NARST 2021 Related papers set

Integrating Computational Thinking in Science Curricula: Professional Development and Student Assessment

Introduction

Contemporary disciplinary research practices have significantly changed due to advancement of computational tools in fields ranging from quantum chemistry to systems biology (Pople 2003; Kohn 2003; Kitano, 2017). Reflecting this change, the Next Generation Science Standards have foregrounded Computational and Mathematical Thinking as one of their eight science and engineering practices (NGSS Lead States, 2013). Engaging students in such contemporary science practices requires designing new curricula that foreground these practices. Such curricula necessitate professional development opportunities for teachers to learn pedagogical strategies to support student learning. Our work focuses on involving teachers in designing Computational Thinking (CT) integrated science curricula, supporting them in implementing those in classrooms, and assessing students' learning with those curricula.

In the context of science education, we use the following operational definition of CT in the context of STEM education based on the work of Weintrop et al. (2016): Computational Thinking is the thinking that STEM professionals use to formulate a problem such that a computational tool can be used to solve it and to interpret the solution appropriately. Learning CT requires engaging students in the practices pertaining to data, modeling and simulation, computational problem solving, and systems thinking. Researchers of science learning argue that there are three benefits of integrating Computational Thinking in school science curricula (Wilensky, Brady & Horn, 2013; Grover & Pea, 2013; Weintrop et al., 2016). Firstly, CT practices are an integral part of contemporary science practice. Secondly, there are pedagogical affordances of using computational tools to learn disciplinary ideas (Blikstein & Wilensky, 2010; Dubovi et al., 2017; Authors and colleagues, 2018a; Horwitz et al., 2010; Levy & Wilensky, 2009; Sengupta & Wilensky, 2011). Finally, CT integration in science curricula ensures that a diverse set of learners can engage in these powerful ideas, which are otherwise limited to those who specially opt for computer science classes in the K-12 school setting. To achieve these three benefits of CT integration in school science curricula we have been partnering with high school teachers in a design-based implementation research project that focuses on design, pedagogy and assessment of student learning.

The papers in this related paper set stem from a multi-year project about designing and implementing CT integrated science units (Authors and colleagues, 2018b, Authors and colleagues, 2020a). The current phase of the project engaged two cohorts of science and math teachers in a professional development (PD) program where teachers worked with researchers to co-design computationally enriched units. Teachers then implemented their new units in their classrooms with researcher support. In papers 1 and 2, we present how participation in learning and co-design activities in the PD program helped teachers think about CT integration and relevant pedagogical considerations. Paper 3 describes changes in teachers' perceptions regarding CT integration and its benefits after they participated in the PD program. In paper 4 and 5, we discuss student outcomes using computational mixed-methods approaches that assess students' engagement in CT activities and attitudinal change as they participated in CT integrated co-designed lessons.

Paper 1: Positioning Teachers as Co-designers To Integrate CT Practices in STEM

Subject/Problem

Jointly with Papers 2 and 3, this paper investigates how a professional development can help teachers learn to integrate CT practices into their STEM classrooms and enhance their *pedagogy*–their teaching of disciplinary ideas through engaging with computational tools and practices. Prior work shows that a pedagogical shift towards integration of CT requires curricular reforms with pedagogically effective technological tools and novel methodological approaches to design curricula (Jeong & Hmelo-Silver, 2016; Windschitl, Thompson, Braaten, & Stroupe, 2012). For this purpose, we have undertaken a unique approach to PD, in which teachers are positioned as design partners on CT integrated curricula (Authors and colleagues, 2020a). This work builds on several successful curricula with computational tools that have been designed and implemented in a range of scientific subjects including chemistry, physics, biology, mathematics and materials science (Authors and colleagues, 2003; 2009; 2019).

Our co-design approach foregrounds teachers' views on how the curriculum aligns with learning objectives, teaching strategies, and expectations for student learning (Severance, Penuel, Sumner & Leary, 2016). In this paper, we present a four-week PD in which teachers work closely with researchers to co-design computationally enriched STEM curricula that align with individual teacher's views and goals. To identify the value of designing a PD focused on co-design, we use post-PD survey data to investigate what teachers learned after participating.

Design and Procedure

The PD, called CT-STEM Summer Institute (CTSI), was designed as a four-week program to help teachers co-design curriculum to be implemented in the following school year (2020-2021). Eleven teachers from Chicago area public schools were placed into co-design teams by subject area: biology (2), chemistry (2), environmental science (2), math (3) and physics (2). Each co-design team included at least one computational researcher and one undergraduate research assistant. In addition, one manager and undergraduate assistant provided support across teams.

We implemented CTSI as an in-person workshop in the previous summer (2019) that met from 10am-3pm, but transitioned to online in 2020 due to the COVID-19 pandemic. To facilitate online learning without changing the hours teachers were expected to participate, we restructured each session of CTSI based on practices in online teaching. Many sessions became partly or completely asynchronous. Table 1 shows a full schedule of all CTSI sessions.

Table 1. All CTSI 2020 sessions by week and day, with synchronous sessions in blue, italics text.

Week	Monday	Tuesday	Wednesday	Thursday	Friday
1	*Introductions (45 min)* Intro to Project + Goal setting (45 min) *Discussion (30 min)* *Lesson 0: Intro to Computational Models (2 hr)*	*CTSI 2019 Teacher Panel (1 hr)* Explore CT-STEM units* (1 hr) Intro to CT Practices (1 hr) *Discussion (1 hr)*	Intro to Programming* (1 hr) *Discussion (45 min)* Computational Tool 1* (1 hr) *Discussion* (1 hr)*	Computational Tool 2* (1 hr) *Discussion* (1 hr)* *CT-STEM Pedagogy (2 hr)*	Intro to Co-design (30 min) *Co-design team meeting (1 hr)* *Reflection (30 min)* Co-design (~2 hrs)
2-4	Co-design (~3 hrs) Review team's work (~1 hr)	Team feedback (~45-60 min) Co-design (~2-3 hrs)	*Workshop (~45-60 min)* Co-design (~2-3 hrs)	Co-design (~3.5 hrs) *Teacher-teacher feedback (30 min)*	Co-design (~3.5 hrs) *Reflection (30 min)*

*Optional workshops for three returning teachers, who participated in the in-person CTSI 2019

The first week of CTSI provided a four-day introduction into computational practices and tools. The first two days focused on building the CTSI community and introducing the project and CT-STEM practices through videos, discussions, and CT integrated curriculum (e.g., participating in a CT integrated lesson as students). The next two days focused on learning about specific CT tools and how they could be integrated into curricula. The latter three weeks focused on co-design. Each co-design team primarily worked on curricula synchronously on Zoom or asynchronously with virtual check-ins, plus attended a few whole-group synchronous meetings for feedback and reflection on their curricula and pedagogy.

Analysis and Findings

To identify what was beneficial about the design of CTSI, we analyzed responses to a post-PD survey question that asked, "What did you learn from CTSI?" (n = 11). Most teachers mentioned that they learned how to integrate CT into their classroom. Some teachers named specific CT tools that they learned about during the four-day introduction and integrated into their units, e.g., "I learned how to use [three CT tools], how to integrate such models into a content-heavy, nuanced unit, and a lot about collaboration." Others described learning about CT and their content through co-designing their curriculum:

> I was able to develop a new unit around infectious diseases, this led to a lot of content knowledge about particular diseases, as well as CT knowledge of how to model and think about these diseases. In working with my team I was able to break down specific knowledge points for kids to figure out and develop models to help them do that.

Others mentioned multiple aspects of CTSI and how their perspective on pedagogy changed:

> OMG- I have learned to embrace co-design [...] the basics of coding and manipulating code [...] I also learned that my science pedagogy is in need of a shift, or rather a new lens to look at science through- that of CT. I am grateful to be energized by all the possibilities and potential accomplishments this will translate into for my students.

Contributions

Our findings suggest that workshops and co-design experiences helped teachers learn about CT and design CT curricula for their science classrooms. These experiences helped teachers not only learn about programming, coding, and CT-related practices, but also gain science content knowledge and rethink their pedagogy. Such teachers are likely more able to support students in CT practices (Paper 2), make lasting changes to their teaching practices (Paper 3), and increase students' confidence in using CT practices to learn science (Paper 5).

Paper 2: Teachers' Sensemaking of CT Integration and Pedagogical Approaches

Subject/Problem

Integrating computational thinking (CT) into science curriculum requires teachers to understand what CT integrated learning activities are and how to support student learning in those activities. This study focuses on a professional development activity for teachers designed to understand CT integration and pedagogical approaches to teach CT integrated curricula. We call this lesson, Lesson 0: Introduction to Computational Modeling (link blinded). The goal of Lesson 0 is to provide an introduction to CT activities in the context of science curricula (Authors, 2020b). This paper analyzes how science teachers made sense of CT practices and pedagogy and how those understandings helped them envision CT-integration in their own classrooms.

Design and Procedure

During the first day of CTSI (see Paper 1, Table 1), 11 high school STEM teachers engaged in Lesson 0 as an introduction to CT. Teachers completed Lesson 0 in groups representing physics, biology, chemistry, math, and environmental science content areas. We recorded and transcribed three breakout discussion sessions from each group.

Lesson 0 involves using, modifying, and debugging computational models to simulate a forest fire (see: http://tinyurl.com/netlogofire; Wilensky, 1997). Users also collected and analyzed data related to fire spread. Teachers were instructed to engage in all Lesson 0 computational activities as students, with the exception of the final two questions of each session: 1) "List computational activities that students would engage in as they go through the questions on this page. Explain how they are expected to participate in those activities." and 2) "How would you support student learning through computational activities integrated into this page?" To understand teachers' sensemaking of CT in a science context, we analyzed their discussions of these two questions and coded their utterances for CT practices and pedagogy:

Codes for *CT*. Utterance about CT involves at least one of the following practices (Weintrop et al., 2015): computational modeling and simulation practices, computational data practices, and/or computational problem solving practices.

Codes for *Pedagogy*. Utterance about pedagogy involves mention of student engagement in the activity and ways to support their learning.

Three coders coded all utterances (n = 120) for CT and Pedagogy with high reliability (Cohen's kappa > 0.7). Disagreements were resolved through discussion. We used distributions of the frequency of CT and Pedagogy utterances to identify teams for further analysis. Biology and physics groups showed more frequent engagement with CT and pedagogy. Below, we share how teachers discussed CT and pedagogy in their groups, using teacher pseudonyms.

Findings and Analysis

Teachers found it difficult to imagine what CT might look like in disciplinary contexts at the start of Lesson 0. For example, one physics teacher, Peter, shared, "I'm having a hard time with the word computational because it means collecting data and getting some numerical data. And none of that is visible in these models. And so how do you expect the students to do some computational thinking?" Here, Peter struggles to draw a connection between Lesson 0 and CT in his own physics classroom.

However, as the lesson progressed, participating in computational activities as learners helped teachers envision pedagogical supports that they would design and enact. After exploring the forest fire computational model, a biology teacher, Bridget, noted, "I would ask a lot of questions too because students, um, special ed students could try it once and not try to move those, you know, the density or the speed of the simulation. So I would ask those questions." This demonstrates how Bridget connected computational activities regarding *modeling and simulation* from Lesson 0 to her specific classroom context to support student learning of science content. Paul, a physics teacher, also talked about supporting learning with CT tools in the context of his classroom. He said, "I would have them graph density, you know, as a function of percent burned. And I would actually just also say what kind of you know what kind of statistical variation you have for, you know, 42%?" This showed how Paul used his experience as a student in Lesson 0 to think about how to engage students in *data visualization and analysis*.

Contributions

Findings from this study provide insight into how teachers made sense of CT content and pedagogy through engaging in CT activities themselves. Curricular reforms, such as CT integration, require teachers to understand the processes of integrating CT in specific curricular contexts and practices to support student learning. Evidence from this study suggests that participating in CT activities through Lesson 0 helped teachers to identify how they might integrate CT in their own science classrooms.

Paper 3: Teachers' Perceptions of the Contribution of Computational Thinking to Science and Math Classrooms

Subject/Problem

This paper reports on high school STEM teachers' perceptions of the contribution of CT to their classrooms following the first CT-STEM co-design summer institute, CTSI 2019. Most teachers have little knowledge of the skills involved in CT and the ways in which these skills can be incorporated in the classroom (Chang & Peterson, 2018; Fessakis & Prantsoudi, 2019; Sands et al., 2018). Although teacher development programs have proven effective in promoting CT-related content knowledge and pedagogical content knowledge (Bower et al., 2017; Haines et al., 2019; Kong et al., 2020; Morreale & Joiner, 2011; Yadav et al., 2014), this change may be limited, as such programs are relatively short (often a week-long or less).

As an extensive four-week professional development, CTSI may be able to help teachers gain CT knowledge that translates to their classroom. Thus, we use a qualitative analysis of the participating teachers' exit interviews to answer the following research question: What are STEM teachers' perceptions of the contribution of CT to teaching and learning after CTSI?

Design and procedure

We used post-workshop semi-structured interviews to collect data from CTSI teachers (N=8). Interviews lasted between 30-50 minutes, during which the teachers were asked general and domain-specific questions about their perceptions of CT; beliefs on teaching; existing and new pedagogic content knowledge; perceptions of learning from the workshop; and influence of domain of practice. We used the Direct Content Analysis method (Hsieh & Shannon, 2005) to analyze the interview scripts, with variables derived from ISTE Standards for Educators (International Society for Technology in Education, 2017).

Analysis and Findings

Four main themes regarding teachers' perceptions about CT integration emerged in our analysis. We discuss each theme below with example quotes from teachers, using pseudonyms.

***Pedagogy**: CT Enables New and Effective Ways for Teaching and Learning, yet Assessment Remains Traditional.* Following CTSI, teachers became aware of different ways in which CT could change teaching and learning; the most prominent ways this change could be manifested is by introducing students to coding, engaging them with more open-ended experimentation, and acquiring them with tools relevant to 21st century skills. As such, teachers described CT as enabling "students [to be] more involved in the design and collecting phase of experimentation" (Phillip), allowing "more experiences where students design their own experiments" (Peter). However, when asked about assessing the new CT-enriched units, teachers mentioned traditional assessment methods, relying on tests and class discussions for assessing content knowledge and not CT knowledge: "I'm going to give them the same test that they had before" (Betty).

***Authenticity**: CT Provides Opportunities for Different, Suitable, Authentic Learning*. Interacting with the CT integrated activities made teachers appreciate the way this kind of learning helps students familiarize themselves with scientific inquiry as scientists practice it; "We know that's what scientists do in the field to give us results. So I think students should be seeing that process in the classroom" (Brenda). Teachers indicated that engaging with CT-based models may help students deeply understand simulated phenomenon; a teacher, reflecting upon his students having "a hard time getting to understand why the [acceleration] graph is the way it is" stated that visualization using a computational model may help "reinforcing what [the] philosophy [of] acceleration [is]" (Phillip).

Equity*: CT Gives Students Agency to Discovery and Will Help them Become Active Citizens.* The teachers claimed that their students should change the way they learn because it will help them in their future world. As one of the participants put it, the value of incorporating CT in the science classroom is "to get [the students] thinking about that now and how that skill could serve them in the future" (Carrie). Teachers perceived CT integration as an opportunity to facilitate student-centered exploratory learning: "[Students] are going to have an influence on how are we going to get the information from a […] set of data, and they're going to have to tell it what to do" (Philip).

Broadening Participation*: CT Encourages Collaboration Among Teachers Within Schools.* Participating teachers understood that developing effective CT-enhanced units requires multiple people with different expertise. This led them to think of collaborating with their peers in a way that would impact students' learning; one of the teachers mentioned sharing "vertically within our school," which "gives more of a nice flow between courses" (Brenda).

This understanding made teachers consider initiating and leading such collaborations; "I was wondering if maybe we could think about doing [...] a workshop with some other teachers from our department. […] It's a time for that discussion" (Carrie).

Contributions

By positioning teachers as both CT learners and equal collaborators to design CT-enhanced curricula (Paper 1), we see evidence of teachers re-evaluating and expanding their ideas of how technology and CT can positively impact them and their students in ways that can transform their classrooms. The four main themes found (pedagogy, authenticity, equity, and broadening participation) align with some of the core ideas of incorporating CT in STEM classrooms (Weintrop et al., 2016). The evidence that our findings bring to these ideas is encouraging from at least two points of view. First, the current research highlights the importance of well-designed professional development programs with longer duration for promoting a meaningful change in teachers' perspective on CT. Second, it emphasizes that CT may be associated with various aspects of teaching and learning in science and mathematics. As such, it provides opportunities for broadening participation in deep learning of these disciplines, supporting students with skills that will help them to become active citizens. That is, CT may help decrease societal gaps at large (Tran, 2018).

Paper 4: Identifying Evidence of Student Engagement in CT via Automated Response Analysis

Subject/Problem

A main argument for the integration of CT into science classrooms is that of *authenticity*– that science should engage students in real science, which is becoming increasingly computationally dependent. While many efforts have been made to design new curricula to incorporate authentic learning experiences by integrating CT into STEM classrooms, little work has been done in using student data to identify examples of engagement in authentic CT science practices (Grover et al., 2014; Swanson et al., 2019; Arastoopour Irgens et al., 2020; Tang et al., 2020). How can we find evidence of students engaging in authentic scientific practice in CT infused classrooms? In this paper, we discuss an automated coding process that identifies evidence of student engagement in CT practices.

Design and Procedure

We base our analysis on the Computational Thinking in Mathematics and Science Practices Taxonomy (Weintrop et al., 2016), which consists of twenty-two specific practices that are divided into four categories: data, modeling and simulation, computational problem solving, and systems thinking. For this analysis, we exclusively focus on the first category, data practices, in order to prototype our computational methods.

We use response data from 51 students who participated in a two-week CT integrated biology unit focused on experimental design. In the unit, students interact with both physical and computational models, while answering multiple-choice and free-response questions on an online platform. We limited our corpus to the 84 free-response questions that students responded to over the course of the curriculum, resulting in a corpus of 4436 responses. To identify student responses that included evidence of students engaging in data practices, we used the taxonomy to derive what a main key – sub-key search structure. The program then searches student responses for instances of the the main key, in the case of data practices was simply 'data,' and also a number of sub-keys are a stemmed verb derived from the taxonomy (e.g."collect" would allow finding "collection", "collecting", "collected", etc.). The sub-keys we used in this analysis are collect*, creat*, manipulat*, analy*, visuali*. These are then computationally searched for in each of the responses.

Analysis and Findings

Using this strategy, we identified 187 responses that contained the main key and at least one sub-key for data practices. To verify our autocoder, four human coders coded a randomized subset of 50 responses in the corpus to identify evidence of a data practice: any responses that indicate students' engagement in collecting, creating, manipulating, analyzing or visualizing data with an explicit or implicit involvement of a computational tool, plus responses that mention automated ways related to data handling. Average Cohen's Kappa across coders, was 0.717, which indicates moderately high agreement.

Table 1. Sample responses identified by the autocoder.

By using those tools you are able to gather your data very quickly and have all the mean and standard deviations in front of you ready to go. The data collecting process is much faster and you can prove your experiment right or wrong very fast as well.
I used google sheets to collect and organize my data. By doing this, I was able to calculate different values such as mean, std. deviation, std. error and chi square. When all my data was collected, I was able to make a graph to visually show my findings.
By creating a larger sample size of 50 subjects, I am making the data more accurate. After 1000 ticks, you can see how the moist environment is preferred by the subjects. 41 subjects entered the moist chamber and 9 entered the dry chamber.
The data is recorded quickly and completely accurately; there is absolutely no room for human error in the computationally automated data collected tool

Contributions

This type of automated response identification is a first step in understanding students' engagement in authentic CT science practices in the classroom to better understand how our designs evoke student engagement in CT. Our automated coding process is generalizable in order to enable future work in identifying evidence of other CT practice engagement. Additionally, we plan to apply the autocoder to the question texts. This question coding, paired with the coding of the responses, could be utilized to identify particular question archetypes that might effectively trigger student responses that include evidence of CT engagement. Finally, because the process is automated, such analysis can be completed on-the-fly, meaning teachers could flag student responses that contain rich CT engagement and use them as in-class discussion points. Alternatively, teachers could use the autocoder as a first pass to identify possibly rich student responses to assess for CT proficiency.

Paper 5: Students' Attitudinal Change After Participating in a CT integrated Biology Unit

Subject/problem

One of the arguments for integrating CT into science curricula is to reach a wider audience, especially women and students underrepresented in computational fields (Weintrop et al., 2016). Since a higher number of students take science classes as compared to students who take Computer Science classes, CT integration in science would provide more students with opportunities to learn CT. Such opportunities could potentially impact students' attitudes towards their interest in and ability to use computational tools in their educational and professional lives. In this paper, we investigate students' attitudinal changes after they participated in a CT integrated biology unit.

Design/procedure

~~Table 1: A subset of survey statements used for quantitative analysis of students' attitudinal change regarding their confidence and affect.~~

	Survey statements
1	I feel confident in my ability to use computational models to learn scientific phenomena.
2	I feel confident in my ability to use computational models to test a hypothesis.
3	I feel confident in my ability to create my own computational model.
4	I enjoy learning science using computational models.
5	I feel confident in my ability to design a method to analyze data using computational tools.
6	I feel confident in my ability to understand and create different ways of data visualization.
7	I feel confident in my ability to use computational tools to analyze data.
8	I enjoy finding patterns in data using computational tools.
9	I enjoy using computers to learn about science or math.
10	I think I'm good at science and technology.

The data used in this paper is from high school students that participated in a two-week long CT integrated biology unit. We collected 41 complete sets of pre/post paired responses. The survey featured Likert scale items (1 = Strongly Disagree to 5 = Strongly Agree) about students' attitudes regarding science, science learning, and use of computational tools for science. We tested pre/post differences in a subset of survey statements pertaining to their confidence in using computational models and other tools (Table 1) using the Wilcoxon-Pratt signed-rank test.

We also conducted principle component analysis (PCA) using k-means clustering to identify students with positive attitudinal changes (Figure 1b). Then, we qualitatively analyzed these students' responses to open-ended survey questions regarding what they learned and enjoyed in the unit.

Findings and analysis

There were statistically significant changes ($p < 0.05$) in students' attitudes regarding their confidence in using computational models and performing data analysis using computational tools (items 1, 2, 3, 5, 6, and 10) (Figure 1a). This indicates their confidence in CT practices increased after their participation in the unit. However, there were no significant changes related to students' perceptions of enjoyment of using CT tools in the context of science (items 4, 8 and 9). Also, there was statistically significant negative change (item 10) regarding students' self efficacy about being good in science and technology. This may be because this unit could have made students aware of various ways of using technology in science that they were not familiar with.

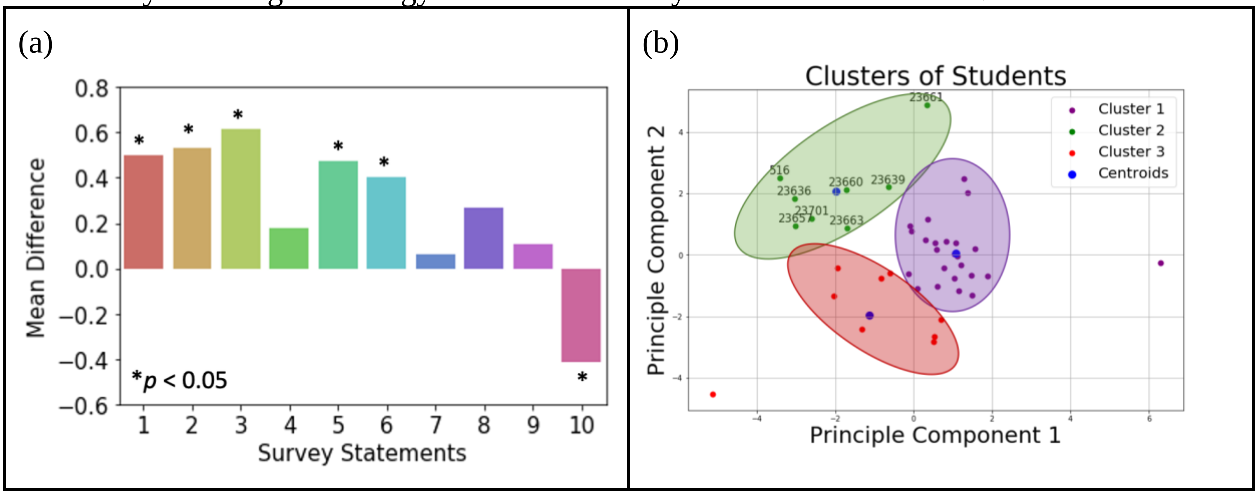

Figure 1: (a) Pre/post comparison of survey statements regarding confidence and affect (statements in table 1) (b) Principal component analysis of students' attitudinal change. The green cluster has all the students that have mean positive difference in the pre/post comparison

Student responses to descriptive questions corroborate our quantitative findings. Students who considered themselves "not techy" felt confident and proud of their participation. A white female student wrote: "*I am not a very techy person, so I was proud of myself because I was able to figure everything out.*" Another student (asian female) wrote specifically about the computational coding part in the unit: "*I enjoyed challenging myself with coding since I am not very good with technology.*" A third student (white female) wrote, "[this experience will] *definitely benefit me in my future; this applies to whether or not I choose a career in science or math."*

Contributions

Our analysis of students' attitudinal change after they participated in a CT integrated biology unit showed a statistically significant increase in students' confidence in their abilities to use computational modeling and data practices. Qualitative analysis of responses of students with positive attitudinal change revealed how some female students who considered themselves "not techy" enjoyed the unit and felt proud of their learning experience. This work supports one of the core arguments of CT integration (Weintrop et al., 2016) that it could broaden participation of student groups that are typically underrepresented in computational fields.

Organization of the Paper-set and its Contribution to the Interests of NARST Members

Integrating Computational Thinking into science curricula is both an opportunity and a challenging problem for science education researchers, teacher educators, and administrators. It is an opportunity to provide authentic and pedagogically effective learning experiences to a wider set of students. However, designing CT-integrated units that teachers can use effectively in their classrooms to support and assess student learning has been a challenging problem in the field of Computational Thinking Education (Angeli & Giannakos, 2020). Our paper set presents five studies from a multi-year CT integration project that addresses two specific aspects of this challenge, namely, professional development of teachers and assessment of student learning and attitudinal change. With the paper set format, we can contextualize the papers' findings within the larger project, making the results and implications clear and meaningful for session attendees.

The Co-design approach discussed in Paper 1 will be of interest to science education researchers who work with teachers on implementing curriculum reforms. The analysis of CTSI and the introductory CT integrated Lesson 0 in Paper 2 will help NARST teacher educators and researchers to design similar activities for helping teachers integrate CT into science contexts. Our presentation will provide additional examples of how specific sessions helped teachers learn about CT and how teacher teams co-designed their curriculum. Analysis of changes in teacher perceptions after they participated in the CT-STEM PD in the third paper will be of interest to teacher education researchers.

While there are increasing efforts to integrate computational thinking across science courses in K-12 education in the NARST community and outside, there are still many unanswered questions on how to assess student engagement in these CT practices in the classroom from a) a researcher's perspective and b) a teacher's perspective. The last two papers in the set are about assessing student engagement in CT and their attitudinal changes. In paper 4, we present an easy-to-understand computational analysis framework to identify student engagement in CT practices. Attitudinal changes in female students, especially regarding confidence in using CT tools and considering career prospects in computational fields will interest the NARST members who focus on equity in the context of science education. Our presentation should appeal to both researchers and educators interested in devising ways to support teacher learning and assess student learning with CT integrated curricula.

References

Angeli, C., & Giannakos, M. (2020). Computational thinking education: Issues and challenges. Computational Thinking Education: Issues and Challenges, 105(4).

Arastoopour Irgens, G., Dabholkar, S., Bain, C., Woods, P., Hall, K., Swanson, H., Horn, M., & Wilensky, U. (2020). Modeling and Measuring High School Students' Computational Thinking Practices in Science. Journal of Science Education and Technology, 29(1), 137-161.

Authors and colleagues (2003).

Authors and colleagues (2009).

Authors and colleagues (2018a).

Authors and colleagues (2018b).

Authors and colleagues (2019).

Authors and colleagues (2020a).

Authors and colleagues (2020b).

Blikstein, P., & Wilensky, U. (2010). MaterialSim: A constructionist agent-based modeling approach to engineering education. In Designs for learning environments of the future (pp. 17-60). Springer US.

Bower, M., Wood, L. N., Lai, J. W. M., Howe, C., & Lister, R. (2017). Improving the computational thinking pedagogical capabilities of school teachers. Australian Journal of Teacher Education, 42(3), 53–72.

Chang, Y.-H., & Peterson, L. (2018). Pre-service teachers' perceptions of computational thinking. Journal of Technology and Teacher Education, 26(3), 353–374.

Dubovi, I., Levy, S. T., & Dagan, E. (2017). Now I know how! The learning process of medication administration among nursing students with non-immersive desktop virtual reality simulation. Computers & Education, 113, 16-27.

Fessakis, G., & Prantsoudi, S. (2019). Computer science teachers' perceptions, beliefs and attitudes on computational thinking in Greece. Informatics in Education, 18(2), 227–258.

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. Educational Researcher, 42(1), 38-43.

Grover, S., Cooper, S., & Pea, R. (2014). Assessing computational learning in K-12. In Proceedings of the 2014 conference on Innovation & technology in computer science education (pp. 57-62).

Haines, S., Krach, M., Pustaka, A., Li, Q., & Richman, L. (2019). The effects of computational thinking professional development on STEM teachers' perceptions and pedagogical practices. Athens Journal of Sciences, 6(2), 97–122.

Horwitz, P., Gobert, J. D., Buckley, B. C., & O'Dwyer, L. M. (2010). Learning genetics from dragons: From computer-based manipulatives to hypermodels. In Designs for learning environments of the future (pp. 61-87). Springer, Boston, MA.

Hsieh, H.F. & Shannon, S.E. (2005). Three approaches to qualitative content analysis. *Qualitative Health Research, 15*(9), 1277-1288.

International Society for Technology in Education. (2017). ISTE Standards for Educators. Journal of Research in Technology in Education.

Jeong, H. & Hmelo-Silver, C. E. (2016). Seven affordances of CSCL Technology: How can technology support collaborative learning. Educational Psychologist, 51, 247-265.

Kitano, H. (2017). Biological Complexity and the need for Computational Approaches. In Philosophy of Systems Biology(pp. 169-180). Springer, Cham.

Kohn W (2003). Nobel lectures, chemistry 1996–2000. World Scientific Publishing Co, Singapore, p 213.

Kong, S. C., Lai, M., & Sun, D. (2020). Teacher development in computational thinking: Design and learning outcomes of programming concepts, practices and pedagogy. Computers and Education, 151, 103872.

Levy, S. T., & Wilensky, U. (2009). Students' learning with the Connected Chemistry (CC1) curriculum: navigating the complexities of the particulate world. Journal of Science Education and Technology, 18(3), 243-254.

Morreale, P., & Joiner, D. (2011). Changing perceptions of computer science and computational thinking among high school teachers. The Journal of Computing Sciences in Colleges, 26(6), 71–77.

NGSS Lead States. (2013). Next Generation Science Standards: For states, by states. Washington, DC: The National Academies Press.

Pople J (2003). Nobel lectures, chemistry 1996–2000. World Scientific Publishing Co, Singapore, p 246.

Sands, P., Yadav, A., & Good, J. (2018). Computational thinking in k-12: In-service teacher perceptions of computational thinking. In M. S. Khine (Ed.), Computational thinking in the STEM disciplines: Foundations and research highlights (pp. 151–164). M. S. Khine.

Sengupta, P., & Wilensky, U. (2011). Lowering the learning threshold: Multi-agent-based models and learning electricity. In Models and modeling (pp. 141-171). Springer, Dordrecht.

Severance, S., Penuel, W. R., Sumner, T., & Leary, H. (2016). Organizing for teacher agency in curricular co-design. *Journal of the Learning Sciences*, *25*(4), 531-564.

Swanson, H., Anton, G., Bain, C., Horn, M., & Wilensky, U. (2019). Introducing and assessing computational thinking in the secondary science classroom. In Computational Thinking Education (pp. 99-117). Springer, Singapore.

Tran, Y. (2019). Computational thinking equity in elementary classrooms: What third-grade students know and can do. *Journal of Educational Computing Research*, *57*(1), 3-31.

Wilensky, U. (1999). NetLogo. http://ccl.northwestern.edu/netlogo/. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.

Wilensky, U., Brady, C. E., & Horn, M. S. (2014). Fostering computational literacy in science classrooms. Communications of the ACM, 57(8), 24-28.

Windschitl, M., Thompson, J., Braaten, M., & Stroupe, D. (2012). Proposing a core set of instructional practices and tools for teachers of science. Science education, 96(5), 878-903.

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining Computational Thinking for Mathematics and Science Classrooms. Journal of Science Education and Technology, 25(1), 127–147.

Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S., & Korb, J. T. (2014). Computational thinking in elementary and secondary teacher education. ACM Transactions on Computing Education, 14(1).