

Scratch programming as a tool for teaching angle as a rotation

KONSTANTINOS KAKAVAS, KONSTANTINOS ZACHAROS, IRINI SKOPELITI,
VASSILIS KOMIS

*Laboratory of Natural Sciences
Mathematics and ICT Didactics
Department of Educational Sciences
and Early Childhood Education
University of Patras
Greece
kkakavas@upatras.gr
zacharos@upatras.gr
eskoopel@upatras.gr
komis@upatras.gr*

ABSTRACT

The present study is part of a larger research related to the cultivation of the concept of angle with the use of visual programming. This article introduces the teaching of the angle as a rotation using the Scratch programming language. A semi-structured digital microworld was designed to represent a screw that should be screwed into a piece of wood. The research involved 35 6th grade students who were asked to digitally simulate the actual movement of the screw. It is shown that visual programming can be an effective and dynamic tool for teaching rotation as an angle dimension.

KEYWORDS

Angle, rotation, simulation, Scratch

RÉSUMÉ

La présente étude fait partie d'une étude plus large liée à la culture du concept d'angle avec l'aide de la programmation visuelle. Cet article présente l'enseignement du concept d'angle comme une rotation à l'aide du langage de programmation Scratch. Un microcosme numérique semi-structuré a été conçu pour représenter une vis qui devait être vissée dans un morceau de bois. A la recherche ont pris part 35 élèves de la 6^{ème} classe qui ont été invités à simuler numériquement le mouvement réel de la vis. Ainsi, à révéler que la programmation visuelle peut être un instrument efficace et puissant pour l'enseignement de l'angle comme la rotation.

MOTS-CLÉS

Angle, rotation, simulation, Scratch

INTRODUCTION

Angle is a mathematical concept found in education in the field of Geometry and Trigonometry. However, the definition of angle is a complex task, as it is a multidimensional concept and is related to the context in which it is approached. Thus, according to Henderson

& Taimina (2005), there is not a formal definition of the angle that can include all aspects of our experience of what is defined as an angle.

The difficulty of definition is also emphasized by Mitchelmore & White (2000) who state that each definition emphasizes a specific dimension of the angle and that the various mathematical structures require a different definition. Consistent with the above, Devichi & Munier (2013) state that an angle can be defined in three different ways: a) as a rotation angle, b) as a sector angle, and c) as half lines extending from a common point. Even more detailed is the categorization proposed by Mitchelmore & White (1998), according to which the angle can include seven subcategories-aspects (rotation, meeting, inclination, corner, turning, direction, opening). Thus, arises the need to teach the various aspects of the angle.

The present study presents a small part of a larger research that aims to teach rotation, as an aspect of the concept of angle, using visual programming. Students were asked to use the Scratch programming language in order to simulate the actual movement of a screw that should be screwed into a piece of wood.

TEACHING ANGLE WITH DIGITAL TECHNOLOGIES

Numerous studies have shown that young students have difficulty understanding the meaning of angle. The difficulties are related both to the difficulties that arise in defining the concept of angle, and to the incorrect generalizations, oversimplifications, illustrations and diagrams presented in mathematics textbooks, which can often lead to misconceptions (Bütüner & Filiz, 2016; Mitchelmore & White, 2000).

Recent research supports the use of digital technologies to teach the concepts of Geometry and consequently the teaching of the concept of angle (e.g. Balomenou, Komis, & Zacharos, 2017). One of the tools used for this purpose is the LOGO programming language and digital activities that implement the features of this language. Research using such technologies has focused on drawing trajectories and paths of moving objects, mazes, the construction of shapes and polygons, and their transformation (e.g. Bartolini, Bussi, & Baccaglioni, 2015; Morgan & Alshwaikh, 2012).

Dynamic Geometry Environments (DGE), such as Geometer's Sketchpad, Cabri Geometry II and Geogebra, are also particularly popular in teaching Geometry. Their main common feature is dragging, which allows students to dynamically manipulate geometric shapes and their properties, but also to make various measurements (e.g. angle, length, area, perimeter, etc.). Various studies that have used these digital technologies (e.g. Fahlgren & Brunström, 2014; Kursat & Aydogan, 2011) have observed significant benefits for students in understanding geometric concepts and in cultivating their spatial thinking.

Other researchers (e.g. Foerster, 2016) have incorporated various visual programming languages into activities that promote the understanding of geometric concepts. These languages use visualized commands in the form of blocks that are conveniently connected like puzzles and give immediate feedback to students on the outcome of the commands they use each time. The present research applies the Scratch programming language to a structured mathematical environment to cultivate the concept of angle.

RESEARCH QUESTIONS

The present research aims to answer the following questions:

1. Can students digitally simulate the motion of an object rotating around a center of rotation using Scratch?

2. How do they relate the numerical value of the angle to the specific characteristics of the rotation mode (rotation direction, number of rotations, minimum rotation, maximum rotation) in the digital environment?

METHODOLOGY

Sample

The research involved 35 sixth grade students (11 years old) who attended two public primary schools in a large city in Western Greece. These students had been taught the concept of angle up to the 5th grade through the standard Mathematics curriculum in a formal classroom. Although basic computer skills have been taught in the Informatics course, they had no previous experience in coding and programming activities. Also, they have not previously been taught mathematics using computer applications or software. Therefore, their participation in the research is their first contact for the connection of computers and mathematics. However, it should be noted that the simulation presented in this article, which the students were asked to develop, is the fourth simulation they developed in the entire study. The previous three simulations they had design related to the angle as a rotation. In this context they simulated the movement of a fan, the movement of a door handle and the movement of a key in the lock. Thus, they were already familiar with the procedures for developing real-life angle simulations using Scratch. These students were selected as we believe that, at this age level, they can handle Scratch effectively, which is the main tool for conducting research.

Material

The categorization of the concept of angle proposed by Mitchelmore & White (1998) was used to implement the research. According to them, the angle includes 7 categories (rotation, meeting, inclination, corner, turning, direction, opening), of which the present research presents the case of rotation. This categorization is particularly useful, as it presents the angle in the real environment and thus gives the possibility for designing such real situations using digital tools. The students were given a semi-structured digital microworld designed by the researchers with Scratch. This microworld resembled a screw and a piece of wood into which the screw should be screwed. Students were asked to work individually on their computer and complete the code that already existed so that their simulation would show the correct rotation of the screw that screws into the wood. They could choose between simple commands such as "turn right" or "turn left" defining the size of the rotation angle themselves, but also choose iteration structures by defining the number of repetitions themselves (e.g. "repeat 2 times", "repeat forever", etc.). In the environment there were also commands that did not serve the purpose of the students to rotate (e.g. "resize", "play sound" etc.). In essence, this microworld is a half-baked microworld (Kynigos & Moystaki, 2014), where students were able to freely choose the commands so that the movement of the simulation represented the real experience.

Procedure

The students worked individually, each on a separate computer. They were connected to an individual internet account so that they could access and edit the semi-structured microworld. They had one teaching hour to complete the simulation. Each student's screen was captured with Bandicam software for better understanding of the students' solution strategies.

RESULTS

The results obtained after the qualitative analysis of the code developed by the students for the presentation of the rotational movement of the screw are presented in Table 1. The effort of each student was characterized as "Correct" when the commands of the code were correct, and the simulation was similar to the realistic movement (e.g.: "repeat 5 times [turn 360° right]"). When the code commands were partially correct but did not sufficiently simulate realistic motion, they were labeled "Incomplete" (e.g.: "forever [turn 15° right]"). When the code commands were incorrect or/and unrelated to realistic motion, they were labeled "Incorrect" (e.g.: "turn 2° right").

TABLE 1
Students' performance in screw movement simulation

| Angle Situation | Degree of Simulation Success | Number of Students |
|-----------------|------------------------------|--------------------|
| Screw | Correct | 29 |
| | Incomplete | 6 |
| | Incorrect | 0 |
| TOTAL | | 35 |

It turns out that 29 of the 35 students (83%) managed to develop a set of commands so that the screw could be screwed into the wood. In contrast, 6 students (17%) developed an incomplete set of commands that cannot properly represent the rotational movement of the screw, which will be described below.

An important finding is the fact that all students (even those who developed incomplete sets of commands) used only commands that rotate the screw clockwise, that is, in the direction that the screw actually screws. It seems that the students used their real experience in this state of rotation to transfer the movement of this model to the digital environment.

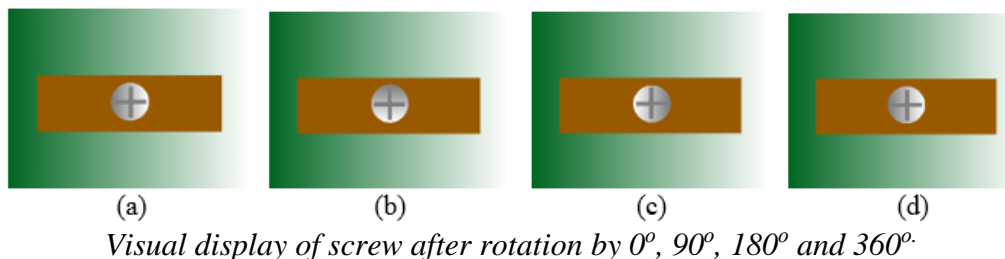
The recording of the students' screen shows that the first attempt made by the majority of students (21 students) is to test the state of rotation. Initially, they try to simply turn the screw to detect movement. For example, these students use the command "turn 10° right" or "turn 15° right" or "turn 50° right" and observe how the model moves after their intervention. This first command helps students to make a second and third attempt to improve the movement of the screw. So, it seems that initially they want to test the motion of the model given to them using the "turn" command at a certain angle. A smaller group (12 students) seems to be more fluent in developing the code for rotating the screw and using repetitive structures (loops). The iteration structures they use have a limited number of iterations and include the "turn" command. Two examples of code for students in this group are: a) "repeat 5 times [turn 15° right]" and b) "turn 90° right, wait 1 second, turn 90° right, wait 1 second, turn 90° right, wait 1 second" etc. Finally, there are 2 students who, initially, use the repetition structure "forever", which, however, cannot represent the correct rotation of the screw, as it will show it rotating constantly. Specifically, both students choose the command "forever [turn 15° right]".

Also remarkable are the results obtained from the analysis of the students' overall efforts to develop the correct simulation of the screw movement. It seems that many of the students' efforts include commands with the basic angles they already know. Specifically, out of the 121 total attempts made by the students, 39 attempts (32.2%) contain the standard angles 90°, 180° and 360°. For the first two angles, students use them either independently (e.g. "turn 90°" or "turn 180°") or in a repetition structure (e.g. "repeat 2 times [turn 90°]" or "repeat 2

times [turn 180°] »). As for the 360° angle, students only use it simply (turn 360°) and never in a repetition structure. However, 4 attempts were found in which the students constructed a complete rotation within a repetition structure (e.g. "repeat 10 times [turn 36°]" or "repeat 36 times [turn 10°]"). The rest of the students' efforts include rotations at simple angles of different sizes (e.g. "turn 60°") or rotations in iterative structures without trying to construct the known angles 90°, 180° and 360° (e.g. "repeat 10 times [turn 40°]").

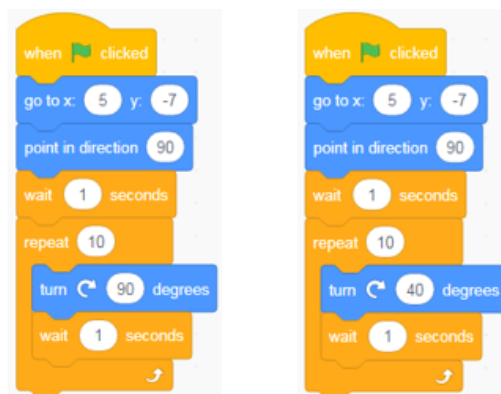
It is also important to note that students effectively used the strengths and weaknesses of Scratch to design the simulation. A major weakness of Scratch is that it shows only the initial position of the screw (before rotation) and its final position (after rotation), without showing the intermediate positions through which it passes. For example, if the screw rotates 90°, its visual image is not quite different, since the screw appears as a cross in its initial position and as a cross in its final position. A small difference arises only from the change in the color of the screw, which students should notice. Figure 1 shows the screw rotations by (a) 0°, (b) 90°, (c) 180° and (d) 360°.

FIGURE 1



An even more difficult case is the rotation of the screw by 360°, since the initial and final position of the screw are identical. Nevertheless, the students handled these difficulties effectively. Thus, some students chose these commands, although the rotation of the screw was not clearly visible. This indicates that they were aware of the effect of the rotation that was occurring even though it did not appear adequately on their screen. As soon as other students noticed Scratch's inability to present the intermediate positions of the screw, they gave instructions so that they could show the intermediate positions and show the rotation. They achieved this by selecting a small numerical value for the angles and using in the middle of the code the "wait 1 second" command in order to stop for a while the screw from rotating and showing its intermediate positions. Figure 2 shows the code of a student that used this strategy. In code (a) he found that the rotation is not visually correct and then used code (b) to make it look even better.

FIGURE 2



Improved code for better visual display of rotation

Finally, 5 of the 6 students who used an incomplete set of commands showed the rotation of the screw with the command "turn forever". Thus, the simulation they designed does not correspond to the characteristics of the movement of the screw, since it cannot rotate forever. One student used the command "turn 2° right" which although it is a correct code choice to rotate the screw, the numerical value she chose is so small that cannot make it screw into the wood. However, they did not use other irrelevant commands that they could select from all the Scratch commands (e.g. resize, play round, etc.), but chose the "turn" command.

CONCLUSIONS

Research has shown that most students are able to develop a set of commands to demonstrate the movement of the screw. At the same time, those who designed an incomplete set of commands used the "turn forever" command, ignoring the fact that the screw cannot be rotated continuously. However, no one used a command completely irrelevant to the model requirements, such as the "resize" command etc.

The simulation designed by the students takes place in the angle frame as a rotation and the students seem to consider the specific characteristics of the rotation they are asked to develop. Thus, they understand that the model must rotate by an angle and not show constant motion. At the same time, from their experience it seems that they know the clockwise screwing of the screw and use the command "turn right" to present this screwing in a realistic way. They also avoided using small angles in the movement of the model (with the exception of a schoolgirl who turned the screw by 2°), as they apply their real experience, according to which to screw a screw must be rotated at a "reasonable" angle which will allow it to be kept in the wood. Therefore, it seems that students in their attempt to develop the simulation use their daily knowledge and real experience, which provide them with all that useful and necessary information for the presentation of the realistic screw movement.

Also, research has shown that some students, although using commands with which the screw can be screwed (e.g.: "turn 180°"), try to reduce the size of the numerical value of the angle in order to present its movement even more realistically on their screen. Thus, it seems that the size of the angle acquires meaning within the context of rotation and the specific characteristics of the situation. The process of programming and improving the code in combination with the realistic representation of the simulation seems to strengthen the understanding of the concept of angle, in general, but also of specific angles, in particular (0°, 180° and 360°) which according to other researchers (Bütüner & Filiz, 2017; Devichi & Munier, 2013) create misunderstandings in students about them.

In conclusion, the research has shown that visual programming using Scratch can help students understand angle as rotation through real-world simulation activities. It also turned out that students can easily manipulate the size of the angle, which is approached by them dynamically and not as a static property of an object or construction, overcoming the limitations of conventional tools used in the teaching of geometry. At the same time, they realized that the concept of angle as rotation and its numerical value are the basic elements for the successful simulation of the rotation model.

The results refer only to the specific sample of students who participated in the research and to the specific characteristics of their age group. The development of the research aims at extracting results from simulation activities of different real-life angle situations and combining these results.

REFERENCES

- Balomenou, A., Komis, V., & Zacharos, K., (2017). Handling signs in inequalities by exploiting multiple dynamic representations – the Case of ALNuSet. *Digital Experience in Mathematics Education*, 3(5), 39-69.
- Bartolini Bussi, M. G., & Baccaglioni-Frank, A. (2015). Geometry in early years: Sowing the seeds towards a mathematical definition of squares and rectangles. *ZDM Mathematics Education*, 47(3), 391-405.
- Bütüner, S., & Filiz, M. (2017). Exploring high-achieving sixth-grade students' erroneous answers and misconceptions on the angle concept. *International Journal of Mathematical Education in Science and Technology*, 48(4), 533-554.
- Devichi, C., & Munier, V. (2013). About the concept of angle in elementary school: Misconceptions and teaching sequences. *Journal of Mathematical Behavior*, 32(1), 1-19.
- Fahlgren, M., & Brunström, M. A. (2014). A model for task design with focus on exploration, explanation, and generalization in a dynamic geometry environment. *Technology, Knowledge and Learning*, 19(3), 1-29.
- Foerster, K. (2016). Integrating programming into the Mathematics curriculum: Combining Scratch and Geometry in grades 6 and 7. In *SIGITE'16* (pp. 91-96). Boston, MA, USA.
- Henderson, D. W., & Taimina, D. (2005). *Experiencing geometry. Euclidean and non-Euclidean with history*. New York: Cornell University.
- Kursat, A., & Aydoğan, A. (2011). The effect of inquiry-based explorations in a dynamic geometry environment on sixth grade students' achievements in polygons. *Computers & Education*, 57(4), 2462-2475.
- Kynigos, C., & Moustaki, F. (2014). Designing digital media for creative mathematical learning. In O. Iversen, E. Elbæk, B. Thomsen, P. Markopoulos, F. Garzotto & C. Dindler (Eds.), *Proceedings of the Interaction Design and Children 2014* (pp. 309-312). Aarhus, Denmark.
- Mitchelmore, M., & White, P. (1998). Development of angle concepts: A framework for research. *Mathematics Education Research Journal*, 10(3), 4-27.
- Mitchelmore, M., & White, P. (2000). Development of angle concepts by progressive abstraction and generalization. *Educational Studies in Mathematics*, 41(3), 209-238.
- Morgan, C., & Alshwaikh, J. (2012). Communicating experience of 3D space: Mathematical and everyday discourse. *Mathematical Thinking and Learning*, 14(3), 199-225.