South Bohemia Mathematical Letters
Volume 25, (2017), No. 1, 48-65.

AUTOMATED REASONING TOOLS IN GEOGEBRA: A NEW APPROACH FOR EXPERIMENTS IN PLANAR GEOMETRY

ZOLTÁN KOVÁCS

Abstract. Computing numerical checks of certain relations between geometric objects in a planar construction is a well known feature of dynamic geometry systems. GeoGebra's newest improvements offer symbolic checks of equality, parallelism, perpendicularity, collinearity, concurrency or concyclicity. Also dragging of locus curves, defined explicitly or implicitly (or as an envelope curve) is a new feature in GeoGebra to visually check conjectures in planar geometry. By combining plotting and proving we can focus on some new possibilities to teach Euclidean geometry in the classrooms.

Introduction

Automated deduction of known or not yet discovered mathematical results has a wide literature since the appearance of the first computers. In particular, in planar Euclidean geometry the first successful attempts date back to the 1950's [10] that led to a line of work within the artificial intelligence context. Another important approach that was based on algebraic geometry methods was started by Wen-Tsün Wu and his followers including Chou [9], and—focusing on the Gröbner bases method—Kapur [13], Kutzler and Stifter [19], among others.

Dynamic geometry software tools became very popular and well known since the breakthrough of home use of personal computers. Beginning with the *Geometric Supposer* [22] in 1981, widely used tools were available commercially including *The Geometer's Sketchpad* [12], *Cabri Geometry* [2] and *Cinderella* [14]. Another breakthrough, *GeoGebra*'s [11] free availability for millions of users, opened the road to consider dynamic geometry as a natural education tool in the classroom.

Combining automated deduction in geometry (ADG) and dynamic geometry (DG) is a somewhat newer topic, but it was already present as more than a research concept since the 1990's in the first versions of several DG tools [25]. The first approaches, however, mostly used numerical and statistical methods for verifications. Just later, since the second half of the 1990's appeared software tools that used pure symbolic methods [26, 3, 20] in proving or visualizing geometric facts.

Today's wide availability of DG software led to a natural requirement: to have symbolic support in the most popular DG applications. In this paper we focus on this movement, in particular, on the extension of GeoGebra with automated reasoning tools (ART). This movement is just the most recent step of a long story, including Gelernter's legacy and Wu's pioneer research, and the work of many more,

---

Date: 15 January 2018.
2010 *Mathematics Subject Classification.* 14A10, 13P10, 68T15, 68N17.
Key words and phrases. Dynamic geometry, automated deduction in geometry, Gröbner bases, mathematics education, locus, envelope.

not mentioned here. (See [5] for a more detailed overview.) Among others, symbolic proof support in GeoGebra was initiated by Recio, Botana and Abánades—as a very first step, they supported Sergio Arbeo, a student programmer participating in the Google Summer of Code project in 2010.

The main focus of this paper is on demonstrating how GeoGebra ART could be used as an education tool. After some theoretical overview in Section 1 some concrete examples will be shown in Section 2.

1. Some theoretical overview

The main idea about proving planar geometry statements by using indirect proof is to use an algebraic translation of the construction and describe the hypotheses $h_1, h_2, \ldots$ by using algebraic equations. The thesis t will also be considered as an algebraic equation. Now an algebraic step, namely the Rabinowitsch trick [21] is used to negate t, and by eliminating the variables corresponding to the free coordinates, an algebraic constraint can be obtained that describes sufficient conditions for the statement to be true. The method is shown in details in [24] and [16].

Similarly, to obtain a symbolic equation to describe a locus equation ℓ depending on a geometric construction conducted by a mover point M, we need to use elimination again (see [3]). Here the algebraic set ℓ contains all possible points of the traced point T while M sweeps all possible positions under its constraints.

Locus equations can be further developed in two directions. The first way is to compute an envelope equation e that describes a geometric set which is defined by its tangent lines t'—they behave exactly like the tracer points T in locus equations, that is, a mover point M conducts the movement of t' and the envelope of t' is to be found. More on this concept can be found in [4] and [15].

The other extension is to compute an *implicit* locus equation. In some sense this is also an extension of proving, but in a less general way: there is only one free (or semifree) point P given, and the computer needs to discover it to have a prescribed geometric property p fulfilled. Usually, the found set is a curve, in particular, an algebraic one, which is computed again by using elimination. Further details can be found in [1].

We call these four kinds of challenges *proving*, *explicit locus*, *envelope* and *implicit locus* computations. We illustrate the first two ones with the following simple examples.

Example 1.1. In Fig. 1 a possible way to set up the algebraic equations as hypotheses is listed to prove that the heights of a triangle are concurrent.

First of all, the free points will be described by six free cordinates, namely $A(v_1, v_2)$, $B(v_3, v_4)$ and $C(v_5, v_6)$. To define two heights as vectors $\overrightarrow{BB_1}$ and $\overrightarrow{CC_1}$, we need two points having defined: $B_1(v_7, v_8)$ and $C_1(v_9, v_{10})$, and then we compute the rotation equations by using vectors $\overrightarrow{AC}$ and $\overrightarrow{BA}$, rotating them by 90 degrees counterclockwise. The required equations are: $h_1 = -v_8 - v_5 + v_4 + v_1 = 0$, $h_2 = -v_7 + v_6 + v_3 - v_2 = 0$, $h_3 = -v_{10} + v_6 + v_3 - v_1 = 0$ and $h_4 = -v_9 + v_5 - v_4 + v_2 = 0$. Now let $D(v_{11}, v_{12})$ be the intersection of the heights BB_1 and CC_1. Since D, B, B_1 and D, C, C_1 are collinear points, we can set up two more equations to describe D: $h_5 = v_{11}v_8 - v_{12}v_7 - v_{11}v_4 + v_7v_4 + v_{12}v_3 - v_8v_3 = 0$ and $h_6 = v_{11}v_{10} - v_{12}v_9 - v_{11}v_6 + v_9v_6 + v_{12}v_5 - v_{10}v_5 = 0$.

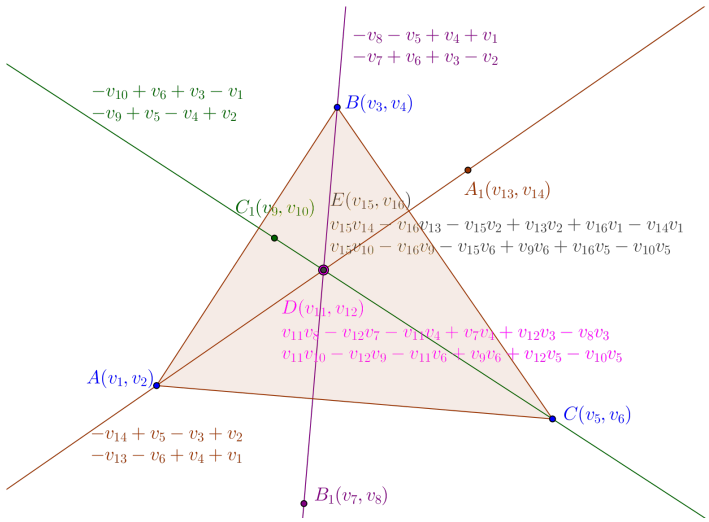

FIGURE 1. Proving that the heights of a triangle are concurrent.

In a similar manner the point $A_1(v_{13}, v_{14})$ can also be defined and for the vector $\overrightarrow{AA_1}$ the equations $h_7 = -v_{14} + v_5 - v_3 + v_2 = 0$ and $h_8 = -v_{13} - v_6 + v_4 + v_1 = 0$ can be stated. Analogously, the point $E(v_{15}, v_{16})$ can be defined as intersection of lines AA_1 and CC_1, that is, $h_9 = v_{15}v_{14} - v_{16}v_{13} - v_{15}v_2 + v_{13}v_2 + v_{16}v_1 - v_{14}v_1 = 0$ and $h_{10} = v_{15}v_{10} - v_{16}v_9 - v_{15}v_6 + v_9v_6 + v_{16}v_5 - v_{10}v_5 = 0$.

The thesis will now assert that $D = E$, that is, $(v_{11}, v_{12}) = (v_{15}, v_{16})$. The negation of this statement is $(v_{11}, v_{12}) \neq (v_{15}, v_{16})$, that is $v_{11} \neq v_{15} \vee v_{12} \neq v_{16}$. By using Rabinowitsch's trick it is easy to see that this is equivalent to the solvability of $\tilde{t} = ((v_{11} - v_{15}) \cdot v_{17} - 1) \cdot ((v_{12} - v_{16}) \cdot v_{17} - 1) = 0$, after introducing the technical variable v_{17}.

Without loss of generality it can be assumed that $A = (0, 0)$, that is, $v_1 = v_2 = 0$. Now by eliminating all variables from $h_1, h_2, \ldots, h_{10}$ and $\tilde{t}$ except the free ones, the equation system

$$\begin{aligned}
v_4 \cdot (v_6 - v_4) \cdot (v_5^2 + v_6^2) &= 0,\\
v_5 \cdot (v_6 v_4 + v_5 v_3 - v_3^2 - v_4^2) &= 0,\\
v_4 \cdot (v_3 v_5 - v_5^2 + v_4 v_6 - v_6^2) &= 0,\\
v_3 v_6 - v_5 v_4 &= 0
\end{aligned}$$

can be obtained which is surely not solvable if the last equation is not solvable, that is, $v_3 v_6 - v_5 v_4 \neq 0$. This final inequality states that the points A, B, C are not collinear. That is, to sum up, if they are not collinear (that is, the triangle is non-degenerate), then the equation system does not have a solution, and ad absurdum it follows that the heights of the triangle are collinear.

Example 1.2. A possible definition of a parabola can be described as given in Fig. 2. The directrix can be defined by joining free points $A(v_1, v_2)$ and $B(v_3, v_4)$, and the focus point given as point $C(v_5, v_6)$. The task is to express point $H(x, y)$ as an algebraic equation of the variables x and y if $v_1, v_2, \ldots, v_6$ are substituted by some rational numbers. Here point H plays the role of the tracer point ("T") and D the mover point ("M").

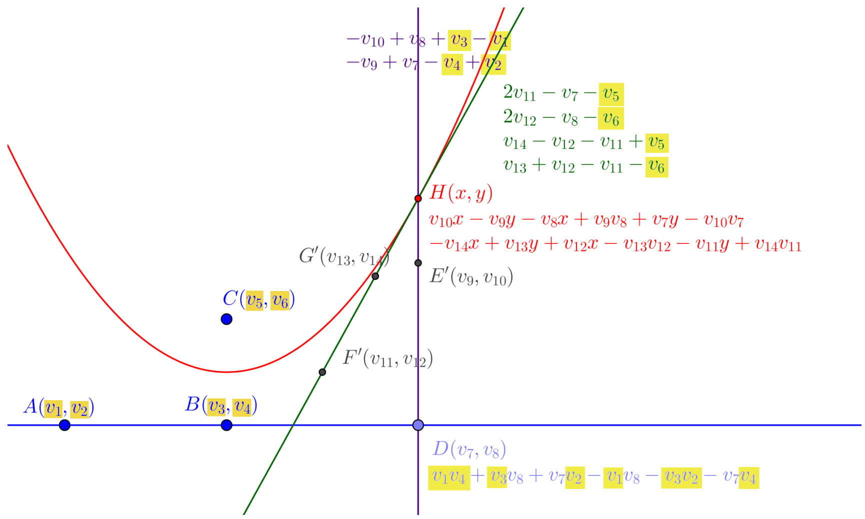

FIGURE 2. A definition of a parabola as an explicit locus. The variables with yellow labels will be substituted by the exact coordinates on each dragging of the free points.

Clearly, the following equations can be stated: Since $D(v_7, v_8)$ is a point lying on the directrix, $h_1 = v_1v_4 + v_3v_8 + v_7v_2 - v_1v_8 - v_3v_2 - v_7v_4 = 0$ holds. Let $\overrightarrow{DE'}$ be the rotation of $\overrightarrow{AB}$ by 90 degrees counterclockwise. To achieve this, the equations $h_2 = -v_{10} + v_8 + v_3 - v_1 = 0$ and $h_3 = -v_9 - v_7 - v_4 + v_2 = 0$ can also be written if point E' has coordinates v_9 and v_{10}. On the other hand, $F'(v_{11}, v_{12})$ can play the role of the midpoint of CD if the equations $h_4 = 2v_{11} - v_7 - v_5 = 0$ and $h_5 = 2v_{12} - v_8 - v_6 = 0$ are given. By defining $G'(v_{13}, v_{14})$ with two further equations, $h_6 = v_{14} - v_{12} - v_{11} + v_5 = 0$ and $h_7 = v_{13} + v_{12} - v_{11} - v_6 = 0$ we claim that $\overrightarrow{F'G'}$ is a rotation of $\overrightarrow{CF'}$ by 90 degrees counterclockwise.

Since the lines DE' and $F'G'$ are now explicitly given, their intersection can be found by using two more equations, namely $h_8 = v_{10}x - v_9y - v_8x + v_9v_8 + v_7y - v_{10}v_7 = 0$ and $h_9 = v_{14}x + v_{13}y + v_{12}x - v_{13}v_{12} - v_{11}y + v_{14}v_{11} = 0$ to express that $H \in DE'$ and $H \in F'G'$, respectively.

Now, by using elimination after A, B and C are explicitly defined, an algebraic equation will be obtained. Here the eliminated variables will be all appearing ones but x and y. A concrete example can be seen in Fig. 3 using substitutions $A = (1, 1)$, $B = (3, 1)$, $C = (3, 4)$—the coordinates of these free points can be changed as an online experiment at `https://www.geogebra.org/m/HUtpsN7P`.

We must emphasize that in all four challenges effective computations are very important. That is, fast computation of an elimination task is the main bottleneck to have a useful tool also in the classroom. Here we refer to some milestones in the practical development of the theory of Gröbner bases computation [7, 17, 15].

2. Automated Reasoning Tools in the classroom

2.1. **GeoGebra commands.** The four challenges mentioned above are implemented in the dynamic mathematics tool GeoGebra as the high level **Relation**

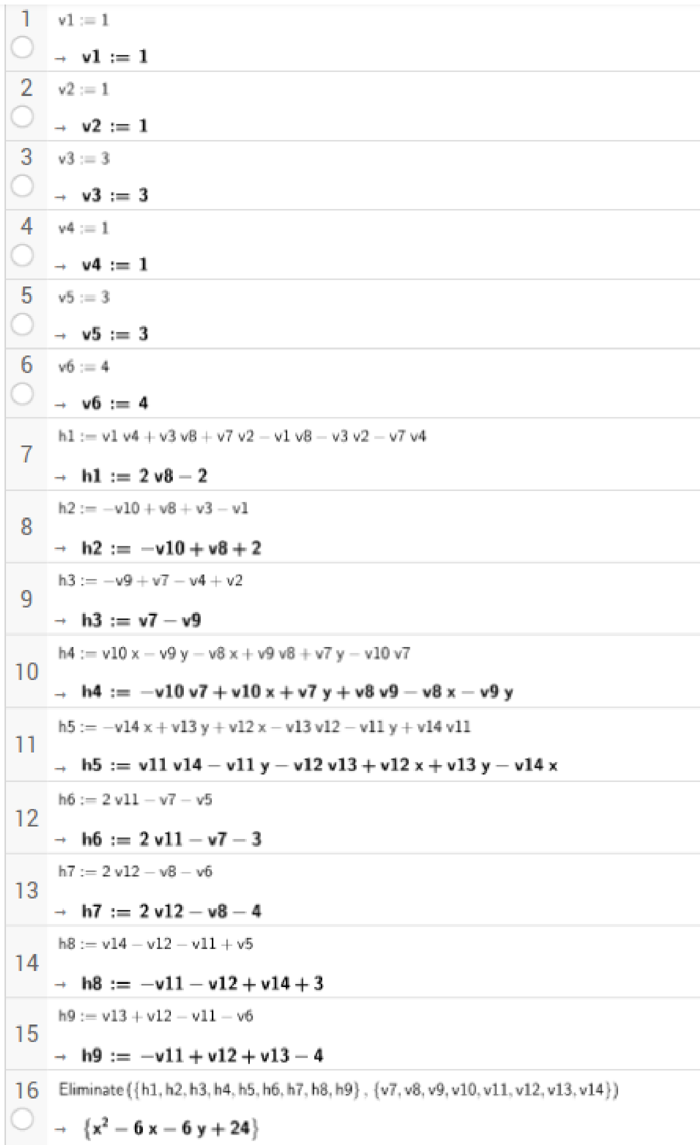

FIGURE 3. Using GeoGebra CAS to compute an elimination task manually. In this particular case $\ell = x^2 - 6x - 6y + 24 = 0$.

tool (and command), the **LocusEquation** command (for both the explicit and implicit cases), and the **Envelope** command. They are fully documented in [18]. In this paper we show some possible classroom scenarios to highlight some benefits of using symbolic computation in geometry reasoning with the help of GeoGebra.

In a nutshell we summarize the suggested way of using these novel methods. For all methods the user needs to construct a geometric drawing first, based on some free points, and then adding some dependent points, by following the traditional Euclidean way of construction. That is—not considering some extra features like,

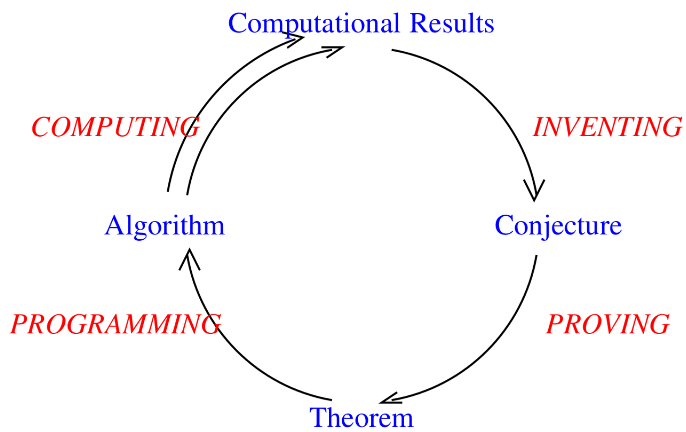

FIGURE 4. Buchberger's concept, the creativity spiral.

for example, the availability of regular n-gons for arbitrary positive integer n—only constructions are supported that can be drawn by compass and ruler.

(1) To *prove* a geometric statement, the user should select the **Relation** icon from the toolset, select two objects and a numerical comparison will be automatically obtained. In case GeoGebra finds a numerical coincidence, that is, for example, two lines are parallel, by pressing the "More..." button, the user can initiate a symbolic check on the numerically conjectured property.
(2) To get a symbolic equation of an *explicit locus*, the user needs to issue the command `LocusEquation(`T`,`M`)` where T is the tracer and M is the mover point.
(3) To get a symbolic equation of an *envelope*, the user needs to issue the command `Envelope(`t'`,`M`)` where t' is the tracer line and M is the mover point.
(4) To get a symbolic equation of an *implicit locus*, the user needs to issue the command `LocusEquation(`p`,`P`)` where p is the prescribed property and P is the free (or semifree) point.

2.2. **Buchberger's workflow.** We refer to Buchberger's concept on the learner's workflow that is communicated as the *creativity spiral* ([8], see Fig. 4).

According to this concept, a continuous workflow can be identifed from having *computational results*, and, by invention, *conjectures* can be obtained. Then, after proving the conjecture, a *theorem* will be found. Buchberger emphasizes that new *algorithms* are usually clarified by using programming, even more in the modern era and for even wider audience. By applying the algorithm, new *computational results* will be obtained. Then, the spiral continues in further inventions and other conjectures.

Of course, this process describes not only the learner's attitude on knowing mathematics better, but the researcher's position as well. We also recall Halmos' quote that "the only way to learn mathematics is to do mathematics".

By following this concept, we refer to [18] in summarizing a possible novel approach to use automated reasoning in the classroom:

(1) Some *computations* are performed with GeoGebra. In many cases this results in using GeoGebra for experiments, randomly, or planned by the teacher. We will focus on computing an implicit locus in this first step.
(2) A *conjecture* for the output curve is made by the student.
(3) The conjecture is checked by the **Relation** tool or command in GeoGebra. We accept this result without further verification as a basic step since the symbolic result in GeoGebra is reliable from the mathematical point of view.
(3+) Occasionally, however, the *proof* can be worked out by paper and pencil as well, if applicable.
(4) The next activity, suggested by Buchberger, "programming", can be interpreted as designing new applets based on new *algorithms* that use the theorem, and are capable of doing additional computations, including the discovered ones and perhaps new ones also.
(1) In such a way, the theorem can be generalized by plotting further implicit loci with GeoGebra—as further experiments for the student, being controlled by the teacher or not.

Then, the process continues from the 2. step again.

2.3. **Examples.**

Example 2.1. In [18] Section 4.3 a worked out example can be found on discovering the midline theorem with GeoGebra ART. As a generalization of that example, that is, in some sense as "a second round" in the creativity spiral, we can do further discoveries with the intercept theorem. It states that *"if two intersecting lines are cut by parallel lines, the line segments cut by the parallel lines from one of the lines are proportional to the corresponding line segments cut by them from the other line"*. This theorem is well known from middle school lessons.

Similarly, the converse of the intercept theorem states (see Fig. 5) that *"if the two intersecting lines are intercepted by two arbitrary lines and $a : b = c : d$ holds then the two intercepting lines are parallel"*.

In a framework based on the creativity spiral, when considering the ratios above, the piece of information that *parallelism plays a key role*, should be kept secret. For this reason we will start our discoveries by considering the converse form, and leaving the discovery of parallelism to the student. That is, let us draw a triangle as seen in Fig. 5, split the left side by segments a and b, split the right side by segments c and d, and let us call the splitting point P.

Computational results. The first experiments should be simple numerical checks by dragging point P in different positions. After some attempts GeoGebra can be asked where to put point P exactly in order to have $a : b = c : d$ by typing `LocusEquation(a/b==c/d,P)` (see Fig. 6).

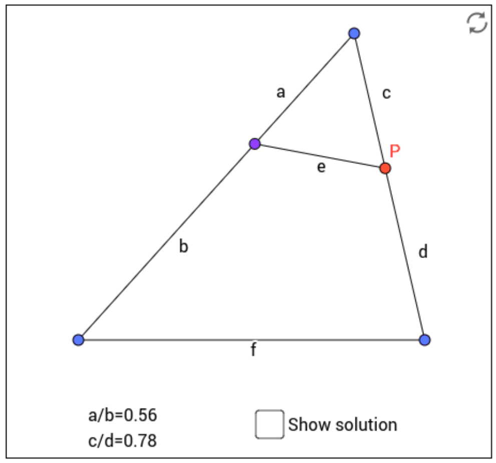

FIGURE 5. Discovering the converse of the intercept theorem.

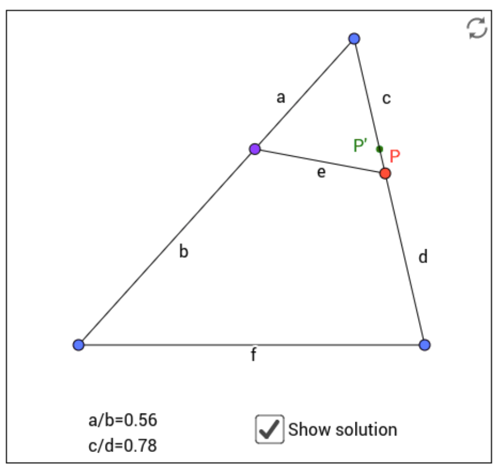

FIGURE 6. Discovering the converse of the intercept theorem (with a single solution on the right side).

For some students, after dragging the other points in the figure, it may be clear that P needs to have a special position to make the expected property hold, namely that e must be parallel to f. But this precondition can be probably better visualized in an alternative way as shown in Fig. 7: here we do not assume that P must lie on the right side of the triangle, but it can be an arbitrary point in the plane.

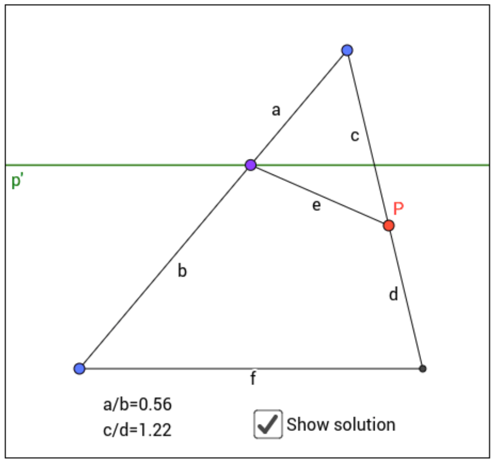

FIGURE 7. Discovering the converse of the intercept theorem (with all possible solutions in the plane).

Using the whole plane for the possible domain of P has one more benefit: by dragging different points of the figure the resulting precondition is visible and well achievable. Now we have a **conjecture**.

At this point the non-converse form of the intercept theorem can also be mentioned, what is more, it is easier to construct a figure where parallelism is already a given condition, and to conclude that the appearing proportions must be equal.

Proof. Fig. 8 shows a possible way to prove the intercept theorem by using GeoGebra ART.

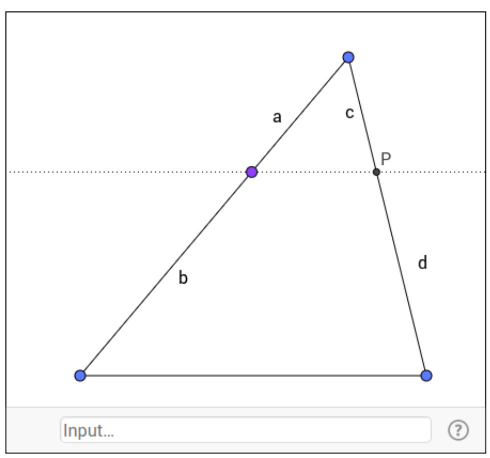

FIGURE 8. Proving the intercept theorem by using the **Relation** command.

The user needs to type `Relation(`a/b`,`c/d`)` and first a numerical check will be performed as seen in Fig. 9.

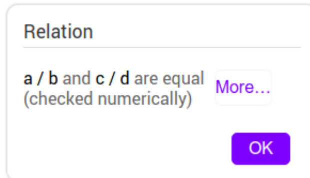

FIGURE 9. Proving the intercept theorem by using the **Relation** command (numerical check).

Then, by clicking "More..." a symbolic check will be done (Fig. 10). It is communicated that on some degeneracies the statement may not hold. In this step, if GeoGebra finds a sufficient condition which has a simple geometric meaning, it will be also communicated to the user.

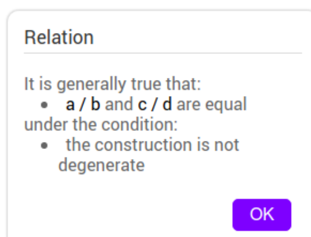

FIGURE 10. Proving the intercept theorem by using the **Relation** command (symbolic check).

In the next step our students will want to play with the topic on their own, maybe with some guidance given by the teacher. To achieve new results, they need to formulate different questions, type different commands, that is, to do some simple **programming**. If their questions require some redrawing of the figure, then their changes on the applet will be even more essential. Either they want to specialize the intercept theorem, or to generalize it, or to find some related questions which are somewhat different. Among others, some ideas where to start playing:

(1) Fig. 11 shows that there may be *two* parallel lines which are the output of the **LocusEquation** command. Explain the situation. For which configuration is there only one parallel line?

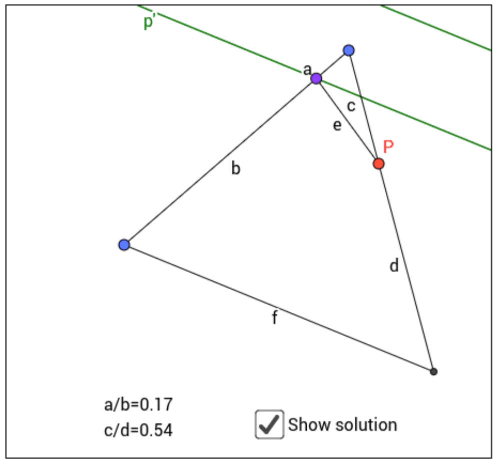

FIGURE 11. Converse of the intercept theorem can have more solutions.

(2) Let us assume the case of parallelism, that is, we collect some consequences of the intercept theorem (Fig. 12). How can we express e/f in terms of a and b?

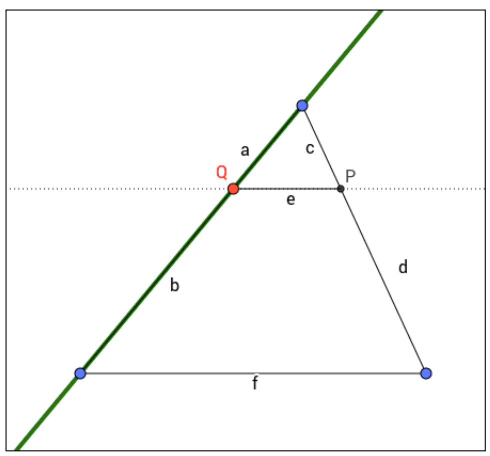

FIGURE 12. Obtaining more information in the case of parallelism. Having the point Q on the left side of the triangle constrained, the command `LocusEquation(`a`/(`a`+`b`)==`e`/`f`,`Q`)` yields the green line, that is, all points on the left side (plus on the line it contains) are the possible solutions of this puzzle. Actually, all of them are indeed solutions.

(3) It may be far-fetched why we consider the ratio of a and b since the sum, the difference or the product of them seem to be more natural. What happens if we prefer considering these other three operations instead of division?

Fig. 13 shows the output if we use multiplication instead. It can be surprising geometrically, but it clarifies that even simple questions may result in difficult answers: here a sextic curve appears which can be factored as the product of two cubics.

As a final remark in this example, we, of course, admit that for children this is advanced mathematics and will not appear in school in the current curricula. Also for teachers some of these results may be difficult to interpret. But is this a good reason to not support children to just play and discover? Such questions are very easy to ask, and if the answer is difficult, we should put more effort in trying to understand it.

For some further details on this worked out example we refer to `https://www.geogebra.org/m/WDMMPR5c#material/ESPcY39G`.

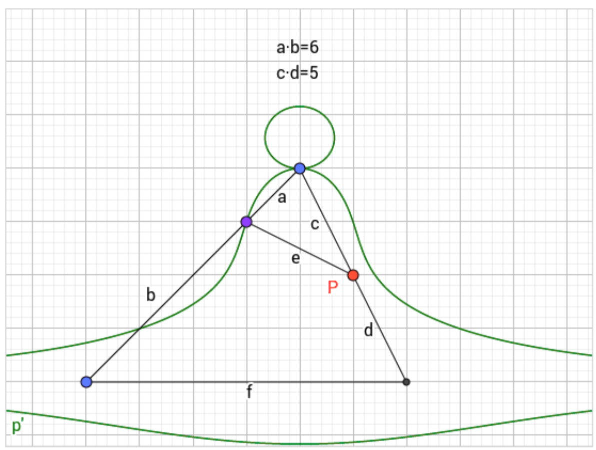

FIGURE 13. The output of the command `LocusEquation(`a `b==`c d`)`.

Example 2.2. Another example in [18] Section 3.2.3 is to define an ellipse as the envelope of perpendicular bisectors of a fixed point D and a constrained point C lying on a fixed circle given with center A and circumpoint B—this example is well known in some middle school textbooks [6, p. 104] as well. By dragging point D and putting it outside of the circle, the ellipse changes to a hyperbola. (See also `https://www.geogebra.org/m/Xu3ZJWGB` for an online version of the concept.) One of the further possible experiments can be to find positions for A, B and D to get a different kind of curve… maybe, a parabola.

Let us compose a "round" in the creativity spiral by extending the simple activity in [18]. After some experimenting it seems hardly possible to drag A, B and D to a configuration to get a parabola for the envelope. Therefore, we may think of using a degenerate circle, one having "infinite radius". That is, we have an **algorithm**.

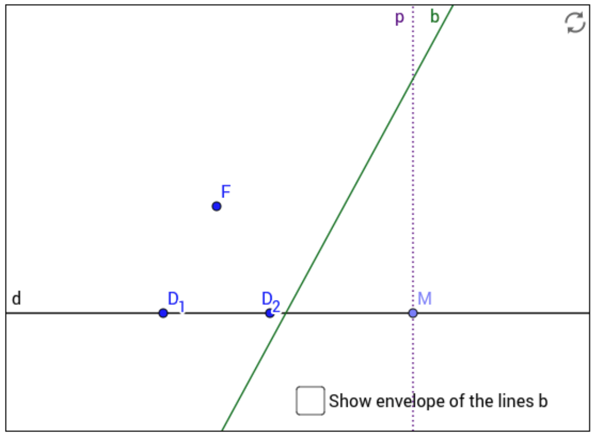

Figure 14. Preparation for constructing a parabola as an envelope.

By following the notations of `https://www.geogebra.org/m/WDMMPR5c#material/TjjaUhh9` (see Fig. 14), we construct the line d joing the free points D_1 and D_2 (this plays the role of the circle, with "infinite radius", that is, D_1 corresponds to A and D_2 to B), and create the free point F (it corresponds to point D). Now by constraining M (which corresponds to C) to d, we can create again the bisector b of FM, and the envelope of b can be computed and plotted by using the command `Envelope(`b`,`M`)` (Fig. 15).

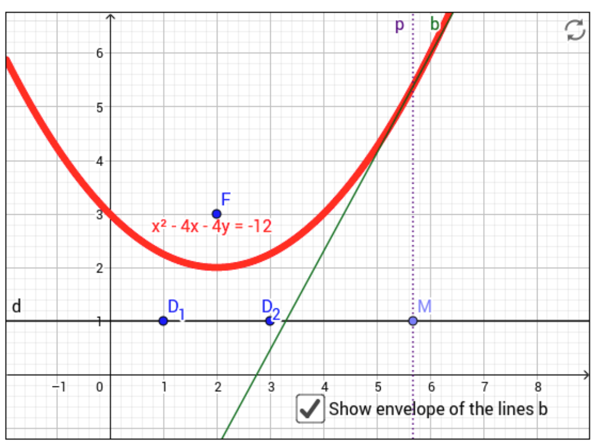

Figure 15. A parabola constructed as an envelope.

Computations. In this case it is useful to enable showing the grid in the coordinate system to check the exact positions of the points D_1, D_2 and F. In the figure, they are $D_1 = (1, 1)$, $D_2 = (3, 1)$ and $F = (2, 3)$. The envelope equation is, computed by GeoGebra, $x^2 - 4x - 4y = -12$. Is this a parabola? Well, yes, because it can be transformed into the equation $y = \frac{1}{4}x^2 - x + 3$, which is obviously one. Of course, this is just one piece of evidence that *in general* we will obtain a parabola.

Conjecture. By dragging the free points randomly, the curve remains looking like a parabola, and the envelope equation seems to be a quadratic equation (Fig. 16). Unfortunately, being quadratic does not imply that the curve is a parabola—it can also be any other conic including ellipses, hyperbolas, circles or even a set of two lines or just single points. Anyway, we can have a conjecture that in “many cases” we obtain a parabola by this process.

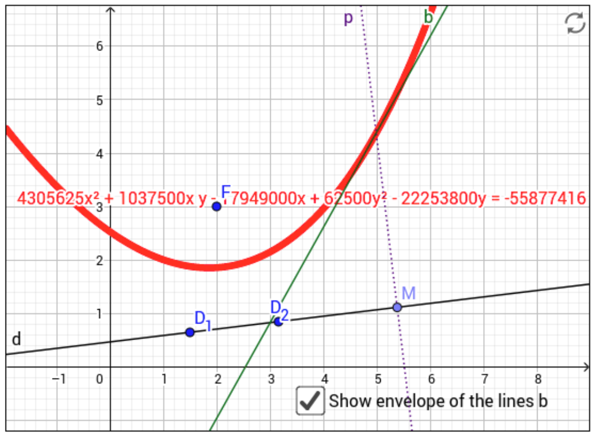

FIGURE 16. A quadratic envelope equation that seemingly describes a parabola.

Proof. We construct a new drawing as seen in Fig. 17.

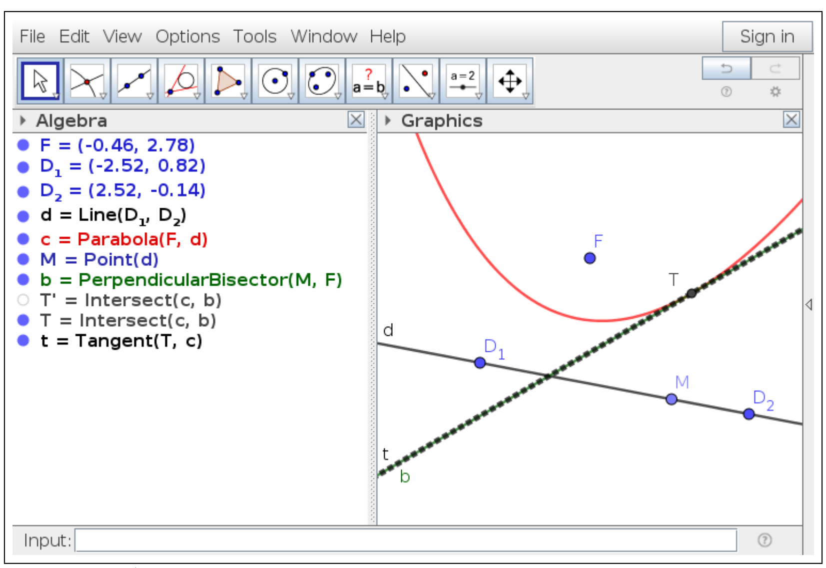

FIGURE 17. A synthetic construction to prepare for proving the statement.

Now by clicking on the **Relation** tool and selecting lines b and t, GeoGebra first performs a numerical check if they are equal. (Surprisingly, there may be cases when the numerical approximation identifies this statement as false, see Fig. 18. In such cases the free points should be dragged to different positions—eventually aligned to the grid—and restart comparison.) Fig. 19 shows a positive result of the numerical check.

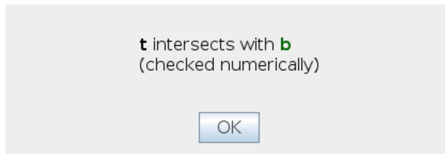

FIGURE 18. The numerical approximation may identify the statement as false in some cases.

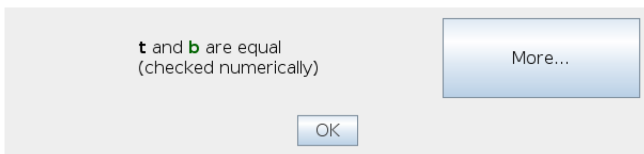

FIGURE 19. A successful run of the numerical check.

When more information is needed, by clicking "More…" a symbolic check will be done and—after a little time, depending on the performance of the machine—a detailed report is shown as seen in Fig. 20.

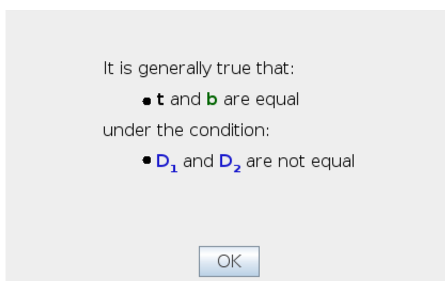

FIGURE 20. The symbolic computation identifies the statement as true in general. Also a possible degeneracy condition is found which can be avoided, and finally we get an *explicitly stated theorem*.

Indeed, based on machine proof, the line d will define the directrix of the parabola, and point F will be its focus.

New algorithm. One round done, but multiple new questions arise. We changed the circle in the initial setting to be a line—what happens if the circle is changed to some different curve? A possible new setup, seen in Fig. 21, can be used to begin with a parabola. In this case, however, the graphical output heavily depends on the position of point D. Anyway, the result seems to be a quintic curve in many cases, in the particular case in Fig. 21 it is

$4\,y^5 - 2\,x\,y^4 + 2\,x^2\,y^3 - 27\,x^4 - 105\,y^4 + 56\,x\,y^3 - 104\,x^2\,y^2 + 36\,x^3\,y + 64\,x^3 + 1000\,y^3 - 232\,x\,y^2 + 514\,x^2\,y - 1084\,x^2 - 4530\,y^2 - 8\,x\,y + 1242\,x + 10084\,y - 9161 = 0.$
Again, we are confronted with a non-trivial question which is far beyond the middle school curriculum. But exploring higher degree algebraic curves is just a surprisingly small step if we have a useful tool which supports explorations.

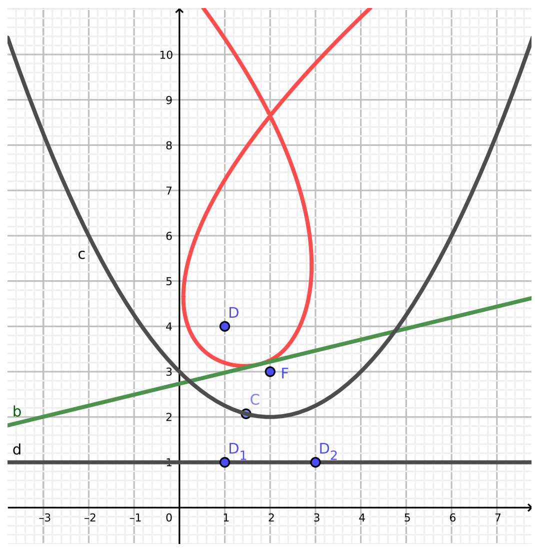

FIGURE 21. Changing the circle to a parabola.

At the end of this section we remark that the creativity spiral can be started at various entry points. One can start even by explicitly telling a theorem. In our opinion, however, if it is possible, better learning results can be achieved if the students find the conjecture on their own (see also [23] on the role of setting up a conjecture as a first step in the learning process).

3. Conclusion and further work

We discussed two examples of middle school geometry problems by extending them to further activities by using GeoGebra's Automated Reasoning Tools. It was possible to obtain surprising results which are rarely analyzed at higher level education, either, but are relatively easy to observe by using dynamic mathematics. We emphasize that the shown examples are difficult to study without symbolic computations because for both the implicit loci and the envelope curves the numerical approximations may lead to improper or incomplete results. Symbolic support for such geometric problems is, in some sense, a new way, and therefore it requires further investigations to describe how they are indeed useful, especially in the classrooms.

We admit that these novel tools may be still slow on many systems including smartphones and tablets. On faster hardware, however, they may be used to get immediate response from the system when the free points are changed on dragging them by the mouse [15]. On the other hand, different formulations of the same problem may lead to success or failure, too. For example, when constructing the drawing in Fig. 17 in a different way, like in Fig. 22, we cannot get a positive result during the symbolic computation because the underlying system sets up an equation system which is too difficult to manipulate, and times out. However, GeoGebra communicates this failure that the statement is “possibly generally true” (Fig. 23).

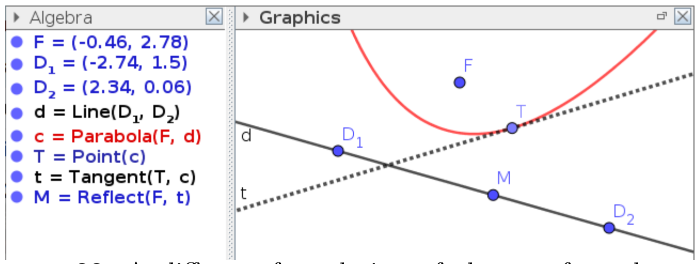

FIGURE 22. A different formulation of the proof as shown in Fig. 17. Here the **Relation** tool can still compare M and d numerically.

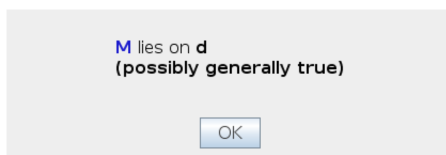

FIGURE 23. The failure of a symbolic check is communicated that the statement is “possibly generally true”.

Also, the second round (and the further investigations) may not be completely supported by GeoGebra ART. For example, higher degree curves than conics cannot be used as inputs in the current version (as of 5.0.417.0). Thus, obtaining strict proofs for more advanced situations are not yet possible.

Finally we refer to the GeoGebraBook `https://www.geogebra.org/m/WDMMPR5c` where additional examples are shown.

REFERENCES

[1] Abánades, M., Botana, F., Kovács, Z., Recio, T. and Sólyom-Gecse, C.: Development of Automatic Reasoning Tools in GeoGebra. *ACM Commun. Comput. Algebra* 50.3, 85–88. 2016.

[2] Baulac, Y., Bellemain, F., Laborde, J. M.: *Cabri geometry II*. Dallas: Texas Instruments. 1994.

[3] Botana, F. and Valcarce, J.: A dynamic-symbolic interface for geometric theorem discovery. *Computers and Education* 38, 21–35. 2002.

[4] Botana, F. and Kovács, Z.: New tools in GeoGebra offering novel opportunities to teach loci and envelopes. *CoRR abs/1605.09153*. 2016.

[5] Botana, F., Hohenwarter, M., Janičić, P., Kovács, Z., Petrović, I., Recio, T. and Weitzhofer, S.: Automated Theorem Proving in GeoGebra: Current Achievements. *Journal of Automated Reasoning* 55.1, 39–59. 2015.

[6] Boxhofer, E., Hubert, F., Lischka, U. and Panhuber, B.: *mathematiX 4*. Veritas, Linz. 2013.

[7] Buchberger, B.: *An Algorithm for finding a basis for the residue class ring of a zero-dimensional polynomial ideal.* PhD thesis, Math. Inst. University of Innsbruck, Austria. 1965.
[8] Buchberger, B. and the Theorema Working Group: Theorema: Theorem proving for the masses using Mathematica. Invited Talk at the Worldwide Mathematica Conference, Chicago, June 18-21, 1998.
[9] Chou, S.-C.: *Mechanical geometry theorem proving.* Kluwer Academic Publishers Norwell, MA, USA. 1987.
[10] Gelernter, H.: Realisation of a geometry-proving machine. *Proceedings of the International Conference on Information Processing, Paris, June 15-20, 1959.*, 273–282. 1959.
[11] Hohenwarter, M.: *Ein Softwaresystem für dynamische Geometrie und Algebra der Ebene.* Master thesis, Paris Lodron University, Salzburg. 2002.
[12] Jackiw, N. R. *The Geometers Sketchpad, v3.0.* Berkeley, CA: Key Curriculum Press. 1995.
[13] Kapur, D.: Using Gröbner bases to reason about geometry problems. *Journal of Symbolic Computation*, 2(4):399–408.
[14] Kortenkamp, U.: *Foundations of Dynamic Geometry.* PhD thesis, ETH Zürich. 1999.
[15] Kovács, Z.: Real-time animated dynamic geometry in the classrooms by using fast Gröbner basis computations. *Mathematics in Computer Science* 11.1. 2017.
[16] Kovács, Z.: *Computer Based Conjectures and Proofs in Teaching Euclidean Geometry.* PhD thesis. Johannes Kepler University, Linz. 2015.
[17] Kovács, Z. and Parisse, B.: Giac and GeoGebra – Improved Gröbner Basis Computations. In: Gutierrez, J., Schicho, J. and Weimann, M. (Eds.): *Computer Algebra and Polynomials. Lecture Notes in Computer Science 8942*, 126–138. Springer. 2015.
[18] Kovács, Z., Recio, T. and Vélez, M.P.: gg-art-doc (GeoGebra Automated Reasoning Tools. A Tutorial). A GitHub project. `https://github.com/kovzol/gg-art-doc`. 2017.
[19] Kutzler, B. and Stifter, S.: On the application of Buchbergers algorithm to automated geometry theorem proving. J. Symb. Comput., *Journal of Symbolic Computation* 2(4):389–397.
[20] Oldenburg, R.: FeliX – mit Algebra Geometrie machen (German). *Computeralgebra Rundbrief, Sonderheft zum Jahr der Mathematik, April 2008*, 15–17. 2008.
[21] Rabinowitsch, J.: Zum Hilbertschen Nullstellensatz. *Math. Ann.*, 102(1), 520. 1929.
[22] Schwartz, J. L. and Yerushalmy, M.: *The Geometric Supposer.* Sunburst Communications, Pleasantville, NY. 1983.
[23] Ufer, S. and Heinze, A.: . . . mehr als nur die Lösung formulieren. Phasen des geometrischen Beweisprozesses aufzeigen. *Mathematik lehren*, 155, 43–49. 2009.
[24] Vélez, M.P. and Recio, T.: Automatic Discovery of Theorems in Elementary Geometry. *Journal of Automated Reasoning* 23, 63–82. 1999.
[25] de Villiers, M.: Exploring Loci on Sketchpad, *Pythagoras* 46(47), 71–73. 1998.
[26] Ye, Z., Chou, S. C. and Gao, X. S.: An introduction to Java Geometry Expert. *Automated Deduction in Geometry, 7th International Workshop, ADG 2008, Shanghai, China, September 22-24, 2008, Revised Papers, Lecture Notes in Computer Science, volume 6301*, 189–195. Springer. 2011.

The Private University College of Education of the Diocese of Linz, Salesianumweg 3, 4020 Linz, Austria
E-mail address: `zoltan@geogebra.org`