Towards educational patterns for teaching computational thinking using robots[*]

Christian Glahn[1]

[1] *Zurich University of Applied Sciences, Schloss 4, 8820 Wädenswil, Switzerland*

Abstract

Computational thinking is considered as a key competence for the digital age. Computational thinking (CT) has been defined as “procedural alignment”, “problem structuring”, and “logical reasoning”. In school curricula these aspects of CT are often represented as primarily cognitive skills because it concerns operations that happen inside a computer. When introducing robots into education, we observe several approaches: the most common approach is the combination of mechanics and computers. This means that the full development of a robot and its control logic needs to be developed as part of a curriculum.

The present study tackles the educational complexity by introducing a multi-layered learning design patterns to identify design principles for playful CT education using robots. This study introduces a pattern-based approach for “educational programs” that allows to analyze and to compare different didactic approaches beyond subject matter principles and specific tools. This contribution applies these design patterns to these programs in order to address these challenges and allows to connect and compare well under-stood educational design concepts to teaching robotics and CT.

Keywords

Computational Thinking, Robots, Educational Design, Adaptivity, Inclusion

1. Introduction

Computational Thinking is considered as a key competence for the digital age. Computational thinking (CT) has been defined as “procedural alignment”, “problem structuring”, and “logical reasoning” [1]. In school curricula these aspects of CT are often represented as primarily cognitive skills because it concerns operations that happen inside a computer. The underlying concept of the related approaches is the rule-set that is represented by a special language. In early education these languages include Logo or Scratch that are later replaced in advanced curricula by languages such as Python, Java, or JavaScript. For any of these languages, computers and their rules are at the center of the interaction. This creates conceptual tensions with modern digital technologies such as AR/XR, smart devices incl. Internet of Things-devices (IoT), smart toys, and robots. The common denominator of these technologies is a rich set of sensors and actuators that shape the possible interactions beyond a screen, mouse and keyboard. This creates new educational opportunities and challenges because these technologies allow CT to be embedded into physical experiences of children and adults.

When introducing robots into education, we observe several approaches among practitioners: the most common approach is the combination of mechanics and computers. This means that the full development of a robot and its control logic needs to be developed as part of a curriculum. Alternatively, and particularly in early education, robots are often used as tangible devices for other educational objectives than CT. A third approach are robots with tangible interfaces to represent logic elements or logic blocks. The robot is in those cases a black box that uses built-in interfaces for altering their logic. These interfaces are typically buttons of different shapes and

[*] *EduRobotX 2025: 3rd International Workshop of the EATEL SIG, September 16, 2025, Newcastle, UK, co-located with the 20th ECTEL2025.*

[1*] Corresponding author.

christian.glahn@zhaw.ch (C. Glahn)

0000-0002-2701-3579 (C. Glahn)

colors. Finally, there are solutions that emphasize on the electronics of a robot, although these solutions are rarely found for applications across different age groups. Between the different kinds of educational robotics, there are few solutions that cover a middle ground. These solutions include the Grove Zero framework [2] or the cubelets framework [3]. The different available tools and solutions create a complex technical landscape for educators to navigate because they need to consider and to incorporate quite different ideas and tools into their practices. As most educational robots can be integrated into different educational strategies, educational practitioners further need to understand the function of a robot within their interventions.

The present study tackles the educational complexity by introducing a multi-layered learning design patterns to identify design principles for playful CT education using robots. Based on this approach, this study targets the extensibility of such patterns towards inclusive interventions to acknowledge the diversity among learners, including cultural, social, cognitive, and physical diversity.

This contribution reports on a design study that addresses the common ground for teaching CT using robots. A design study is a solution-based approach to a real-world problem faced by domain experts that supports solving the problem, validate or compare designs, and to reflect about the lessons learned in order to refine guidelines to practice [4]. The domain experts in this study are teachers and educational designers. The present study introduces a pattern-based approach for "educational programs" (EPs) that allows to analyze and to compare different didactic approaches beyond subject matter principles and specific tools.

2. Problem Statement

This paper is one outcome of the CoTEDI project [5]. CoTEDI stands for "Complutational Thinking Education for Diversity and Inclusion". The project targets the use of playful educational robotics for teaching computational thinking skills with a particular focus on inclusion and diversity of pupils in primary schools. The project collected several EPs for teaching CT concepts using different types of robots. These programs are fully worked out and tested lesson structures for different aspects of CT, subjects, and age groups.

An EP is a complete lesson plan with working materials that allows teachers and child-care-workers to include an educational approach into their practice. In order to create and share EPs, it is crucial not only to document educational concepts, but also to provide the means for comparing and extending them.

In the context of CoTEDI, this implies that practitioners need to understand and identify how a particular program supports inclusion and diversity while teaching CT skills. If a program does not support a practitioner's inclusion, diversity, or CT problem, they need means for adapting or adjusting a program to a situation at hand, or for identifying whether another program would provide a solution to the very problem.

While adaptability is particularly important for facilitating diversity and inclusion, it is also relevant for sharing robotic practices in schools. Educational practitioners have to understand whether or not a particular EP can be implemented with the resources available to them. As many robots support similar sensors and actuators, a program of interest may describe an approach using a system of one brand, but could be implemented with a quite different robot of another brand.

3. Design Patterns and Educational Design

Before we return to robotics, we need to understand the concept of a pattern. A pattern is an incomplete blueprint for solving a problem. A pattern provides key aspects of a solution that are known to work or are part of common practices and professional standards. A pattern is incomplete, so it can be adjusted to the particular needs of a situation or integrated with other patterns to create a complete solution. Through their partial solution, patterns simultaneously pre-structure the design space for a solution and provide recognizable anchors for a solution's verification and consistency.

All EPs defined in CoTEDI follow the environment-play-act model of educational design [6]. This model sets an educational approach in an environment that defines framing conditions of that approach, such as learning materials, tools, or objectives. An educational intervention is then defined as a sequence of activities. Such activity sequence is called a "play" within a lesson plan. Each activity is one episode of an EP. Typically, activities are tied to tasks for the pupils and their educators. This separation of environment, sequences and activities forms the core pattern for EPs. The environment-play-act model can be found at the core of frameworks such as the 4C/ID model [7] or technical specifications such as IMS Learning Design [8] or IMS Simple Sequencing [9].

The play defines the flow of an EP. A very simple flow would be a linear sequence of activities. However, activity-flows can become arbitrarily complex when considering ideas such as collaboration or personalization. A literal translation of such flows can be found in the business process model notation (BPMN) [10] that identifies three core aspects a such flows: the activities, internal and external events that trigger following activities, as well as branching for decisions, transitions, and integrations between activities. BPMN further provide means for highlighting task responsibility as well as resource usage. These five BPMN elements can be directly mapped to the concepts described by Dillenbourg's collaborative orchestration graphs [11]. On top of BPMN several flow patterns were identified and analyzed [12].

A different take on plays and activities is chosen by Engeström's activity theory model [13]. According to this model, plays are mere activities that can be recursively divided into smaller activities of the same structure. In this model, each activity consists of seven parts: actors (orig. subject), resources and tools (orig. instruments), targets and objectives (orig. object), rules, tasks (orig. division of labour), community as well as the activity outcomes. It has been argued that the community part can get generalized to the activity context to account for a wider area of practices [14, 15]. Engeström's model provides an analytical framework for the defining elements of an activity. It frames the design space for activities. By grouping actors and objectives into the interactions of an activity, the rules and tasks into its procedures and processes, and tools and context into the activity environment, the model can be directly connected to the environment-play-act model, mentioned above [15, 16]. In this model, flow patterns are represented at the procedures and process level within the activities. Therefore, such aspects are typically presented external to the activity when approaches such as BPMN are used to model the activity flow.

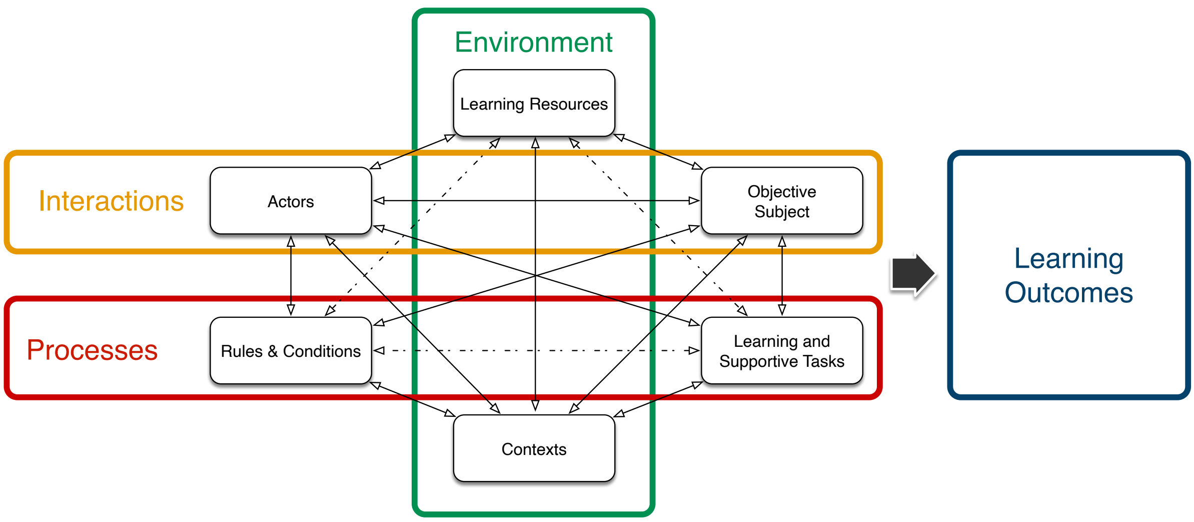

Figure 1: Alignment of Engeström's activity model to the environment-play-act model [16].

Earlier we have argued [17] that the activity theory model is to coarse for modeling the complexity of educational experiences. In order to capture the full educational design space that accounts for a wide range of educational and psychological theories and practices, each activity element requires further devision into sub-elements. Only at this finer level of detail important design elements of a learning experiences become visible and manageable.

The models presented so far define *structural patterns of educational activity*. Any complete EP must consider all structural parts to avoid misinterpretation and failures in execution. The problem with structural patterns in education is the complexity of the design space. Implementation patterns help to mitigate such complexity.

Implementation patters pre-structure the design space at activity level and provide sequences across activities. These patterns provide solutions for common problems or goals. Combined with structural patterns, these patterns preset parts of the design space and thereby remove these aspects from the design process. In education such patterns typically refer to common professional practices.

One example for such implementation patterns is the 8-learning-events model (8LEM) [18]. The model identifies eight common educational practices frequently implemented by teachers and educators. These events include "receive", "explore", "practice", "imitate", "debate", "experiment", "create", and "meta-learning". Each of these "learning events" reflects specific student and teacher motivations and responsibilities. Therefore, these events preset much of the process and the outcome aspects of the activity design structure. By understanding the role and agency of learners and educators in these settings, such patterns highlight the differences of the approaches of different EPs.

Similar implementation patterns exist on the level of the process flow, such as the 4C/ID model [7] or the jigsaw pattern [19]. 4C/ID model defines activity sequence patterns for systematic knowledge and competence development and acquisition, whereas jigsaw patterns has been used for demonstrating the use of the same pattern for formulating and implementing different collaborative learning experiences.

4. Design Method

The foundation of this design study is given by the EPs collected by the CoTEDI Project [5]. These EPs are provided as sequential text and cover typically several educational activities. Educational rationale are available only at macro-level of each EP. As the project aims at using robots for computational thinking as well as supporting diversity and inclusion, the kinds of patterns described above do not cover the specifics of these aspects of an EP. Therefore, it is unclear whether these aspects rely on independent design patterns. This requires revisiting concepts that are already used in computational thinking education, robotics, and inclusive education.

This study applies the design patterns to restructure two EPs to highlight the procedural educational rationale at activity level. Both EPs were classroom tested with different age groups. The two EPs were deliberately selected as they focus on two distinct and empirically well tested approaches: The first EP uses a constructive approach, while the other takes an explorative approach to teaching and learning CT with robots. The EPs were remodeled using the above activity patterns in order to identify the applicability as well as to understand the structural differences of the different educational approaches.

5. Patterns for CT and Robots

The CoTEDI project aims at supporting so called plugged and unplugged learning activities for approaching computational thinking education. This differentiation between plugged and unplugged activities are widely used concepts in the computational thinking domain. However, it is not entirely clear, what plugged and unplugged activities are when autonomous robots and games are used in educational experiences. Despite such ambiguity, the literature provides a rich body of worked examples that are a treasure trove for educational patterns.

When considering plugged and unplugged educational activities, we identified two reasoning approaches that use the same terms. One approach uses the terms plugged and unplugged activities in CT education to distinguish between direct and represented experiences of computational concepts [20]. The other approach ties the terms plugged and unplugged activities to the use of digital tools or non-digital tools to teach computational strategies [21]. Both approaches appear the same on first sight because they use the same terminology. The activity theory model helps to analyze the conceptual differences: direct and represented experiences relate to the learning objective or subject of an activity, while distinguishing between digital and non-digital tools are clearly associated to learning resources and tools.

In practice, the differences between the two approaches are subtle, because direct experience of CT concepts require digital tools, and the use of non-digital tools for teaching CT concepts automatically leads to represented or unplugged experiences. However, when introducing robots as tools to the educational process, the differentiation becomes important for two reasons. Firstly, a modern robot is a digital tool. Secondly, educational robots often represent computational concepts through their actions and behavior that can be influenced through the environment. This leads to the dilemma that the two reasoning schools now create contradicting definitions for the same educational activity. While the objective-centric approach [20] would consider this an unplugged activity, the tool-centric approach [21] suggests a plugged activity. This ambiguity is not helpful for educational robotics in CT education.

The duality of robotic applications in CT education can be resolved by treating the two approaches as distinct views on the same problem. This creates a space of four quadrants (Fig. 2). The first quadrant relates to the use of digital tools for direct CT experiences. All plugged activities fall into this quadrant. The second quadrant refers to the use of non digital tools for direct experiences of CT concepts. This typically refers to CT theory and conceptual abstraction. The third quadrant links non-digital tools to represented experiences of CT concepts. This is the realm of unplugged activities. The final quadrant uses digital tools for represented CT experiences. While this is one application of educational robotics, games, and smart devices, this quadrant is typically not covered by the CT-related literature. In absence of established terminology, I propose to call these activities "augmented experiences".

Within EPs, the type of experience facilitated by a tool may change between activities. Because such roles are a kind of implementation pattern, it can be helpful to emphasize this visually in the description of an EP. Examples of such visual pattern-representations are given in Fig. 2. The core metaphor is the "plug"-symbol coming from the prior labelling of these activities. These visual labels are enriched using "visual annotations". These annotations have been taken from visual cues commonly found in mobile apps. The checkmark stands for the need of a plug and, thus, plugged activities. The cross stands for unavailable plugs, which refers to unplugged activities. Finally, the "dash"-annotation stands for plug not needed to indicate augmented learning activities. This is expected to support practitioners to identify different tool purposes.

	Digital Tools	Non Digital Tools
Direct Experience	**Plugged**	Theory
Represented Experience	**Augmented**	**Unplugged**

Figure 2: Four quadrants of CT approaches with visual cues.

6. Towards Patterns for Inclusion and Diversity

The field of diversity and inclusion is a complex domain on its own that challenges practices in schools and educational facilities. This paper emphasizes on three preliminary aspects that contribute to differences in implementing EPs with this domain in mind. These aspects were identified in the discussion about and comparison of the CoTEDI EPs. Different to the CT-related patterns, inclusion and diversity patterns primarily affect the process and interaction dimension of learning activities.

The identified inclusion and diversity patterns fall into three distinct types along two primary dimensions. These dimensions are important for educators as they distinguish between different objectives in the class experience. The first dimension is the difference between inclusion and diversity. The second is the target for inclusion and diversity.

Differentiating between inclusion and diversity are two extremes of approaching differences among the student population: while inclusion focuses on commons at the cohort or group level, diversity addresses differences at the group or individual level.

The target of inclusion and diversity can be either inherent or external to the class. Again, both aspects at the target dimension are extremes of a continuum. The inherent target tackles commons and differences between students and pupils that are present and observable in a class, while external targets refer to aspects that are absent among the students.

External diversity and inclusion are typically included at the level of educational *objectives* or as part of an EP's *subject matter*. Robots and more recent AI technologies create a new option to address this aspect. As intelligent robots can act as an agent that takes those parts that are missing among the students. This option allows extensions of role-play scenarios that help children to comprehend diversity beyond the scope of their imagination. The danger at this level is avoid stereotypes within the robotic agent and potential discrimination by objectifying individuals.

The *inherent* extreme of diversity and inclusion typically take a different shape within EPs, because it considers those people who are present in a classroom. Under these conditions, inclusion and diversity handle differences at the procedural and interaction level. My initial observations suggest that diversity manifests differently in educational designs than inclusion.

Inherent diversity is found in programs that use adaptation or personalization to address the differences in students talents or challenges. An extreme form of dealing with inherent diversity would be personalized activities for each student.

Inherent inclusion can be found in collaboration where strengths and weaknesses within a group lead to a joint outcome. An extreme form of this approach would be a "not-leaving anybody behind" teaching strategy that treats all students equally and does not consider their individual differences.

Throughout the project we identified that structuring diversity and inclusion helps to compare the scope between and across EPs. However, inherent and external approaches to diversity *and* inclusion could not get harmonized into a unified framework at this state of our work. Therefore, the identification of design patterns is still at a preliminary stage. This is mainly due to the fact that approaches in the project's EPs vary greatly.

7. Application

Patterns serve as a tool in educational design to speed up the development of EPs and to enhance the understanding of the logic of learning experiences across learning activities. Moreover, patterns allow to reconstruct activity parts that are not explicitly stated in the description of a program. Example 1 (Appendix A) refers to the first three learning activities of an EP on developing an understanding how motors drive and steer a mobile robot [22].

The textual description in Example 1 contains some instructions about the motivation of an activity. However, the activity descriptions do not include direct instructions for the educators nor for the students. It also lacks a description on whether the activities are plugged, unplugged, augmented, or theoretical. This makes it hard to understand the logic of this example sequence and requires educators to translate the EP into actual classroom activities with task descriptions and supportive material. By applying educational patterns, it is possible to reconstruct the actual classroom scenario underpinning the EP (Fig. 3). Fig. 3 uses the activity patterns of 8LEM in a BPMN-like process notation.

ACT-OUNL2-0001 is an instruction activity that explains key functions of programming a robot's motor using an integrated development environment (IDE) on a computer. Despite the wording, this activity is unlikely a student led exploration rather than a teacher centric introduction to the systems used in the experience. Therefore, the activity is a receiving activity of the 8LEM pattern with a teacher's motivation "here is something you should know" and the students' motivation "tell me something new". The activity sets the robot and its IDE as the subject of the activity. This sets the this activity in the first quadrant of our CT education approaches. It can be considered as a plugged activity.

ACT-OUNL2-0002 describes the development of a program that is then run with different parameters on the robot. While the description describes this activity as one activity, the classroom experience will cover four activities – one per program. Again the activity description includes neither student nor educator instructions. The activity wording suggests an imitation activity under the 8LEM. This sets the teacher core motivation to "look how this is done" and the student motivation to "let me repeat your example". Again, the robotic concepts are directly concerned, so the activities are plugged.

ACT-OUNL2-0003 repeats the ideas of ACT-OUNL2-0002 using different programming strategies. However, these strategies are implicit here. Under the pattern approach it would be required to make each strategy explicit, so educators understand each step of the classroom experience. For the sake of this example, this activity will be the same as the previous activities.

Fig. 3 shows a pattern-based reconstruction of the EP of Example 1 using the visual syntax as used by BPMN [10].

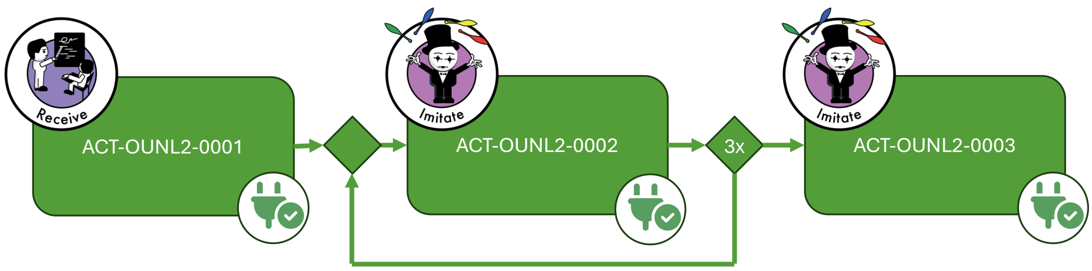

Figure 3: Illustration of a pattern-based flow of Example 1 in BPMN-like notation.

Example 2 (Appendix B) covers a full EP for understanding conditional branching using a modular educational robot [23]. The textual description of Example 2 includes the description of the learning activities as well as some remarks regarding supportive activities by an educator. The classroom activities consists of three activities. These activities are explicit in this program.

ACT-ZHAW-0001 describes the setup of the learning environment and the task to understand the tools available. The subject of the exercise is conditional branching. The educational robot is used in this exercise to represent the control logic that implements the branching. The colors of the pens represent four conditions to branch: black follow the line, blue turn left, green turn right, red stop. As the robot and colors represent CT concepts, the activity is an augmented learning activity. The students need to find out, how the colors influence the robot's behavior by exploring the tools in their learning environment. The 8LEM pattern is explore and sets the teacher's motivation to: "Here are some resources", and the students' motivation to: "let me find out". The teacher's role in this activity is to avoid deadlocks and time keeping.

ACT-ZHAW-0002 asks the students to create a race track for their robot. The task description is open but sets some requirements for the final solution. This activity follows the create pattern with the teachers motivation of “I am at your disposal” and the students' motivation of “Let me construct, produce, invent, and create”. Like in the previous activity, the robot represents the related CT concepts. Therefore, this activity is also an augmented activity. Again the teacher's role is to orchestrate the learning activity and guide the students if their phantasy leads them off track.

ACT-ZHAW-0003 brings the results of the entire class together and each group presents and demonstrates their solution. As most of the class is watching these presentations, this activity follows a receive pattern of the 8LEM. The key difference to ACT-OUNL2-0001 is that the teacher is not presenting. Therefore, the motivation of the presenting students is “here is something you should know” and for the rest of the class to “tell me something”.

Fig. 4 shows the activity flow using the 8LEM activity patterns in BPMN-like notation.

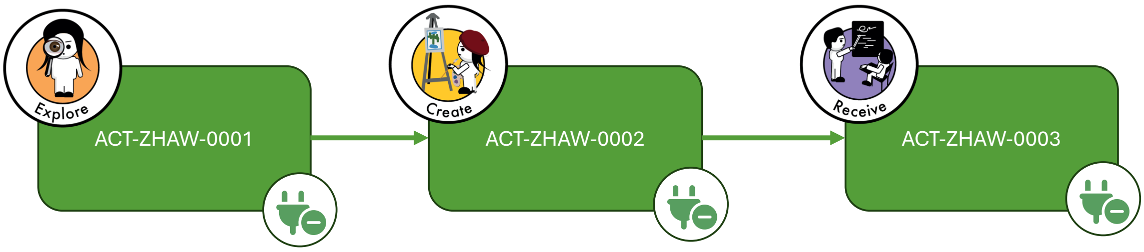

Figure 4: Illustration of the pattern-based flow of Example 2 in BPMN-like notation.

From the mere textual description the didactical differences between the two EPs are hard to spot. By using pattern-based visualizations these differences become more evident and provide already a rough orientation for the classroom experience. This approach also allows comparing different approaches on a higher conceptual level, while pre-structuring the detailed activity planning.

8. Conclusions and Limitations

This paper tackled the challenges of teaching CT using educational robots. By applying the concepts of educational patterns this paper highlighted prior work on structural and procedural patterns as well as implementation patters for classroom experiences. Particularly, structural patterns were developed primarily for analytical purposes. Educational designers can use these patterns as guidelines for creating complete and structured task descriptions, that are applicable for implementing educational programs. Using two examples I illustrated how educational and subject matter implementation patterns can be applied to structure complex learning experiences. Graphical notations can help to highlight arrangements and flows in the classroom.

This paper demonstrates the application of multi-layered patterns for describing and comparing educational approaches. It identified a significant gap in the patterns put forward in computational thinking education. This conceptual gap is created by the available educational technology, including robots and IoT-devices. This paper introduced “augmented learning activities” as a pattern annotation to bridge that gap.

This paper does neither include a full pattern library of educational robotics in CT education, analyze approaches to diversity and inclusion found in EPs, nor does it empirically evaluate teacher experiences when applying those patterns. Both aspects are open to further research. Instead, this paper addresses the application of patterns and the visual representation as a solution to design and implementation problems that emerge from conventional sequential descriptions of educational programs for communicating different educational approaches in educational robotics.

Acknowledgements

The work presented in this paper has been partially funded by Movetia Erasmus+ Funds under the project number 2023-1-NL01KA220-SCH-000152037. The work is related to the CoTEDI Project funded by the European Commission.

Declaration on Generative AI

The author has not employed any Generative AI tools in the context of this work.

References

[1] P.J. Denning & M. Tedre. “Compurational Thinking”. MIT Press, Cambridge, 2019.
[2] seeedstudio, “Grove Zero.” [Online]. Available: https://www.seeedstudio.com/Grove-Zero-Starter-Kit-V2-0-p-4352.html
[3] modrobtics, “Cubelets.” [Online]. Available: https://modrobotics.com/
[4] M. Sedlmair, M. Meyer and T. Munzner, "Design Study Methodology: Reflections from the Trenches and the Stacks," in IEEE Transactions on Visualization and Computer Graphics, vol. 18, no. 12, pp. 2431-2440, Dec. 2012, doi: 10.1109/TVCG.2012.213.
[5] CoTEDI Project. Project Website [Online]. 2025, https://cotedi.eu
[6] R. Koper, “An introduction to learning design”. In R. Koper & C. Tattersall, Learning design, a handbook on modelling and delivering networked education and training. Heidelberg: Springer, pp. 3-20.
[7] J.J.G. Van Merriënboer, R.E. Clark, M.B.M. de Croock: Blueprints for Complex Learning: The 4C/ID-Model. Educational Technology, Research and Development, 50(4), pp. 39-64. 2002.
[8] S. Grocott et al., Learning Design Information Model, 2003. [Online]. Available: https://www.imsglobal.org/learningdesign/ldv1p0/imsld_infov1p0.html
[9] T. Anderson et al., Simple Sequencing Information and Behavior Model, 2003. [Online]. Available: https://www.imsglobal.org/simplesequencing/ssv1p0/imsss_infov1p0.html
[10] Object Management Group (OMG), “Business Process Model and Notation (BPMN), Version 2.0,” Spezifikation. [Online]. Available: https://www.omg.org/spec/BPMN/2.0.2/PDF
[11] P. Dillenbourg, Orchestration graphs: modeling scalable education, 1. ed. Lausanne: EPFL Press, 2015.
[12] N. Russell, W. M. P. Van Der Aalst, and A. H. M. Ter Hofstede, Workflow Patterns: The Definitive Guide. The MIT Press, 2016. doi: 10.7551/mitpress/8085.001.0001.
[13] Y. Engeström. Learning by expanding: an activity-theoretical approach to developmental research (Second Edition). New York: Cambridge University Press, 2015.
[14] M. Sharples, I. Arnedillos, M. Milrad, & G. Vavoula. Mobile Learning: Small Devices, Big Issues. In N. Balacheff, S. Ludvigsen, T. Jong, A. Lazonder, & S. Barnes (Eds.), Technology-enhanced Learning (pp. 233–249). Dordrecht: Springer Netherlands, 2009.
[15] C. Glahn, “Mobile Learning Operating Systems,” in M. Ally and A. Tsinakos (eds) Increasing Access through Mobile Learning, Vancouver: Commonwealth of Learning, 2014, pp. 142–157. [Online]. Available: https://oro.open.ac.uk/44393/1/pub_Mobile%20Learning_web.pdf
[16] C. Glahn & M.R. Gruber Designing for Context-Aware and Contextualized Learning. In: S. Yu, M. Ally, A. Tsinakos (eds) Emerging Technologies and Pedagogies in the Curriculum. Bridging Human and Machine: Future Education with Intelligence. Springer, Singapore, 2020. https://doi.org/10.1007/978-981-15-0618-5_2
[17] C. Glahn and M. R. Gruber, “Mobile Blended Learning,” in Handbuch Mobile Learning, C. de Witt and C. Gloerfeld, Eds., Wiesbaden: Springer Fachmedien Wiesbaden, 2018, pp. 303–320. doi: 10.1007/978-3-658-19123-8_16.
[18] D. Leclercq & M. Poumay The 8 Learning Events Model and its Principles. Liege, Blegium: University of Liege, 2005. http://www.labset.net/media/prod/8LEM.pdf

[19] E. Vives et al., “Learning With Jigsaw: A Systematic Review Gathering All the Pieces of the Puzzle More Than 40 Years Later,” Review of Educational Research, vol. 95, no. 3, pp. 339–384, June 2025, doi: 10.3102/00346543241230064.
[20] F. KaleliOğlu, Y. Gülbahar, and V. Kukul, “A Framework for Computational Thinking Based on a Systematic Research Review,” 2016.
[21] C. P. Brackmann, M. Román-González, G. Robles, J. Moreno-León, A. Casali, and D. Barone, “Development of Computational Thinking Skills through Unplugged Activities in Primary School,” in Proceedings of the 12th Workshop on Primary and Secondary Computing Education, Nijmegen Netherlands: ACM, Nov. 2017, pp. 65–72. doi: 10.1145/3137065.3137069.
[22] N. Fanchamps, “Computational Thinking by SRA-programming with Lego EV3 or Spike robots”, Educational Program, OUNL; CoTEDI Project, 2025.
[23] C. Glahn & D. Bajkam, “Follow the line for modular robots“, Educational Program, ZHAW, CoTEDI Project, 2025.

A. Example 1: Start of an educational program to learn about robotic motor functions

This example is based on the works of [21].

ACT-OUNL2-0001: Basic Instruction. Exploration of the hardware components of the robot concerning motors, sensors, PLC control unit, cables, inputs and outputs. Exploration of the software and drag & drop computer programme, how to create a new programme, what is the work screen, exploration of button bars, what functions. Review of a sample computer programme and creation of a first computer programme.
ACT-OUNL2-0002: Making and a first programme. We make a first programme in which the robot drives forward for 4 seconds at the power of 50. The programme is loaded to the robot

- Making a second programme. We make a second programme in which the robot drives backwards for 8 seconds at a power of 20. The programme is loaded to the robot.
- Making a third programme. We make a third programme in which the robot first drives forward for 4 seconds, makes a 90-degree turn to the right, then drives forward for 2 seconds and then stops.
- Making a fourth programme. We make a fourth programme in which the robot first drives backwards for 3 seconds, makes a 90-degree turn to the left, then drives forwards for 3 seconds and then stops. After the robot stops, it plays a sound for 5 sec.

ACT-OUNL2-0003: Repeating steps. A programme that makes the robot do the same thing in different ways. The challenge is to make a programme as smart as possible.

B. Example 2: Conditional branching using an educational robot

This example is based on the works of [22].

ACT-ZHAW-0001: Functional understanding: Understand the sensor and motor functions of the robot. The students receive a robot kit, four colored pens (Black, Red, Blue, and Green), and a large sheet of paper. Students need to find the line following module and attach it to the basic robot. In groups, the students should figure out the meaning of the colors for controlling the robot’s movement. The teacher asks guiding questions to help the students to identify a testing strategy. The teacher should positively reinforce the students’ insights.

ACT-ZHAW-0002: Create a robot racetrack: In groups, the students create a robot race track using their insights from the first activity. The race track should have at least two crossings and one finish. The students should create the racetrack in a way, that the robot reliably gets to the finish.

Pitfalls:

If lines and color spots are too thin or too small, the robot will ignore them. The teacher needs to tell the students to make the lines big enough, but not too big.

Some students create branches using black lines only. The teacher should ask about the robot's behavior.

ACT-ZHAW-0003: Show and tell: Each group shows its race track and demonstrates a robot run using their robot. The group needs to explain the class the logic of their race track. The teacher asks questions to foster reflection. All groups get an applause.