From Automaton to AI Robot: the Added Value for Learning

Marie Martin, Thomas Deneux and Morgane Chevalier

April 19, 2024

From automaton to AI robot: the added value for learning.*

Marie Martin[1,2][0000−0001−5981−3696], Thomas Deneux[1][0000−0002−9330−7655], and Morgane Chevalier[4][0000−0002−9115−1992]

[1] Université Paris-Saclay, CNRS, Institut de Neurosciences Paris-Saclay, France
{marie.martin.2,thomas.deneux}@cnrs.fr
[2] Université Paris Cité, Paris, France
marie.martin@cri-paris.org
[3] Haute École Pédagogique du canton de Vaud (HEP-VD), Lausanne, Switzerland
{morgane.chevalier}@hepl.ch

Abstract. Since educational robotics has been integrated into school curricula, the number of robots on the market has continued to grow. Today, it is even possible to increase the degree of autonomy of some of these ground robots by connecting them to artificial intelligence software. However, the use of these new functionalities raises the question of their added value for learning. In this position paper, we compare the potential of each category of robot (traditional robot versus AI robot) to promote learning. As a result, we present scenarios where self-learning robots enable students to reflect on their cognitive processes and learn how to learn.

Keywords: Educational Robotics · Artificial Intelligence Education · Machine Learning · Metacognition · Primary School.

1 Introduction

The use of robots in education is not new. Numerous studies have confirmed the relevance of educational robotics learning activities (ERLA) in fostering essential skills for academic success, such as computational thinking (CT) [5] [10] [6] [3] and metacognition [12] [17] [8], as early as elementary school. However, with the increasing prevalence of artificial intelligence (AI) in our daily lives, some educational robot programming platforms now incorporate machine learning (ML) systems. Some offer training of a ML model that can later be used in a Scratch or Scratch-like programming interface (MachineLearningForKids.co.uk; Google's Teachable Machine; MIT's Cognimate; Vittascience, France). Others, such as the AlphAI software, allow the training of learning robots and the visualization of AI algorithms (developed at the CNRS, France [12], and commercialized by Learning Robots, France). A large number of other AI robot initiatives have been

* Supported by the FIRE PhD program funded by the Bettencourt Schueller foundation and the EURIP graduate program (ANR-17-EURE-0012).

proposed in the science education field [21]. This development of robots linked to AI software raises the question of the added value of using AI robots over more traditional robots. This question follows the pedagogical concerns raised by Tedre and colleagues [18], who compared the traditional approach to programming (CT 1.0) with the data-driven approach to programming through ML (CT 2.0). Indeed, the growing complexity of computer science (CS) leads us to rethink the way in which CS is taught at school, introducing new knowledge (version CT 2.0 according to [18]) without replacing the basic knowledge (CT 1.0). In addition to this increase in CS knowledge, the teaching of ML seems to offer the possibility of approaching metacognition through learning activities that compare ML and human learning. Therefore, in this paper, we explore the added value of ERLA and ERLA embedding an AI algorithm (AI-ERLA), with a particular focus on the development of metacognition in primary school students.

2 From automaton to event-based programming robots

As Papert [14] has written, robots are objects to 'think with', because being able to manipulate them accompanies thought in action, as a mediating tool. In fact, from the very first years of school, such tools for mediating thought are essential for overall learning (communication, problem solving, etc.).

For more than 40 years now, robots have been entering the educational sphere in an evolutionary manner, moving from automaton that the student controls to robots equipped with sensors that can adapt their behaviour according to the data collected in their environment. While the former operate in an open loop, the latter operate in a closed loop or sensory-motor loop, which only gives the latter the status of robot [1]. In addition to the machines themselves, it is their programming paradigm (sequential vs. event-based) and problem solving approach (deductive according to [18]) that helps or hinders their entry into schools. Sequential programming seems to be better suited to young students, as sequentiality is dealt with at the start of schooling in the activities involved in learning to read and write. [7]. In addition, human-computer interaction is also at stake. In this respect, visual programming interfaces have the advantage of being able to make the language accessible to young students, by enforcing the correct coding syntax and making it visually explicit [11]. Moreover, some engineers and researchers go so far as to make the programming interface tangible too in order to make robot programming accessible to even younger students and reduce the mental load [16].

Using "objects to think with" (the robot and its tangible programming interface), allows to leverage children's embodied experiences in the world in order to better help them enter the new universe of computer code and digital artifacts [9]. Thus, through the use of automaton and robots to be programmed, students are encouraged to develop 21st century skills [19] such as communication, collaboration, creativity and critical thinking. Added to this are 'information, media and technology skills', particularly knowledge of machines (sensors, actuators, etc.) and knowledge of the languages used to program these machines. In the context of an ERLA, debugging skills are also at stake. Consequently, the three CT components [2] are likely to be mobilised during an ERLA: computational perspectives, computational concepts, computational practices. Of course, certain conditions favour the emergence of CT, such as the teacher temporarily blocking access to programming [5] or delayed feedback for the student on the execution of the program by the robot [4]. These two interventions refer to the metacognitive pause [15] and, besides, some researchers associate the CT skill with metacognitive skills [20]. However, all this knowledge about ERLA from the state of the art is only valid in the CT 1.0 version [18], i.e. under the deductive problem-solving approach. This changes when the problem-solving approach is inductive (CT 2.0), i.e. based on data-driven approach to programming.

3 Shift towards AI robots

AI robots open up a new form of human-machine interaction and thus a new learning relationship in the classroom between student and robot. These robots incorporate a ML algorithm (supervised or reinforcement learning) which, on one hand, impacts on the problem solving approach (inductive, [18]) and, on the other hand, gives them more autonomy than traditional robots (section 2). Indeed, traditionally, in computational problem solving (CT 1.0), the problem is solved by the human and the solution of the problem is then executed by the machine [5] whereas in data-driven approach to programming (CT 2.0, fostered by generative AI [18]) both the problem solving and the solution are generated by the machine. The role of the student has therefore changed and he·she needs to understand how the machine learns, whether in supervised or reinforcement mode, by expanding the scope of his·her knowledge developed in traditional programming to that of ML[18].

Supervised learning is “applicable when we know what output is expected for a set of inputs” [1] (p. 214) whereas, in reinforcement learning,“we do not specify the exact output value in each situation; instead, we simply tell the network if the output it computes is good or not” (ibid.). Through this game of rewards (positive or negative), the robot is obliged to take account of its previous actions, i.e. it learns from delayed feedback, which makes the learning process iterative.

In the case of reinforcement learning, we can clearly see the transition to autonomy of the machine, which solves and executes the given problem itself. This calls into question the role of the student in the problem-solving task. At first sight, this seems to run counter to Papert’s constructivist approach [14] (section 2). Nevertheless, some authors [13] posit this as an opportunity to revisit the work of Papert and his colleagues, who have already highlighted the gain in metacognitive knowledge and skills for student (even young ones) in AI context. Based on the idea of “contrastive learning” [13] (p.22), these authors support the value of highlighting the “relationship between AI and education, not only looking at AI as an applied tool to advance education but also investigating its value as an analogy to human intelligence” (p.1).

In this regard, [12] recently showed that students aged 8 to 11, subjected to an ERLA embedding an AI algorithm (AI-ERLA), significantly improved their metacognitive knowledge, compared with those not subjected to the AI-ERLA. The educational value of AI robots is thus well-founded, but there are still differences in the way they are used. For instance, [12] proposes a teaching scenario in which students i) explore and manipulate the robot in traditional mode, but also ii) observe the AI robot in reinforcement learning mode. The switch from the traditional robot to the AI robot involves a change of task (from exploration/manipulation to observation), with the student being asked to focus more on metacognitive aspects. This change leads to different learning objectives.

4 Four learning scenarios with increasing robot autonomy

As mentioned in the introduction, we have identified 4 main types of scenarios that can be set using different types of robots, with or without embedded AI. Table 1 in Appendix summarizes these 4 scenarios and discusses their respective advantages. We focus our discussion in particular on how students engage in the proposed activities. Indeed, while the ideas of Papert's constructionism imply that the student is the actor or actress of his·her learning and concretely in action in his·her learning environment, the shift to the use of AI robots could run somewhat counter to this idea. In fact, with traditional use of educational robots (columns 2 and 3 in Table 1), the autonomy of the machine is low and the power of the student to program and therefore to reflect and act (computational perspectives and practices) is high. But what happens when the machine becomes increasingly autonomous?

In the first scenario, students simply control robots remotely and learn to associate commands (e.g. buttons) with robot actions and to plan a sequence of actions. Student engagement is high as they are involved in every moment of the robot course, but at the same time this interaction remains trivial, so intellectual engagement may be low as students get older.

The second scenario corresponds to the mainstream use of educational robots, where students are engaged in programming the robot's response to its environment. Their engagement is high as they iteratively i) plan how the robot should behave, code, then ii) observe the robot behaving according to its code and correct the code (debugging, etc.). Among these coding activities, we can distinguish those that are simpler for the younger ones, do not involve sensors, and where students only program an open-loop sequence of actions.

The third scenario is an extension of the second, but involving AI, or more precisely "supervised learning". Here, students no longer program their robots with code, but train them with data thanks to an ML model for a classification task: in the case of image recognition, for example, they need to provide relevant example images associated with given categories in order to train an image recognition model. In the case of an AlphAI robot, students collect data by remotely controlling their robot, and each image from the robot's camera is directly associated with an action that the robot should perform, so that the robot can be switched to autonomous driving as soon as this training is complete: using such an AI robot can be simpler than traditional robot programming and therefore more accessible to young students. In fact, the underlying task is different because the approach to solving the problem changes from deductive to inductive. In this context, data collection leads to an understanding of what is relevant data. For instance, in the case of platforms such as machinelearningforkids.com, students integrate their trained model to a Scratch code, therefore they manage to program their robot using both code (scenario #2) and data (scenario #3). Altogether, despite the high degree of autonomy available to the AI robot in scenario #3 and the AI model's ability to learn and memorize, the student's power to act remains high (similar to scenario #2, i.e. action of iterative debugging). It is noteworthy also that such activities are more accessible to young students, and that they allow the programming more "exciting" robot behaviours, such as circuit racing.

In the fourth scenario, the robot learns 'on its own' by trial and error. Its goal during learning is to maximise a score or 'reward' that it receives after each action it performs. The activities become more subtle because they include waiting periods during which the robot learns by accumulating experience. The power of the student to act seems to be reduced, since it is the AI robot that solves the problem on its own. Nevertheless, it is possible for the student to manipulate: he·she can help the robot to learn faster by forcing it to make certain explorations [12]; he·she can also manipulate the reward given to the robot to adjust the learning goals, as is the case in the study by Zhang and colleagues [21]. The task proposed in this study is designed according to constructionist principles, and the results of the study show a high level of student engagement and learning, similar to what was described in scenarios #2, #3 and #4 in Table 1 (understanding AI and CT concepts). Moreover, the students' engagement may also lie in the transition from manipulation to critical observation, as the analysis of how the robot learns is useful for learning about one's own learning: in scenario #4, the students observe that the robot, like humans, needs time, curiosity, acceptance of mistakes and thus perseverance to learn successfully. This use follows the recommendation of Ojeda and colleagues [13] to use reinforcement learning as a tool for metacognitive reflection, allowing children to "reflect on their own thinking and learning".

5 Conclusion, Limitations, and Perspectives

The purpose of the present paper was to explore the added value of educational robotics and AI in the learning process, with a particular focus on the development of metacognition in primary school students.

Based on the state of the art and on practitioners'feedback, we built and proposed a synoptic table regarding 4 different scenarios with robots and AI. It shows the added value of AI robots compared with traditional robots in primary schools. It emerges that the 4 scenarios identified with robots do not all offer the same level of i) robot autonomy in solving the problem, ii) type of student's engagement in learning, iii) targeted educational objective, iv) teacher intervention during learning.

In this paper, continuing Tedre and colleagues's pedagogical concerns about CT versioning [18], we argue that all different types of robots represent an opportunity for learning. The challenge, therefore, is to train teachers in this area. In the previous sections, we recalled the recognized benefits of the tangible aspect of robots for learning and showed how data visualization, a neural network, and the *in situ* behavior of an AI robot also contribute to student learning. Consistent with the work of Ojeda and colleagues [13], this analysis highlights the need to use AI robots not as a replacement for traditional robots, but as a complement to them. Constructed learning is simply not the same and depends more on the scenarios, the nature of the task, and how the tool is mediated by the teacher than on the level of autonomy of the robot. This shift in students' agency needs to be understood by teachers so that they can make good use of each of these tools according to their pedagogical objectives.

References

1. Ben-Ari, M., Mondada, F.: Elements of robotics. Springer Nature (2017)
2. Brennan, K., Resnick, M.: New frameworks for studying and assessing the development of computational thinking. In: Proceedings of the 2012 annual meeting of the American educational research association, Vancouver, Canada. vol. 1, p. 25 (2012)
3. Catlin, D., Woollard, J.: Educational robots and computational thinking. In: Proceedings of 4th International workshop teaching robotics, teaching with robotics & 5th International conference robotics in education. pp. 144–151 (2014)
4. Chevalier, M., Giang, C., El-Hamamsy, L., Bonnet, E., Papaspyros, V., Pellet, J.P., Audrin, C., Romero, M., Baumberger, B., Mondada, F.: The role of feedback and guidance as intervention methods to foster computational thinking in educational robotics learning activities for primary school. Computers & Education **180**, 104431 (2022)
5. Chevalier, M., Giang, C., Piatti, A., Mondada, F.: Fostering computational thinking through educational robotics: A model for creative computational problem solving. International Journal of STEM Education **7**(1), 1–18 (2020)
6. Eguchi, A.: Ai-powered educational robotics as a learning tool to promote artificial intelligence and computer science education. In: International Conference on Robotics in Education (RiE). pp. 279–287. Springer (2022)
7. Greff, É.: Le jeu de l'enfant-robot: une démarche et une réflexion en vue du développement de la pensée algorithmique chez les très jeunes enfants. Ph.D. thesis, Paris VII (1996)
8. Havenga, M., Jordaan, T.: Introducing coding and robotics: Prospective mathematics teachers' metacognitive thinking in scratch. In: International Conference on Robotics in Education (RiE). pp. 17–26. Springer (2023)
9. Horn, M., Bers, M., et al.: Tangible computing. The Cambridge handbook of computing education research **1**, 663–678 (2019)
10. Hsu, T.C., Chang, S.C., Hung, Y.T.: How to learn and how to teach computational thinking: Suggestions based on a review of the literature. Computers & Education **126**, 296–310 (2018)

11. Magnenat, S., Shin, J., Riedo, F., Siegwart, R., Ben-Ari, M.: Teaching a core cs concept through robotics. In: Proceedings of the 2014 conference on Innovation & technology in computer science education. pp. 315–320 (2014)
12. Martin, M., Chevalier, M., Burton, S., Bonvin, G., Besançon, M., Deneux, T.: Effects of introducing a learning robot on the metacognitive knowledge of students aged 8–11. In: International Conference on Robotics in Education (RiE). pp. 169–183. Springer (2023)
13. Ojeda-Ramirez, S., Rismanchian, S., Doroudi, S.: Learning about ai to learn about learning: Artificial intelligence as a tool for metacognitive reflection (2023)
14. Papert, S.A.: Mindstorms: Children, computers, and powerful ideas. Basic books (1980)
15. Romainville, M.: Conscience, métacognition, apprentissage: le cas des compétences méthodologiques. In: La conscience: perspectives pédagogiques et psychologiques. Presses de l'université du Québec (2007)
16. Sapounidis, T., Demetriadis, S.: Tangible versus graphical user interfaces for robot programming: exploring cross-age children's preferences. Personal and ubiquitous computing **17**, 1775–1786 (2013)
17. Socratous, C., Ioannou, A.: Using educational robotics as tools for metacognition: an empirical study in elementary stem education. In: Immersive Learning Research Network Conference, UK. pp. 64–75 (2019)
18. Tedre, M., Denning, P., Toivonen, T.: Ct 2.0. In: Proceedings of the 21st Koli Calling International Conference on Computing Education Research. pp. 1–8 (2021)
19. Voogt, J., Roblin, N.P.: A comparative analysis of international frameworks for 21st century competences: Implications for national curriculum policies. Journal of curriculum studies **44**(3), 299–321 (2012)
20. Yadav, A., Ocak, C., Oliver, A.: Computational thinking and metacognition. TechTrends **66**(3), 405–411 (2022)
21. Zhang, Z., Lavigne, K., Church, W., Sinapov, J., Rogers, C.: Introducing reinforcement learning to k-12 students with robots and augmented reality. In: International Conference on Robotics in Education (RiE). pp. 351–365. Springer (2023)

Appendix

Table 1: Comparison of the 4 robotics scenarios (with and without AI).

Key elements of the scenario	**Scenario #1: Remote Control**	**Scenario #2: Programming**	**Scenario #3: Supervised Learning**	**Scenario #4: Reinforcement Learning**
Example of robots that can be used	Automaton robots (i.e. no sensor needed): Blue Bot, Thymio in Pink mode	Ground robot (i.e. with sensors): Thymio, mBot, Dash, …	Ground robot, connected to an AI software	Ground robot, connected to an AI software
Type of task and related approach according to [18]	Remote control the robot to move from point A to point B. **Deductive approach**	Program robot to: move from A to B, avoid obstacles, solve a maze, line tracking, etc. **Deductive approach**	Train AI with data to the same tasks (avoid obstacles, etc.) but also tasks involving complex sensors (camera). **Inductive approach**	Set a goal and a reward system, then the AI learns "all by itself". **Inductive approach**
What does the student do?	He·She gives orders to the robot, he·she controls it (remotely).	He·She programs the robot according to the situations he·she anticipates. He·She analyzes bugs and adjusts the program, based on the CCPS model [5] for instance.	He·She identifies all possible cases and gives relevant examples to the robot. He·She analyzes bias and regulates with new examples.	He·She observes, analyzes and explains how the robot achieves its goals. *He·She engages in reflexive practices, comparing his/her learning process with that of the robot and drawing conclusions.*
Type of teacher's intervention (Teacher's mediation)	Help students manipulate the remote control.	Help students with the programming language and identification of all cases.	Help students identify all the cases and relevant examples needed.	Help the student 1) identify the strategies the robot uses to solve the problem, *2) draw parallels and differences between the way the robot learns and the way the student learns.*
level of complexity of the task for the student	Very low Interaction is step by step (low anticipation).	High The difficulty lies in the programming task, which requires anticipation and knowledge of the programming language (computational concepts)[1].	Medium No need to know the programming language. However debugging is made more difficult because it is impossible to discern why a neural network has generated an output [18].	High The difficulty lies in knowing how the robot learns and having enough distance (maturity) from one's own way of learning as a student. Putting things into words can be an obstacle for students.
What does the robot do?	It executes orders regardless of its environment (open loop).	It executes the program according to the situations encountered in its environment (closed loop). These situations have been defined.	It learns from labeled examples provided by the student. It analyzes these examples to understand the relationships between inputs and outputs, and then uses this knowledge to make decisions autonomously, including generalizing decisions to new, unlabeled data.	It learns by trial and error. It learns continuously "on its own" from its experiences, thanks to the reward system provided by humans.
Can we talk about AI? Why ?	No ...because the robot executes orders when asked to do so. It is not autonomous.	No ...because the robot executes a program according to the precise situations programmed. It may be autonomous, but it is not off the beaten track.	Yes ...because what is called AI today involves "machine learning": here, the robot learns from the examples it memorized.	Yes ...because the machine explores and fumbles around in order to implement a strategy to achieve the set goal. This also involves "machine learning".
Links with everyday objects used by students	Remote-controlled car.	Coffee machine, automatic car wash, domestic machines.	Industrial robots using computer vision. Self-driving cars.	A robot like Dreamer[2] that adapts the way it walks according to the state of the ground.
			AIs such as ChatGPT and other chatbots and audio assistants that have been trained using both supervised and reinforcement learning.	
Knowledge at stakes	Technical aspects of the machine	Technical aspects of the machine + CT 1.0 (including CS)	AI concepts, especially the importance of data in modern life + CT 2.0	AI concepts + MK + CT 2.0
Level of robot autonomy	No autonomy	Autonomy No learning	Autonomy Memory Learning	Autonomy Memory Learning (a more difficult one!) Exploration
Student's engagement	High Direct handling	High Direct handling + CT 1.0	High Direct handling + regulation + CT 2.0	High Active observation during the time robot learns, reflection, and metacognition + CT 2.0

1 An intermediate level of difficulty consists in programming only sequences of actions, without using any sensor.
2 See at <https://danijar.com/project/daydreamer/>.