The Future of Engineering Education
2024 Annual Conference & Exposition
Oregon Convention Center ASEE
Portland, OR . June 23 - 26, 2024
Paper ID #43400

WIP: Using Games and Robotics to Teach Computer Programming in High School STEM Classes: A Collective Case Study

Leslie Anna Brown, Utah State University

Leslie Brown is an Engineering Education Master's student at Utah State University. Her thesis focuses on training high school STEM teachers to implement engineering curriculum into their classrooms.

Dr. Marissa A Tsugawa, Utah State University

Marissa Tsugawa is an assistant professor at Utah State University who leverages mixed-methods research to explore neurodiversity and identity and motivation in engineering. They completed their Ph.D. in Engineering Education where they focused on motivation and identity for engineering graduate students.

WIP: Using Games and Robotics to Teach Computer Programming in High School STEM Classes: A Collective Case Study

Abstract

The purpose of this WIP research paper is to explore high school STEM teachers' perspectives on teaching computer programming through games and robotics in their classes. In response to high demands for software engineering and a decreased availability of programmers, student interest in computing needs to begin before higher education. Initiatives such as the creation of the Next Generation Science Standards (NGSS) call for cross-cutting engineering and technology instruction in K-12 STEM classes. We investigate how high school STEM teachers can incorporate games, underutilized educational technology (e.g., robots, smartphones), and block-based programming to increase motivation and engagement in computer programming among high school students through a qualitative, collective case-study design. Our case study implements an intervention where we train high school teachers to teach computational thinking through a game that leverages problem solving, block programming, and a 'robotics in healthcare' theme. After observing high school STEM teachers facilitate the designed learning unit, we will interview them about their perceptions on this teaching framework and we will observe their teaching when implementing the game lesson. Data will then be coded and analyzed using thematic analysis to find out the change in preparedness and engagement towards teaching computer science.

Introduction

In response to the lack of engineering and computer science education in high school, the Next Generation Science Standards (NGSS) were created in 2014 by twenty-six states with twenty states adopting these standards [1]. The NGSS shifted science instruction to incorporate cross-cutting (utilizing common themes among STEM disciplines) engineering standards and expand on computational thinking skills [2]. However, as technology and computing have advanced, the NGSS do not reflect the modern skills needed for computing to address a sizable demand for computer programmers in the workforce [3], [4]. Particularly, the US Bureau of Labor Statistics [5] anticipates a 15% increase in computer science related careers, but a 10% decrease in the computing workforce. One potential factor in this predicted decrease could be that high school students lack exposure to computer science as upward of 89% of high school science teachers never include computer science in their teaching [6]. To expose high school students to computing, future iterations of the NGSS should integrate computer science standards that align with computing skills needed to enter college computer science programs. Furthermore, STEM high school teachers need supportive programs and partnerships to bring computer science into their classrooms.

While there are no current computer science standards within the NGSS, current studies have researched the most popular forms of teaching code: utilizing gamification of learning and/or problem-based learning (PBL) [7]. In this study, we will explore the implementation of computer science instruction leveraging the gamification framework and instructional technology. We seek to answer the following research question: How can teachers combine gamification of learning and robotics to teach computer programming?

Researcher Positionality

I (Leslie Brown) am currently an engineering education graduate student at Utah State University. I worked as a software engineer after completing a mechanical engineering undergraduate degree. Because of this transition from mechanical engineering to software engineering, I realized how the skill sets between the two professions overlapped and the importance of exposure to multiple engineering fields. Additionally, I was raised by a public school teacher and have worked with various K-12 teachers throughout my life. I see the effort and dedication teachers put into their craft. Because of this background, through my research, I have built rapport with high school teachers to support them in implementing national and state standards (such as the NGSS) to expose students to critical thinking and problem-solving skills.

I (Marissa Tsugawa) am a tenure-track faculty of engineering education at Utah State University. with a research focus on both neurodiversity in engineering and implementing engineering through gamification in K-12 STEM classrooms. During my mechanical engineering Master’s program, I participated in a nationally funded fellowship where I partnered with science middle and high school teachers to bring engineering into their classrooms. I developed engineering lessons that followed the NGSS and supported science learning. My experience also showed me the limited time teachers have to learn and prepare extra content they were not trained to teach.

Literature Review

Using problem-based gamification aims to increase a student’s problem-solving and logical thinking skills, which are both necessary for computer science and engineering professions [8]. In computer science education, research has shown that problem-solving type games significantly increased students' motivation, self-efficacy, and interest in the subject matter [9], [10]. Teachers implementing gamification have noticed an increase in complexity throughout gamified computer science student projects [11].

Currently, research has not made conclusions on effective educational frameworks for teaching computer science in K-12 spaces [12]. Unlike the NGSS guidelines for engineering, there are no national standards for K-12 computer science education [4]. However, some researchers have explored the use of proper instructional technology when teaching computer science to K-12 students. Researchers have found that manipulating physical objects (such as the LEGO® SPIKE™ robot) helps students visualize code more effectively compared to their 2D counterparts (e.g. Scratch) [13], [14]. Not only was the visualization of components effective, but students reported higher levels of active listening, active learning, and peer collaboration when using LEGO® robotics.

Using a LEGO® SPIKE™ robot and block-based coding, teachers can overcome challenges they face such as motivating students by giving them a physical tool that represents visual coding practices [13], [14]. This physical and visual tool can also assist in structuring game-based problem-solving challenges while minimizing syntax and code structure difficulties [15]. Using a proper game-based approach to teaching computer science using a LEGO® SPIKE™ robot, will support high school teachers in generating interest in computer programming among students [10], [16] .

Pedagogical Framework: Gamification of Learning

Gamification of learning is defined as "the use of game elements in non-game contexts" [17, p. 2] and should include four main components, a game goal, game dynamics, game mechanics, and game elements (defined in Table 1 in the Methods section, [18]). Researchers have shown that gamification increases learning achievement, motivation to learn, and generates positive attitudes among students [10], [19]. Specifically, this paper focuses on problem-solving type games to foster problem-solving skills and computational thinking. Problem-solving games have effectively increased intrinsic motivation to learn STEM concepts [20], [21]. While utilizing gamification has worked in non-STEM K-12 classes and undergraduate computer science classes, little is known about applying gamification to engineering and computer science in K-12 classes [21] despite its potential to strengthen skills such as problem solving and self-efficacy [8].

Methods

We utilize a collective case study approach to understand teacher perspectives towards using gamification and instructional technology for computer science instruction [22]. Specifically, we trained high school teachers on implementing a robotics lesson in their science classroom that introduces simple coding (Scratch) and physical manipulatives (LEGO® SPIKE™). We utilized both observational and interview data to understand the teachers' perspectives on this instructional framework and physical manipulatives to teach computer science.

Case Definition

For this case study, we explore the practical application of teaching computer science in the high school classroom using gamification and instructional technology. We define each case for this paper as a single high school STEM teacher. Each case will explore an in-depth understanding of high school STEM teachers and their perspectives on utilizing technology and gamification to teach computer science related coursework. We are collecting both handwritten observations and interviews from the STEM teachers to build each case and answer the research questions. Both, within-case (analysis of a single case) and cross-case examination (analysis across multiple cases) techniques will be used to understand the teacher's perspectives of utilizing technology and gamification to teach computer science [22].

Intervention

In this intervention, the participating teachers will teach a robotics unit utilizing both gamification and a LEGO® SPIKE™ robotics kit. The designed lesson plan (created by Leslie Brown) is titled 'Robots in Healthcare' where students will be tasked with coding the LEGO® SPIKE™ robot so that it can maneuver around a hospital to deliver medicine and food to patients. In the first week of the learning unit, students will practice using pseudocode. In the second week of the learning unit, students will practice using block-based code through the LEGO® SPIKE™ application and robotics kit.

‘Robots in Healthcare’ Lesson Plan Overview

In this lesson, students will learn how code and robotics have many applications within society and specifically in healthcare. The ultimate goal of this learning unit is to direct the LEGO® SPIKE™ robot through obstacles (such as around machines or hallways in hospitals) to its destination (to deliver food and/or medicine to patients’ rooms). For more specific game mechanics, see Table 1. Within the lesson plan, we have broken down the curriculum into two sections: the pseudocode section and the block-based code section. Information about both sections are provided below.

Pseudocode Section

During this week of the learning unit, students will first learn about the different coding structures (e.g. functions, if-statements, while and for loops) and they will learn how to structure pseudocode. In this section, students may have difficulty understanding the difference between how a computer reads code versus how a human understands instructions. Students will compare differences between an ‘instruction list’ and ‘computer pseudocode’. For this week’s game, the course instructor will act as the ‘robot’ and the students will work in pairs to develop pseudocode in the form of handwritten notes. The goal of this initial game is to get the ‘robot’ from the back of the room to the front of the room without bumping into any obstacles. If a student finishes one ‘level of difficulty’ the teacher can then place more obstacles in the pathway that students must try to direct the robot to avoid. When the students finish writing their pseudocode, they will then ‘test’ their code against the robot. This initial week is intended to help students strengthen their computational thinking and computer logic skills before introducing any coding syntax or challenges with using new technology.

Block-Based Code Section

For the next week of the learning unit, students will use their pseudocode notes, and translate this code into the LEGO® SPIKE™ application. This application looks similar to Scratch, where students can use visual blocks to construct their code. This application also connects to the LEGO® SPIKE™ robot via Bluetooth for testing their code (see fig 1). Once the student pairs have completed their code, they can then test their code using the LEGO® SPIKE™ robot. This physical representation helps students visualize where they may need to debug or revise their code. The goal in this section is for the robot to reach the patient’s room to deliver food and/or medicine. Once students have completed the ‘first level’ each challenge again increases in difficulty by either adding multiple stops to patient rooms or adding more obstacles to navigate around.

Table 1. An outline of game elements and how they relate to the game-based learning unit

Game Component	Definition	Example
Game Goal	The objective of the game	To reach the patient rooms using the most efficient code possible
Game Dynamics	What the players must do to accomplish the goal	Solving the puzzle: Students must work in pairs to develop and test code using the LEGO® SPIKE™ robot
Game Mechanics	The rules of the game and how players interact in the game	Students must work in pairs to complete their code. Students are also given group

		roles within their team and are assigned certain tasks for building their code. Once students have completed a 'game map level' they must move to the next 'game map' which is more challenging than the last.
Game Elements	The "look and feel" of the game	The theme of the game is 'robots in healthcare' and the game takes place in a hospital. Thus, giving a 'realistic' feel to the game.

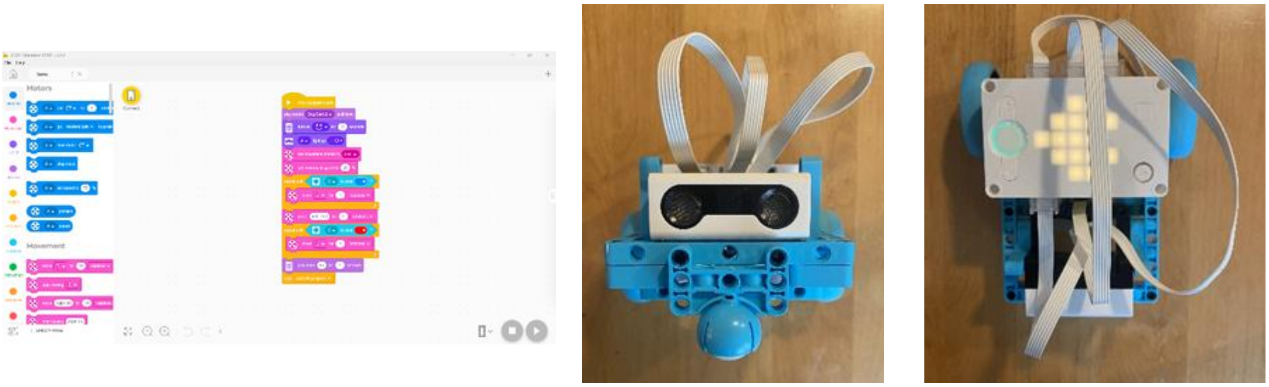

Fig 1. The LEGO® SPIKE™ block-based application and a possible robot configuration

Participants and Recruitment

We will recruit two to three high school science teachers via email to teach the 'Robots in Healthcare' lesson plan to their students. Due to funding constraints, teachers must be working at a school associated with Gaining Early Awareness and Readiness for Undergraduate Programs (GEAR-UP) which is administered by Utah State University. The high schools participating in this program are within and across the state. This particular state's mandated science and engineering standards are similar to the NGSS.

Data Collection and Analysis

We plan to collect observations of and interviews with the teachers then code the data and generate themes. First, we will observe the teachers as they teach the 'Robots in Healthcare' lesson to identify common patterns as they instruct the students. Our observational protocol includes both descriptive and interpretive notes that are time-stamped. Following the observation, we will conduct semi-structured interviews with the teacher that will be audio recorded. Interviews will be transcribed then analyzed using open-coding methods to develop a codebook and identify patterns. [23]. Once we have completed the coding process, we will conduct a thematic analysis to find common themes among the teachers and to better understand a proper framework to teach computer science using gamification and instructional technologies [24].

Preliminary Results

Currently, we have observed and interviewed one chemistry high school teacher teaching the 'Robots in Healthcare' lesson for the high school's STEM club. As a brief overview of the first observation and interview, our first participant, Morgan (pseudonym), demonstrated familiarity with engineering and the engineering problem-solving process during our observations. Morgan was able to answer the students' questions related to programming and how the robot should function. In the interview, Morgan described their experience with teaching engineering problem-solving and its differences and similarities with the scientific method. They stated:

"So it's a little bit different with ya know, the scientific process of you're having like, this prediction or this idea and then you test. Umm.. You're kind of doing more of, you have a problem, you're designing something whatever that problem type is and then you're finding a uhh… like a solution to answer that problem."

However, considering their experience with gamification, Morgan described not having interest in using the gamification framework in their teaching practice.

"[gamification] can kind of lead into the problem I have in the classroom where students… can't focus for a few minutes… gotta keep having the next exciting thing were um…. Sometimes I feeling like I can be more of a performer, instead of a teacher."

We are in the process of scheduling the intervention with two other science teachers at different high schools (a physics teacher and an engineering/earth science teacher). After all data are collected, we can understand how high school STEM teachers can integrate engineering standards through gamification of learning methods.

Implications

After completing our intervention with the next teachers, we will be able to generate themes within and across the cases of teachers' experiences teaching computer programming using gamification and robotics. Both within-case and across-case results will contribute to the teaching and learning scholarship by elucidating how gamification can be utilized to teach computer science at the K-12 level. For example, by conducting this learning unit, STEM teachers may see themes in required skills to teach open-ended problem solving and computational thinking that may differ from scientific inquiry [25]. This case study also provides insight on proper ways to implement a computer science framework from the teacher's perspective, as there is no 'best practice' provided [12].

This research is also meant to combine the use of instructional technology with gamification framework to provide insights into best computer science instructional practices. The research will help provide insight on methodology, proper tools, and practices for supporting teachers to teach computer science in high school. This way, teachers will be better equipped to help students develop the proper problem-solving skills and computational thinking skills required to become engineers or computer programmers.

Conclusion

The proposed gamified lesson plan utilizing instructional technology is meant to address the lack of proper teaching framework and training for K-12 computer science instruction. By completing this study, we aim to provide an in-depth understanding of popular frameworks (such as problem-solving and gamification) for computer science instruction. Not only will this research provide a framework for computer science instruction, but it will also explore supporting instructional technology when teaching computer science. Thus, this research will provide insight on teacher perceptions and preparedness to teach computer science in high school after receiving proper support. Preparing teachers to teach computer science and utilize problem-solving skills and computational thinking will in turn better prepare and expose high school students to engineering and computer science disciplines.

References

[1] National Science Teachers Association (NSTA), “About the Next Generation Science Standards,” *National Science Teachers Association (NSTA)*, 2016. https://ngss.nsta.org/About.aspx (accessed Aug. 12, 2023).

[2] Next Generation Science Standards Lead States, *Next generation science standards: For states, by states*. Washington DC: The National Academies Press, 2013.

[3] W. Aminger *et al.*, “Preservice secondary science teachers’ implementation of an NGSS practice: Using mathematics and computational thinking,” *J. Sci. Teacher Educ.*, vol. 32, no. 2, pp. 188–209, Feb. 2021.

[4] R. S. N. Lindberg, T. H. Laine, and L. Haaranen, “Gamifying programming education in K-12: A review of programming curricula in seven countries and programming games,” *Br. J. Educ. Technol.*, vol. 50, no. 4, pp. 1979–1995, Jul. 2019.

[5] Bureau of Labor Statistics, U.S. Department of Labor, “Occupational Outlook Handbook,” *Computer Programmers*, Sep. 08, 2022. https://www.bls.gov/ooh/computer-and-information-technology/computer-programmers.htm#tab-6 (accessed May 30, 2023).

[6] E. R. Banilower, P. S. Smith, K. A. Malzahn, C. L. Plumley, E. M. Gordon, and M. L. Hayes, “Report of the 2018 NSSME+,” *Horizon Research, Inc.*, Dec. 2018, Accessed: Jul. 28, 2023. [Online]. Available: http://files.eric.ed.gov/fulltext/ED598121.pdf

[7] M. I. Calderon Ribeiro and O. M. Passos, “A Study on the Active Methodologies Applied to Teaching and Learning Process in the Computing Area,” *IEEE Access*, vol. 8, pp. 219083–219097, 2020.

[8] C.-S. Chang, C.-H. Chung, and J. A. Chang, “Influence of problem-based learning games on effective computer programming learning in higher education,” *Educ. Technol. Res. Dev.*, vol. 68, no. 5, pp. 2615–2634, Oct. 2020.

[9] M. Moradi and N. F. B. Noor, “The Impact of Problem-Based Serious Games on Learning Motivation,” *IEEE Access*, vol. 10, pp. 8339–8349, 2022.

[10] M. Gamarra, A. Dominguez, J. Velazquez, and H. Páez, “A gamification strategy in engineering education—A case study on motivation and engagement,” *Comput. Appl. Eng. Educ.*, vol. 30, no. 2, pp. 472–482, Mar. 2022.

[11] V. F. Martins, I. de Almeida Souza Concilio, and M. de Paiva Guimarães, “Problem based learning associated to the development of games for programming teaching,” *Comput. Appl. Eng. Educ.*, vol. 26, no. 5, pp. 1577–1589, Sep. 2018.

[12] R. P. Medeiros, G. L. Ramalho, and T. P. Falcão, “A Systematic Literature Review on Teaching and Learning Introductory Programming in Higher Education,” *IEEE Trans. Educ.*, vol. 62, no. 2, pp. 77–90, May 2019.

[13] M. Lykke, M. Coto, C. Jantzen, S. Mora, and N. Vandel, “Motivating Students Through Positive Learning Experiences: A Comparison of Three Learning Designs for Computer Programming Courses,” *JPBLHE*, vol. 3, no. 2, 2015, doi: 10.5278/ojs.jpblhe.v0i0.1130.

[14] C. Qin, Y. Liu, and H. Zhang, “Scratch versus LEGO robots: Which engages undergraduates more in programming education?,” *J. Comput. Assist. Learn.*, vol. 39, no. 3, pp. 935–953, Jun. 2023.

[15] S. Altiok and E. Yükseltürk, “Pre-Service Information Technologies Teachers’ Views on Computer Programming Tools for K-12 Level,” *International Journal of Computer Science Education in Schools*, vol. 2, no. 3, Aug. 2018, Accessed: Dec. 11, 2023. [Online]. Available: http://files.eric.ed.gov/fulltext/EJ1207430.pdf

[16] E. Afari and M. S. Khine, “Robotics as an educational tool: Impact of lego mindstorms,”

International Journal of Information and Education Technology, vol. 7, no. 6, pp. 437–442, 2017.

[17] S. Deterding, R. Khaled, L. Nake, and D. Dixon, "Gamification: Toward a definition," in *gamification workshop proceedings*, Vancouver, Canadá, May 2011, pp. 1–79.

[18] S. Boller and K. Kapp, *Play to Learn: Everything You Need to Know About Designing Effective Learning Games*. Association for Talent Development, 2017.

[19] I. Yildirim, "The effects of gamification-based teaching practices on student achievement and students' attitudes toward lessons," *The Internet and Higher Education*, vol. 33, pp. 86–92, Apr. 2017.

[20] D. Alt, "Assessing the benefits of gamification in mathematics for student gameful experience and gaming motivation," *Comput. Educ.*, vol. 200, p. 104806, Jul. 2023.

[21] H. Dehghanzadeh, M. Farrokhnia, H. Dehghanzadeh, K. Taghipour, and O. Noroozi, "Using gamification to support learning in K-12 education: A systematic literature review," *Br. J. Educ. Technol.*, May 2023, doi: 10.1111/bjet.13335.

[22] B. Johnson and L. Christensen, "Chapter 15 Narrative Inquiry and Case Study Research," in *Educational Research: Quantitative, Qualitative, and Mixed Approaches*, 7th ed.SAGE Publications Inc., 2019, pp. 397–420.

[23] J. Saldaña, *The Coding Manual for Qualitative Researchers*. SAGE Publications, 2021.

[24] V. Braun and V. Clarke, *APA Handbook of Research Methods in Psychology*, vol. 2. American Psychological Association, 2012.

[25] R. W. Bybee, "NGSS and the Next Generation of Science Teachers," *J. Sci. Teacher Educ.*, vol. 25, no. 2, pp. 211–221, Apr. 2014.