

Computational Thinking into K-12 Classrooms: Experiences and Challenges from Professional Learning Experiences

Zhichun Liu, Zarina Gearty, Eleanor Richard,
Chandra Orrill, Shakhnoza Kayumova, Ramprasad Balasubramanian
zliu11@umassd.edu, zmadan@umassd.edu, erichard1@umassd.edu, chandra.orrill@umassd.edu,
skayumova@umassd.edu, r.bala@umassd.edu

Kaput Center for Research and Innovation in STEM Education, University of Massachusetts Dartmouth

Abstract: Teachers often find it challenging to learn computational thinking (CT) and integrate it with classroom learning. In this systematic review, we focus on how professional learning experiences have supported K-12 teachers to integrate CT into their classrooms. The findings suggest some effective strategies for building professional learning experiences but highlight the need for more agreement about the nature of CT.

Introduction

Computational thinking (CT), defined as problem solving processes, skills, and dispositions needed to develop software (e.g., ISTE & CSTA, 2011), has garnered national attention as important skills and processes required for all students. While CT is tied to CS, it can be developed as a problem-solving approach. Despite the importance of CT for 21st-century education, opportunities to engage in CT are often present only in informal learning settings such as after-school clubs and limited to coding skills (Sengupta et al., 2018). Teachers often find it challenging to learn about CT and integrate it with classroom learning (Ketelhut et al., 2020). To this end, we sought to understand what kinds of learning experiences teachers had related to CT. To this end, we conducted a systematic review of the literature synthesizing research on how CT-focused PLEs have supported K-12 teachers to integrate CT into their classrooms. Specifically, we pursued two research questions:

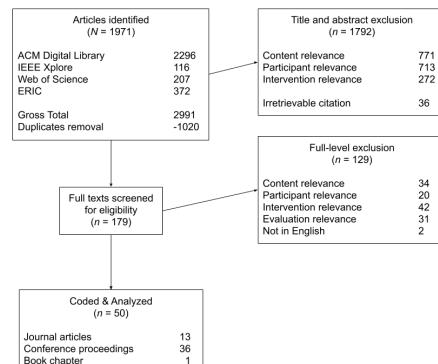
- What are the learning goals and conceptualizations of the existing CT PLEs?
- How did prior studies support teachers to integrate CT into classrooms?

Method

We conducted a systematic literature review based on PRISMA protocol of recent research on CT PLEs across four bibliographic databases: ERIC, Web of Science, IEEE Xplore, and ACM digital library. The search terms combined CT + professional-development-related terms or CT + outcome-related terms. In total, we screened 1,792 articles with pre-established eligibility criteria. After reviewing all articles at the title and abstract level, 179 were determined to have met our initial criteria. Three reviewers examined those articles at the full-text level. Full-text screening resulted in 50 articles that reported CT-related PLEs with K-12 teachers and 27 of which reported PLEs with component(s) on CT integration (see Figure 1 for the literature review process).

For each of the 50 articles, we recorded basic study and participant information. Then, we coded the details of the PLE and the outcomes. All the articles were discussed and coded collaboratively by three reviewers. The consensus was established through iterative discussion, and discrepancies were resolved at group meetings.

Figure 1
The PRISMA Flowchart





RQ1: Goals and conceptualizations

Overall, the goals of the PLEs focused on teachers' understanding of CT-related concepts, tools, and pedagogies. The goals included advancement of knowledge, change of perception or awareness, and improvement of skills. We saw a wide range of conceptualizations of CT and CT integration. The most prevalent goals included CT principles (c.f., ISTE & CSTA, 2011), which were viewed as problem-solving processes involving skills such as pattern recognition, problem decomposition, algorithmic thinking, abstraction, etc. Other goals focused on specific knowledge and skills in Computer Science ($n = 5$), computing concepts or practices ($n = 4$), and robotics ($n = 2$). The wide spectrum of foci reflects CT education in the K-12 world. On the one hand, CT can be viewed as a development of higher-order thinking skills that support problem solving (e.g., ISTE & CSTA, 2011). On the other hand, CT can be operationalized as specific content knowledge associated with different computing tools.

RQ2: Supporting CT integration

We identified two themes in terms of supporting teachers to integrate CT into teaching practices. First, certain supports were reported to be effective. Studies that involve prolonged engagement with teachers often report positive reactions and substantial change at a behavioral level (e.g., Hestness et al., 2018; Ketelhut et al., 2020). Additionally, personalized mentoring has been reported to support teachers' professional growth. Such support can take various forms (e.g., coach in Hamner et al., 2016). Aside from the technical supports, teachers also need pedagogical supports to actively integrate CT into their classrooms (Dong et al., 2019). CT integration has often resulted from supports from a community of practice (e.g., Hestness et al., 2018). Second, we identified the specific experiences, provided through the supports, that are beneficial to teachers. For example, many studies addressed the role of feedback and reflection in PLE in supporting teachers' CT integration (e.g., Ouyang et al., 2018). Teachers need to develop conceptual understanding and have opportunities for practice to support CT integration. Thus, teachers need to actively plan and teach in CT-integrated classes.

Discussion and Conclusion

Overall, CT integration can be challenging because supporting teachers in changing their attitude, knowledge, beliefs, and practices take time (Ketelhut et al., 2020). Although CT is closely related to coding, we need to incorporate broader theoretical perspectives. Further, more research is needed on computational literacy in K-12 contexts (Kafai & Proctor, 2021). While the field is beginning to understand how to create PLEs that support teachers in teaching certain aspects of CT, more work is needed to understand what should be taught in K-12 CT and how best to support teachers in that endeavor.

References (use Heading 1)

- Dong, Y., Catete, V., Jocius, R., Lytle, N., Barnes, T., Albert, J., Joshi, D., Robinson, R., & Andrews, A. (2019). PRADA: A practical model for integrating computational thinking in K-12 education. *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, 906–912. <https://doi.org/10.1145/3287324.3287431>
- Hestness, E., Jass Ketelhut, D., McGinnis, J. R., & Plane, J. (2018). Professional knowledge building within an elementary teacher professional development experience on computational thinking in science education. *Journal of Technology and Teacher Education*, 26(3), 411–435.
- ISTE., & CSTA. (2011). Operational definition of computational thinking for K-12 Education. <https://www.iste.org/explore/computational-thinking/computational-thinking-all>
- Kafai, Y. B., & Proctor, C. (2021). A revaluation of computational thinking in K–12 education: Moving toward computational literacies. *Educational Researcher*. <https://doi.org/10.3102/0013189X211057904>
- Ketelhut, D. J., Mills, K., Hestness, E., Cabrera, L., Plane, J., & McGinnis, J. R. (2020). Teacher change following a professional development experience in integrating computational thinking into elementary science. *Journal of Science Education and Technology*, 29(1), 174–188. <https://doi.org/10.1007/s10956-019-09798-4>
- Ouyang, Y., Hayden, K. L., & Remold, J. (2018). Introducing computational thinking through non-programming science activities. *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, 308–313. <https://doi.org/10.1145/3159450.3159520>
- Sengupta, P., Dicks, A., & Farris, A. (2018). Toward a phenomenology of computational thinking in STEM education. In M. Khine (Ed.), *Computational Thinking in the STEM Disciplines: Foundations and Research Highlights* (pp. 49–72). Springer International Publishing. https://doi.org/10.1007/978-3-319-93566-9_4