Technological tools to teach the idea of optimality

Reinhard Oldenburg

Angaben zur Veröffentlichung / Publication details:

Oldenburg, Reinhard. 2024. “Technological tools to teach the idea of optimality.” *Qeios*, I1Y5A5. https://doi.org/10.32388/i1y5a5.

Open Peer Review on Qeios

Technological Tools to Teach the Idea of Optimality

Reinhard Oldenburg[1]

1 Universität Augsburg

Funding: No specific funding was received for this work.

Potential competing interests: No potential competing interests to declare.

Abstract

Optimization is a fundamental idea of mathematics, yet it is not that much in the focus of widespread digital teaching tools. The paper first explores the relevance of optimality and then shows how the idea is supported in the FeliX dynamic geometry online system. The idea is to include optimality statements together with equality statements as means to describe geometric scenes. Some reports from teaching experiences are discussed as well.

Reinhard Oldenburg

Augsburg University, Mathematics Institute, Germany;reinhard.oldenburg@math.uni-augsburg.de

Keywords: Optimization, Dynamic geometry, Algebraic modelling, GeoGebra.

Introduction

Optimization problems are central in mathematics; for example, there are several scientific journals that are devoted exclusively to optimization, such as Optimization, Journal of Optimization Theory and Applications, Applied Mathematics & Optimization, Open Journal of Mathematical Optimization. However, in mathematics education, their role is much less prominent. Of course, there are a lot of high school problems that can be solved by an application of differential calculus, and there is also literature on solving optimality problems by means of inequalities, e.g. (Schupp, 1992), but many others miss the connection, e.g., OECD (2016). Some research has investigated certain cognitive aspects of solving optimization problems (e.g., Malaspina & Font, 2010). Moreover, there is some research investigating optimization with technology, e.g., Bushmeleva et al. (2018). And, of course, there are lots of GeoGebra applets showing solutions to specific optimization problems. However, in these approaches, optimization is not used directly as a modelling tool; i.e., technology does not mediate between the context and the solution but between a modified problem description and the solution. Therefore, in this paper, we show how optimization can be built directly into a dynamic geometry environment. It is not meant to say that other approaches are not sensible, but that the integration shown here widens the didactical repertoire in teaching optimization.

Theory

Bruner (1960) put forth the idea of fundamental ideas. Vohns (2016) has investigated the reception of this idea in mathematics education and reports that at least some researchers include optimization in the list of fundamental ideas of mathematics, e.g., Schreiber (1979). Of course, there are plenty of arguments to justify this judgement:

- Optimality is used in the characterization of many concepts, e.g., the integer divisor as the largest integer making a product not exceed a given number, the least common multiple, greatest common divisor, derivative as slope of the locally best fitting linear function, integration as approximating functions by step functions, mean value of numbers as the number with minimal square distance, regression, line segments as shortest paths between two points in the Euclidean plane.
- Many fundamental mathematical objects that are (typically) not characterized by optimality properties nevertheless have optimal properties, e.g., the square as the rectangle with the greatest area given a fixed circumference.
- Optimization problems have been around since the earliest days of mathematics; consider e.g., Dido's problem or the problem of the Brachistochrone.
- Optimization is a large domain with many subdomains, such as constraint optimization, linear optimization, discrete optimization, and geometric optimization, and optimization appears in most disciplines of mathematics and many fields of applied mathematics (e.g., operations research).
- Optimality is a concept that allows one to coin concepts in many applied disciplines; e.g., in physics, one can derive the equations of mechanics, electrodynamics, general relativity as minimizers of certain functionals, and the quantum versions of these theories as variations around this classical solution, and Mill's moral theory renders the choice of the morally adequate action as an optimization problem.

Regarding school mathematics, one may hold the view (e.g., Klein) that functions and functional thinking are central. There are three basic operations one can do with a single function with values in an ordered set:

1. Evaluation: Given an input, one determines the value of the function.
2. Solving: Find an input that produces a certain output, e.g., a zero.
3. Optimization: Find a minimum or maximum.

These three operations can be understood in a wide sense. Calculation of function values includes e.g., all formulas calculated in spreadsheets, the determination of the position of a constructed element in a dynamic geometry environment (which is, essentially, a spreadsheet with a different user interface), programs in a functional programming language. Solving and optimization in this wider sense can be achieved in Excel by the plug-in called "Solver" and in dynamic geometry by guided dragging (Arzarello et al., 2002) to a position that fulfills some property. However, the standard mode of working of these tools is by function evaluation.

Note that additional operations (additions, composition, etc.) occur when more than one function comes into play, but I will ignore this further complexification. Instead, I will cross these operations with Kieran's activities in algebra (Kieran, 2004). She distinguished generational activities, transformational activities, and global/meta-level activities. Crossing relations with the three operations given above are displayed in Table 1.

Table 1. Kieran's activities and operations with functions

	Evaluating	Solving	Optimization
Generational	Setting up an expression	Setting up an equation	Setting up an optimization problem
Transformational	Equivalent expressions, e.g., simplification	Equivalent transformation of equations	
Global/meta	e.g., structural investigations (being symmetric in variables)	e.g., classification by types of solutions (algebraic)	

The two cells left blank in Table 1 indicate that the relevant topics are usually not taught in school. It is not meant to say that there is nothing to fill into these cells – quite to the contrary. Transformational activities for optimization include the following: An extremum is not affected by applying a strictly monotonic increasing function. This includes the special cases of adding a constant, multiplying by a positive constant, taking squares, square roots, or logarithms. Applying a decreasing function exchanges minima and maxima. Or the composition rule: $f(g(x))$ has a maximum in x_0 if f has a maximum in y_0 and $\exists x_0 : y_0 = g(x_0)$. All these could be dealt with in school, including the discovery and justification of these transformation rules. Global and meta-activities could center around the number of maxima or the question under which conditions one may conclude that between two maxima there exists a minimum. Moreover, the relevance of convexity for optimization could be reflected.

By now, it should be clear that in typical math curricula, operations with optimization are neglected in comparison to the other two operations. Correspondingly, most digital tools used in school mathematics do not support optimization very well. For optimization problems in one variable, GeoGebra can, of course, plot graphs and automatically find minima and maxima. More advanced computer algebra systems offer implementations of numerical (and sometimes symbolic) optimization algorithms that can often also include constraints set by equations or inequalities. For example, the classical problem of what cylinder with 1000cm^3 volume has the least surface area (and thus – up to some idealization – least use of material) can be solved in Mathematica by the command `FindMinimum[{2*Pi*r^2+2*Pi*r*h, Pi*r^2*h==1000}, {{r,1}, {h,1}}]`

and in Maxima by:

```
fmin_cobyla(2*%pi*r^2+2*%pi*r*h, [r, h], [1,1],
 constraints= [%pi*r^2*h=1000]);
```

These tools are very flexible and powerful, and teaching should aim to educate students who are both competent in using them and in understanding them (e.g., knowing how their way of working implies certain limitations). However, to develop a more intuitive understanding of the meaning and properties of optimization, it seems desirable to have tools at hand that allow for a more direct interaction with the optimization problem. The current paper presents a possible implementation of optimization within a relational dynamic geometry environment.

The FeliX-Systems

FeliX is an algebraic relational dynamic geometry environment that exists in various versions (Oldenburg, 2007; Oldenburg, 2022). There are versions built on top of Mathematica, and they provide some advanced symbolic analysis tools, but here I restrict to two versions that are available online. The first is a realization in JavaScript that is available online via https://myweb.rz.uni-augsburg.de/~oldenbre/jsfelix/F2d/jxfelix.html. Its user interface consists of three main components (see Figure 1): A geometry view of the Euclidean plane with a Cartesian coordinate system, a table that shows all points and their current coordinates, and an equation table that may, in fact, take equations, inequalities, and expressions that are built up from the coordinate variables of the points.

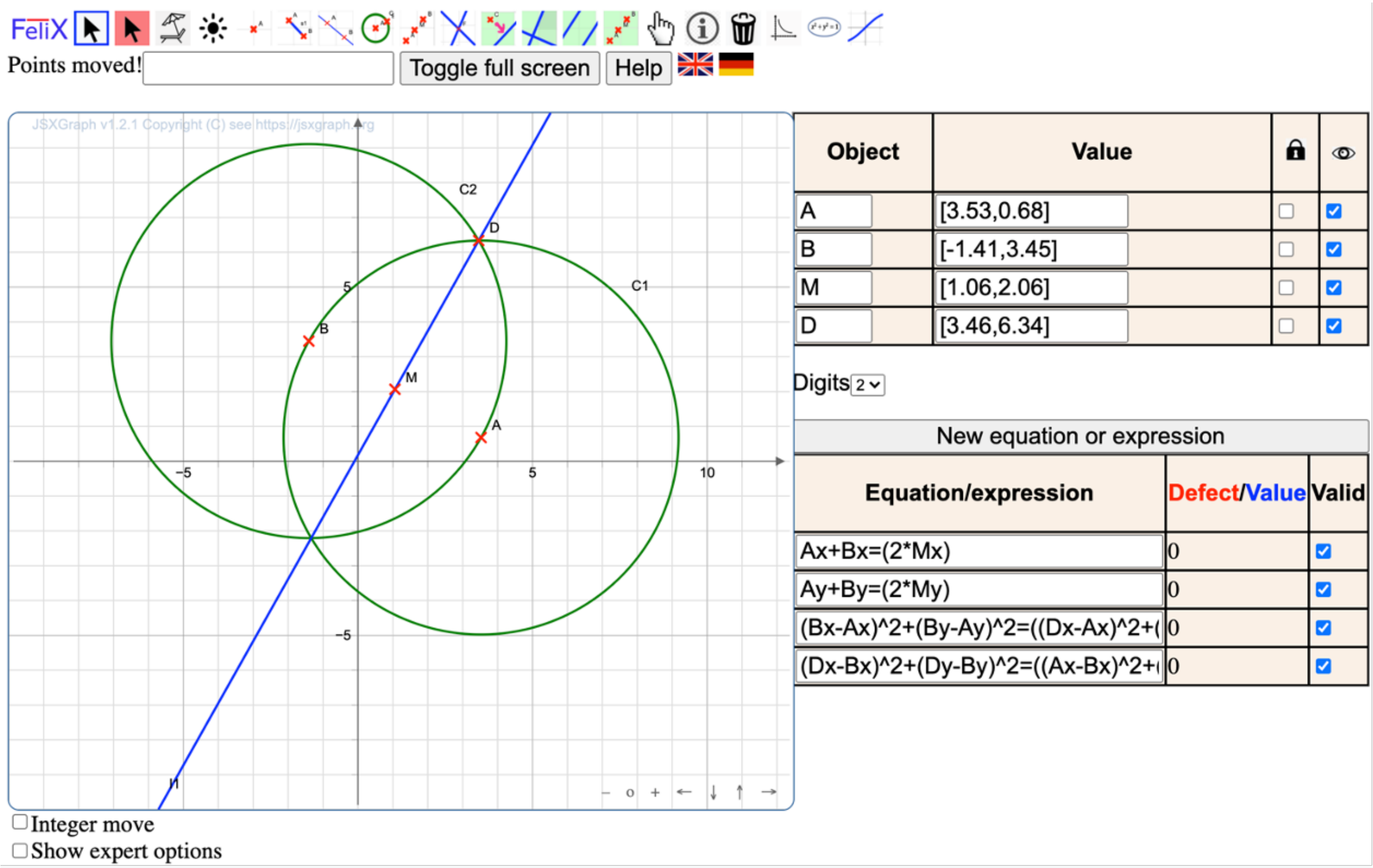

Figure 1. The FeliX window

For a point P, the variables Px and Py refer to its Cartesian coordinates. The equations and inequalities that are entered into the equation list are respected while dragging. For example, if one has three points A, B, C and enters the equations 2*Bx=Ax+Cx and 2*By=Ay+Cy, then B will be the midpoint of A and C. Still, all three points can be dragged with the mouse. There is no distinction between free points and constructed points, and neither the order of object creation nor the order of equations has an effect on the dragging behavior: There is just a set of objects and a set of equations. For convenience, there are functions to calculate the length of segments and angles between them, but of course, all this could be achieved by using the standard formulas learned in high school explicitly. The same holds for the green buttons that automate the process of entering equations that let certain lines be parallel or orthogonal. Besides equations and inequations, the equation editor can also hold expressions. By default, they have no influence on the dragging behavior; their value is calculated and displayed in the table. If, however, one activates the checkbox “valid” for an expression, then this expression will be minimized. Applications will be shown later.

The second realization, GGBFeliX, of FeliX that shall be discussed here is built on top of GeoGebra and is available under https://myweb.rz.uni-augsburg.de/~oldenbre/GGB/ggbFelix.html. Its user interface (see fig. 2) consists of an unmodified GeoGebra applet that is controlled by a JavaScript layer. Equations are entered into a single text field with multiple equations separated by semicolons. Simple test cases are to enter Ax=Bx or Ax>2*By or the like. While coordinates of points, equations, and inequalities work exactly as in FeliX, there is a difference regarding segments and angles. If there is a segment named x, then the variable Sgx represents the segment's length, and Sgxang represents the segment's angle with respect to the x-axis. Thus, one can make right angles either by using the expression for dot products or by setting e.g. Sgeang= Sgfang+pi/2 to set segments e and f orthogonal. Expressions to be minimized have to be given inside a mini(…) function. Figure 2 shows an example: First, a quadrilateral has been drawn from four segments, and this has been turned into a parallelogram by setting opposite sides to have the same length. This parallelogram is restricted to become a rectangle by setting one of the angles to 90°. Finally, its area is fixed to 25 by Sgf*Sgi=25, and it is asked to minimize the circumference by adding mini(Sgf+Sgi+Sgg+Sgh). This gives, of course, a rigid square that can still be dragged around.

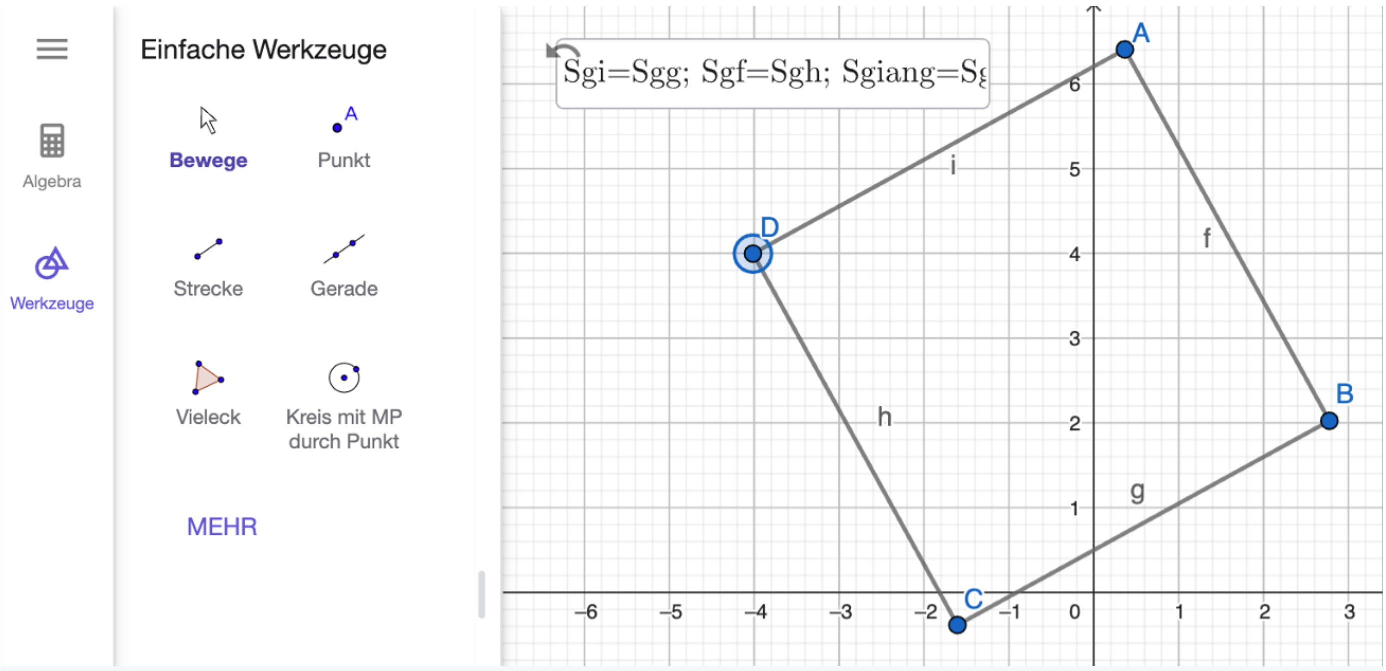

Figure 2. The GGBFeliX window. The full text in the box is:

Sgi=Sgg; Sgf=Sgh; Sgiang=Sghang+pi/2; Sgf*Sgi=25; mini(Sgf+Sgi+Sgg+Sgh)

Examples of optimizations

The example from Fig. 2 showed already a first use case. Here, some further easy examples shall demonstrate how the tool can be used to investigate optimality conditions. In Fig. 3 (left), three points A, B, C have been created and set to be fixed. The segments to a fourth point D are constructed, and by dragging D around and watching the sum of the lengths of these segments, one may get the idea that there is a unique point that minimizes this sum of distances. This is, in fact, true (it is the Fermat-Torricelli point), and the command mini(Sgf+Sgg+Sgh) moves D directly to the optimal position. In the FeliX versions not based on GeoGebra, one may still drag the points A, B, C and observe where the point lies, and this might also give support to the hypothesis that the angles between the three segments are always 120°. Minimizing the sum of squared lengths, i.e., by mini(Sgf^2+Sgg^2+Sgh^2), one gets the center of the triangle.

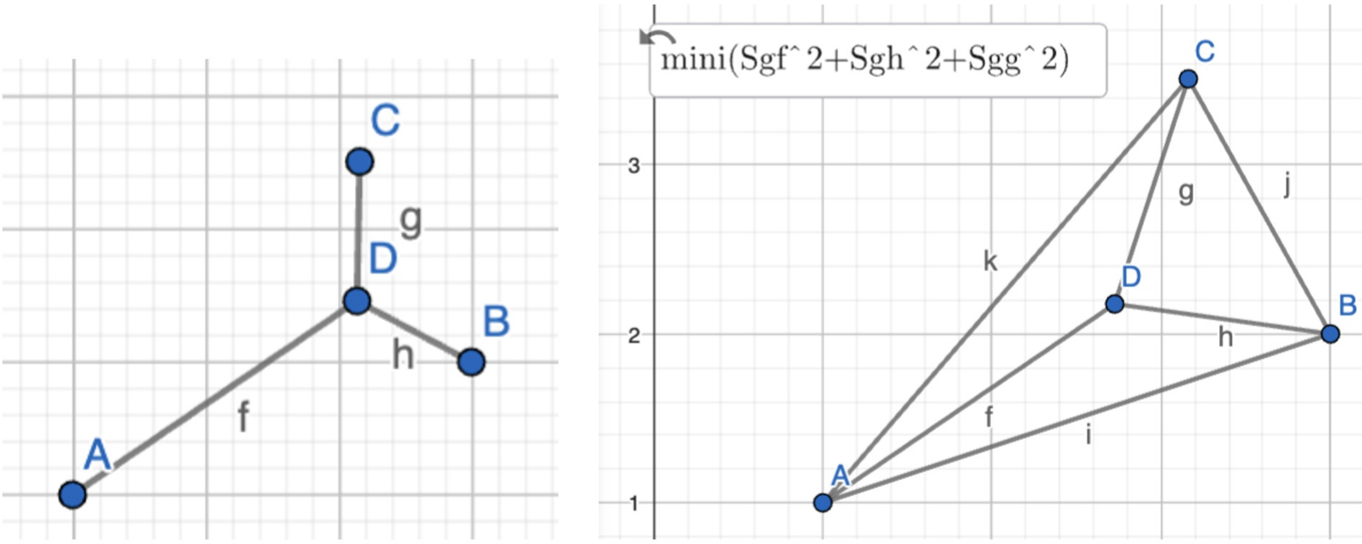

Figure 3. The Fermat-Point of a triangle (left) and the center (right) of a triangle

Figure 4 (left) gives a physical application: A swing consisting of two massless strings f, h of length 4 that are fixed at two points A, B and a wooden beam g of length 2. At rest, the swing will hang in such a way that the potential energy of the beam is minimized. This potential energy is minimal if the center of mass, which lies in the point $\left(\frac{1}{2}C_x + \frac{1}{2}D_x, \frac{1}{2}C_y + \frac{1}{2}D_y\right)$, is as low as possible, thus it suffices to minimize Cy+Dy. The whole model is given by Ax=1;Ay=5;Bx=6;By=5; Sgg=2; Sgf=4; Sgh=4; mini(Cy+Dy).

Another physical example shows the refraction of light: Points A and C are fixed, and the time required for a ray to travel from A to C is to be minimized, assuming that the speed in the part below the x-axis is only half as high as above. This situation can be modelled by the description Ax=1;Ay=3; By=0; Cx=3;Cy=-3; mini(Sgf+2*Sgg), which results in the configuration of Fig. 4 (right). Again, in versions other than that based on GeoGebra, the points A and C can be moved to see the dynamics of the solution.

Figure 5 shows a discrete hanging chain. Points A and E are fixed (recall that they still can be moved explicitly with the mouse), and B, C, D are unrestricted points. The segments s1, s2, s3, s4 between AB, BC, CD, DE are all set to have length 3. Their midpoints F, G, H, I are constructed, and the expression Fy+Gy+Hy+Iy that corresponds to the potential energy of the chain is minimized. It is interesting how natural the chain behaves when, e.g., E is raised further or moved horizontally.

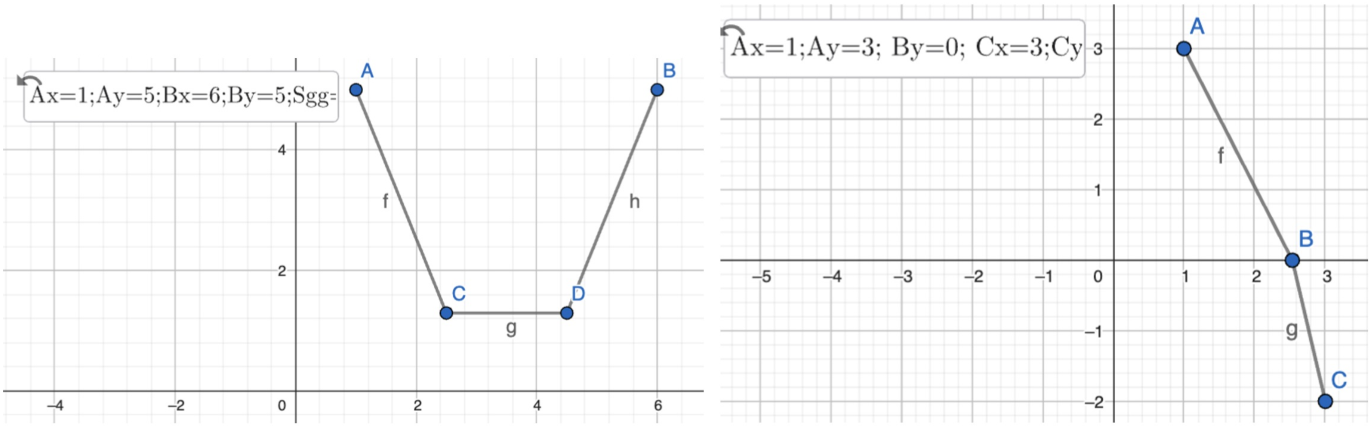

Figure 4. A swing and light refraction

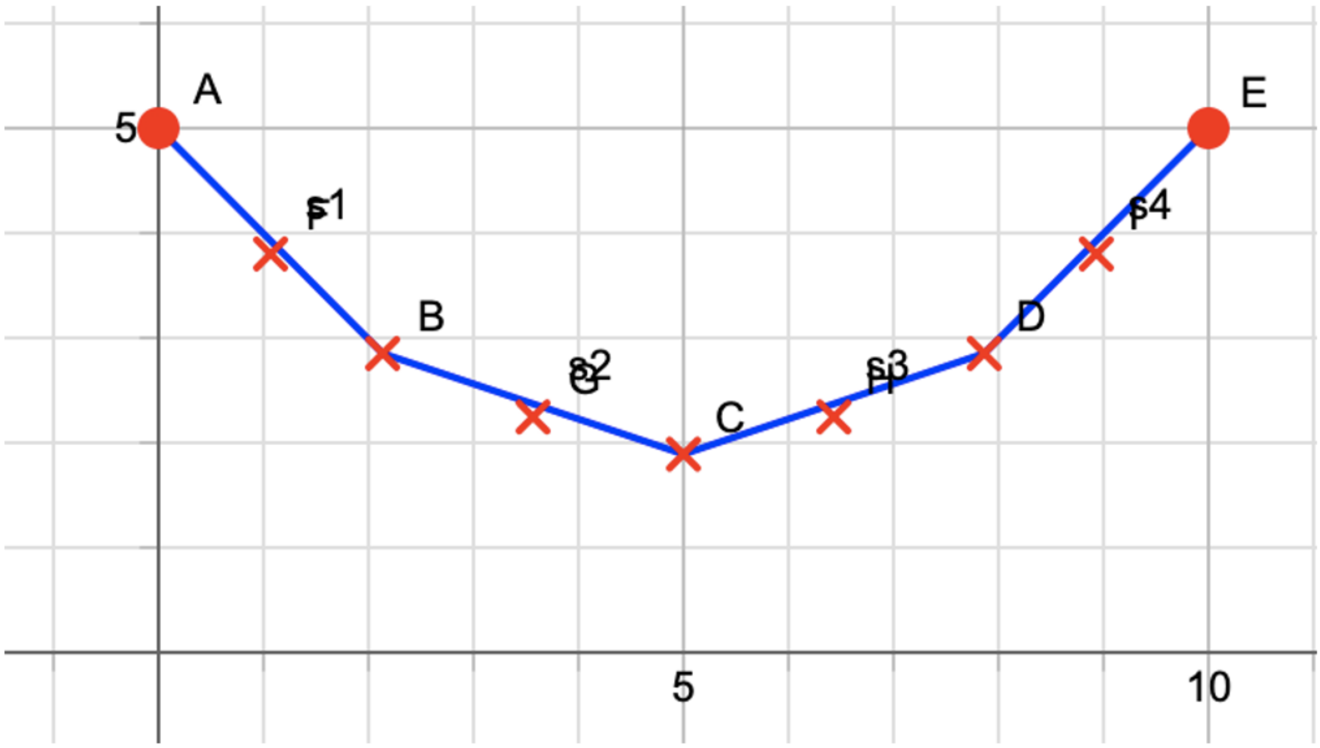

Figure 5. Discrete hanging chain

Teaching experiences

So far, the FeliX versions with built-in optimization have only been used with teacher students at the university. Students were introduced to the possibilities of optimization, and they had to solve certain problems such as finding the Fermat-Torricelli point. It turned out that no specific problems occurred: Specifying an optimality condition seemed as natural to students as constraining a construction by equations or inequations. However, occasionally students got stuck when they entered an expression to be minimized that was not bounded from below, and hence the minimization process did not converge. However, such situations can be used to discuss questions of existence and uniqueness of optimizers, and this can give insights that are valuable beyond the use of this specific tool.

Conclusion

The position of points in Euclidean geometry can be fixed by calculating their coordinates from certain functions (that may correspond to classical geometric rules-compass-constructions), as is done in dynamic geometry systems, by solutions of certain equations, as is done in relational geometry systems such as parametric CAD programs, or, last but not least, by optimality conditions. This work has shown that it is indeed possible to realize a dynamic geometry system that allows optimality conditions as means to specify configurations. It is yet an open question if this can be successfully used by younger students, and thus further research is necessary. However, experience with students suggests that this might be an adequate way to enrich the possibilities to interact with mathematical concepts.

References

- Arzarello, F., Olivero, F., Paola, D. et al. A cognitive analysis of dragging practises in Cabri environments *Zentralblatt für Didaktik der Mathematik* 34, 66–72 (2002). https://doi.org/10.1007/BF02655708
- Bruner, J.S. (1960). *The process of education*. Harvard University Press.
- Bushmeleva, N. A., Sakhieva, R. G., Konyushenko, S. M., & Kopylov, S. M. (2018). Technology for Teaching Students to Solve Practice-Oriented Optimization Problems in Mathematics. *Eurasia Journal of Mathematics, Science and Technology Education*, 14(10). https://doi.org/10.29333/ejmste/93678
- Kieran, C. (2004). The Core of Algebra: Reflections on its Main Activities. In: Stacey, K., Chick, H., Kendal, M. (eds) *The Future of the Teaching and Learning of Algebra The 12th ICMI Study* New ICMI Study Series, vol 8. Springer. https://doi.org/10.1007/1-4020-8131-6_2
- Malaspina, U., Font, V. (2010). The role of intuition in the solving of optimization problems *Educ Stud Math* 75, 107–130. https://doi.org/10.1007/s10649-010-9243-8
- OECD (2016), *Equations and Inequalities: Making Mathematics Accessible to All* PISA, OECD Publishing, Paris, https://doi.org/10.1787/9789264258495-en.
- Oldenburg, R. (2007). Algebraic modeling using a dynamically linked geometry and computer algebra system. In D. Pitta-Pantazi & G. Philippou (Eds.), *Proceedings of the Fifth Congress of the European Society for Research in Mathematics Education*, 1480-1488, University of Cyprus.
- Oldenburg, R. (2021, September). FeliX. https://myweb.rz.uni-augsburg.de/~oldenbre/jsfelix/F2d/jxfelix.html

- Oldenburg, R. (2022). Relational thinking supported by an algebraic modeling tool on the web in: U. Th. Jankvist, R. Elicer, A. Cark-Wilson, H.-G. Weigand, M. Thomsen (Eds.) *Proceedings of the 15th International Conference on Technology in Mathematics Teaching* (ICTMT 15), 89-96.
- Schreiber, A. (1979). Universelle Ideen im mathematischen Denken—ein Forschungsgegenstand der Fachdidaktik. *Mathematica Didactica*, 2, 165–171.
- Schupp, H. (1992). *Optimieren*. BI.
- Vohns, A. Fundamental Ideas as a Guiding Category in Mathematics Education—Early Understandings, Developments in German-Speaking Countries and Relations to Subject Matter Didactics. *J Math Didakt* 37 (Suppl 1), 193–223 (2016). https://doi.org/10.1007/s13138-016-0086-4