education sciences

MDPI

Perspective

Computational Thinking and Coding for Young Children: A Hybrid Approach to Link Unplugged and Plugged Activities

Daisuke Akiba

Queens College & The Graduate Center, The City University of New York, New York, NY 11367, USA; dakiba@qc.cuny.edu

Abstract: In our increasingly technology-dependent society, the importance of promoting digital literacy (e.g., computational thinking, coding, and programming) has become a critical focus in the field of childhood education. While young children these days are routinely and extensively exposed to digital devices and tools, the efficacy of the methods for fostering digital skills in the early childhood classroom has not always been closely considered. This is particularly true in settings where early childhood educators are not digital experts. Currently, most of the efforts in standard early childhood settings, taught by teachers who are not digital experts, appear to revolve around "unplugged" activities that do not directly involve digital tools or devices, and it is not entirely clear how well these "unplugged" lessons promote the corresponding skills in "plugged" settings, such as coding and programming. This article discusses how, through further research, we may be able to devise an effective method for seamlessly building digital literacy among young children, transcending the "unplugged vs. plugged" barriers effortlessly.

Keywords: coding; computational thinking; computing education; digital education; digital literacy; early childhood education; programming; technology education

check for updates

Citation: Akiba, D. Computational Thinking and Coding for Young Children: A Hybrid Approach to Link Unplugged and Plugged Activities. *Educ. Sci.* **2022**, *12*, 793. https://doi.org/10.3390/educsci12110793

Academic Editors: Brian M. McSkimming and João Piedade

Received: 6 October 2022
Accepted: 2 November 2022
Published: 8 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

Computational thinking (CT) and coding have frequently been cited among the critical 21st century competencies that all children are encouraged to acquire and polish in order to flourish in our increasingly technology-dependent society [1,2]. Wing [3] has defined CT as " . . . thought processes involved in formulating a problem and expressing its solution(s) in such a way that a human or machine can effectively carry out". CT places a combined emphasis on problem solving and on the articulation of solutions that others, humans or non-humans, can replicate. Closely related, the concept of coding refers to the processes of " . . . creating the step-by-step instructions a computer understands and needs in order for its programs to work" [4]. This definition of coding incidentally coincides considerably with the definition of algorithm design, which is among the cornerstones of CT. Thus, it is perhaps unsurprising that scholars often consider CT and coding to be intricately intertwined with each other, and growth in one should, theoretically, facilitate growth in the other [5,6].

Furthermore, given the prevalence of the increasingly central roles that digital tools play in our lives, many have argued that the early introduction to activities pertaining to digital literacy, such as CT and coding, should be encouraged, especially in early childhood education settings, in order to ensure that young children learn and develop in contexts in which digital literacy skills are seamlessly integrated into children's routine activities and discoveries [7–9]. This idea, according to Bers [10], has its roots in the constructionist framework, introduced over 40 years ago by Seymour Papert [11], and a number of learning tools have been developed as a direct result of this perspective. Among these are ScratchJr (https://www.scratchjr.org, accessed on 5 October 2022) and TangibleK Robotics Program [10].

While these resources have been investigated and adopted relatively extensively with success in many contexts, there may be notable concerns that may constrain their infiltration into the standard early childhood classroom. For one, ScratchJr, a younger-children-friendly

derivative of Scratch3.0 (scratch.mit.edu, accessed on 5 October 2022), which is a visually driven programming tool, primarily targets elementary-age children, and its utility in early childhood settings may, therefore, be limited, especially if the teachers do not have sufficient digital literacy [12]. TangibleK [10,13] has successfully been implemented in early childhood classrooms over the last decade [14]; however, it requires teachers to have relatively sophisticated digital literacy—far more so than ScratchJr—and the lessons are quite extensive, making its incorporation into the ordinary early childhood education classroom, taught by regular educators, challenging. In addition, since the core activities have been designed around robotics, they might not necessarily have an inherent appeal to all children. These potential constraints may partially explain why resources such as ScratchJr and TangibleK, for example, have not exactly taken the world of early childhood digital education by storm, despite their highly desirable properties. Taken together, the hurdle may be too high for these potentially effective resources to be incorporated into everyday early childhood classrooms, taught by teachers that are not necessarily digital experts. It is, therefore, not surprising that most early childhood scholars and educators have been prompted to rely almost entirely on "unplugged" (or non-digital) methods for digital literacy integration into curricula [15–17].

Joohi Lee [8,18] has emerged as a leader in the unplugged digital literacy integration into early childhood classrooms taught by non-technology-expert teachers, and she and her team have proposed several unplugged activities designed to expose children to CT- and coding-related concepts and skills. For instance, in Lee's 2020 article [8], she describes how the use of directional words (e.g., turn left, go ahead, etc.) and their corresponding symbolic representations (e.g., ↰, ⬆) in daily conversations and instruction can strengthen young children's understanding and comfort with these directional concepts. In the same manuscript, she further proposes that using sequential words (e.g., first, second, third, etc.; today, tomorrow, the day after tomorrow, etc.) routinely would help young children to develop a sense of step-by-step progression, thereby building the foundational skills to engage in tasks involving CT and coding. When consistently built into daily classroom activities across different contexts, Lee and Junoh [18] posit that these practices will have synergetic effects on young children's learning, better preparing them for CT and coding. Other scholars and practitioners have also developed methods of integrating unplugged activities in early childhood education to promote CT and coding [19,20], although: (a) they typically seem to require the teachers to be adequately trained in digital literacy [21]; and (b) the effectiveness of these activities remains to be fully validated [22,23].

While these suggestions are compelling and appear to be developmentally appropriate in general, some challenges have been identified in implementing these practices to pertain directly to plugged settings. For one, early childhood educators frequently seem to lack the fluency to deliver, let alone design, activities to foster digital literacy in plugged contexts [21,24]. Additionally, there are no widely accepted assessment methods for CT- and coding-related skills for use in early childhood education settings, making it difficult to assess the outcomes of CT and coding integrations [25]. Another major concern has been raised by some researchers, such as Kristin Gunckel and her colleagues [16], that the successful transition from advances in unplugged activities to plugged skills remains unclear, especially in early childhood. In other words, with the attention in regular early childhood classrooms almost exclusively being around unplugged activities, there has not been notable evidence that unplugged lessons designed to promote CT- and coding-related skills among young children effectively translate into fluency in plugged contexts [1]. One potential method for making the unplugged-to-plugged transition may be to provide tools that would permit young children and their teachers to readily navigate between the unplugged and plugged aspects of given activities.

Specifically, Lee and her colleagues ([17], see Figure 1 on P. 200) proposed an unplugged activity, whereby young children and their teachers design a treasure map using directional arrows, based upon which they can engage in an indoor treasure hunting activity in the classroom. Young children frequently show inherent interest in treasure

hunts [17,26], which makes this activity ideal for children in this age range. Furthermore, the effectiveness of linking the manually constructed treasure map to the physical activity of treasure hunting is well corroborated by Seymour Papert's classic constructionist framework that digital literacy is enhanced when related activities are integrated into the learners' mundane learning, discovery, interaction, and exploration [11], which is a view that remains influential today [7,10]. To further support young children's learning of digital literacy, it would be pedagogically spectacular for them to be able to instantaneously witness, firsthand, how their unplugged activities (e.g., treasure maps and treasure hunts) may manifest into plugged contexts using digital tools (e.g., treasure maps being transcribed into codes, which may result in a game-like treasure hunt scenario that the children can experience in real time on a digital device). Unfortunately, resources that can be utilized to create such lessons (e.g., ScratchJr), as discussed earlier, appear to be too: (a) cognitively demanding for learners in their early childhood; and (b) challenging for ordinary early childhood teachers to implement [27]. This makes the proposed unplugged-plugged "hybridization" difficult to embed into the early childhood classroom, which is taught by educators that are not necessarily digital experts [12,16].

As argued so far, it is feasible to posit that unplugged-plugged hybrid activities that allow children to readily alternate between the unplugged and plugged aspects of a given set of tasks would represent an ideal method of promoting digital literacy in early childhood classrooms, primarily by removing the potential gaps between unplugged activities and plugged skills to which previous research has alluded [16]. In addition, in order for such activities to be widely, readily, and effectively implemented into early childhood education settings, they need to be sufficiently simple and intuitive so that teachers with non-expert-level technological inclinations can guide their pupils competently [24]. While no published research or publicly disseminated scholarly works appear to have directly addressed these points, as identified through citation databases, such as PsycINFO, ERIC, Academic Search Complete, and JSTOR, there is a promising product that may contribute greatly to our collective efforts to make the transition from unplugged to plugged activities more seamless for early childhood learners and teachers. This product may give us clues as to how best to incorporate unplugged–plugged hybrid activities into the everyday early childhood classroom in a pedagogically sound manner, which even non-digital-expert teachers can facilitate.

Pocky is a snack food, originally introduced in Japan in 1966 by Ezaki Glico Co., Ltd. (Osaka, Japan), which has been available primarily in the Northern Hemisphere, including North America, Asia, and Europe (in Europe, it is marketed as "Mikado" by Mondelēz International). A Pocky is a thin stick biscuit, approximately 13 cm in length, mostly covered in chocolate, with the exception of about 2.5 cm on one end, designed to allow consumers to hold the stick without staining their hands (see Figure 1). Given the small portion of each stick, it is popular among young children and young adults alike (https://pocky.glico.com/global/about/history.html (accessed on 5 October 2022)).



Figure 1. Pocky sticks.

In 2016, Glico released a free smartphone application called Glicode (https://cp.glico.com/glicode/en, accessed on 5 October 2022), targeting young children. This application takes children through the process of strategically placing Pocky sticks to create "unplugged codes," which could be captured by smartphone cameras to generate Scratch-like commands (see Figure 2) to control the movements of a bird-like figure called Hug Hug, whose primary mission focuses on giving a virtual child a hug (see Figure 3). In other words, the sequential placements of Pocky sticks (i.e., unplugged activity), when captured by smartphones, get translated into clearly displayed algorithms instantaneously, which subsequently control how the characters move in a plugged, game-like environment. Thus, children can directly and instantaneously witness how algorithms are generated based on unplugged information, what transcribed codes look like, how codes behave, and how they can manipulate algorithms through the sequential placement of Pocky sticks (i.e., back to the unplugged aspect of the same activity).

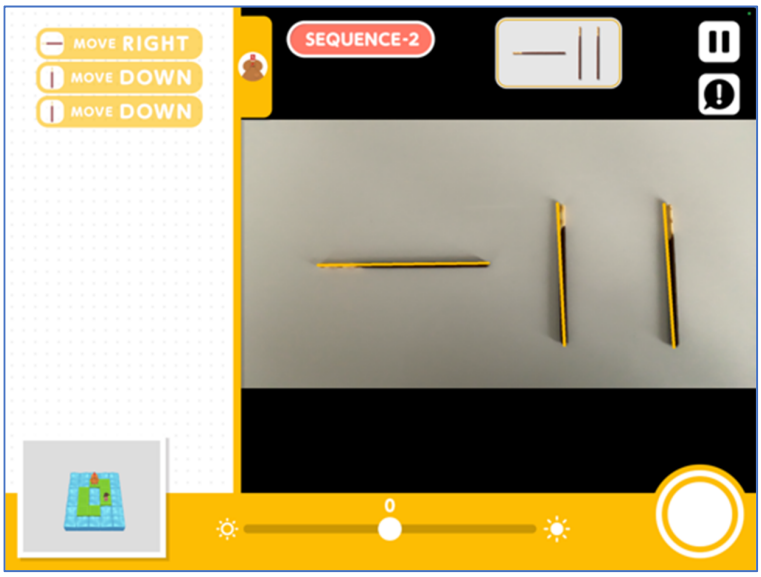

Figure 2. Screenshot of a real-time conversion of Pocky stick placement to Scratch-like commands through the Glicode application, clearly revealing the commands created with the sticks.



Figure 3. Screenshot of the Hug Hug character, who, as a result of the commands initiated through the strategic placement of Pocky sticks, approaches the virtual child to give her a hug.

When errors are made, Glicode would permit an unlimited number of adjustments to be made to the Pocky placements. These steps correspond directly to the cornerstones of CT and coding, including debugging [2]. Upon successful completion of each sequence, Glicode presents a new scenario for children to explore while guiding Hug Hug's movements through planning and maneuvering the placements of Pocky sticks for a total of 24 increasingly challenging scenarios.

In Figures 1 and 2, for example, the Pocky stick on the left signals "move right by one square," and the ones in the middle and on the right correspond to "move down by one square." In addition to directional commands, placing the sticks at certain angles can signal "begin loop" and "end loop" commands, and combining the direction of the sticks with certain tapping actions represents an "if–then" command. Although the range of the functions may seem limited, Glicode contains 24 incremental steps, ranging from the simplest to the most complicated, and children would be able to tackle these challenges on their own, with peers, or under adult supervision. Essentially, Glicode offers a user-friendly entry point for children and adults alike to experience ScrachJr- or Scratch-like coding via a seamless analog-to-digital conversion process that is visible to the users.

While presumably unavailable outside of the Japanese market, three other food products by Ezaki Glico could be used with the Glicode application. A product called *Bisco* can make Hug Hug jump in a variety of ways, depending on the placement, and another product, Almond Peak, functions to alter the size of the Hug Hug character. Finally, with Asobi Glico, children can create an array of random and irregular moves, such as backflipping, spinning, hiding, rotating, and turning objects upside down (https://www.glico.com/jp/newscenter/pressrelease/12705, accessed on 5 October 2022). Although this is Glicode's seventh year since its release, there appear to be no scholarly works examining the efficacy of Glicode. However, our research team's initial attempts at having a number of children, aged 4 to 5, as well as early childhood educators that are not digital experts, experiment with Glicode have proved to be promising, as both constituents have shown great interest in the use of Glicode, with very little frustration expressed by the users. We plan to report the results once sufficient data have been obtained. With many early childhood and elementary teachers expressing discomfort with technology in general, not very many of them approach these topics with enthusiasm or self-efficacy [24,28]. A user-friendly tool such as Glicode may serve as an easy entry point to encourage them to gain comfort and self-efficacy around digital literacy instruction, infusing the unplugged and plugged aspects of the activities with relative ease. In summary, a tool of this nature would be exceedingly helpful because: it (a) not only features properties that optimize digital literacy education among young children; but also (b) provides educators who lack an advanced fluency in technology with an effective platform upon which to implement digital literacy lessons. Thus, such a tool should help enhance digital literacy among both children and educators.

Of course, the recommendation here is not necessarily to adopt Glicode in the early childhood classroom. For example, the use of food products, such as Pocky, may present a great distraction for some children, especially prior to developing sufficient impulse control, and it may also be unsanitary for classroom use. To this end, one could argue that more "classroom-friendly" inedible manipulatives could easily be created. In addition, expanding the scenario beyond Hug Hug and the virtual child to include more flexibility for customization would be desirable since such variations may contribute to efforts toward diversity and inclusion by allowing educators and children to create contents that may appeal to children from a wide range of demographic backgrounds and exceptionalities. Therefore, Glicode can serve as an important clue for education researchers and practitioners to develop tools, akin to Glicode, to promote CT- and coding-like activities that allow children to directly experience the transparent connection between unplugged and plugged activities simultaneously.

The use of such tools to embed digital literacy into children's learning would also be compelling because it directly aligns: (1) with the classic constructionist framework

proposed by Papert, discussed earlier [11]; as well as (2) with the challenges that are being suggested by the recent research on ensuring the unplugged–plugged continuity [7,9,16]. There may be other commercial products of this nature available that require no advanced knowledge or skills in digital literacy for teachers to incorporate them into early childhood education settings. However, our pursuit to identify such products has not been successful, as similar tools seem to decidedly require educators to have, at a minimum, a rudimentary understanding of coding.

The perspective that is proposed here also signals the importance of academia–industry collaborations in tackling challenging topics in research and practice in general. A promising tool such as Glicode appears to be virtually unknown to early childhood scholars and educators, and, in turn, corporations such as Ezaki Glico may be unaware of the profoundly dynamic ways in which their products could be used to contribute to scholarly conversations. Through synergetic collaborations between academia and the industry, we may be able to develop ideal tools and methods that even non-digital-expert educators can readily utilize to promote digital literacy education in the early childhood classroom.

Finally, and perhaps most importantly, the perspective presented in this manuscript may represent an important, yet realistic, step towards further promoting diversity, equity, and inclusion in education. This is because underfunded urban schools in the United States are typically unable to secure sufficient digital resources, including teaching staff with training in digital literacy education, which has resulted in children from socioeconomically disadvantaged backgrounds not being provided with the range of opportunities their wealthier counterparts are likely to enjoy [29]. Taken together, further research is warranted on the development and utilization of effective tools to support digital literacy education in standard early childhood classrooms with teachers who are not digital experts, with the ultimate goal of equipping all young children with the skills and aptitudes to become fully participating members of the digital society.

Funding: This work was supported by the Robin Hood Foundation (Grant Number A1N-5A000000IYl2QAA).

Institutional Review Board Statement: Ethical review and approval were waived for this perspectives manuscript because this work falls outside of the definition of human subject research, according to the Human Research Protection Program (HRPP) of The City University of New York.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The author would like to thank Line Saint-Hilaire, Michelle Fraboni, Christopher Wagner, and Grace Pai, all based at The City University of New York, for their input, support, and guidance throughout various phases of the manuscript completion process.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Fenty, N.S.; Pierce, A.; Schildwachter, J. Coding Is Lit: Integrating Coding and Literacy in Early Childhood Inclusive Settings. *Teach. Except. Child.* **2022**, *54*, 276–285. [CrossRef]
2. Wing, J.M. Computational Thinking, Ten Years Later. In *Microsoft Research Blog*; Microsoft Corporation: Redmond, WA, USA, 2016; Volume 2020, pp. 10–11. Available online: https://www.microsoft.com/en-us/research/blog/computational-thinking-10-years-later (accessed on 5 October 2022).
3. Wing, J.M. Computational Thinking Benefits Society. In *Social Issues in Computing 40th Anniversary Blog*; DiMarco, J., Ed.; University of Toronto: Toronto, ON, Canada, 2014; Volume 2020. Available online: http://socialissues.cs.toronto.edu/index.html%3Fp=279.html (accessed on 5 October 2022).
4. McLennan, D.P. Creating Coding Stories and Games. *Teach. Young Child.* **2017**, *10*, 18–21.
5. International Society for Technology in Education. Computational Thinking Competencies, Integrate CT across Disciplines, with All Students: CT Competencies for Educators. Available online: https://www.iste.org/standards/computational-thinking (accessed on 5 October 2022).
6. Sun, L.; Hu, L.; Zhou, D. Which way of design programming activities is more effective to promote K-12 students' computational thinking skills? A meta-analysis. *J. Comput. Assist. Learn.* **2021**, *37*, 1048–1062. [CrossRef]

7. Bers, M.U.; González-González, C.; Armas–Torres, M.B. Coding as a playground: Promoting positive learning experiences in childhood classrooms. *Comput. Educ.* **2019**, *138*, 130–145. [CrossRef]
8. Lee, J. Coding in early childhood. *Contemp. Issues Early Child.* **2020**, *21*, 266–269. [CrossRef]
9. Lee, J.; Joswick, C.; Pole, K. Classroom Play and Activities to Support Computational Thinking Development in Early Childhood. *Early Child. Educ. J.* **2022**, 1–12. [CrossRef]
10. Bers, M.U. The Tangiblek robotics program: Applied computational thinking for young children. *Early Child. Res. Pract.* **2010**, *12*, n2.
11. Papert, S. *Mindstorms: Children, Computers, and Powerful Ideas*; Basic Books: New York, NY, USA, 1980.
12. Chou, P.-N. Using ScratchJr to Foster Young Children's Computational Thinking Competence: A Case Study in a Third-Grade Computer Class. *J. Educ. Comput. Res.* **2020**, *58*, 570–595. [CrossRef]
13. Bers, M.U.; Flannery, L.; Kazakoff, E.R.; Sullivan, A. Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Comput. Educ.* **2014**, *72*, 145–157. [CrossRef]
14. Horn, M.S.; Crouser, R.J.; Bers, M.U. Tangible interaction and learning: The case for a hybrid approach. *Pers. Ubiquitous Comput.* **2012**, *16*, 379–389. [CrossRef]
15. Code.org. Courses by Code.org Grades K-5. Available online: https://code.org/student/elementary (accessed on 22 October 2022).
16. Gunckel, K.L.; Covitt, B.A.; Love, G.; Cooper-Wagoner, J.A.; Moreno, D. Unplugged to Plugged In. *Sci. Teach.* **2022**, *89*, 46–51.
17. Lee, J.; Joswick, C.; Pole, K.; Jocius, R. Algorithm design for young children. *Contemp. Issues Early Child.* **2022**, *23*, 198–202. [CrossRef]
18. Lee, J.; Junoh, J. Implementing Unplugged Coding Activities in Early Childhood Classrooms. *Early Child. Educ. J.* **2019**, *47*, 709–716. [CrossRef]
19. Lo, C. Designing Unplugged and Plugged Activities to Cultivate Computational Thinking: An Exploratory Study in Early Childhood Education. *Asia-Pac. Educ. Res.* **2020**, *29*, 55–66. [CrossRef]
20. Otterborn, A.; Schönborn, K.J.; Hultén, M. Investigating preschool educators' implementation of computer programming in their teaching practice. *Early Child. Educ. J.* **2019**, *48*, 253–262. [CrossRef]
21. Strawhacker, A.; Lee, M.; Bers, M. Teaching tools, teachers' rules: Exploring the impact of teaching styles on young children's programming knowledge in ScratchJr. *Int. J. Technol. Des. Educ.* **2018**, *28*, 347–376. [CrossRef]
22. Pugnali, A.; Sullivan, A.; Bers, M.U. The Impact of User Interface on Young Children's Computational Thinking. *J. Inf. Technol. Education. Innov. Pract.* **2017**, *16*, 171. [CrossRef]
23. Relkin, E.; Bers, M. TechCheck: Development and Validation of an Unplugged Assessment of Computational Thinking in Early Childhood Education. *J. Sci. Educ. Technol.* **2020**, *29*, 482–498. [CrossRef]
24. Avcı, C.; Deniz, M.N. Computational thinking: Early childhood teachers' and prospective teachers' preconceptions and self-efficacy. *Educ. Inf. Technol.* **2022**, *27*, 1–25. [CrossRef]
25. Unahalekhaka, A.; Bers, M.U. Evaluating young children's creative coding: Rubric development and testing for ScratchJr projects. *Educ. Inf. Technol.* **2022**, *27*, 6577–6597. [CrossRef]
26. Outdoor Treasure Hunt. *Scholast. Early Child. Today* **2004**, *18*, 8.
27. Haduong, P.; Brennan, K. Helping K-12 Teachers Get Unstuck with Scratch: The Design of an Online Professional Learning Experience. In Proceedings of the 2019 ACM Technical Symposium on Computer Science Education, Minneapolis, MN, USA, 27 February–2 March 2019; pp. 1095–1101. [CrossRef]
28. Thiel, O.; Lundheim, R.; Hanssen, S.M.; Moe, J.; Vaz Rebelo, P. Using self-made automata to teach STEM in early childhood teacher education. *J. Learn. Dev. High. Educ.* **2020**, *18*. [CrossRef]
29. Mouza, C.; Barrett-Greenly, T. Bridging the app gap: An examination of a professional development initiative on mobile learning in urban schools. *Comput. Educ.* **2015**, *88*, 1–14. [CrossRef]