions in natural language, formulas and diagrams; thus, it is much more difficult to quantify its readability through formulas, then in the case of regular text. Even more complex is the problem of the readability of mathematical proofs produced by automatic provers that are often presented in a form that can only be read by experts.

In this paper, we will introduce both a language to formulate readability criteria for formal proofs produced by automated theorem provers for geometry, based on the area method [2, 10] (see Appendix 1), and a novel criterion based on our modernisation of Lemoine's *Geometrography* [14, 22, 24]. We will show how this new criterion is consistent with the results of the other already existing criteria, but that it is also more general and expressive compared to the others.

The proposed language allows for an easy formulation of new readability criteria as well as for an easy implementation of those criteria in repositories such as the *Thousand of Geometric problems for geometric Theorem Provers* (TGTP),² thereby collecting data relevant to the further development of the area of automated theorem proving in geometry. The data will help strengthen the use of automatic tools not only in research but also in applications like in mathematics education, where the use of automated deduction is already making its way [7, 23]. Therefore, as in the Automath project, the formulation of a readability criterion will allow the definition of a threshold below which "people will start using them (the proofs produced by automated theorem provers³) for serious work" (see §2.2.2).

Overview of the paper. The paper is organised as follows: first, in §2, the known readability criteria will be discussed. In §3, Lemoine's *Geometrography*, its modernisation and a formal language employed to study readability of formal proofs produced by automated theorem provers for geometry, based on the area method, will be analysed. In §4, a new readability criterion that uses *Geometrography* will be presented, providing also some examples of its application. In §5 conclusions are drawn and future work will be discussed.

2 Criteria of Readability (by Experts)

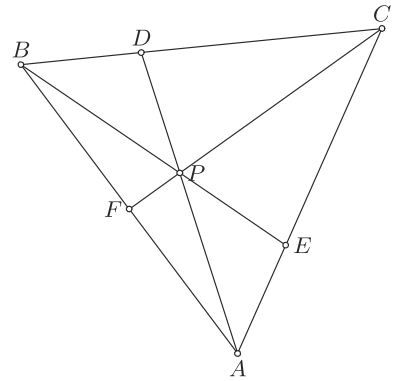
To the best of our knowledge there are two precise proposals to measure the readability of a proof. The first one is that proposed by Chou et al. [1, p. 452], while the second is that proposed by Freek Wiedijk and is known as the *de Bruijn factor* [4, 27].

¹ We will not consider here the problem of the readability of geometric proofs from the Mathematics Education point of view. We will address some issues related to that context in the conclusions.

² <http://hilbert.mat.uc.pt/TGTP/index.php>.

³ In the original sentence, "Automath like Systems".

Fig. 1 Geometric Construction, Ceva's Theorem



2.1 Maxt-Lems Criterion

Chou et al. [1, p. 452] proposed a way to measure how difficult it is to read a formal proof, obtained by using an automated theorem prover for geometry (GATP) implementing the area method. The Maxt-Lems (ML) criterion considered the following pair (*maxt*, *lems*), where:

- *maxt* is the number of terms of the maximal polynomial occurring in the machine's proof. Thus, *maxt* measures the number of computations needed in the proof;
- *lems* is the number of elimination lemmas used to eliminate points from geometric quantities. In other words, *lems* indicates the number of deduction steps in the proof.

Using those two elements and analysing all the proofs done by their GATP, they managed to determine an indicative threshold for readability. According to [1, p. 452] a formal proof, which employs the area method, is considered readable if one of the following conditions holds:

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

It is interesting that according to their corpus:⁴ 66.9% of the proofs have $\text{maxt} \leq 5$, 42.6% have $\text{lems} \leq 10$ and 73.2% have $\text{lems} \leq 20$.

Let us consider, for example, the *Thousand of Geometric problems for geometric Theorem Provers* (TGTP) repository, specifically, problem GEO0001, the *Ceva's Theorem*.

Theorem 1 (Ceva's Theorem) *Let $\triangle ABC$ be a triangle and P be any point in the plane. Let $D = AP \cap CB$, $E = BP \cap AC$, and $F = CP \cap AB$. Show that: $\frac{AF}{FB} \times \frac{BD}{DC} \times \frac{CE}{EA} = 1$. P should not be in the lines parallel to AC , AB and BC and passing through B , C and A respectively [10].*

With respect to the *ML criterion*, considering the proof made by the *Geometry Constructions L^AT_EX Converter (GCLC)* [11] GATP (see Appendix 1), the values are: $\text{maxterm} = 1$, and $\text{lems} = 3$. Therefore, this would be considered a readable proof.

⁴ They considered 478 machine solved geometry problems.

Table 1 Ceva's Theorem Proof by *GCLC* Area Method GATP, *de Bruijn factor*

	Informal	Formal	de Bruijn factor
Uncompressed	125KB	137KB	1.09
Compressed	124KB	136KB	1.09

2.2 The de Bruijn Factor

The *Automath* project had the goal of developing a system that would allow to write entire mathematical theories in such a precise fashion that verification of the correctness of theorems in such theories could have been carried out by formal (mechanical) operations applied directly to the text [4]. This was a first effort in the direction of the Formalisation of Mathematics that is now pursued by researchers working in systems like *Coq*, *Isabelle* and *Mizar*.⁵

In “*A Survey of Project Automath*”, de Bruijn introduced the *loss factor* between the size of an ordinary mathematical exposition and its full formal translation inside a computer. The *loss factor* expresses what someone loses, in terms of shortness, when translating informal mathematics into *Automath*. Wiedijk developed the concept and called it the *de Bruijn factor*. The *de Bruijn factor* was developed for a situation where a proof is entered in a computer in full detail in such a way that the computer can check its correctness, e.g. when an existing informal mathematical text is taken and it is translated into a computer representation (using a system like *Automath*). So the *de Bruijn factor* measures how efficient a system is [27]. Wiedijk noted that non-meaningful questions about formatting could affect the calculus of the loss factor, for example: if indentation is performed employing the tab key, then such indentation can be eight times smaller compared to situations in which the indentation is done using the space key; also the \TeX macro name for the ‘ \Leftrightarrow ’ symbol uses 15 characters, while an encoding like “ $\langle = \rangle$ ” uses only 3. To further smooth formatting choices, Wiedijk proposed to compress the files before calculating the ratios of their sizes. Wiedijk calls the ratio of the uncompressed file sizes the *apparent de Bruijn factor*, and the ratio of the compressed file sizes the *intrinsic de Bruijn factor* [27].

We claim that the *de Bruijn factor* can be used, in a broader sense, to measure the efficiency of an automated theorem prover and a given axiomatisation. Whenever a informal proof is known for a given theorem, it can be compared with the formal proof produced by the automated theorem prover, using a specified axiomatisation. This is particularly true in geometry where a given informal geometric proof can be compared with an, also geometric, formal proof produced by a geometric automated theorem prover.

Using again the Ceva's Theorem as an example, the readability of its formal proof, with respect to the *de Bruijn factor* can be calculated⁶ (see Table 1).

Wiedijk also introduced the *de Bruijn threshold*, i.e., a limit below which “the people will start using them (*Automath* like system) for serious work”. We will consider the value of 2 as a readability threshold. Further studies are needed in order to establish a readability threshold for automated proofs, using the de Bruijn factor. Moreover, a broader comparison between formal proofs and informal proofs is needed.

Considering the quotient of the size of the compressed formal proof (area method) and the size of the informal proof, the *de Bruijn factor* of Ceva's Theorem is 1.09. It would therefore be sensible to consider the *GCLC* area method proof, readable.

⁵ <https://coq.inria.fr/>, <https://isabelle.in.tum.de/>, <http://mizar.org/>.

⁶ We used the proof found in https://artofproblemsolving.com/wiki/index.php/Ceva's_theorem as source for the informal proof.

2.3 ML and de Bruijn Factor's Limits

Analysing the previous criteria, we can note a first limit for both the *ML criterion* and the *de Bruijn factor*: they assume that *readability by expert* is being considered, i.e., a geometer expert in the language of the prover that produces the proof.

A second limit emerges when the following [22] classification of formal geometric proofs produced by GATPs is taken into consideration:⁷

1. no readable proof, only a proved/not proved output;
2. non-synthetic proof (i.e., a proof without a corresponding geometric description, e.g. algebraic methods);
3. semi-synthetic proof with a corresponding prover's language rendering;
4. (semi-)synthetic proof with a corresponding natural language rendering;
5. (semi-)synthetic proof with a corresponding natural language and visual rendering;

Relating the *ML criterion* with this classification, we can note that such criterion only allows the definition of a threshold for semi-synthetic proofs that employ the area method (level 3). The direct applicability of the *ML criterion* to other synthetic methods, e.g. full-angle methods or the deductive database method [3, 28], would be possible, considering the number of deduction steps of the proofs and adapting the condition regarding the maximal term in the proofs.

The *de Bruijn factor* can be used directly in all levels above 1, although it is more meaningful on levels greater or equal than 2.3. Considering the (GCLC) and its integrated GATPs based on the area method, Wu's method and Gröbner Basis method [9], it is possible to calculate the readability of the proofs developed using the different GATPs. It is indeed possible to imagine, extrapolating from the results with the area method, that all those proofs would be readable, and this would hold even though the *de Bruijn factor* requires informal proofs to be provided.⁸

The two criteria analysed are very different, the first is very specific while the second is very generic, although both criteria require readability by experts. We can therefore ask ourselves if it is possible to define a new criterion which does not require readability by experts, which is also more natural and expressive than the previous ones, and which can be generalised to various proof methods.

3 Looking for a More Natural Readability Criterion

The new criterion that we want to propose is based on our modernisation of Lemoine's *Geometrography* [14, 22, 24]. We will begin by explaining what *Geometrography* is and what its modernisation consists of.

Geometrography, "alias the art of geometric constructions", aims at providing a tool: (i) to designate every geometric construction by a symbol that manifests its simplicity and

⁷ GATPs can be of two major types: algebraic, the proof, if it exist, is done recurring to an algebraic reasoning (e.g. Gröbner basis); geometric (synthetic), the proof, if it exist, is done recurring a set of axioms and inference rules of geometry, without the use of coordinates. Semi-synthetic methods, e.g. the area method, use also the axioms of a field of characteristic different from 2.

⁸ The Wu's method and the Gröbner basis method are both algebraic methods, from the geometric point of view their proofs are unreadable (level 2).

exactitude;⁹ (ii) to teach the simplest way to execute an assigned construction; (iii) to discuss a known solution to a problem and eventually replacing it with a better solution; (iv) to compare different solutions for a problem, by deciding which is the most exact and the simplest solution from the point of view of *Geometrography* [14–17, 20, 22, 24].

3.1 Classical Geometrography

In Lemoine’s *Geometrography* two coefficients are defined to measure the relative difficulty to perform some geometric constructions. The approach is applied to ruler and compass geometry, i.e., geometric constructions made only with the help of a ruler and a compass. Considering the modifications proposed by Mackay [16], the following **Ruler and Compass** constructions and the corresponding coefficients can be analysed.

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

Then a given construction is measured against the number of uses of those elementary steps. For a given construction expressed by the equation:

$$l_1 R_1 + l_2 R_2 + m_1 C_1 + m_2 C_2$$

where l_i and m_j are coefficients denoting the number of times any particular operation is performed. The number $(l_1 + l_2 + m_1 + m_2)$ is called the *coefficient of simplicity* (cs) of the construction, and it denotes the total number of operations performed. The number $(l_1 + m_1)$ is called the *coefficient of exactitude* (ce) of the construction, and it denotes the number of preparatory operations on which the exactitude of the construction (made with the help of physical, inaccurate, tools) depends [16, 17].

3.2 Geometrography in Dynamic Geometry

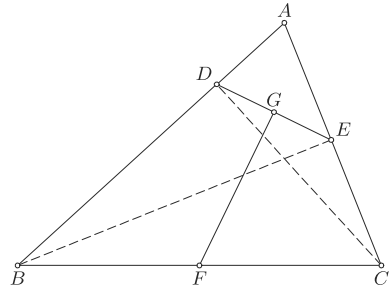
Classical *Geometrography* applies to geometric constructions made with the help of a ruler and a compass. Its modernisation, proposed in [22, 24] uses the tools of the dynamic geometry systems (DGS). In [22] it was shown how to modernise *Geometrography* using *GCLC*, in [24] the generality of the approach is shown, using *GeoGebra* [8].

Considering the operations: *define a point, anywhere in the plane, D* and *define a given object, using other objects, C*, the following values for the *GCLC* basic constructions are obtained:

- point – fix a point in the plane. D
- line – uses two points. $2C$
- circle – uses two points $2C$
- intersec – uses two lines $2C$
- intersec – uses four points $4C$
- intersec2 – uses a circle and a circle or line $2C$

⁹ Exactitude, or the lack of it, in this context refer to the possible inaccuracy introduced by physical tools such as ruler and compass.

Fig. 2 Geometric Construction, TGTP problem GEO0369



$$CS_{gcl} = 19 = 3D + 16C$$

$$CF_{gcl} = 6$$

midpoint – uses two points	2C
med – uses two points	2C
bis – uses three points	3C
perp – uses a point and a line	2C
foot – uses a point and a line	2C
parallel – uses a point and a line	2C
onsegment – uses two points	2C
online – uses two points	2C
oncircle – uses two points	2C

In the modernisation (extrapolation) of the *Geometrography*, considering the “tools” of dynamic geometry systems, the *coefficient of exactitude* loses its meaning, the constructions will be executed by the DGS, so they are accurate (exact). However, the *coefficient of simplicity* of the constructions can still be useful, it can be used to classify constructions by levels of simplicity. A new dimension can also be added, the *coefficient of freedom* (cf), given by the degree of freedom a given geometric object has, e.g. “a point in a line” has one degree of freedom, a point in the plane has two degrees of freedom, etc. This new coefficient will give a value to the dynamism of the geometric construction. The degrees of freedom are measured against the point definitions. The *point* definition, defines a point with two degrees of freedom, the *onsegment*, *online* and *oncircle* constructions, define points with one degree of freedom. For the *GCLC* constructions contained in *TGTP* an average value of simplicity (CS_{gcl}) of 20.8 was obtained. Using the k-means clustering function implemented in the statistics package of *Octave*,¹⁰ three classes of geometric constructions describing an increasing level of complexity were defined: simple constructions, $1 \leq CS_{gcl} \leq 18$; average complexity constructions, $18 < CS_{gcl} \leq 28$; complex constructions, $CS_{gcl} > 28$.

TGTP contain 71 simple constructions; 81 average complexity constructions; 28 complex constructions.

For example (*TGTP* problem’s GEO0369): “In triangle $\triangle ABC$, let F be the midpoint of the side BC , and D and E the feet of the altitudes on AB and AC , respectively. FG is perpendicular to DE at G . Show that G is the midpoint of DE ”, has a geometric construction with coefficient of simplicity 19 (see Fig. 2), so an average complexity construction. The value

¹⁰ GNU Octave, version 6.1.1, package octave-statistics, function kmeans <https://octave.sourceforge.io/statistics/function/kmeans.html>.

of 6 for its coefficient of freedom is given by the fact that only the three points A , B , and C are free in the plane, while all the other points are completely bind, by construction.

3.3 Geometrography in Automatic Theorem Proving

The same approach can be (again) extrapolated to take into consideration synthetic geometric proofs, i.e., proofs based on a geometric axiomatic theory, using geometric inference rules.

Considering the proofs produced by the GATP *GCLC*, implementing the area method [9, 10],¹¹ the coefficient of simplicity for all the axioms and lemmas of the theory can be calculated.

Apart from the geometric constructions in which the proof is based (with coefficient of simplicity $n\mathbf{Cnst}$), there are other steps to be considered.

- (Elementary) Algebraic Simplification. (**AS**)
- (Elementary) Geometric Simplification. (**GS**)
- Application of the Area Method Lemma n (**AML_n**)

A given proof can thus be measured against the number of those steps.¹² For a given proof expressed by the equation:

$$n_1\mathbf{Cnst} + n_2 \times \mathbf{AS} + n_3 \times \mathbf{GS} + \sum_{j=l_1}^{l_k} \mathbf{AML}_j$$

where n_1 is the *coefficient of simplicity* of the geometric construction, n_2 is the number of algebraic simplifications and n_3 is the number of geometric simplifications.

The coefficient of simplicity for the proof would be:

$$CS_{\text{proof}} = n_1 + n_2 + n_3 + \sum_{j=l_1}^{l_k} CS_{\text{proof}}(\mathbf{AML}_j)$$

The coefficient of freedom has no meaning in this setting.

Each lemma of the area method, \mathbf{AML}_j , has a corresponding simplicity coefficient, the term, $\sum_{j=l_1}^{l_k} CS_{\text{proof}}(\mathbf{AML}_j)$, is the sum of all those values, for all the lemmas used in the proof. In order to achieve this for each lemma of the area method the corresponding coefficients of simplicity were calculated [21].

For example, the proof of Lemma 9 will have the following coefficient of simplicity, $CS_{\text{proof}}(\mathbf{AML}_9) = 74$.

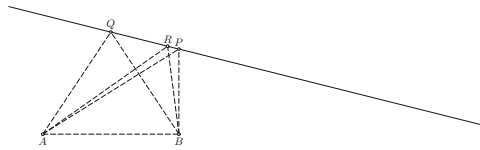
Lemma 1 (AML₉) *Let R be a point on the line PQ . Then for any two points A and B it holds that $S_{RAB} = \frac{PR}{PQ}S_{QAB} + \frac{RQ}{PQ}S_{PAB}$.*

¹¹ The proofs developed by GATPs based on the Area Method are formal proofs. The method itself was formalised, and proved sound, using the *Coq* proof assistant. The GATP developed by J. Narboux, as a *Coq* tactic, can have the proofs verified by *Coq*. The *GCLC* area method, do not have, explicitly, that possibility, but, it would be a matter of developing a filter from the *GCLC* language to the *Coq* language (see Appendix 1).

¹² By elementary algebraic simplification it is understood the basic algebraic operations: addition, subtraction, multiplication, division, and their properties of commutativity, associativity and distributivity. By elementary geometric simplification it is understood the direct application of the definition of the area method quantities. We call them *trivial steps*.

The following is a shorter version of its proof with the elementary algebraic and geometric simplifications condensed (the expanded version can be see in [21]).

Geometrography of Lemma 9 (AML₉)



$$CS_{gcl} = 22 = 4D + 18C$$

$$CF_{gcl} = 8$$

- $s = S_{ABPQ}$, initial construction;
- 1 × **GS**, areas of triangles with the same orientation, $S_{RAB} = s - S_{ARQ} - S_{BPR}$;
- 1 × **AML₁₄**, lemma 14, $\frac{PR}{PQ} = r$ (**AML₁₄** = 8);
- 1 × **AML₅**, lemma 5, $\frac{S_{ARQ}}{S_{APQ}} = \frac{RQ}{PQ}$ (**AML₅** = 18);
- 1 × **GS**, segments with the same orientation, $\frac{RQ}{PQ} = \frac{PQ-PR}{PQ}$;
- 2 × **AS**, algebraic simplifications, $\frac{PQ-PR}{PQ} = (1-r)$ and $S_{ARQ} = (1-r)S_{APQ}$;
- 1 × **AML₅**, lemma 5, $\frac{S_{BPR}}{S_{BPQ}} = \frac{PR}{PQ}$ (**AML₅** = 11);
- 2 × **AS**, algebraic simplifications, $S_{BPR} = rS_{BPQ}$ and $S_{RAB} = s - (1-r)S_{APQ} - rS_{BPQ}$;
- 2 × **GS**, areas of triangles with the same orientation, $S_{RAB} = s - (1-r)(s - S_{PAB}) - r(s - S_{QAB})$;
- 7 × **AS**, algebraic simplification, $S_{RAB} = s - s + rs + S_{PAB} - rS_{PAB} - rs + rS_{QAB}$, $S_{RAB} = rS_{QAB} + (1-r)S_{PAB}$ and $S_{RAB} = \frac{PR}{PQ}S_{QAB} + \frac{RQ}{PQ}S_{PAB}$.

Geometrography for the demonstration: 4D + 18C + 4GS + 11AS + 1AML₁₄ + 2AML₅

$$AML_9 \begin{cases} CS_{proof} = 74 = 22 + 4 + 11 + 8 + (18 + 11) \\ CS_{gcl} = 22 \end{cases}$$

with **AML₁₄** = 8 and **AML₅** = 18 (first application) and **AML₅** = 11 (second application).

It is considered that, from the second application of a lemma onward, its proof is accepted, so, only its adaptation to the new configuration is needed, i.e., the pattern matching of the lemma configuration to a new setting. For that reason, in any second, third, etc. application of a lemma, only the CS_{gcl} coefficient values are considered.

Given that a mathematical proof is a sequence of steps, in addition to the coefficient of simplicity, it would be useful to have other coefficients: e.g., the total number of steps in the proof; the value of the most difficult step in the proof; the number of different steps of high difficulty in the proof; the number of different types of steps (lemmas) in the proof; a proof script; a numerical description of the proof; and a corresponding line chart or proof trace.

Therefore, to fully characterise a formal synthetic proof produced by a GATP, we can define and consider the following coefficients:

- CS_{proof} , the simplicity coefficient (as above), it gives the simplicity coefficient for the overall proof;

- CT_{proof} , the total number of steps in the proof;
- CS_{proofmax} , the highest simplicity coefficient of the lemmas/definitions applications, it gives the simplicity coefficient for the most difficult step of the proof;
- $CD_{\text{typeproof}}$, the number of different types of lemmas used in the proof;
- $CD_{\text{highproof}}$, the number of different steps of high difficulty in the proof;
- The proof script, as defined above;
- The corresponding line chart or proof trace in *tikz* format.¹³

It is important to note that to obtain the coefficient $CD_{\text{highproof}}$ (hp) the area method lemmas implemented in the GATP *GCLC* were analysed, and, using the k-means clustering function implemented in the statistics package of *Octave*, divided into three categories: low difficulty ($hp < 284$), medium difficulty ($284 \leq hp < 1848$) and high difficulty ($hp \geq 1848$).

Using the defined coefficients above, we have the following values for the proof of **AML9**:

$$\text{AML9} \left\{ \begin{array}{l}
 CS_{\text{proof}} = 74 = 22 + 4 + 11 + 8 + (18 + 11) \\
 CS_{\text{gel}} = 22 \\
 CT_{\text{proof}} = 19 \\
 CS_{\text{proofmax}} = 18 \\
 CD_{\text{typeproof}} = 2 \\
 CD_{\text{highproof}} = 0
 \end{array} \right.$$

The GATP *GCLC* implementation of the area method [9, 10] is able to produce proof scripts. Using the command `prooflevel` it is possible to have control over the level of detail of the proof script. Two programs¹⁴ were implemented to calculate the *Geometrography* of the proofs. The *Geometrography* of the construction is calculated by a *bash* script, `gclcGeometrography.bash`, that analyses the *GCLC* geometric construction (not considering all the rendering commands). The *Geometrography* of the proof script (minus the geometric construction) is calculated by, `csproof`, a parser that analyse the proof script counting the algebraic steps and the geometric steps in sequence and also the lemmas and definitions of the area method with the respective coefficient of simplicity.

Using the program `csproof` on an arbitrary geometric proof, it can be obtained: a CSV file¹⁵ with the values regarding the *Geometric Readability Coefficient of Proofs* (see Sect. 4); a file with the coefficient of simplicity of the geometric construction; a file with a line chart, a graphical representation of the proof done by the GATP *GCLC*.

To better understand some details, let's consider again the Ceva's theorem (see Theorem 1). Using the GATP *GCLC*, with the full level of detail, the proof script of Ceva's theorem has all the details explained and it fills two pages, almost three pages, if the notes about the non-degeneracy conditions and about the proof itself are taken into consideration (see Appendix 1). The line chart is shown in Fig. 3. In it, the sequences of algebraic, or geometric, simplifications are condensed in only one step (for a more condensed view of the graph).

¹³ <https://ftp.eq.uq.pt/software/TeX/graphics/pgf/base/doc/pgfmanual.pdf>.

¹⁴ The open source codes are available in the GitHub project <https://github.com/GeoTiles/Geometrography/tree/master/GeometrographyProofs>.

¹⁵ Comma Separated Values format.

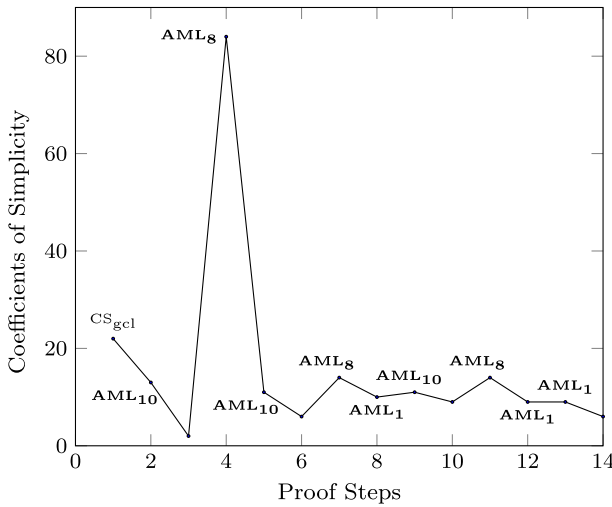


Fig. 3 Ceva’s Theorem, Geometrography Proof Trace

Therefore, the *Geometrography* of Ceva’s Theorem Proof is the following: $4D + 18C + 23AS + 3AML_1 + 3AML_8 + 3AML_{10}$.

$$\text{Ceva's Theorem} \left\{ \begin{array}{l} CS_{\text{proof}} = 220 \\ CS_{\text{gcl}} = 22 \\ CT_{\text{proof}} = 32 \\ CS_{\text{proofmax}} = 84 \\ CD_{\text{typeproof}} = 3 \\ CD_{\text{highproof}} = 0 \end{array} \right.$$

4 A Geometrographic Criterion

It is interesting to note how the Geometrographic coefficients highlight many salient aspects of the proof, aspects that could be used to analyse the readability of such proofs. Furthermore, it is interesting to stress how the proof trace constitutes a sort of electroencephalogram of the machine while proving the theorem. Just as an electroencephalogram can be useful for measuring a brain’s electrical activity, the line chart helps to understand some features of the proof by looking at its trace.

Applying the *Geometrography* to the area method proofs contained in the repository *TGTP*, using the *GATP GCLC* with the full level of detail, and using the geometrographic coefficients we can argue in favour of the following new readability coefficient: Geometrographic Readability Coefficient of Proofs (GRCP)

$$GRCP = ((CS_{\text{proof}} - CT_{\text{proof}}) \times (CD_{\text{highproof}} + CD_{\text{typeproof}}))$$

This coefficient relates four quantities: the simplicity coefficient of the proof, the total number of steps in the proof, the number of different steps with high-difficulty in the proof, the number of different lemmas used in the proof.

The first factor, $(CS_{\text{proof}} - CT_{\text{proof}})$, gives an approximation to the overall coefficient of simplicity of the non-trivial steps in the proof. Note that CT_{proof} count the number of

steps rather than the coefficient of simplicity of each step. By contrast, in CS_{proof} , it is the coefficient of simplicity that counts. Each trivial step has a coefficient of simplicity equal to one, and the coefficients of simplicity for non-trivial steps, such as the construction and the lemmas, are much greater than one. In the light of this, it can be concluded that the difference between CS_{proof} and CT_{proof} emphasises the complexity of the proof, disregarding its length.

The second factor, $(CD_{\text{highproof}} + CD_{\text{typeproof}})$, gives an account of the difficult steps. Steps that, potentially, make the proof much harder to follow, steps where the normal flow of the proof would be interrupted to jump to the proof of the lemma, resuming after completing the lemma’s proof. The addition of the number of high-difficulty steps with the number of different lemmas used in the proof, gave a multiplying factor for the overall complexity of the proof. A final note about this second factor: a high-difficulty step is, for sure, a lemma application, nevertheless we felt that the high-difficulty nature of the lemma is a sufficient reason for this double counting.

Multiplying these factors, the approximation for the overall simplicity coefficient and the number difficult steps—both elements that we believe characterise the readability of a proof—we obtain a readability coefficient of a proof.

Therefore, considering 71 theorems and their area method proofs, from the *TGTP* repository and using, again, the k-means clustering function from *Octave*, the proofs can be divided into the following classes of Geometrographic readability:¹⁶ readable (*high – readability*), $GRCP \leq 48000$; medium-readability $48000 < GRCP \leq 135000$; low-readability, $GRCP > 135000$.

The *GRCP* for GEO0001, Ceva’s proofs is: $GRCP_{\text{GEO0001}} = (220 - 32) \times (0 + 3) = 564 \leq 48000$, so a readable (high-readability) proof.

GRCP Medium-readability Example.

TGTP problem’s GEO0021:

Theorem 2 (Circumcenter of a Triangle) *The circumcenter of a triangle can be found as the intersection of the three perpendicular bisectors*

has the following values for the different coefficients.

$$\begin{array}{l}
 \text{GEO0021} \left\{ \begin{array}{l}
 CS_{\text{proof}} = 8554 \\
 CS_{\text{gcl}} = 11 \\
 CT_{\text{proof}} = 591 \\
 CS_{\text{proofmax}} = 2807 \\
 CD_{\text{typeproof}} = 13 \\
 CD_{\text{highproof}} = 3
 \end{array} \right. \\
 48000 < GRCP = 127408 \leq 135000
 \end{array}$$

By the *GRCP* criterion, this is a medium-readability problem. It can be seen that it has 13 different lemmas, 3 high-difficulty step, a long proof with a significant difference between the CS_{proof} and the number of steps of the proof (see Figs. 4 and 6).

4.1 GRCP low-readability Example.

TGTP problem’s GEO0020:

Theorem 3 (Distance of a line containing the centroid to the vertices) *Given a triangle ABC and a point X, the sum of the distances of the line XG, where G is the centroid of ABC, to*

¹⁶ The actual values were rounded for better readability.

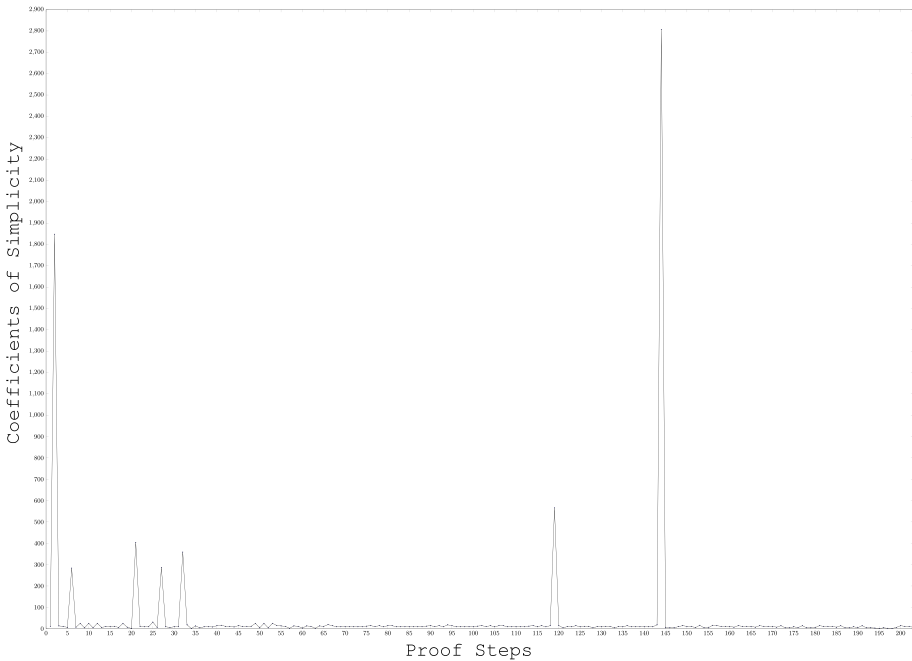


Fig. 4 *TGTP*, GEO0021, Circumcenter of a Triangle

the two vertices of the triangle situated on the same side of the line is equal to the distance of the line from the third vertex.

has the following values for the different coefficients:

$$\text{GEO0020} \left\{ \begin{array}{l} \text{CS}_{\text{proof}} = 19989 \\ \text{CS}_{\text{gcl}} = 26 \\ \text{CT}_{\text{proof}} = 4119 \\ \text{CS}_{\text{proofmax}} = 2807 \\ \text{CD}_{\text{typeproof}} = 13 \\ \text{CD}_{\text{highproof}} = 4 \end{array} \right.$$

$GRC P = 269790 > 135000$, so a low-readability problem. It can be seen that it has 13 different lemmas, 4 high-difficulty step, a long proof, with a very high value of overall complexity (see Figs. 5 and 7).

4.2 Comparing the Different Criteria

The *Geometrography Readability Coefficient of Proofs* criterion takes into consideration all the significant aspects of a formal proof, its overall difficulty, its number of steps, the number of difficult steps and the number of different lemmas that must be applied. The other criteria consider fewer aspects. The *de Bruijn criterion*, given its different goal, takes only in consideration the size of the proof and it needs to have an informal proof to compare with. The *ML* criterion considers the number of different lemmas applied and uses the number of

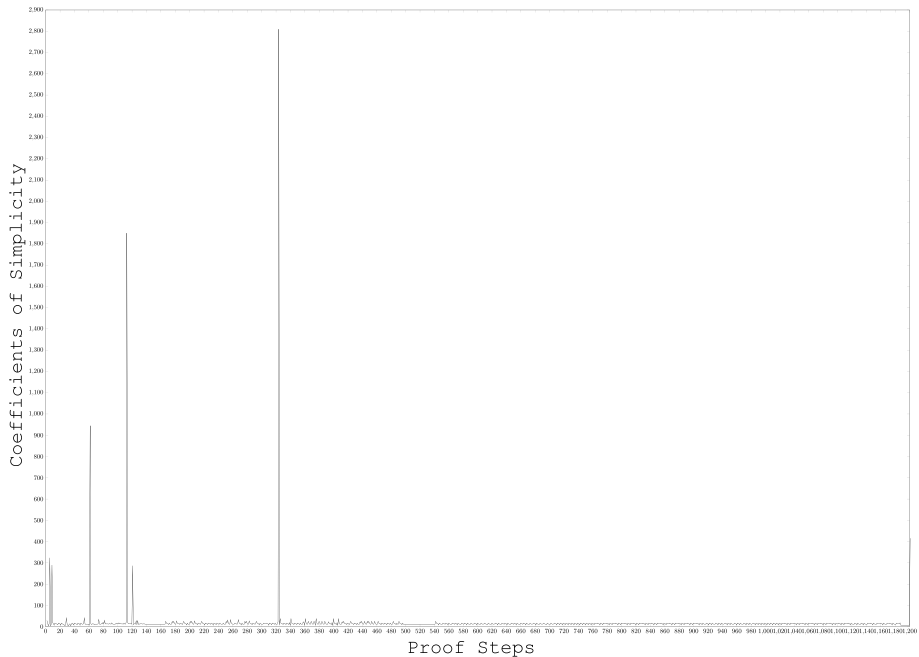


Fig. 5 TGTP, GEO0020, Distance of a line containing the centroid to the vertices

terms of the maximal polynomial as a way to have an approximation to the complexity of the proof.

Alongside the *ML* criterion, in the *GRCP* criterion, the number of lemmas in the proof is considered: in the *GRCP* criterion as a multiplicative factor, in the *ML* criterion as one of the conditions for readability. In the *ML* criterion the number of terms in the maximal polynomial are considered, but, as its authors remarked, this measures the number of computations needed in the proof, not its readability. This is weakly related to the number of steps in the proof. It approaches the number of steps needed to decompose those long polynomials occurring in the proof to a simple expression.

Regardless of this criteria comparison, we want to emphasise that the Geometric approach proposed in this paper has a more general scope. Although the *GRCP* criterion is a reasonable proposal, the elementary quality of the Geometric approach, through the analysis of various coefficients of the proofs, the proof scripts and the proof traces, makes it possible to have a language or a tool that can be used by non-experts to formulate other criteria weaker or stronger than the one we propose. The contribution of this paper is therefore not only that of a Geometric criterion, but of a Geometric approach to the problem of measuring the readability of formal proofs in automated deduction in geometry, an approach that offer an environment in which to analyse the proofs in detail by proposing and test readability criteria. To the best of our knowledge, it is the first time that the community has access to such a general tool to formulate and to study the readability of formal proofs in automated deduction in geometry. It is also interesting to note that our criterion offers a classification of proofs that is in line, when the fundamental points are considered, with the classifications given by the other two criteria. i.e., proofs that are classified as difficult to read according to the new criterion are also classified as difficult to read for the others, and the same applies to proofs that are easy to read (Table 2).

Table 2 Comparison of the three criteria

<i>TGTP</i>	ML	de Bruijn	GRCP
GEO0001	$3 < 5$, deduction steps easy	$1.6 < 2$ easy	$564 \leq 48000$ easy(high)
GEO0021	$13 > 5$ deduction steps & number of terms > 5 difficult	$37.63 > 2$ difficult	$127408 \leq 135000$ difficult(medium)
GEO0020	$13 > 5$ deduction steps & number of terms > 5 difficult	$47.31 > 2$ difficult	$269790 > 135000$ difficult(low)

Finally, we have to say that all the criteria proposed here have no empirical validation through the submission of tests to students, experts, etc. Nevertheless, the great advantage that our approach offers is that it allows to formulate criteria that can be implemented in repositories such as *TGTP* and can be evaluated experimentally in a very simple way.

5 Conclusions

In this paper we have analysed the problem of measuring the readability of formal proofs in automated deduction in geometry. We have introduced two known criteria and highlighted some of their limitations. We have then introduced a third criterion that seems to overcome the problem of readability by expert, therefore being more natural than the previous ones, and seems to be easily generalised. One possible generalisation is given by the possibility to formulate weaker, or stronger, criteria, using the proposed language. Another possible avenue is given by the generalisation of our approach to other GATPs (e.g. the JGEx integrated GATPs, area method, full-angle method and deductive databases method [3, 28, 29], ArgoCLP, coherent logic prover [25]) and any other ATP that has a proof script based on axioms, lemmas applications and, eventually, elementary steps (algebraic, geometric, etc.). It is a matter of calculation of the coefficients of simplicity for the axioms and lemmas of the base theory in consideration.

As we pointed out, the great advantage that our approach offers is that it allows to formulate criteria that can be implemented in repositories such as *TGTP* and evaluated experimentally. For this reason, an important work that we are planning is an experiment to be submitted to mathematicians, computer scientists, educationalists and students providing an adequate empirical test for our Geometrographic criterion.

Acknowledgements The authors wish to thank Salvatore Florio, Victor Pambuccian, and Mirko Tagliaferri for their comments on an earlier draft of this work.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Appendix

A Ceva’s Theorem, GCLC Area Method Proof

The area method for Euclidean constructive geometry was proposed by Chou, Gao and Zhang in the early 1990’s [2]. The method can efficiently prove many non-trivial geometry theorems and is one of the most interesting and most successful methods for automated theorem proving in geometry. In [10] a variant of the original axiom system was presented, based on that axiomatisation all the lemmas needed by the method were formally proved and the soundness of the method was established, using the *Coq* proof assistant [18].¹⁷

The GCLC implementation of the area method is able to produce formal proofs. If the highest level of details is chosen, `prooflevel 7`, it would be possible to (an appropriated filter has to be built) formally verify those proofs using a proof assistant, e.g. *Coq*. The \LaTeX proof scripts that GCLC produces (by default, at `prooflevel 2`) are a natural language rendering, to be read by mathematicians.

The area method axiomatic system for Euclidean plane geometry (within first order logic with equality), has just one primitive type of geometrical objects: points. Variables can also range over a field $(F, +, \cdot, 0, 1)$, where F is any field of characteristic different from 2. The axioms of the theory of fields used in GCLC area method proofs, are standard.

The Ceva’s proof presented below is a \LaTeX proof script produced by GCLC, at `prooflevel 7`, edited to include the GRCP values.

GCLC Prover Output for conjecture “cevaGEO0001”, Area method used

$$\begin{aligned} \left(\left(\frac{\vec{AF}}{\vec{FB}} \cdot \frac{\vec{BD}}{\vec{DC}} \right) \cdot \frac{\vec{CE}}{\vec{EA}} \right) &= 1 \quad \text{by the statement } (0) \\ &\quad \mathbf{22 = 4 \times D + 18 \times C} \\ \left(\left(\left(-1 \cdot \frac{\vec{AF}}{\vec{BF}} \right) \cdot \frac{\vec{BD}}{\vec{DC}} \right) \cdot \frac{\vec{CE}}{\vec{EA}} \right) &= 1 \quad \text{by Lemma } 10 \quad (1) \\ &\quad \mathbf{AML_{10} = 13} \\ \left(\left(-1 \cdot \frac{\vec{AF}}{\vec{BF}} \right) \cdot \left(\frac{\vec{BD}}{\vec{DC}} \cdot \frac{\vec{CE}}{\vec{EA}} \right) \right) &= 1 \quad \text{by right association } (2) \\ &\quad \mathbf{1 \times AS} \\ \left(-1 \cdot \left(\frac{\vec{AF}}{\vec{BF}} \cdot \left(\frac{\vec{BD}}{\vec{DC}} \cdot \frac{\vec{CE}}{\vec{EA}} \right) \right) \right) &= 1 \quad \text{by right association } (3) \\ &\quad \mathbf{1 \times AS} \\ \left(-1 \cdot \left(\frac{S_{APC}}{S_{BPC}} \cdot \left(\frac{\vec{BD}}{\vec{DC}} \cdot \frac{\vec{CE}}{\vec{EA}} \right) \right) \right) &= 1 \quad \text{by Lemma 8 (point } F \text{ eliminated)} \quad (4) \\ &\quad \mathbf{AML_8 = 84} \\ \left(-1 \cdot \left(\frac{S_{APC}}{S_{BPC}} \cdot \left(\frac{\vec{BD}}{\vec{DC}} \cdot \left(-1 \cdot \frac{\vec{CE}}{\vec{AE}} \right) \right) \right) \right) &= 1 \quad \text{by Lemma } 10 \quad (5) \\ &\quad \mathbf{AML_{10} = 13} \\ \left(-1 \cdot \frac{\left(S_{APC} \cdot \left(\frac{\vec{BD}}{\vec{DC}} \cdot \left(-1 \cdot \frac{\vec{CE}}{\vec{AE}} \right) \right) \right)}{S_{BPC}} \right) &= 1 \quad \text{by multiplication of fractions } (6) \\ &\quad \mathbf{1 \times AS} \end{aligned}$$

¹⁷ Narboux, J.: Formalization of the area method. *Coq* user contribution (2009). http://dpt-info.u-strasbg.fr/~narboux/area_method.html.

$$\frac{\left(-1 \cdot \left(S_{APC} \cdot \left(\frac{\vec{BD}}{DC} \cdot \left(-1 \cdot \frac{\vec{CE}}{AE}\right)\right)\right)\right)}{S_{BPC}} = 1 \quad \begin{array}{l} \text{by multiplication of frac-} \\ \text{tions } \mathbf{1} \times \mathbf{AS} \end{array} \quad (7)$$

$$\frac{\left(-1 \cdot \left(S_{APC} \cdot \left(-1 \cdot \left(\frac{\vec{BD}}{DC} \cdot \frac{\vec{CE}}{AE}\right)\right)\right)\right)}{S_{BPC}} = 1 \quad \begin{array}{l} \text{by associativity and com-} \\ \text{mutativity } \mathbf{1} \times \mathbf{AS} \end{array} \quad (8)$$

$$\frac{\left(-1 \cdot \left(-1 \cdot \left(S_{APC} \cdot \left(\frac{\vec{BD}}{DC} \cdot \frac{\vec{CE}}{AE}\right)\right)\right)\right)}{S_{BPC}} = 1 \quad \begin{array}{l} \text{by associativity and com-} \\ \text{mutativity } \mathbf{1} \times \mathbf{AS} \end{array} \quad (9)$$

$$\frac{\left(1 \cdot \left(S_{APC} \cdot \left(\frac{\vec{BD}}{DC} \cdot \frac{\vec{CE}}{AE}\right)\right)\right)}{S_{BPC}} = 1 \quad \begin{array}{l} \text{by multiplication of con-} \\ \text{stants } \mathbf{1} \times \mathbf{AS} \end{array} \quad (10)$$

$$\frac{\left(S_{APC} \cdot \left(\frac{\vec{BD}}{DC} \cdot \frac{\vec{CE}}{AE}\right)\right)}{S_{BPC}} = 1 \quad \begin{array}{l} \text{by multiplication by } 1 \\ \mathbf{1} \times \mathbf{AS} \end{array} \quad (11)$$

$$\frac{\left(S_{APC} \cdot \left(\frac{\vec{BD}}{DC} \cdot \frac{S_{CPB}}{S_{APB}}\right)\right)}{S_{BPC}} = 1 \quad \begin{array}{l} \text{by Lemma 8 (point} \\ E \text{ eliminated)} \\ \mathbf{AML}_8 = \mathbf{14} \end{array} \quad (12)$$

$$\frac{\left(S_{APC} \cdot \left(\frac{\vec{BD}}{DC} \cdot \frac{S_{CPB}}{S_{APB}}\right)\right)}{(-1 \cdot S_{CPB})} = 1 \quad \begin{array}{l} \text{by Lemma } 1 \\ \mathbf{AML}_1 = \mathbf{10} \end{array} \quad (13)$$

$$\frac{\left(S_{APC} \cdot \left(\left(-1 \cdot \frac{\vec{BD}}{CD}\right) \cdot \frac{S_{CPB}}{S_{APB}}\right)\right)}{(-1 \cdot S_{CPB})} = 1 \quad \begin{array}{l} \text{by Lemma } 10 \\ \mathbf{AML}_{10} = \mathbf{11} \end{array} \quad (14)$$

$$\frac{\left(S_{APC} \cdot \left(\left(1 \cdot \frac{\vec{BD}}{CD}\right) \cdot \frac{S_{CPB}}{S_{APB}}\right)\right)}{(1 \cdot S_{CPB})} = 1 \quad \begin{array}{l} \text{by ratio cancellation} \\ \mathbf{1} \times \mathbf{AS} \end{array} \quad (15)$$

$$\frac{\left(S_{APC} \cdot \left(\frac{\vec{BD}}{CD} \cdot \frac{S_{CPB}}{S_{APB}}\right)\right)}{(1 \cdot S_{CPB})} = 1 \quad \begin{array}{l} \text{by multiplication by } 1 \\ \mathbf{1} \times \mathbf{AS} \end{array} \quad (16)$$

$$\frac{\left(S_{APC} \cdot \left(\frac{\vec{BD}}{CD} \cdot \frac{S_{CPB}}{S_{APB}}\right)\right)}{S_{CPB}} = 1 \quad \begin{array}{l} \text{by multiplication by } 1 \\ \mathbf{1} \times \mathbf{AS} \end{array} \quad (17)$$

$$\frac{\left(S_{APC} \cdot \frac{\left(\frac{\vec{BD}}{CD} \cdot S_{CPB}\right)}{S_{APB}}\right)}{S_{CPB}} = 1 \quad \begin{array}{l} \text{by multiplication of frac-} \\ \text{tions } \mathbf{1} \times \mathbf{AS} \end{array} \quad (18)$$

$$\frac{\left(S_{APC} \cdot \left(\frac{\vec{BD}}{CD} \cdot S_{CPB}\right)\right)}{S_{APB}} = 1 \quad \begin{array}{l} \text{by multiplication of frac-} \\ \text{tions } \mathbf{1} \times \mathbf{AS} \end{array} \quad (19)$$

$$\frac{\left(S_{APC} \cdot \left(\frac{\vec{BD}}{CD} \cdot S_{CPB}\right)\right)}{(S_{APB} \cdot S_{CPB})} = 1 \quad \begin{array}{l} \text{by multiple fraction sim-} \\ \text{plification } \mathbf{1} \times \mathbf{AS} \end{array} \quad (20)$$

$$\frac{\left(S_{APC} \cdot \left(\frac{\vec{BD}}{CD} \cdot 1\right)\right)}{(S_{APB} \cdot 1)} = 1 \quad \begin{array}{l} \text{by ratio} \\ \mathbf{1 \times AS} \end{array} \text{cancellation (21)}$$

$$\frac{\left(S_{APC} \cdot \frac{\vec{BD}}{CD}\right)}{(S_{APB} \cdot 1)} = 1 \quad \begin{array}{l} \text{by multiplication by } 1 \\ \mathbf{1 \times AS} \end{array} \text{(22)}$$

$$\frac{\left(S_{APC} \cdot \frac{\vec{BD}}{CD}\right)}{S_{APB}} = 1 \quad \begin{array}{l} \text{by multiplication by } 1 \\ \mathbf{1 \times AS} \end{array} \text{(23)}$$

$$\frac{\left(S_{APC} \cdot \frac{S_{BPA}}{S_{CPA}}\right)}{S_{APB}} = 1 \quad \begin{array}{l} \text{by Lemma 8 (point} \\ D \text{ eliminated)} \\ \mathbf{AML_8 = 14} \end{array} \text{(24)}$$

$$\frac{\left(S_{APC} \cdot \frac{S_{BPA}}{(-1 \cdot S_{APC})}\right)}{S_{APB}} = 1 \quad \begin{array}{l} \text{by Lemma} \\ \mathbf{AML_1 = 9} \end{array} \text{(25)}$$

$$\frac{\left(S_{APC} \cdot \frac{S_{BPA}}{(-1 \cdot S_{APC})}\right)}{(-1 \cdot S_{BPA})} = 1 \quad \begin{array}{l} \text{by Lemma} \\ \mathbf{AML_1 = 9} \end{array} \text{(26)}$$

$$\frac{\frac{(S_{APC} \cdot S_{BPA})}{(-1 \cdot S_{APC})}}{(-1 \cdot S_{BPA})} = 1 \quad \begin{array}{l} \text{by multiplication of frac-} \\ \text{tions} \\ \mathbf{1 \times AS} \end{array} \text{(27)}$$

$$\frac{\frac{(1 \cdot S_{BPA})}{(-1 \cdot 1)}}{(-1 \cdot S_{BPA})} = 1 \quad \begin{array}{l} \text{by ratio} \\ \mathbf{1 \times AS} \end{array} \text{cancellation (28)}$$

$$\frac{\frac{S_{BPA}}{(-1 \cdot 1)}}{(-1 \cdot S_{BPA})} = 1 \quad \begin{array}{l} \text{by multiplication by } 1 \\ \mathbf{1 \times AS} \end{array} \text{(29)}$$

$$\frac{\frac{S_{BPA}}{-1}}{(-1 \cdot S_{BPA})} = 1 \quad \begin{array}{l} \text{by multiplication by } 1 \\ \mathbf{1 \times AS} \end{array} \text{(30)}$$

$$\frac{(-1 \cdot S_{BPA})}{(-1 \cdot S_{BPA})} = 1 \quad \begin{array}{l} \text{by fraction with number} \\ \text{denominator} \\ \mathbf{1 \times AS} \end{array} \text{(31)}$$

$$1 = 1 \quad \begin{array}{l} \text{by ratio} \\ \mathbf{1 \times AS} \end{array} \text{cancellation (32)}$$

Q.E.D

NDG conditions are:

$S_{BPA} \neq S_{CPA}$ i.e., lines BC and PA are not parallel (construction based assumption)

$S_{APB} \neq S_{CPB}$ i.e., lines AC and PB are not parallel (construction based assumption)

$S_{APC} \neq S_{BPC}$ i.e., lines AB and PC are not parallel (construction based assumption)

$P_{FBF} \neq 0$ i.e., points F and B are not identical (conjecture based assumption)

$P_{DCD} \neq 0$ i.e., points D and C are not identical (conjecture based assumption)
 $P_{EAE} \neq 0$ i.e., points E and A are not identical (conjecture based assumption)
 Number of elimination proof steps: 3
 Number of geometric proof steps: 6
 Number of algebraic proof steps: 23
 Total number of proof steps: 32
 Time spent by the prover: 0.001 seconds

Enlarged Proof Traces

TGTP: GEO0021 See Fig. 6.

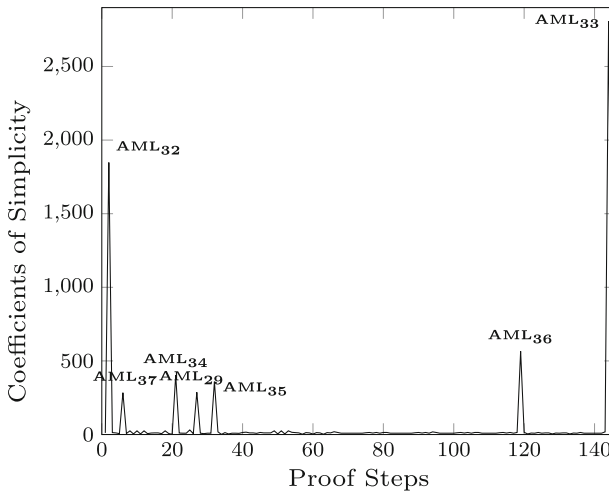


Fig. 6 Medium-readability—*TGTP*, GEO0021, the first 146, out of 203, steps of the proof

TGTP: GEO0020 See Fig. 7.

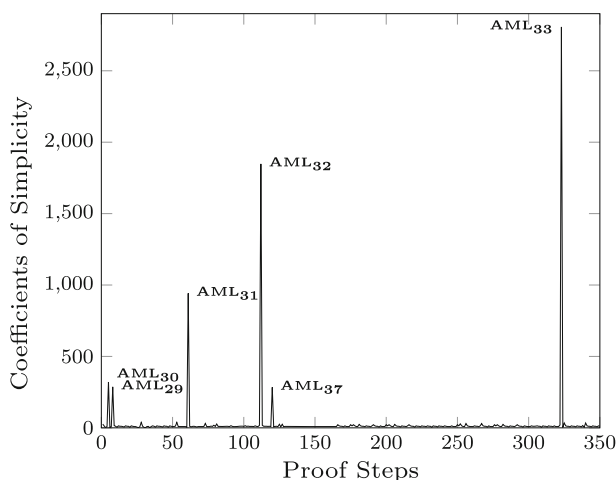


Fig. 7 Low-readability—*TGTP*, GEO0020. The first 350, out of 1200, steps of the proof

References

1. Chou, S.C., Gao, X.S., Zhang, J.Z.: *Machine Proofs in Geometry*. World Scientific, Singapore (1994)
2. Chou, S.C., Gao, X.S., Zhang, J.Z.: Automated generation of readable proofs with geometric invariants, I. Multiple and shortest proof generation. *J. Automat. Reason.* **17**, 325–347 (1996). <https://doi.org/10.1007/BF00283134>
3. Chou, S.C., Gao, X.S., Zhang, J.Z.: Automated generation of readable proofs with geometric invariants, II. Theorem proving with full-angles. *J. Automat. Reason.* **17**(3), 349–370 (1996). <https://doi.org/10.1007/BF00283134>
4. de Bruijn, N.: *Selected Papers on Automath, Studies in Logic and the Foundations of Mathematics*, vol. 133, chap. A survey of the project Automath, pp. 141–161. Elsevier, Amsterdam (1994). [https://doi.org/10.1016/S0049-237X\(08\)70203-9](https://doi.org/10.1016/S0049-237X(08)70203-9)
5. DuBay, W.H. (ed.): *The Classic Readability Studies*. Impact Information (2006)
6. Gao, H., Li, J., Cheng, J.: Measuring interestingness of theorems in automated theorem finding by forward reasoning based on strong relevant logic. In: 2019 IEEE International Conference on Energy Internet (ICEI), pp. 356–361 (2019). <https://doi.org/10.1109/ICEI.2019.00069>
7. Hanna, G., Reid, D., de Villiers, M. (eds.): *Proof Technology in Mathematics Research and Teaching*. Springer, Berlin (2019)
8. Hohenwarter, M.: *Geogebra—a software system for dynamic geometry and algebra in the plane*. Master’s thesis, University of Salzburg, Austria (2002)
9. Janičić, P.: GCLC - A tool for constructive Euclidean geometry and more than that. In: A. Iglesias, N. Takayama (eds.) *Mathematical Software - ICMS 2006, Lecture Notes in Computer Science*, vol. 4151, pp. 58–73. Springer (2006). https://doi.org/10.1007/11832225_6
10. Janičić, P., Narboux, J., Quaresma, P.: The area method: a recapitulation. *J. Automat. Reason.* **48**(4), 489–532 (2012). <https://doi.org/10.1007/s10817-010-9209-7>
11. Janičić, P., Quaresma, P.: System description: GCLCprover + GeoThms. In: U. Furbach, N. Shankar (eds.) *Automated Reasoning, Lecture Notes in Computer Science*, vol. 4130, pp. 145–150. Springer (2006). https://doi.org/10.1007/11814771_13
12. Jiang, J., Zhang, J.: A review and prospect of readable machine proofs for geometry theorems. *J. Syst. Sci. Complexity* **25**(4), 802–820 (2012). <https://doi.org/10.1007/s11424-012-2048-3>
13. Johnson, D.A.: The readability of mathematics books. *Math. Teach.* **50**(2), 105–110 (1957). <https://doi.org/10.5951/MT.50.2.0105>
14. Lemoine, É.: *Géométrie ou Art des constructions géométriques*. No. 18 in *Scientia, Série Physico-Mathématique*. C. Naud, Éditeur, Paris (1902). <http://catalogue.bnf.fr/ark:/12148/cb36049032t>
15. Loria, G.: *La geometrografia e le sue trasformazioni*. *Period. Mat.* **3**(6), 114–122 (1908)
16. Mackay, J.S.: The geometrography of Euclid’s problems. *Proc. Edinb. Math. Soc.* **12**, 2–16 (1893). <https://doi.org/10.1017/S0013091500001565>

17. Merikoski, J.K., Tossavainen, T.: Two approaches to geometrography. *J. Geom. Graph.* **13**(1), 15–28 (2010)
18. Narboux, J.: A decision procedure for geometry in Coq. *Lect. Notes Comput. Sci.* **3223**, 225–240 (2004). <https://doi.org/10.1007/b100400>
19. Pak, K., Schubert, A.: The impact of proof steps sequence on proof readability—experimental setting. In: *Workshop and Work in Progress Papers at CICM 2016* (2016)
20. Pinheiro, V.A.: *Geometrografia 1*. Gráfica Editora Bahiense (1974)
21. Quaresma, P., Graziani, P.: The geometrography’s simplicity coefficient for the axioms and lemma of the area method. Technical Report TR 2021-001, Center for Informatics and Systems of the University of Coimbra (2021)
22. Quaresma, P., Santos, V., Graziani, P., Baeta, N.: Taxonomy of geometric problems. *J. Symbol. Comput.* **97**, 31–55 (2020). <https://doi.org/10.1016/j.jsc.2018.12.004>
23. Richard, P., Vélez, M., Vaerenbergh, S.V. (eds.): *Mathematics Education in the Age of Artificial Intelligence, Mathematics Education in the Digital Era*, vol. 17. Springer (2022). <https://doi.org/10.1007/978-3-030-86909-0>
24. Santos, V., Baeta, N., Quaresma, P.: Geometrography in dynamic geometry. *Int. J. Technol. Math. Educ.* **26**(2), 89–96 (2019). https://doi.org/10.1564/tme_v26.2.06
25. Stojanović, S., Pavlović, V., Janičić, P.: A coherent logic based geometry theorem prover capable of producing formal and readable proofs. In: P. Schreck, J. Narboux, J. Richter-Gebert (eds.) *Automated Deduction in Geometry, Lecture Notes in Computer Science*, vol. 6877, pp. 201–220. Springer, Berlin (2011). https://doi.org/10.1007/978-3-642-25070-5_12
26. Wang, K., Su, Z.: Automated geometry theorem proving for human-readable proofs. In: *Proceedings of the 24th International Conference on Artificial Intelligence, IJCAI’15*, pp. 1193–1199. AAAI (2015). <http://dl.acm.org/citation.cfm?id=2832249.2832414>
27. Wiedijk, F.: The de Bruijn factor. Poster at International Conference on Theorem Proving in Higher Order Logics (TPHOL2000) (2000). Portland, Oregon, USA, 14–18 August 2000
28. Ye, Z., Chou, S.C., Gao, X.S.: Visually dynamic presentation of proofs in plane geometry: Part 2. Automated generation of visually dynamic presentations with the full-angle method and the deductive database method. *J. Automat. Reason.* **45**(3), 243–266 (2010). <https://doi.org/10.1007/s10817-009-9163-4>
29. Ye, Z., Chou, S.C., Gao, X.S.: An introduction to Java geometry expert. In: T. Sturm, C. Zengler (eds.) *Automated Deduction in Geometry, Lecture Notes in Computer Science*, vol. 6301, pp. 189–195. Springer, Berlin (2011). https://doi.org/10.1007/978-3-642-21046-4_10

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.