



HAL
open science

Comparing three different approaches to teach programming and computational thinking in 5th grade

Kevin Sigayret, Nathalie Blanc, André Tricot

► To cite this version:

Kevin Sigayret, Nathalie Blanc, André Tricot. Comparing three different approaches to teach programming and computational thinking in 5th grade. The 27th Annual JURE Conference for Research on Learning and Instruction, Aug 2023, Thessaloniki, Greece. hal-04755432

HAL Id: hal-04755432

<https://hal.science/hal-04755432v1>

Submitted on 31 Oct 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Comparing three different approaches to teach programming and computational thinking in 5th grade.

Kevin SIGAYRET, Laboratoire Epsilon, Université Paul Valéry Montpellier 3

Nathalie BLANC, Laboratoire Epsilon, Université Paul Valéry Montpellier 3

André TRICOT, Laboratoire Epsilon, Université Paul Valéry Montpellier 3

Abstract (150-250 words)

Programming and computational thinking are gradually being integrated into school curricula. Many researchers believe that this learning would have a positive impact on certain abilities essential to any citizen in the decades to come, particularly in relation to problem solving abilities. However, there are many ways, both plugged-in and unplugged, to teach programming in school and very few studies attempted to compare these different approaches. For this reason, we conducted an experiment with 319 Grade 5 students in France. They were assigned to three experimental conditions: learning with educational programming software (Scratch-only group), learning with software and an educational robot (Scratch + Thymio group) and unplugged learning (Unplugged group). Our aim was to examine whether significant differences can be observed between the three groups regarding knowledge and understanding of fundamental concepts in programming, ability to solve algorithmic problems, motivation, self-belief and attitude toward science. Our main results show that the Scratch-only group exhibited better learning performance compared to the other groups, with the lowest performance for the unplugged group. Explanations could be sought by looking at the influence of the feedback allowed by the programming software and by evaluating the cognitive load associated with the different learning situations. A plugged-in approach also seems to benefit to students' motivation, compared to the unplugged group which showed a significant drop in motivation at the end of the sessions, but this positive impact is only observable with boys.

Keywords (4) : Programming – Computational Thinking – Primary education – Teaching/Instructional Strategies

Extended summary (600-1000 words)

Theoretical background: Programming Education in K-12 has been developing rapidly in recent years in many countries (Eurydice, 2019; Guo & Ottenbreit-Leftwich, 2020). Indeed, this discipline would allow learning of computational thinking (which refers to "the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent", Wing, 2011), an essential skill for the 21st century (Mohaghegh & McCauley, 2016). Different pedagogical devices are used to teach programming at school. Visual programming languages are supposed to be more accessible for young students and novices (Sáez-López et al., 2016; Scherer, et al., 2020) and could have a positive effect on motivation (Armoni et al., 2015). Among the most popular, Scratch (Resnick et al., 2009) is used in most studies assessing the impact of educational programming software. The use of educational robots also seems to be indicated to develop computational thinking (Atmazidou & Demetriadis, 2016; Ioannou & Makridou, 2018), alone or in addition to educational programming software. Finally, some researchers emphasize unplugged activities, *i.e.* without the use of digital tools, which would decrease the cognitive load linked to the use of machines and would promote an embodied learning of programming (Romero et al., 2018). Experimental studies seem to confirm the positive impact of unplugged programming on computational thinking (Brackmann et al., 2017; Delal & Oner, 2020). However, there are currently very few researches that propose to compare these different devices to assess their relative effectiveness (Zhang & Nouri, 2019; del Olmo-Muñoz et al., 2020) and we lack empirical data to draw strong conclusions on this issue.

Aim: The aim of this study is to evaluate the effect of devices (unplugged vs. visual programming vs. visual programming + robot) on learning performance and other variables related to students' subjective experience (motivation, self-belief and attitude toward science). Results concerning the comparison between unplugged and visual programming (without robot) conditions have already been published (Sigayret et al, 2022).

Method: We conducted an experiment involving 319 5th grade students, all novices in programming. Students were introduced to programming through three different learning devices that correspond to our three experimental groups. Students were recruited near Montpellier, south of France. Each class was randomly assigned to an experimental group. In the unplugged condition, students used paper instructions designed to replicate the blocks available on Scratch (unplugged group). In the plugged-in condition, students used the Scratch 3.0 software without using an educational robot (Scratch-only group) or also using the Thymio II Wireless robot, programmable on Scratch (Scratch + Thymio group). Ten 45-minute sessions were set up so that the concepts and skills taught were the same in all three groups. We designed a mastery of computational concepts test and an algorithmic

test to assess understanding of programming concepts and ability to solve algorithmic problems. These two tests were previously validated with a sample of 449 students, from 4th to 9th grade, showing good reliability ($\omega = 0.79$ and $\omega = 0.85$, respectively) and moderate correlation with students' level of programming practice ($r = 0.60$ and $r = 0.58$, respectively), indicating concurrent validity. In order to assess students' motivation and self-belief, we used items retrieved from Students' Approaches to Learning (SAL) Instrument (Marsh et al., 2006). Attitude toward science was measured using a combination of items from the Modified Attitudes toward Science Inventory (Weinburgh & Steele, 2000) and another test assessing the attitude toward science (Kind et al., 2007).

Results: One-way ANOVA was used to analyse results from the two performance tests. We found significant differences regarding mastery of computational concepts between the three groups ($F(2, 281) = 15.6$; $p < .001$). Scratch-only group performed significantly better than Scratch + Thymio group, which in turn performed better than unplugged group. For the Algorithmic Test, we also found significant differences ($F(2, 277) = 25.5$; $p < .001$). Scratch-only group also showed better performance comparing to the two other groups but this time, differences between unplugged and Scratch + Thymio groups were not significant.

Motivation and self-belief Test and Attitude Toward Science Test were conducted pre and post-test. We analysed the results with Repeated Measure ANOVA. Regarding motivation-only items, post-hoc comparisons showed a drop in motivation at the end of the sessions in the unplugged condition ($t = 3.14$; $p = .028$). Motivation in the Scratch + Thymio group remained stable (Figure 1). Gender seemed to have an impact on motivation, as shown in figure 2. Boys (M on the figure) seemed to be significantly more motivated in both plugged-in conditions (Scratch-only and Scratch + Thymio groups) compared to girls (F). We also found minor results regarding self-belief but no significant results concerning attitude toward science.

Figure 1 : Pre-test/Post-test evolution regarding motivation items in the experimental groups

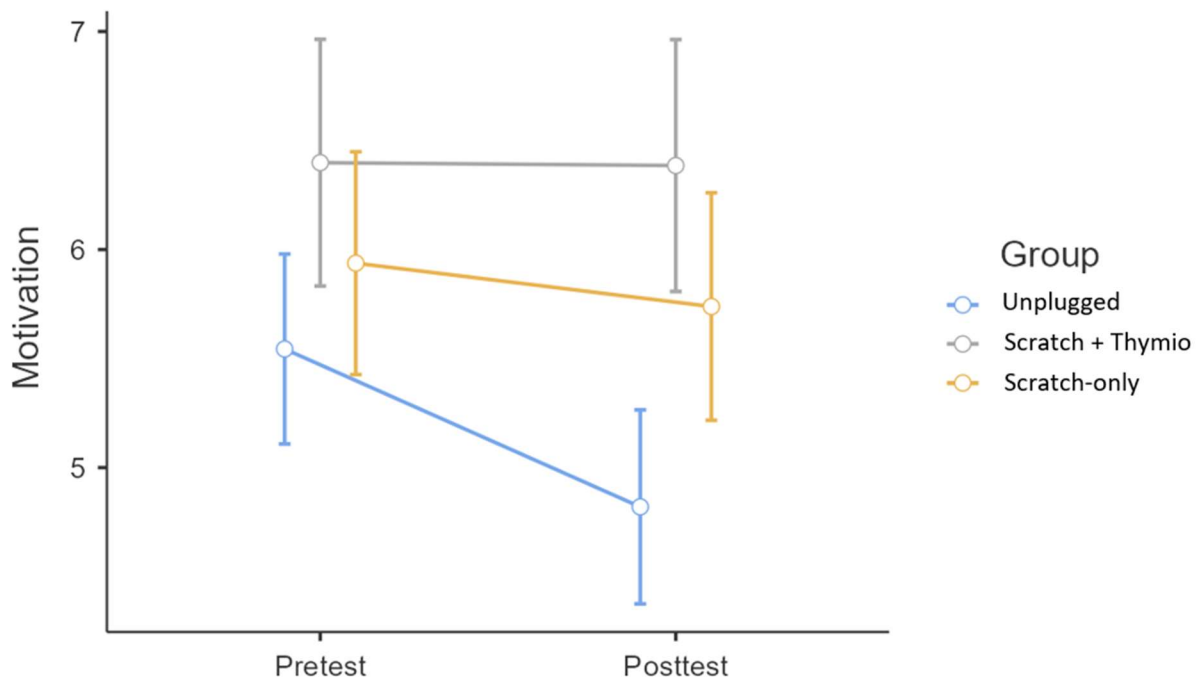
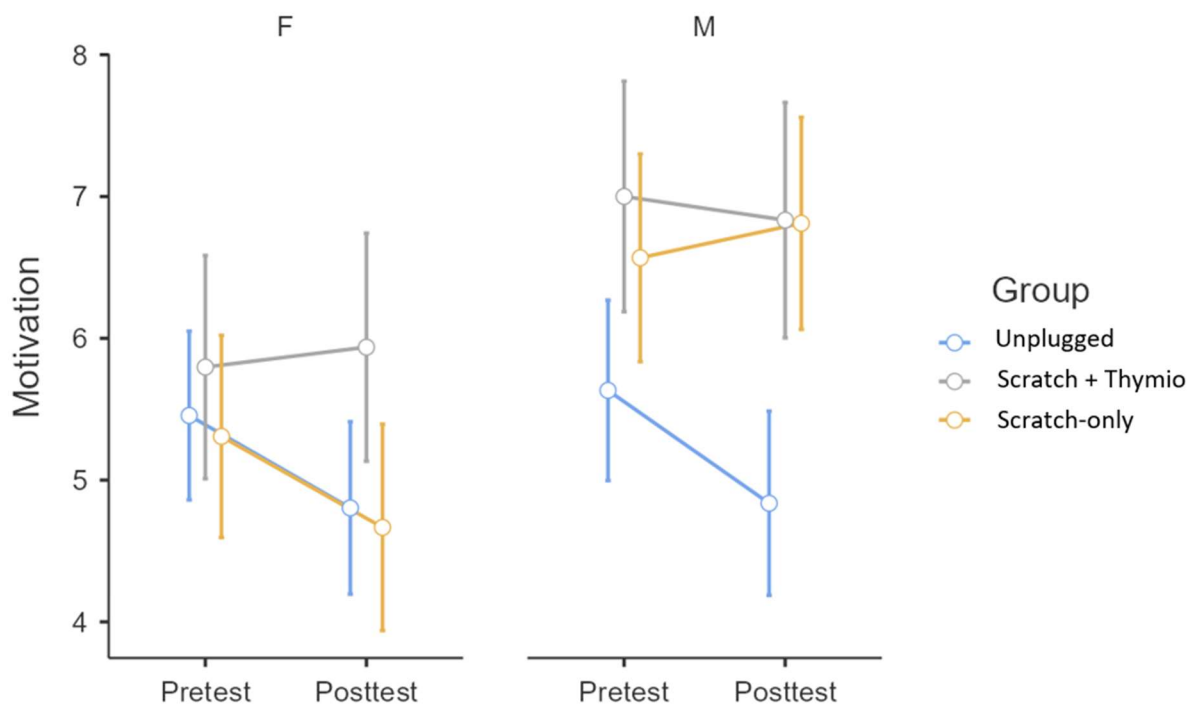


Figure 2: Pretest/Posttest evolution regarding motivation items in the experimental groups by gender.



Discussion: Results show a benefit in using Scratch visual programming software when learning computational concepts and skills linked to computational thinking. These results contradict previous research that reported better performance in the unplugged group (Wohl et al., 2015; Del Olmo-Muñoz et al., 2020). Nevertheless, these studies involved younger students (5 to 7 years old), less

accustomed to computers and software, and are often focused on mixed approaches, comparing unplugged and plugged-in programming versus plugged-in only. A plugged-in approach allows for basic but automatic feedback from the machine that could positively impact learning (Hattie & Timperley, 2007). Interestingly, the addition of educational robots to this setup seems to decrease students' performance. A plausible hypothesis is that the cognitive load (Sweller, 1994) involved in learning computational thinking concepts + manipulating a programming software + manipulating robot is too highly demanding for 5th grade students. However, robots seem to maintain students' motivation at a high level during the sessions, whereas the other two conditions show a drop in motivation (significant for the unplugged group). Results concerning motivation show a positive but gendered impact of a plugged-in learning of programming (in favour of boys). It seems that stereotypes concerning girls and the use of computer tools are already observable at the end of primary education.