Developing Computational Thinking through Project-Based Airplane Design Activities

Dazhi Yang, Ph.D.
Educational Technology Department
Boise State University
Boise, ID USA
dazhiyang@boisestate.edu

Youngkyun Baek, Ph.D.
Educational Technology Department
Boise State University
Boise, ID USA
youngkyunbaek@boisestate.edu

Steve Swanson, Ph.D.
Distinguished Educator in Residence
Boise State University
Boise, ID USA
steveswanson@boisestate.edu

***Abstract*—This study investigated the outcome of project-based, airplane design activities on promoting computational thinking (CT) in sixth grade students in the context of an integrated STEM learning environment. A curriculum unit of airplane design activities was implemented in a sixth grade classroom over 10 days. The students' CT skills measured by the Bebras Challenges were significantly improved after their completion of the airplane design curriculum unit.**

Keywords—computational thinking, integrated STEM, computing education, engineering design, project-based learning

I. Introduction

Computational thinking (CT) is a fundamental skill that involves problem formulation, problem-solving and scientific reasoning [1]. The integration of CT in elementary curriculum has the potential to improve student learning of subject content and problem solving [2]. This study, a sub-study of a large research project, focused on CT in sixth graders within the context of an integrated STEM learning environment. Specifically, this study aimed to investigate the outcome of project-based, airplane design activities on sixth graders' CT skills after they had participated in a curriculum unit centered on airplane design activities. The underlying rationale of this study was consistent with that of the current research on computing education, which is teaching and learning computing skills such as CT does not necessarily involve computers or coding [3].

II. Literature Review

Integrated STEM was defined as an integration to facilitate students working on complex tasks "that require students to use knowledge and skills from multiple disciplines" [4] (NAE & NRC, 2014, p. 52). This approach was intended to teach STEM in a connected manner with real world problem solving. Many studies have reported promising benefits of an integrated approach [2] [5]. In US elementary schools, teachers are responsible for multiple subject areas and an integrated curriculum has a practical value to teach and engage students in computing and computational thinking.

Integrated STEM learning has been beneficial not only for computing education but also for engineering design especially for pre-college students [6]. For example, an integrated STEM learning environment could facilitate in applying engineering design processes with young learners that helped children learn and practice CT skills in [7]. Hynes and colleagues' study also provided an example of what engineering thinking and CT would look like for young students while applying various disciplinary knowledge during design activities.

Project-based learning (PBL) engages students in constructing knowledge and learning skills through an extended period centered around solving real world problems [8]. In PBL, learning activities and objectives are driven by an overall guiding question with students showcasing their products often through a final competition. The PBL approach with hands-on activities allows students to investigate relevant problems, which is consistent with best practices (e.g., inquiry-based activities) for STEM learning [9]. For example, instead of teacher's lecturing about the relationship between music and mathematics, students can be guided to compose music to discover the connections for themselves. PBL is also one of the most adopted approaches to integrating CT in various content subject areas [10].

Yang and colleagues explored the practice of CT with upper level elementary students in a project-based, integrated STEM learning environment in an afterschool setting and pointed out that the practice of CT by students seemed to be closely related to specific learning tasks [6]. Their study also called for further investigation of such association. Therefore, this study focused on the investigation of CT in sixth grade students after they had participated in a unit of PBL guided airplane design activities in the context of an integrated STEM learning environment in a formal classroom setting, which required students to apply the subject knowledge of engineering, science, mathematics and technology as well as CT to solve a design challenge. The research question guiding this study was: Could integrated airplane design activities guided by PBL promote computational thinking in sixth grade students in a formal classroom setting?

III. Theoretical Framework

The authors chose eleven CT components as the foundation for this study based on relevant literature (e.g., [11] [12] [13]). The components were: a) CT vocabulary such as variables,

This study was funded by the National Science Foundation.

modeling, testing and debugging; b) Abstraction as sense making through reducing complexity and generalizing from specific instances; c) Algorithm as applying set of tools or sequence of steps to solve problems; d) Communication as descriptions supported by graphs, visualizations, and computational analysis; e) Conditional logic as using strategies to clarify problems and solutions; f) Data collection as gathering data to define or solve a problem; g) Data structures, analysis and representation as exploring data to find patterns, causes, trends, or results to facilitate problem solving; h) Decomposition as simplifying problems or specifying steps to solve problems; i) Heuristics as applying experience-based strategies that facilitates problem solving; j) Pattern recognition as recognizing repeated patterns; k) Simulation and modeling as manipulating data or concepts through controlled programs or exercises.

Subsequently, these CT components (see Table 1) were embedded in the airplane design activities by the research team that consisted of an interdisciplinary group of researchers from educational technology, engineering, and mathematics education. The airplane design activities further were aligned with the curriculum standards. The alignment between the design activities and the content standards of science, engineering and technology was determined by the teacher teaching the specific grade level.

TABLE I. CT EMBEDDED IN THE BRIDGE BUILDING PROJECT

CT Component	Description
CT vocabulary and terminology	Such as variables, data, modeling, testing and debugging, iterative [11] [13]
Abstraction	Reducing complexity and generalizing from specific instances to make sense of things. The abstraction process allows building complex designs and large systems [14] [1]
Algorithm	Applying specific set of tools or sequence of steps (processes) to solve problems [15] [16]
Communication	Written and oral descriptions supported by graphs, visualizations, and computational analysis [17]
Conditional logic	Using strategy such as an “if-then-else” construct to clarify problems and solutions [1]
Data collection	Gathering data to define or solve a problem [18] [19]
Data structures, analysis and representation	Exploring data to find patterns, causes, trends, or results to facilitate the knowledge construction and problem solving [19] [18]
Decomposition	Simplifying problems or specifying steps to solve problems [20]
Heuristics	Applying experience-based strategy that facilitates problem solving, such as "trial and error” [16]
Pattern recognition	Recognizing repeated patterns such as iteration or recursion [12] [19]
Simulation and modeling	Manipulating data or concepts through controlled programs or exercises or creating such programs for data manipulations [18]

IV. METHOD

The study participants consisted of 51 sixth graders from two classes at a suburban elementary school. The PBL guided curriculum unit consisted of eight airplane design lessons for 10 days, geared towards developing CT and problem-solving skills as well as student learning about how forces (drag, thrust, lift and gravity) work on an airplane. Table II lists the learning objectives in the form of guiding questions and specific student activities regarding the airplane design. The students worked in small groups of four to five students, and two teachers led and facilitated their own class. Both teachers had been trained in CT and the content of the airplane design activities, and had facilitated a longer version of the PBL curriculum unit in an eight-week afterschool program with small groups of 4th to 6th grade students the previous semester [6].

TABEL II. STUDENT AIRPLANE DESIGN ACTIVITIES

Date	Guiding Question	Activity
Day 1	1. How do airplanes fly? How do teams solve problems?/ What makes a good team?	• Team building • Introduce the Airplane Design Unit • Review CT components
Day 2	What makes an airplane fly?	• Create a hypothesis about what makes an airplane fly • Creating a paper airplane prototype • Test prototype • Debrief on successes and failures
Day 3	1. What are the four forces of flight? 2. How do wings keep an airplane in the air?	• Recap Day 2 Activity • Introduce 4 Forces of Flight • Explore center of gravity • Explore Bernoulli’s Principle on flight • Explore a wing’s role in flight
Day 4	1. What is the Bernoulli principle? 2. How does the angle of attack affect lift and drag?	• Recap Day 3 activities • Work Stations (Bernoulli Principle, Center of Gravity, Wing Cross-Section, 4 Forces Reinforcement, and Airplane Parts)
Day 5	How does changing the angle of attack affect the lift and drag of an airplane?	• Recap Day 4 Activity • Build glider/prototype/Test glider • Revise the design/Change angle of attack
Day 6	How can we make an airplane stable?	Same as above
Day 7 & 8	How can we build an airplane that flies the farthest?	Same as above
Day 9	Same as above	Same as above
Day 10	Determine the best airplane	Final contest

During the first session of the PBL unit, the teachers introduced the learning objectives, driving question and purpose of the unit. Most importantly, the teachers explained the Problem Solving Process Chart (PSC) (see Figure 1) that mapped CT with the problem-solving and engineering design processes and activities [6].

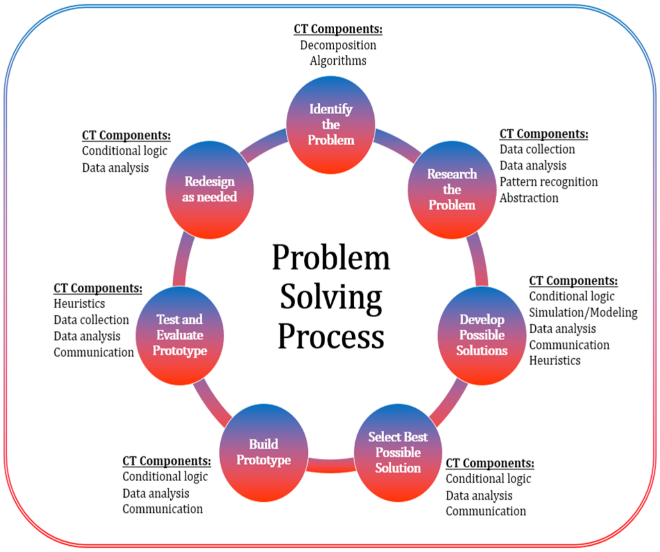

Fig. 1. Problem solving chart.

Various CT components (Table 1) were covered and practiced during various design and inquiry activities in the lessons. For example, abstraction was practiced when students needed to apply the theories of forces during the design of an airplane and its wings. Communication and data analysis were practiced when students needed to test and revise their airplanes.

A Bebras Challenges test consisting of 10 problems with different tasks [21] was administered to all students before and after the curriculum unit. The Bebras Challenges examined students' logic and CT skills through different types of problems with three levels of difficulty which was frequently adapted by researchers to measure elementary students' CT [22]. The examined logic and CT skills in the test were closely related to what the airplane design unit/curriculum focused on. The challenges took about 35 minutes to finish. 39 students completed both tests.

V. RESULT AND IMPLICATION

Data analysis showed that the student CT skills were significantly improved (p=.04) after completing the airplane design unit activities. Table III summarizes the students' CT as measured by the pre- and post-challenge test. The results demonstrated that for those participants the PBL and integrated airplane design activities helped not only teaching the subject content knowledge but also provided an opportunity to learn and practice CT. The airplane design activities integrated with CT seemed to help students better solve the design challenges as students practiced various CT components during decomposition of the problems and finding solutions.

It is interesting to note that although the airplane design activities guided by the PBL approach were not focused on logic and algorithms like coding or programming activities, such design activities do help students learn CT components like conditional logic as tested by the Bebras Challenges. This finding would help researchers expand CT integration beyond the usual coding and programming to a non-coding and programming approach.

TABLE III. PARTICIPANTS' PRE- AND POST- CHALLENGE PERFORMANCE

Pre-Challenge				Post-Challenge			
Min	Max	M	SD	Min	Max	M	SD
1	7	4.28	1.67	1	9	4.75	1.99

The study has limitations. First, the Bebras Challenges test focuses on logic and pattern recognition and is not fully representative of all the CT components embedded (see Table 1) in the curriculum unit. Second, most students had not been trained in completing questions like the Bebras Challenges, which might also help explain the low means for both the pre- and post-challenge performance. Students' unfamiliarity with the type of challenge questions might have particularly affected students' performance on the pre-challenge test (their first time encountering such questions).

Given the increasing popularity of integrating CT in elementary curriculum, this study has important implications for integrating CT in K-12 education, especially for computing education and engineering education. The study shows that it could be feasible to achieve a complex learning goal of computing education and engineering education via project-based learning for elementary students. The study fills a research gap of using PBL guided engineering design activities to develop CT in students.

ACKNOWLEDGMENT

We wish to thank the National Science Foundation for supporting this study.

REFERENCES

[1] J. Wing, "Computational thinking and thinking about computing," Philosophical Transactions. Series A, Mathematical, Physical, and Engineering Sciences, vol. 366, 2008, pp. 3717–3725.

[2] National Research Council. "Committee for the Workshops on Computational Thinking: Report of a workshop of pedagogical aspects of computational thinking.," Washington, D.C., 2011.

[3] C. Harris, "Computational Thinking Unplugged: Comparing the Impact on Confidence and Competence from Analog and Digital Resources in Computer Science Professional Development for Elementary Teachers," A Education doctoral Dissertation in Executive Leadership, 2018.

[4] National Academy of Engineering and National Research Council, "STEM Integration in K–12 Education: Status, Prospects, and an Agenda for Research," Washington, DC: The National Academies Press, 2014.

[5] D. Yang, and S. J. Baldwin, "Using technology to support student learning in an integrated STEM learning environment," in International Journal of Technology in Education and Science, vol 4, 2020, pp. 1-11.

[6] D. Yang, S. Swanson, B. Chittoori, and Y. Baek, "Work-in-Progress: Integrating computational thinking in STEM education through a project-based learning approach," Proceedings of the American Society for Engineering Education (ASEE) Annual Conference and Exposition, Salt Lake City, UH., 2018.

[7] M. M. Hynes, T. J. Moore, M. E. Cardella, K. M. Tank, S. Purzer, M. Menekse, and S. P. Brophy, "Inspiring computational thinking in young children's engineering design activities," In Proceedings of the American

Society for Engineering Education (ASEE) Conference & Exposition, New Orleans, Louisiana, 2016.

[8] Buck Institute of Education (BIE), “Why project based learning?’ 2017.

[9] D. A. Kolb, “Experiential learning: Experience as the source of learning and development (2nd ed.),” Upper Saddle River, New Jersey: Pearson Education, Inc., 2014.

[10] T. C. Hsu, S. C. Chang, and Y. T. Hung, (2018). “How to learn and how to teach computational thinking: Suggestions based on a review of the literature,” Computers & Education, vol 126, 2018, pp. 296-310.

[11] K. Brennan, and M. Resnick, “New frameworks for studying and assessing the development of computational thinking,” In American Educational Research Association meeting. Vancouver, BC, Canada, 2012.

[12] S. Grover, and R. Pea, “Computational thinking: A competency whose time has come,” In S. Sentance, E. Barendsen, & C. Schulte (Eds.), Computer science education: Perspectives on teaching and learning, New York, NY: Bloomsbury Academic, 2018, pp. 19-38.

[13] S. Y. Lye, and J. H. L. Koh, “Review on teaching and learning of computational thinking through programming: What is next for K-12?” Computers in Human Behavior, vol 41, 2014, pp. 51–61.

[14] I. Lee, F. Martin, J. Denner, B. Coulter, W. Allan, J. Erickson, … and L. Werner, “Computational thinking for youth in practice,” Acm Inroads, vol 2, 2011, pp. 32–37.

[15] D. Barr, J. Harrison, and L. Conery, “Computational thinking: A digital age skill for everyone,” Learning & Leading with Technology, vol 38, 2011, pp. 20–23.

[16] A. Yadav, N. Zhou, C. Mayfield, S. Hambrusch, and J. T. Korb, “Introducing computational thinking in education courses,” SIGCSE’11 - Proc. 42nd ACM Tech. Symp. Comput. Sci. Educ., no. 1, pp. 465–470, 2011.

[17] O. Astrachan, and A. Briggs, “The CS principles project,” ACM Inroads, vol 3, 2012, pp. 38–42.

[18] Computer Science Teacher Association (CSTA), & International Society for Technology in Education (ISTE), “Computational Thinking Teacher Resources (Second ed.),” 2011.

[19] S. Grover, and R. Pea, “Computational thinking in K-12: A review of the state of the field,” Educational Researcher, vol 42, 2013, pp. 38–43.

[20] D. Catlin, and J. Woollard, “Educational robots and computational thinking,” In Proceedings of 4th International Workshop Teaching Robotics, Teaching with Robotics & 5th International Conference Robotics in Education, Padova, Italy, 2014, pp. 144–151.

[21] V. Dagienė, “Information technology contests - introduction to computer science in an attractive way,” Informatics in Education, vol 5, 2006, pp. 37–46.

[22] Y. Baek, S. Wang, D. Yang, Y-H. Ching, S. Swanson, and B. Chittoori, “Revisiting second graders’ robotics with an Understand/Use-Modify-Create (U2 MC) strategy,” European Journal of STEM Education, vol 4, 2019: 07.