Promoting Computational Thinking in K-12 students by applying unplugged methods and robotics

Miguel Á. Conde
Grupo de Robótica - University of León
Escuela de Ingenierías. Campus de Vegazana S/N 24071. León. Spain
mcong@unileon.es

Camino Fernández-Llamas
Grupo de Robótica - University of León
Escuela de Ingenierías. Campus de Vegazana S/N 24071. León. Spain
cferll@unileon.es

Francisco J. Rodríguez-Sedano
Grupo de Robótica - University of León
Escuela de Ingenierías. Campus de Vegazana S/N 24071. León. Spain
fjrods@unileon.es

Ángel M. Guerrero-Higueras
Grupo de Robótica - University of León
Escuela de Ingenierías. Campus de Vegazana S/N 24071. León. Spain
am.guerrero@unileon.es

Vicente Matellán-Olivera
Grupo de Robótica - University of León
Escuela de Ingenierías. Campus de Vegazana S/N 24071. León. Spain
vmato@unileon.es

Francisco J. García-Peñalvo
Grupo GRIAL – University of Salamanca
Facultad de Ciencias. Plaza de los caídos S/N 37008. Salamanca. Spain
fgarcia@usal.es

ABSTRACT

[1] Nowadays students live in the digital age and they do not only should learn to speak, write or develop specific skills. Students needs to be successful in their context and a possible way to achieve this is by developing the computational thinking. In the last few years there are several initiatives to promote computational thinking and to define approaches and methods to support it. One of this is the unplugged methods, in which students develop computational thinking skills without using the technology. This paper presents an experiment to promote computational thinking by using unplugged methods and employing robots as teachers as an engagement factor for the students. During the experiment, they have been distributed in two groups. One has carried out unplugged activities to develop computational thinking while the other did not. From the experiment, it is possible to see that results are better for those students that have completed unplugged activities and there are differences depending on age.

TEEM 2017, October 18–20, 2017, Cádiz, Spain

ACM ISBN 978-1-4503-5386-1/17/10…$15.00
https://doi.org/10.1145/3144826.3145355

CCS CONCEPTS

• **Applied Computing** → **Education** → Computer Assisted instruction • **Human-centered computing** → Laboratory experiments; • **Computer systems organization** → **Embedded and cyber-physical systems** → Robotics

KEYWORDS

Educational Robotics, Unplugged Science, Computational Thinking, Child education.

1 INTRODUCTION

Computational Thinking in the digital society has become an essential skill to be acquired by children in order to his something essential to increase individuals success of children in their current contexts and increase their performance and employability once they reach the labor market [1-4]. There are many proposals to encourage computational thinking [2, 5-7] in K-12 curricula. Many projects have explored different programming environments, developed tools, and generated material to teach computing to K-12 students. For instance, some of the authors of these work are involved in the Taccle3 Coding project[1] were a website of ideas and resources is being develop together with in-service training courses and other staff development events.

A possible approach to promote computational thinking are the unplugged methods. Unplugged Computing is the approach to teaching computing concepts using constructivist, often kinesthetic, activities away from computers [8]. It has successfully been used for teaching computing at all age groups from primary school to university level [9-11]).

Given this context the present project aims to assess how the use of unplugged methods to promote computational thinking can affect students results. In order to do so an experiment was carried out. The experiment involves k-12 students from different age ranges. They were randomly distributed in a control an experimental group. Students in the control group were going to carry out a computational thinking activity and the control group did not. Later, both groups carried out another unplugged activity after a robot has explained their students what a loop and a conditional is. We have used the robot because previous experiments have shown us that their use increase students motivation and engagement which helps them to better understand the concepts.

This paper is going to present the experiment and the first results obtained. In order to do so the paper is organized as follows. Second section formulates the hypothesis analyzed in this paper. Third section describes in detail the experiment designed to evaluate these hypotheses. Section four presents the statistical analysis of the data gathered, which are discussed in the fifth section. Finally, section six summarizes the conclusions obtained in this research and the further work envisioned.

2 AIMS OF THE RESEARCH

When this experiment was defined the first idea was to try to discover the benefits to the development of computational thinking by applying unplugged methods. However, as commented above this have been tested in several experiments.

Given this context what is interesting for us is to check two hypotheses:

- H1) The participation in a previous computational thinking activity can affect to students' performance in future similar activities.
- H2) There is a relationship between students' self-perceived knowledge on programming and the abilities shown during the exercise.
- H3) Students perceive that robots can act properly as teachers and this increase their motivation.

In this research work we are going to deal with the first two of these hypotheses.

3 EXPERIMENTAL SETUP

In order to check the hypotheses an experiment was carried out. The experiment involved a group of 129 students aged between 6 and 17 years. These students were in a summer technology camp at the University of León.

It consists of two main activities:

- The first activity (A1) is computational thinking problem that students should solve by using colored game tokens.
- The second activity (A2) is carried out by students after an explanation about programming concepts by a robot teacher. Students should solve a problem by using some cards that represent programming concepts.

In order to check H1, the students were randomly split in two groups. Those in the experimental group carried out A1 and A2. The students in the control group only completed A2. Students in the control group were supposed to have better grades in A2 because they have carried out other similar computational thinking activity.

H2 was checked by comparing A2 grade with the students self-perceive knowledge on programing.

Activities A1 and A2 are described in the following subsections.

3.1 First computational thinking activity – A1

During this activity, the students should fulfil a form with personal information about them. We asked them their age, school course, school name and their previous knowledge in programming. In order to obtain this information, we ask them if they know any programming language and they should write a sentence using it. They could also describe their skill level by choosing one in a four-value Likert scale.

Once they fulfill the form a teacher presents them the exercise to solve. This exercise consisted of writing down the steps they took to complete a figure with colored tokens. For this, the students were given an envelope with the form and the color tokens (Figure 1)

All the students that carried out A1 activity should later complete A2.

3.2 Second computational thinking activity – A2

This activity consists in a class where a robot will explain students programming concepts, and after the lesson they should solve and exercise. We decided to use a robot because in previous experiments [12] we have seen that using a robot as a teacher could increase students' motivation.

Given this fact we decided using Baxter Research Robot manufactured by Rethink Robotics. This robot consists of two seven degree-of-freedom (7DOF) arms, with inbuilt feedback systems, although for our experiment we only used one of the arms. It was developed to allow collaborative human-robot co-work and to improve Human-Robot Interaction.

In order to use Baxter as a teacher, voice capabilities were added using a free software for building speech synthesis2 and external speakers. The display of the robot also played an important role. A friendly image with moving eyes was used in order to ease up the robot's appearance and improve human-robot interaction. shows Baxter during a lecture. Figure 2 shows Baxter during a lecture.

[2] http://www.cstr.ed.ac.uk/projects/festival

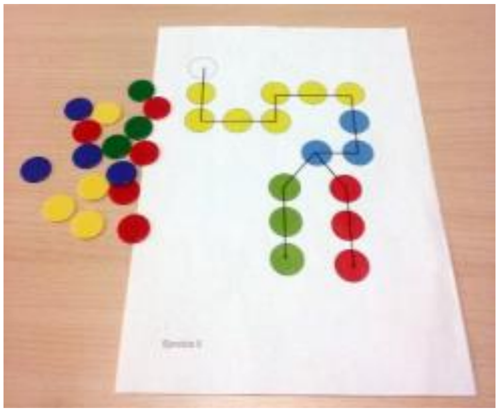

Figure 1: Material delivered to the students of the A1 activity.

Figure 2: Baxter robot lecturing

In this class, the robot explained two basic programming concepts: Repetition and Condition. To do this, Baxter used the slide support in which students could see a scheme of both concepts. In addition, Baxter would demonstrate as a demo of these two concepts. In the demo, to explain the repetition, Baxter grabs a ball from a ramp and drops it into a box. This action repeats even if there are no balls on the ramp. To explain the condition, Baxter takes a ball from a point on the table and leaves it in the box, but in this case, only if there is a ball at that point.

At the end of the explanation, the robot asked the students to complete an exercise. The exercise consisted in ordering a set of cards to obtain a figure formed by colored tokens. The figure was similar to that of the exercise performed by one of the groups of students in the class with the human teacher.

To carry out the exercise, each student was given an envelope containing the statement of the exercise and different cards (Figure 3). These cards represented the tokens of the different colors, numbers and the "if" and "repeat" statements. The students could not solve the exercise without using these two cards (if and repeat) because there were not enough cards of colored tokens in the envelope. After completing the exercise, students also had to fill out a questionnaire.

In Figure 4 it is possible to see students working on this activity.

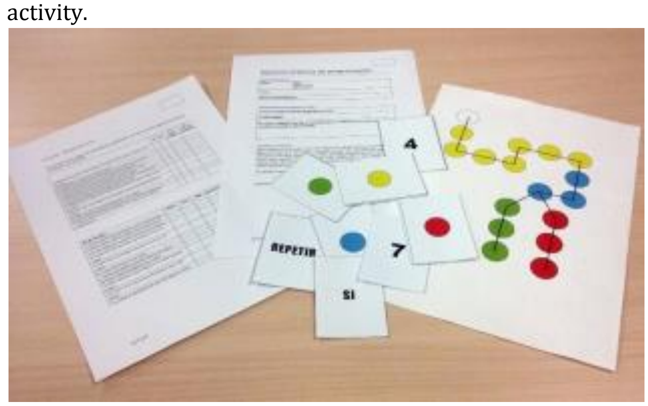

Figure 3: Material delivered to the students of the class with the robot teacher

Figure 4: Students working during the experiment

3.3 Methodology

Experimental setup has an important role to analyze the influence of previous knowledge and previous activities of students in their performance. In this particular case, we can contrast what happens when some of them carry out an activity while the other will not. In order to define the experiment methodology, we should take into account that we have students distributed in groups as commented above, some are going to be part of the treatment group and the rest will be included in the control group. This type of distribution fits with a between subject experimental design [13]. Although between-subject design has several limitations, with these types of designs, as long as group assignment is random, causal estimates are obtained by comparing the behavior of those in one experimental condition with the behavior of those in the other [14].

In order to carry out the experiment each group of students was escorted by their monitor at the summer camp. During the experiment, besides several human teachers and Baxter robot, there was at least another person from the organization for assisting the whole process as well as the monitor.

Monitors were in charge of providing students with the exercise sheets, ball-pens, and anything they would need. They would also answer questions about the exercises and the questionnaires.

The experiment was repeated three times along three consecutive weeks with different groups of students.

4 EXPERIMENTAL SETUP

The total number of students involved in the experiment was 129. There was a 3.30 ratio of male/female (90/39). It is remarkable that male was prevalent in all age groups, as shown in Figure 5.
58 of them carried out A1 (experimental group) and the remaining 71 did not complete this experience (control group). After this, the students in both groups participate in A2.

Students' age range was from 6 to 17. Experiment results have been analyzed as a whole and also by groups. Groups division were based on the Spanish Educational Law groups. We will name students from 6 to 9 as Primary school 1, students from 10 to 13 as Primary school 2, and students from 14 to 17 as Secondary school.

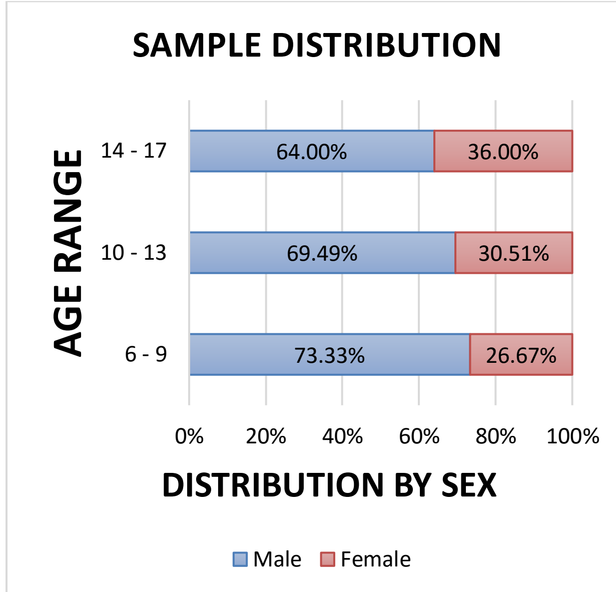

Figure 5: Sample distribution in age range and gender

Table 1 shows the number of students in each age range.

Table 1: Number of students per age range

	Male	Female
6 – 9 – PRIM1	33	12
10 – 13 – PRIM2	41	18
14 – 17 - SEC	16	9

The idea is to check the posed hypothesis by analyzing the data gathered during the experiment. This information is quantitative, so descriptive analysis of results is presented in the following tables and also the correlation between several variables.

In table 2 we can see the average grade obtained in the final computational thinking activity carried out by all the students. It is possible to see the average grade differencing between males and females. As the number of female students is very low in comparison with male students it is not easy to compare results, so we are going to explore the full sample without making gender based conclusions.

Table 2: Sample size, average grade and standard deviation for the students of the experiment distributed by gender

	N	$\bar{x}$	σ
Total	129	6,64	2,786
Male	90	6,36	0,304
Femal e	39	7,31	0,392

As we have used a between-subject experimental design, it is necessary to have similar number of students in both the control and the experimental group This has meant that the sample size should be reduced to 54 participants in each group, that is, a sample with 108 students. The selection of the students to remove was carried out randomly. The average for each group with 54 students can be seen in Table 3.

Table 3: Sample size, average grade and standard deviation for the students of the experiment distributed by control and experimental group

	N	x̄	σ
Total	108	6,66	2,782
C. Group	54	6,54	2,793
E. Group	54	6,78	2,793

In table 4 it is possible to see the average grade for each age range and taking into account if students have participated in the experimental group (symbolized with an E) or in the control group symbolized with (symbolized with an C).

Table 4: Sample size, average grade and standard deviation for the students of the experiment and control group distributed by age range

	N_E	N_C	$\overline{X_E}$	$\overline{X_C}$	σ_E	σ_C
PRIM I	11	14	7,27	5,64	1,191	2,373
PRIM II	33	26	6,64	7,00	3,210	2,828
SEC	10	14	6,70	6,57	2,710	3,081

5 DISCUSSION

From the data gathered during the experiment it is possible firstly to see than independently of the age range there are much more men than woman involved. This fact is something common in computational thinking initiatives, and there are lot of efforts and funds devoted to balance this distribution [15-17].

When comparing the experimental and the control group it is possible to see that there is a difference in the grade of those students in the experimental group. Something that could be understood as normal because they have dealt with a computational thinking activity previously to Baxter explanation and the final unplugged activity. However, the difference is so small that it cannot be understood as significant, as can be checked by applying a Student T test (Signification of 0.665 > 0,05).

However, it is possible to study also differences between students in different age groups. The greatest difference is for the younger students. This can be motivated because in this age they change their way to learn. Piaget [18] saw the age of 7 as a major cognitive turning-point; around this age children make the important transition from preoperational to the more advanced concrete operational stage. Children begin to learn about classifications and temporal relations. Therefore, for this group of students, dealing with a computational thinking problem such as A1, that is similar to A2 exercise, could help them to improve their performance.

There are also differences between students in PRIM II or SEC groups, however in these cases the grade can be influenced by their familiarity with technology or how computational thinking skills are being developed in their learning institutions. This means that hypothesis 1 should be retained, that is, from this experiment it is not possible to conclude that "The participation in a previous computational thinking activity can affect to students' performance in future similar activities". Although it seems that for younger students the hypothesis can be accepted although the hypothesis should be check with more students (only 25 are involved in the analysis).

Regarding the second hypothesis we have carried out a correlation analysis between the previous experience in programming that students say to have and the grade obtained in the final activity.

We Spearman Rho statistical technique because the experience and the exercise scores are ordinal variables. This statistical test returns a value of 0,219 which means that there is a medium correlation between students' self-perception of experience and the results in the exercise. In order to check this also a One-Way ANOVA is applied. The one-way ANOVA requires to assure normality and homogeneity, although normality is not a strict requirement. Taking this into account, homogeneity was checked using the Levene test. This test returns a signification of 0,001 which is lower than 0,05. This means that there is not homogeneity between groups, so we applied a Welch ANOVA test. The results of this test show a signification of 0,002 which is lower that 0,05. This allow us to say that there is a relationship between previous expertise and the result in their final activity. Something similar has been found in other research works related with computational thinking [19, 20] and other works has found out that there is not relationship [21], so as a future research work it would be desirable to involve more students and to study the relationship for each age group

6 CONCLUSIONS AND FURTHER WORK

In our current context students should be able to solve problems in their daily life with their daily tools and in most of the cases by using ICT. In order to be successful in a very technological society as the current one, it is necessary that they develop computational thinking skills. This may also help them to address other kind of problems with several benefits. In this sense, there are several initiatives and in this work, we have presented one of them.

In our case, we aimed to check two issues: 1) If previous interaction with computational thinking activities without any programming explanation could help students to solve more complex computational thinking problems; and 2) If previous programming knowledge has associated better students' performance.

For the first issue, and with the limitations of our experiment we can say that previous interaction does not seem to have an impact in future computational thinking activities. However, we would be possible to assert this positively with young students. In future experiments, we should explore this issue with more students and more detail.

For the second issue, it seems that there is some relationship between previous knowledge in programming and better performance. In order to check this also with new experiments should be carried out, and it would be interesting to study this relationship taking into account gender and specific age ranges.

REFERENCES

[1] García-Peñalvo, F. J., Reimann, D., Tuul, M., Rees, A. and Jormanainen, I. *An overview of the most relevant literature on coding and computational thinking with emphasis on the relevant issues for teachers*. Belgium, 2016.

[2] García-Peñalvo, F. J. What computational thinking is. *Journal of Information Technology Research*, 9, 3 (2016), 5-8.

[3] Pinto-Llorente, A. M., Casillas-Martín, S., Cabezas-González, M. and García-Peñalvo, F. J. Building, coding and programming 3D models via a visual programming environment. *Quality & Quantity* (April 22 2017).

[4] Wing, J. Computational thinking benefits society. *40th Anniversary Blog of Social Issues in Computing* (2014).

[5] Bers, M. U., Flannery, L., Kazakoff, E. R. and Sullivan, A. Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers & Education*, 72 (2014), 145-157.

[6] Llorens Largo, F., García-Peñalvo, F., Vendrell Vidal, E. and Molero Prieto, X. Educación en Informática sub 18 (EI< 18). *ReVisión*, 10, 2 (2017), 13-18.

[7] Llorens-Largo, F., García Peñalvo, F. J., Molero-Prieto, X. and Vendrell-Vidal, E. La enseñanza de la informática, la programación y el pensamiento computacional en los estudios preuniversitarios. *2017*, 18, 2 (2017).

[8] Bell, T. A low-cost high-impact computer science show for family audiences. In *Proceedings of the Proceedings 23rd Australasian Computer Science Conference. ACSC 2000* (Canberra, AU, 2000).

[9] Curzon, P. and McOwan, P. W. Engaging with computer science through magic shows. *SIGCSE Bull.*, 40, 3 (2008), 179-183.

[10] Cutts, Q. I., Brown, M. I., Kemp, L. and Matheson, C. Enthusing and informing potential computer science students and their teachers. *SIGCSE Bull.*, 39, 3 (2007), 196-200.

[11] Curzon, P., McOwan, P. W., Plant, N. and Meagher, L. R. Introducing teachers to computational thinking using unplugged storytelling. In *Proceedings of the Proceedings of the 9th Workshop in Primary and Secondary Computing Education* (Berlin, Germany, 2014). ACM.

[12] Conde, M. Á., Fernández-Llamas, C., Rodríguez-Lera, F. J., Rodríguez-Sedano, F. J., Matellán, V. and García-Peñalvo, F. J. Analysing the attitude of students towards robots when lectured on programming by robotic or human teachers. In *Proceedings of the Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality* (Salamanca, Spain, 2016). ACM.

[13] Bordens, K. S. and Abbott, B. B. *Research Design and Methods: A Process Approach*. McGraw-Hill Education, 2007.

[14] Charness, G., Gneezy, U. and Kuhn, M. A. Experimental methods: Between-subject and within-subject design. *Journal of Economic Behavior & Organization*, 81, 1, 1-8.

[15] Margolis, J. and Fisher, A. *Unlocking the clubhouse: Women in computing*. MIT press, 2003.

[16] Grover, S. and Pea, R. Computational thinking in K–12: A review of the state of the field. *Educational Researcher*, 42, 1 (2013), 38-43.

[17] Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S. and Korb, J. T. Computational thinking in elementary and secondary teacher education. *ACM Transactions on Computing Education (TOCE)*, 14, 1 (2014), 5.

[18] Piaget, J. *The Child's Conception of the World*. Routledge, London, 1929.

[19] Kurland, D. M., Pea, R. D., Clement, C. and Mawby, R. A study of the development of programming ability and thinking skills in high school students. *Journal of Educational Computing Research*, 2, 4 (1986), 429-458.

[20] Byrne, P. and Lyons, G. *The effect of student attributes on success in programming*. ACM, 2001.

[21] Wiedenbeck, S. Factors affecting the success of non-majors in learning to program. In *Proceedings of the Proceedings of the first international workshop on Computing education research* (2005). ACM.