A Multi-Case Study Exploration of Teachers' Choices of Computer Science Tools Following Professional Development

Purpose and Perspectives

Although schools have offered computing since 1980's, the basis of this education has largely been the ability to use computers and software as opposed to creating with them (Guzdial et al., 2014; Repenning et al., 2015). However, there has been a shift in the educational landscape as denoted by the adoption of CS standards into the curriculum in 34 states (Code.org, 2019), highlighting the expansion of interest in computer science (CS) within the US education system. This shift has been spurred by several factors, including economic needs, the proliferation of CS across domains, the need for understanding computers and computing for participation in public life, and a drive for more equitable access and participation in CS (Blikstein, 2018; Ericson et al., 2008).

Despite states' recognition of these needs and their adoption of CS standards, barriers remain in implementing CS learning within the classroom. Most notably, fewer than one in three teachers has had any CS education, formal or informal (Century et al., 2013). This means that very few teachers at any level have the background knowledge necessary to implement CS learning in their classrooms. Moreover, even fewer teachers at the elementary and middle school levels are prepared to incorporate CS within their lessons (Barr & Stephenson, 2011; Bilkstein, 2018).

Preparing the next generation of students necessitates preparing these teachers through professional development (PD) in the CS content, pedagogy, and computing skills required to bring CS into their lessons. However, the research on CS PD is still in its emerging stages, with the majority of literature focusing on PD for high school teachers (Menekse, 2015). What is more, there are significant gaps in this body of knowledge. Most notably, the body of work regarding CS PD is largely absent of exploring teachers' classroom practice after taking part in PD, with those studies that do relying on self-reported teacher data (Kay et al., 2014; Liu et al., 2011; Authors, 2017).

This work examines a research-based PD program designed to support elementary and middle school teachers to integrate CS within their classroom. The PD incorporates two elements: a week-long Summer Institute followed by in-classroom support. The Summer Institute focuses on CS content, pedagogy, skills, and culturally relevant approaches to CS education designed to make CS more equitable for *all* students (Authors, 2021; Madkins et al., 2019). Follow-up support is provided by CS undergraduates who partner with teachers, as part of a service-learning course (see Authors, 2017). Specifically, this work explores the following question: *After taking part in a CS PD program, what CS instructional tools did teachers use with and without CS undergraduate support?*

Context

This study was conducted as part of a larger NSF-funded project designed to increase CS instruction in a Mid-Atlantic state by providing CS focused PD to in-service teachers followed by classroom support delivered by CS undergraduates through a field experience course (Authors, 2015). This study focuses on the PD and follow up support provided through the field experience course after two consecutive summer PD sessions.

The PD program that served as the basis for this work was designed for teachers interested in incorporating CS into their existing curricula. Instruction and materials revolved around four main themes: (a) learning CS content and pedagogy; (b) developing skills for integrating CS principles into existing curricula; (c) establishing a community of practice; and (d) expanding access to and participation in CS to a wider demographic of students (examples of the PD for summer A can be found in Table 1 and for Summer B can be found in Table 2). After participation in PD, teachers were offered the opportunity to receive classroom-based support focusing on the implementation of CS from CS undergraduate students.

Insert Table 1

Insert Table 2

Methods

Participants

Study participants were selected from elementary and middle school teachers attending the PD. Teachers voluntarily attended the PD, suggesting their interest in learning how to implement CS within their classrooms. Near the conclusion of each summer PD program, teachers were invited to participate in the study and receive follow-up support by partnering with CS undergraduates enrolled in the field-experience course.

Seven teachers, from different schools, who took part in the PD (Summer A or Summer B) participated in the study. Three participated in Summer A with two continuing into Summer B. Additionally, four teachers joined the study after Summer B. (see Table 3 for a timeline of participation).

Insert Table 3

Data Collection

Data collected included: Classroom implementation data, consisting of classroom observations and undergraduate weekly reflections, and teacher interviews. Each teacher's class was observed 5-12 times each semester they were involved in this work. Observations were conducted without a structured protocol, but attention was paid to the types of CS tools teachers, undergraduates, and students utilized within the classroom. Additionally, as a weekly assignment for the field experience course undergraduates wrote reflections on their time in teachers' classrooms. Reflection is an important component of service-learning which helps connect what undergraduates learn in the classroom with their experience of real-world conditions (Bringle & Hatcher, 1999). Reflections were structured around three prompts: (a) field activities enacted, (b) identified successes and challenges, and (c) recommended steps for future action (Authors, 2015, 2016, 2020). In addition, all teachers were interviewed at two points in time throughout their participation. (see Table 4 for a timeline of data collection).

Insert Table 4

Data Analysis

All data were uploaded into Dedoose software for coding. An iterative process using an emergent thematic approach was used to establish a coding framework. Responses were categorized using a constant comparative method (Glaser & Strauss, 1967). Themes were tested with samples and modified until categories fit with the overall data set through multiple rounds of coding (Given, 2008). Multiple data sets were used in determining themes and triangulating findings when confirming or dismissing themes (Merriam & Grenier, 2019).

After determining themes, codes were applied to all data, to identify patterns of themes from within each teacher's classroom and any overall patterns that were present. Following the coding process, approximately 10% of the data were recoded to check for intra-rater reliability. Discrepancies led to application when codes were missed and deletion when incorrectly applied. Thereafter, the data were recoded and re-checked for intra-rater reliability until coding reliability for all categories using Cohen's Kappa (Cohen, 1960) were above K=0.80, which is considered strong agreement (McHugh, 2012).

Findings

Tool Usage

Findings indicated that teachers fit into one of three categories: heavy-volume users, moderate-volume users, and low-volume users. Heavy-volume users were defined as teachers who used more than two CS tools with their classes. Moderate-volume users were defined as teachers who used no more than two CS tools with their classes with only one being an online CS tool and only one being a physical CS tool. Low-volume users were defined as teachers who used no tools themselves during observations but whose undergraduates may have led lessons in which tools were used (see Table 5 for data on tool usage).

Insert Table 5

Two teachers, Maria and Denise, fell into the heavy user category using ten and five tools respectively over the course of the study. Three teachers, Kim, Erica, and Evelyn, fell into the moderate-volume user category, each using one or two tools. The final two teachers, Victoria and Stephanie fell into the low-volume user category. Neither used any digital or physical CS tools with their classes when observations were undertaken. However, Stephanie referenced using Scratch with her students in her interview, and Victoria's undergraduate partners ran classes using both Scratch and Micro:bits.

Findings further indicated that the majority of tools teachers used were those featured in the PD, and that each teacher used at least one of the CS tools featured in the PD.

Reasons for Usage

Findings indicated three primary factors that influenced teachers' choice of tools. First, teachers most readily used tools which were available to them at no cost, but when they were able to procure funding, they did use additional tools with their students. Second, teachers' choice of tools was influenced by recommendations from trusted sources. Finally, teachers' use of tools was influenced by their ease of use.

Funding/Access

Findings support that teachers' use of tools was partially affected by their access to funding. The majority of tools used in teachers' classrooms could be classified as Open Educational Resources (OER), free for teachers use, and all teachers who used CS tools with their classes used at least one OER, while Maria and Denise, the heavy-volume users, used multiple OER with their classes. Despite the prevalence of OER, there were some exceptions; Maria, Kim, and Evelyn used tools purchased for CS instruction. However, Evelyn and Maria both purchased these tools primarily using grant money they secured themselves. Of the seven teachers in the study, only Kim was provided funding within her school budget for purchase of these tools. Finally, Erica and Victoria were able to access tools based on their relationships. Victoria's undergraduate partners brought Micro:bits, small programmable computing devices, to use with her class, and Erica worked closely with members of her district office who loaned her Spheros to use with her class (see table 6).

Insert Table 6

Influence of Knowledgeable Others

Interview data showed that a large portion of teachers' CS tool choices were influenced by recommendations from the PD or from their undergraduate partners while they provided follow up support. Additionally, teachers reported that knowledgeable peers within their schools and online communities were influential in their choices (see table 7 for teachers' reported influences).

Insert Table 7

Teachers Content Knowledge and Technical Skills

Finally, evidence demonstrates that teachers' use of tools was further influenced by their ease of use. Interview data showed that with limited exceptions, teachers felt that their CS content knowledge and ability to use CS tools was relatively low. As a result, teachers primarily chose tools that were highlighted during the PD, and that were simple to use. For example, most teachers used Scratch with their students, and all teachers who used Scratch did so in conjunction with "Scratch Cards," step-by-step programming directions provided at the Scratch website, or with the Google CS First platform, a digital platform with video directions and premade projects for students to complete. This was indicative of how teachers used these CS tools, leveraging resources that were available in conjunction with the tools to support gaps in their content knowledge and technical skills.

Discussion and Significance

Findings suggest that when teachers choose CS tools to use with their classes, they are relying on more knowledgeable others to guide them. Based on teachers' reports and classroom observations, teachers' use of CS tools seems to be heavily influenced by their participation in PD. Demonstrating that CS PD likely has an influence on classroom practice.

However, there were other indications within the data that further helped explain why teachers used these tools. Teachers' use of OER can be explained by the absence of cost and the materials supporting their use. However, findings also indicated that when teachers were able to procure their own funding or had access to other tools through outside contacts, they often used these tools as well. This may demonstrate that teachers are the driving force behind CS integration, and that school administrators have not yet prioritized CS integration as demonstrated by teachers' need to procure their own funding.

Finally, findings indicated that ease of use as well as the availability of sample curricular materials is an important consideration that influences teachers' choices. This finding is consistent with prior work indicating that curricular resources coupled with digital tools' ease of use have been shown to support technology adoption within a school setting (Aldunate & Nussbaum, 2013; Martin, 2003).

Together, these findings offer insight into how PD can influence teachers' choice of CS tools used in their classrooms which may foster the development of PD that supports teachers' implementation of CS.

References

Authors. (2015).

Authors. (2016).

Authors. (2017).

Authors. (2020).

Authors. (2021).

Aldunate, R., & Nussbaum, M. (2013). Teacher adoption of technology. *Computers in Human Behavior*, *29*(3), 519-524.

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: What is involved and what is the role of the computer science education community?. *Acm Inroads*, *2*(1), 48-54.

Blikstein, P. (2018). Maker movement in education: History and prospects. In M.J. de Vries & I. Mottier (Eds.), *Handbook of technology education*, (pp. 419-437) Springer International Publishing.

Bringle, R. G., & Hatcher, J. A. (1999). Reflection in service learning: Making meaning or experience. *Educational Horizons*, 179.

Century, J., Lach, M., Wilson, C., Guzdial, M., Chapman, G., & Astrachan, O. (2013). Expanding access to K-12 computer Science Education: research on the landscape of computer science professional development. In *Proceedings of the 44th ACM Technical Symposium on Computer Science Education* (pp. 541-542). ACM.

Code.org.. (2019). *State of computer science education: Equity and diversity.* Retrieved from https://advocacy.code.org/2019_state_of_cs.pdf.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, *20*(1), 37-46.

Ericson, B., Guzdial, M., & Biggers, M. (2007). Improving secondary CS education: progress and problems. *ACM SIGCSE Bulletin*, *39*(1), 298-301.

Given, L. M. (Ed.). (2008). *The Sage encyclopedia of qualitative research methods*. Sage publications.

Glaser, B., & Strauss, A. (1967). The discovery of grounded theory: Strategies for qualitative research. Mill Valley.

Guzdial, M., Ericson, B., Mcklin, T., & Engelman, S. (2014). Georgia computes! An intervention in a US state, with formal and informal education in a policy context. *ACM Transactions on Computing Education (TOCE)*, *14*(2), 1-29.

Kay, J. S., Moss, J. G., Engelman, S., & McKlin, T. (2014). Sneaking in through the back door: Introducing K-12 teachers to robot programming. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education* (pp. 499-504).

Liu, J., Lin, C., Hasson, E. P., & Barnett, Z. D. (2011). Introducing computer science to K12 through a summer computing workshop for teachers. In *Proceedings of the SIGCSE'11: The 42nd ACM Technical Symposium on Computer Science Education.* (pp. 389–394).

Madkins, T. C., Martin, A., Ryoo, J., Scott, K. A., Goode, J., Scott, A., & McAlear, F. (2019, February). Culturally relevant computer science pedagogy: From theory to practice. In *2019 research on equity and sustained participation in engineering, computing, and technology (RESPECT)* (pp. 1-4). IEEE.

Martin, M. H. (2003). *Factors influencing faculty adoption of web-based courses in teacher education programs within the State University of New York* (Publication No. 3089087) [Doctoral dissertation, Virginia Polytechnic Institute and State University] PQDT Open. https://vtechworks.lib.vt.edu/bitstream/handle/10919/27399/MartinDiss2.pdf?sequence=

Menekse, M. (2015). Computer science teacher professional development in the United States: a review of studies published between 2004 and 2014. *Computer Science Education, 25*(4), 325-350.

McHugh, M. L. (2012). Interrater reliability: The kappa statistic. *Biochemia Medica, 22*(3), 276-282.

Merriam, S. B., & Grenier, R. S. (Eds.). (2019). *Qualitative research in practice: Examples for discussion and analysis*. John Wiley & Sons.

Repenning, A., Grover, R., Gutierrez, K., Repenning, N., Webb, D., Koh, K., Nickerson, H., Miller, S. B., Brand, C., Horses, I. H. M., Basawapatna, A., Gluck, F., Grover, R., Gutierrez, K., & Reprenning, N. (2015). Scalable game design: A strategy to bring systemic computer science education to schools through game design and simulation creation. *ACM Transactions on Computing Education, 15*(2), 1–31.

Table 1

Summer Institute PD Schedule (Summer A)

Time	Monday	Tuesday	Wednesday	Thursday	Friday
9:00-10:15	Introductions, Program Purpose & Overview + CS Unplugged Icebreaker	Scratch and Paired Programming	Data Evaluation & Sources Unplugged- Outbreak Data Power Data Sources	CS Toys: Exploring Different CT/Curriculum Kits	Lesson Development
10:15-10:30	Break				
10:30-11:45	Introduce CSTA Standards, Ex Lesson Unit: Mars Rescue	Creativity with Scratch	Cybersecurity and Data	Culturally Responsive (1/2 hour) & Peer Feedback	Lesson Sharing
11:45-12:30	Lunch				Adjourn
12:30-1:00	Broadening Participation in Computing	Culturally Responsive CS	Broadening Participation in Computing	CS Unplugged- Network Simulation	
1:00-2:15	Continuation of Mars Rescue and Discussion	CSP: Creativity- Assessing Programs for Learning and Creativity	Micro: bit Unplugged-Hack-a-Ball	CSP: Internet- Teaching Web Programming	
2:15-2:30	Break				
2:30-3:45	Culturally Responsive (1/2 hour), Lesson Planning	Lesson Development & Peer Feedback	Culturally Responsive (1/2 hour), Lesson Development & Peer Feedback	Projects for Web Programming	
3:45-4:00	Reflection on Learning	Reflection on Learning	Reflection on Learning	Reflection on Learning	

Table 2

Summer Institute PD Schedule (Summer B)

Time	Monday	Tuesday	Wednesday	Thursday	Friday
9:00-10:15	Introductions, Program Purpose & Overview + CS Unplugged Icebreaker	Algorithms: CS Unplugged Dice Game	Assessing Scratch Products: (Assessing for Creativity) Rubrics and Dr. Scratch	Begin Creating a VR scene using A-Frame	CS Unplugged: Abstraction and Lesson Poster Finalization
10:15-10:30	Break				
10:30-11:45	Programming Ozobots with two different languages-Color lines and block-based	Algorithms and Culturally Relevant Pedagogy (continued)	Data Abstraction with the CORGIS Visualizer	Programming and Creativity with Micro:bits (Rock Paper Scissors Game)	Lesson Sharing
11:45-12:30	Lunch				Adjourn
12:30-1:00	Broadening Participation in Computing	CS First	CS Unplugged: Boolean Searches with Guess Who	CS Unplugged-Cybersecurity	
1:00-2:15	Continuation of Ozobots: Focus on Creativity.	CS First	Digital Art in Pixels	CS Toys: Exploring Different Computational Tools/Curriculum Kits	
2:15-2:30	Break				
2:30-3:45	Introduction to Cultural Relevance	CS First	Cultural Relevance and Lesson Planning	Cultural Relevance and Lesson Planning	
3:45-4:00	Reflection on Learning	Reflection on Learning	Reflection on Learning	Reflection on Learning	

Table 3

Teachers, Schools, and Dates of Participation in PD and In-Class Partnerships with Undergraduates

Teacher	Summer A PD	Fall A Partnership	Spring A Partnership	Summer B PD	Fall B Partnership	Spring B Partnership*
Victoria (MS Teacher)	X	X	X			
Maria (MS Teacher)	X	X	X	X	X	
Denise (MS Teacher)	X	X	X	X	X	
Kim (MS Teacher)	X			X	X	
Evelyn (Elementary Teacher)				X	X	
Erica (Elementary Teacher)				X	Observed w/o Undergraduate Students	
Stephanie (Elementary Teacher)				X	Observed w/o Undergraduate Students	

* All partnerships between teachers and CS undergraduate as well as observations were canceled due to the Covid-19 pandemic.

Table 4

Teachers and Timeline for Data Collection

Teacher	S1 Partnership	S2 Partnership	S3 Partnership
Victoria	14 Observations Interview following S2		
Maria	10 Observations Interview following S2		12 Observations Interview following S3
Denise	11 Observations Interview following S2		6 Observations Interview following S3
Kim			6 Observations Interview following S3
Evelyn			8 Observations Interview following S3
Erica			6 observations (w/o Partnership) Interview following S3
Stephanie			8 observations (w/o Partnership) Interview following S3

Table 5

Tools Used in Teachers' Classrooms and Context of their Use

Teacher	Teacher Chosen CS Tools Observed or Included in Undergraduate Reflections	CS Tools Referenced in Interviews but not observed or included in reflections.	Undergraduate Facilitated CS Tools Observed or Included in Reflections.
High Volume Users			
Maria	Code.org, **Scratch** (with **Google CS First** and without), Bitsbox, **Micro:bits***, **Ozobots***, **Makey Makey*, Sphero***	NA	Lego Boost*
Denise	Pencil Code, **Scratch**, Code.org, Excel, Hour of Code	NA	NA
Low Volume Users			
Victoria	NA	NA	**Scratch**, **Micro:bits***
Stephanie	None	**Scratch**	NA
Moderate Volume Users			
Erica	**Scratch** (with **Google CS First**)	Sphero	NA
Kim	**Edison Bots***, **Scratch** (with **Google CS First**)	NA	**Edison Bots***

Evelyn	**Code n' Go Mice* (Analogous to Beebots and significantly less expensive)**	**Ozobots* (Purchase intended)**	NA

Note. Bolded Text denotes Tools Presented in the Summer PD Sessions. * denotes physical CS tools.

Online CS tools are tools that allow for creation and programming exclusively in an online environment.

Physical CS tools have tangible components which allow for online or physical programming of systems that interact in physical space.

Definitions of Tools

Scratch: Low-floor, high-ceiling, block based programming environment developed at MIT.
Google CS First: A Google initiative providing multi-step online lessons within Scratch designed to support students learning to code.
Pencil Code: An online, block based coding environment for designing and drawing.
Code.org: Online curriculum and programming exercises to support CS concepts.
Excel: Microsoft's spreadsheet program which incorporates programming using formulas.
Hour of Code: Website of coding puzzles and exercises designed as an introduction to programming.
Bitsbox: A subscription based service of follow along directions for creating online programs.
Micro:bit: A programmable, pocket sized physical computing device with LED matrix and multiple sensors.
Ozobot: Small robots designed to introduce programming concepts physically and online.
Makey Makey: A circuitry kit that ties in with programs such as Scratch allowing online environments to extend into the physical world.
Sphero: Spherical programmable robots.
Edison Bot: Small programmable robot with online and physical coding capabilities.
Lego Boost: Programmable Lego sets.

Table 6

No Cost and Purchased Tools Used

Teacher	No Cost Tools Used	Purchased Tools Used	Accessible Tools Used
Maria	Code.org, Scratch	Lego Boost, Bitsbox, Micro:bits, Ozobots, Makey Makey, Sphero	NA
Denise	Pencil Code, Scratch , Code.org, Excel, Hour of Code (website)	NA	NA
Victoria	Scratch	NA	Micro:bits
Stephanie	NA	NA	NA
Erica	Scratch	NA	Sphero
Kim	Scratch	Edisonbots (only tool purchased with school funds)	NA
Evelyn	NA	Code & Go Mice	NA

Table 7

Influences for Teachers' Choices of Tools

Teacher	**Fellow Teacher Recommendations**	**Professional Development**	**Undergraduate Partners**	**Professional Communities**
Maria	NA	Micro:bits, Ozobots, Makey Makey, Sphero, Scratch	NA	Lego Boost, Bitsbox, Google CS First
Denise	NA	Scratch	NA	NA
Victoria	NA	NA	Scratch, Micro:bits	NA
Stephanie	Code.org (Curriculum)	NA	NA	NA
Erica	Google CS First (Scratch), Sphero	NA	NA	NA
Kim	NA	Edisonbots and Google CS First (Scratch)	NA	NA
Evelyn	NA	Code & Go Mice	NA	NA