

# Developing Implementation Measures for K-12 Computer Science Curriculum Materials

Daisy Rutstein, Yuning Xu, Kevin McElhaney, Marie Bienkowski  
SRI Education  
SRI International  
Menlo Park, CA, USA

[daisy.rutstein@sri.com](mailto:daisy.rutstein@sri.com), [yuning.xu@sri.com](mailto:yuning.xu@sri.com), [kevin.mcelhaney@sri.com](mailto:kevin.mcelhaney@sri.com), [marie.bienkowski@sri.com](mailto:marie.bienkowski@sri.com)

## ABSTRACT

As K-12 computer science (CS) education initiatives scale throughout the U.S., researchers seek to understand the context-specific relationships between CS instruction and student learning. Evaluation of instruction requires valid measures of curriculum implementation. We have developed measures for identifying conditions for successful implementation of an introductory high school computer science curriculum along two-dimensions: teaching quality and curriculum enactment. Additionally, we have defined three types of instructional strategies for teaching quality. Quantitative and qualitative data were collected from 53 teachers through surveys and interviews. Data were aggregated and integrated to derive scaled measures for the instructional strategies and curriculum adaptation, and implementation measures were correlated with student end-of-unit assessment data. We found potential factors that can enhance or impede the successful implementation of CS curriculum materials, and we have identified several broad issues associated with scaling up CS curricular implementation.

## CCS CONCEPTS

Social and professional topics~K-12 education, Social and professional topics~Student assessment

## KEYWORDS

Computational Thinking, Curriculum Implementation, Measurement

## ACM Reference format:

Daisy Rutstein, Yuning Xu, Kevin McElhaney and Marie Bienkowski. 2018. Developing Implementation Measures for K-12 Computer Science Curriculum Materials. In *Proceedings of ACM SIGCSE conference (SIGCSE'19)*. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/3287324.3287424>

## 1 Introduction

The scope of K-12 CS curriculum development and adoption have greatly expanded in the U.S. over the last decade. Researchers and advocacy groups have been successful on several fronts, such as: creating the K-12 CS Framework [14] to inform

the articulation of standards; promoting the inclusion of semester-long or full-year standalone CS courses in high school course offerings; integrating CS concepts and practices into K-12 mathematics and science classes [21]; and launching a successful Advanced Placement course that demonstrates potential to diversify the STEM workforce pipeline [2, 3]. At the same time, many K-12 CS courses are taught by teachers in departments such as math, science, and business [7] due to teacher shortages in STEM subjects, the newness of CS as a K-12 discipline, and schools' and districts' efforts to make CS courses available to their students [23]. These challenges highlight the need for researchers to examine (1) what constitutes high quality CS teaching and successful curriculum enactment and (2) the extent to which high teaching quality and successful curriculum enactment result in positive CS learning outcomes.

K-12 CS teaching may exhibit wide variation due to the diverse backgrounds and levels of experience teachers possess. Moreover, there is also variation in how schools, districts, and states include CS as part of their curricular programs. Characterizing how CS curricula are taught and enacted by different teachers and in different education contexts is centrally important to designing CS curriculum materials and teacher professional development (PD) opportunities [1]. Designers of curriculum materials such as Exploring Computer Science (ECS) [10] provide guidance and supporting resources for teachers to help them successfully enact CS lessons. The CS education research community needs ways to determine whether teachers' teaching strategies and enactment decisions are consistent with the underlying principles of the curriculum materials.

Prior work has addressed the development and validation of measures of computer science learning outcomes, such as students' proficiency with computational thinking concepts and practices [22]. In this paper, we extend that work to the measurement of curriculum implementation, focusing on ECS. We have defined a set of constructs underlying curriculum implementation, used them to develop measures of implementation, and conducted a study to characterize the relationships between aspects of curriculum implementation and student learning outcomes. To examine these relationships, we performed an analysis integrating attributes of teachers and their associated learning contexts, teachers' teaching strategies, and the fidelity of the curriculum enactment [e.g., 17, 20].

## 2 Background

ECS is a comprehensive school reform effort for computing education consisting of three elements: curriculum materials, teacher PD, and policy efforts [10]. It was developed with the goal to broaden the participation of traditionally underrepresented students in precollege computer science. The curriculum is made up of different units, each focusing on different aspects of computing. For example, the first unit focuses on human and computer interaction, while the second unit focuses on problem solving. The design of ECS is guided by the underlying principles of inquiry, equity, and collaboration. Inquiry-based learning constitutes an appropriate guiding principle for the design of ECS activities, as it emphasizes core concepts learned through engagement with disciplinary practices, student-centered teaching approaches, and open-ended, continuous, and formative assessment. [12].

Second, ECS is based on an *equitable* learning model. The curriculum activities reflect the notion that success in CS is attainable by and needed for all students regardless of their cultural, linguistic, or economic backgrounds. To achieve this aim, ECS strives to provide students with instructional experiences that are culturally relevant and meaningful to them and emphasize the impacts of computer science on contemporary society. This approach draws from the “funds of knowledge” perspective [15] and the national Algebra project [16], which include connections to community-based topics and address students’ beliefs about their ability to learn mathematics.

Finally, collaborative work, peer-communication, and group discussion underpin many ECS activities. Having students work together provides the benefits of engaging with peers for learning and engagement. Activities are designed to encourage students to work in a variety of collaborative settings including elbow partners, pair-programming, and group research projects. This collaboration encourages conversations around computing topics.

In support of an initiative to engage K-12 teachers in rigorous secondary computer science teaching, the National Science Foundation and industry supporters funded the preparation and administration of several surveys of secondary CS teachers and teacher PD providers to characterize the CS education landscape [7]. The study found that more than half of teachers surveyed were not full-time CS teachers, and that they most often taught mathematics, technology skills, or science. Some teachers surveyed were drawn from business, information technology, and library science, and many of those surveyed were new to computer science. More than half were the sole CS teacher in their school, and 87% reported having an advocacy role for CS in their schools. The diversity of teacher backgrounds and ways that CS was incorporated into school and district educational infrastructure presents challenges to consistently implementing CS curriculum materials. Goode and Margolis [10] also reported the difficulties teachers face in implementing new curriculum based on their beliefs about (1) how and what students can learn, (2) the utility of inquiry approaches to teaching, (3) nature of science, and (4) the importance of students’ prior knowledge. Further work [12] demonstrated that second-year teachers are less

concerned with learning CS content, express increased readiness to implement pedagogical changes, and benefit from reflection on their teaching practice. These findings suggest context factors that affect the implementation of ECS.

Based on these studies, we identify five important factors that influence teachers’ teaching practices and customizations to curriculum sequences and activities:

1. *Teacher background in CS*: knowledge of CS concepts and familiarity with CS pedagogy;
2. *State secondary school policies*: Graduation requirements, information technology literacy requirements, career technical education contexts;
3. *Regional workforce concerns*: Input from local industry;
4. *District/school CS context*: Initiatives that align with teaching; reform initiatives that could help or hinder (including those linked to accountability); afterschool programs in related topics such as robotics or technology; availability of technology for the classes and availability of AP CS at the school.
5. *Student background and prior achievement*: Socioeconomic status, gender, prior exposure to computing, access to technology, rigor of their academic program, grade in Algebra I.

We frame these factors in terms of the human, social, and technical capital that teachers have at their disposal [19, 4]:

1. *Human capital*: A teacher might believe that they have taught themselves to program and studied the computational thinking practices, or they may view CS as aligning with STEM teaching initiatives at their school. Teachers will have different experiences with ECS PD and subsequent in-school support.
2. *Social capital*: This element includes the expertise that individuals can access to accomplish goals. For example, a teacher may belong to the local CSTA chapter or an online community.
3. *Technical/resource capital*: For CS, access to computers and the Internet with adequate performance can be important. Although ECS uses “unplugged” activities for teaching, many of the activities require the Internet and a programming environment. Another important resource is time: it would be unusual for a secondary CS teacher to be teaching only CS, so ECS teachers will have competing demands on their time.

These factors can shape the adoption of a new course like ECS and can influence the pedagogical approach teachers take once it is in place. Our study looked for evidence of important factors of ECS in enactment: inquiry, equity, and collaboration. Our specific research questions are:

1. What factors predict implementation of the inquiry-, equity-, and collaboration-based instructional strategies supported in the ECS program?
2. What factors predict the extent to which teachers adapt the ECS curriculum materials?
3. What factors enhance or impede the successful implementation of ECS?
4. How does curriculum implementation relate to student outcomes?

### 3 Methods

#### 3.1 Modeling Curriculum Enactment and Teaching Quality

Researching the multiplicity of factors that can influence how a curriculum is implemented over a semester or a year and identifying the characteristics of the teaching of computer science is a complex endeavor. As researchers, we look for a way to make sense of the extensive set of factors and their influence relationships.

As mentioned above, we defined two high-level factors we believed would relate to student outcomes: teaching quality (which we defined as teachers' use of inquiry-, equity-, and collaborative-based teaching strategies) and curriculum enactment (which we defined as related to lessons modified, skipped, and added). We also hypothesized that there are many factors that could influence the degree of teaching quality and curriculum enactment. These factors include the attributes of the learning context (e.g., district, school and classroom characteristics) and teacher attributes (including the teacher's beliefs about the curriculum, the typical strategies the teacher uses in their classrooms, and the teacher's background characteristics.) Figure 1 shows our model for relating teacher and learning context attributes, aspects of implementation, and student outcomes.

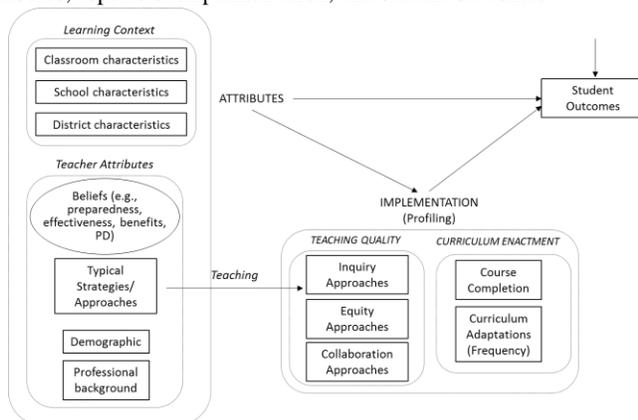


Figure 1: Overall Model for Factors that Could Influence Student Outcomes

Importantly, here we highlight the distinction between *teacher* quality and *teaching* quality [9]. This distinction aims to convey that how teachers employ teaching practices and teaching moves are open to change, improvement, and innovation over time [8, 13], and that teaching decisions are not static attributes of teachers. Our teaching quality measures target the teachers' teaching decisions, rather than characteristics of teachers themselves.

We further specified each of the constructs included in the model according to a set of variables, each of which we hypothesized would have either a positive or negative relationship to student learning outcomes. Measures were developed to capture

these variables. Further information about how this model was operationalized is given below.

#### 3.2 Operationalizing the Model

We developed a set of six surveys: one background survey (administered at the start of the year), one teacher PD survey (administered soon after the teacher PD workshop), and one unit survey for each of the first four ECS units (administered after each unit was taught). We focused on the first 4 units as those are considered the fundamental units (Unit 1 is about human/computer interaction, Unit 2 is about problem solving, Unit 3 focuses on web-design, and Unit 4 on programming). We also developed a student survey that asked questions around the students' demographic background and background with computer science concepts. The questions on the surveys were designed to measure the constructs specified in our model.

Fifty-three teachers completed at least one of the surveys and 34 teachers completed the background survey and at least one unit survey. Table 2 below presents the sample sizes for the teacher surveys and student assessments.

Table 2: Sample Sizes for Teacher Background Survey, PD Survey, End-Unit Surveys, and Student Assessments.

Data Collection Activity	Teacher N	Student N
Background Survey	36	-
Professional Development Survey	11	-
Pre-assessment	-	1133
ECS Unit 1	38	981
ECS Unit 2	37	892
ECS Unit 3	27	811
ECS Unit 4	28	586
Cumulative Assessment	-	780

Once the survey data were collected and coded, we re-examined the results of the survey to create a set of variables that could be included for each of the constructs in the model. These variables were generated using both a top-down approach (using our initial theoretical model as a guide) and a bottom-up approach (using statistical analysis and review of question responses to determine which questions had sufficient variability in responses to be included in the model). The final variables used for analysis can be either a composite variable that combines a set of survey questions or a variable based on a single survey question. Table 1 includes examples of survey questions related to the instructional strategies.

In addition to the set of selection answers on the surveys, there was a set of open-ended responses. These questions asked teachers to elaborate on aspects of their teaching, in particular to describe modifications they made to the curricular materials, or additional activities they used in the courses. These responses were then coded as to whether they indicated the teacher was engaging in one of the three types of teaching practices (inquiry, equity, or collaboration practices) or that they modified the curriculum.

The background survey for the teacher covered a variety of topics, including demographic information, prior CS experience, support for CS at the school level, beliefs about CS and ECS, and

their typical teaching strategies. The PD survey included questions relating to the teacher’s preparedness for teaching, and beliefs about the ECS curriculum.

So as not to overburden teachers with daily or weekly instructional logs, we administered a unit survey at the end of each unit, when teachers administering the end-of-unit assessment to their students. The end-of-unit surveys asked about their teaching assignment (unit 1 only), ECS classes, experience teaching the unit, modifying/skipping of topics, sense of preparedness to teach the unit, instructional activities taught, unit successes and challenges, availability of technology, and their overall ECS experience (units 3 and 4 only).

Table 1: Example survey questions for teaching quality factors

Inquiry-Based Practices
When teaching the unit, how often did you engage students in the following practice?
<ul style="list-style-type: none"> <li>Students develop their own solutions to problems</li> </ul>
Equity-Based Practices
Which of the following instructional successes did you encounter?
<ul style="list-style-type: none"> <li>Opportunities for students to connect computational thinking to cultural backgrounds</li> </ul>
Collaboration-Based Practices
When teaching the unit, how often did you engage students in the following practice?
<ul style="list-style-type: none"> <li>Students collaborate or work in teams</li> <li>Students present or communicate their ideas to peers</li> </ul>

In the next phase we defined the theoretical model. Here we articulated hypotheses about the relationships between the variables to create a model that addresses the research questions. For example, Figure 2 illustrates our hypothesis that the likelihood that the teacher will implement inquiry-based teaching approaches will exhibit a positive relationship (indicated with a +) with a teacher’s prior experience with inquiry and a negative relationship (indicated with a - sign) with number of non-inquiry-based (didactic) teaching approaches they use [5].

For research question 3, we defined successful implementation of the ECS curriculum as having higher instances of the approaches that are associated with teaching quality. We then categorized the types of modifications teachers reported according to whether they align with the pedagogy of the curriculum (e.g. adding activities that relate to the cultural background of the classrooms) or do not align (e.g. skipping activities). We then hypothesized whether these modifications relate positively or negatively to these variables. Here we included aspects that are often outside of the teachers control, such as the availability of technology (see Figure 3).

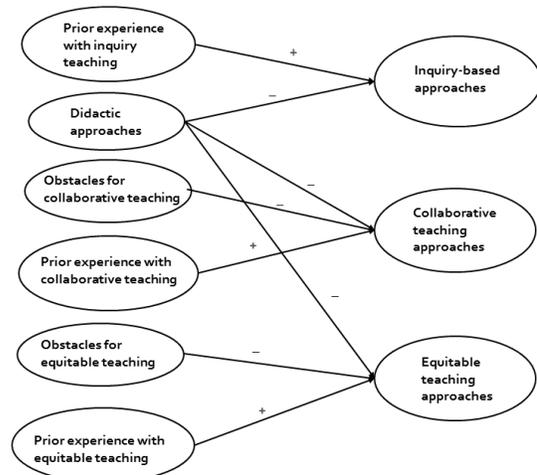


Figure 2: Theoretical model of the factors that predict teachers’ implementation strategies

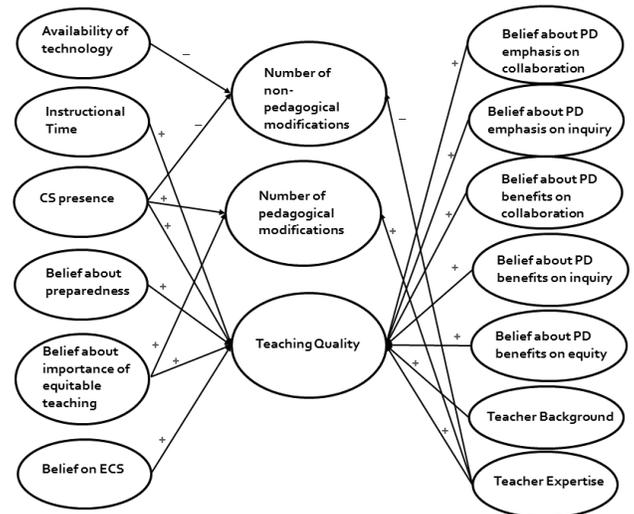


Figure 3: What factors enhance or impede the successful implementation of ECS?

Once we specified these models, we then tested them with the data. We started by determining if any of these variables had statistically significant correlations. We also examined the predictive capabilities of the variables in our theoretical model.

After we determined which factors might be predictors of the teaching quality and curricular enactment we examined the relationships among these variables and the student outcomes. The student outcomes were measured by four end-of-unit assessments, each given to the student at the end of instruction in the corresponding unit. We examined the relationships separately for each of the four unit assessments.

## 4. Results

### 4.1 Teaching Quality

#### 4.1.1 Inquiry-based instructional strategy (IBIS)

We define the measure for IBIS based on both teachers' level on inquiry practices and their evaluations on successful classroom promotion regarding inquiry. Frequency of practices is prioritized over successful promotions in evaluating the implementation of IBIS. Table 4 presents the five levels for IBIS from low to high and the distributions of teachers on these levels by unit.

Table 4: Teacher Distributions on the Defined Measure for Inquiry-Based Instructional Strategy.

Levels of Inquiry-Based Instructional Strategy	Number of Teachers			
	Unit 1	Unit 2	Unit 3	Unit 4
L1: No practice regardless of success	0	0	0	0
L2: Low frequencies of practice + few success	3	2	0	0
L3: Low frequencies of practice + more success	5	4	1	1
L4: High frequencies of practice + few success	12	7	8	5
L5: High frequencies of practice + more success	17	22	18	22
Total N	37	35	27	28

The results showed high IBIS levels across all ECS units, with units 3 and 4 having the highest levels. The differences across units indicate the transition from lower to higher levels for some teachers as they progressed from unit 1 to unit 4. Such changes are likely due to the different nature of the different unit topics and activities.

#### 4.1.2 Collaboration-based instructional strategy (CBIS)

As with IBIS, the measure for CBIS is also defined using the frequency of practices and the successful promotion of collaborative teaching. Table 5 presents teacher distributions on CBIS for the different units.

Table 5: Teacher Distributions on the Defined Measure for Collaboration-Based Instructional Strategy.

Levels of Collaboration-Based Instructional Strategy	Number of Teachers			
	Unit 1	Unit 2	Unit 3	Unit 4
L1: No practice regardless of success	0	0	0	0
L2: Low frequencies of practice + few success	4	4	4	5
L3: Low frequencies of practice + more success	1	3	5	4
L4: High frequencies of practice + few success	7	5	5	3
L5: High frequencies of practice + more success	25	23	13	16
Total N	37	35	27	28

High CBIS levels were found across all ECS units, but less so than IBIS. ECS units 1 and 2 had the highest CBIS levels. In contrast to IBIS, some teachers transitioned from higher to lower levels of CBIS as they progressed from unit 1 to unit 4. This trend might be due to the nature of units 3 and 4 where the focuses were on specific activities such as design and programming.

#### 4.1.3 Equity-based instructional strategy (EBIS)

In the unit surveys, one of the close-ended questions targeted equitable teaching strategies. This question asks if teachers were successful in promoting computational thinking in their ECS classroom by providing opportunities for students to connect computational thinking to personal interests, cultural backgrounds, community problems, or social issues. The table

below summarizes the percent of teachers who indicated successful promotion in each of the units. In all units, over half of the surveyed teachers indicated successful promotions. Unit 3 had a higher percent than the other units.

Table 6: Teacher Distributions on the Defined Measure for Equity-Based Instructional Strategy.

	Total Number of Teachers	% Teacher Indicated Successful Promotion
Unit 1	38	63.16%
Unit 2	37	64.86%
Unit 3	27	74.07%
Unit 4	28	64.29%

#### 4.1.4 Factors enhancing or impeding IBIS, CBIS, and EBIS

For each of the instructional strategies, we correlated the measure for the strategy and the variables for teacher attributes and learning context. The results are presented in Table 7 below.

Teachers' prior experience with inquiry were found to moderately correlate with their ECS IBIS level for the first unit they taught. The levels of implementing EBIS were positively related to teachers' beliefs regarding culture relevance but were negatively related to the number of years of teaching CS and ECS for specific units.

Table 7: Correlations Between Measures for Instructional Strategies and Variables for Teacher Attribute and Learning Context.

	IBIS	CBIS	EBIS
Teacher Attributes	Prior inquiry experience	.49*** (Unit 1)	
	Belief: ECS benefits some or all	.36 (Unit 2)	
	Belief: Culture relevance did not decrease rigor		.38* (Unit 1)
	Belief: Culture relevance improves outcomes		.47* (Unit 2)
	Number of years teaching CS		.51* (Unit 3)
	Number of years teaching ECS		-.59* (Unit 2)
	Industry experience		-.49* (Unit 2)
	Total instruction time	-.31 (Unit 1)	
	% of students with disability	-.39* (Unit 4)	

Note. Correlations smaller than .30 are not presented. \* $p < .05$ , \*\*\* $p < .001$ .

## 4.2 Curriculum Enactment

We operationalize curriculum enactment by examining the number of topics teachers modified, skipped, and added during each unit. Table 8 below shows the summary statistics for these three measures. On average, teachers made more modifications in units 1 and 3, and skipped more topics for units 3 and 4. More than a third of the teachers added their own topics in all units.

Table 8: Summary Statistics for Measures for Curriculum Adaptation

	Total Number of Topics	Average Number of Topics Modified	Average Number of Topics Skipped	Percent of Teachers Adding Any Topic
Unit 1 (N=37)	8	3.57	.19	45.16%
Unit 2 (N=34)	9	3.35	.32	46.67%
Unit 3 (N=27)	11	7.11	.81	34.62%
Unit 4 (N=28)	18	6.54	1.21	42.11%

Relationship between the levels of the instructional strategies and curriculum enactment were examined by correlating the

corresponding measures. Teachers with prior experience with focal ECS instructional strategies (e.g. IBIS, CBIS) appear to be more inclined to modify units, perhaps because those strategies often require teachers to customize materials, so they are comfortable doing so. If there is some kind of well-established CS program at the school as evidenced by CS teaching experience or other CS classes, teachers are more likely to implement lessons "as-is". Additional evidence is needed to help understand the underlying causes.

On the other hand, access to technology matters for how lessons are implemented. If teachers lack technology or experience software issues, they tend to skip or replace certain lessons. If teachers have access to technology, they tend to use the curriculum more "as-is".

#### 4.3 Teacher Implementation and Student Outcomes

We built hierarchical linear models (HLMs) to explore the impact of teacher implementation levels as well as the teacher and learning context attributes on student learning outcomes. At the end of each unit, students received a summative assessment that has been validated through a two-year pilot study. We used student scores on the end-unit assessment as an outcome measure, and measures developed for teacher implementation, teacher attributes, and learning context as classroom-level predictors for the HLM models. Student score on the pre-assessment was included as a student-level predictor in the model as well to account for variation at the student level. For each unit, we started with the student level predictors as defined by the theoretical model, then included the classroom level predictors and determined if there were interaction effects. This model was used to examine if the predictors were significant.

The results found students whose teachers reported higher unit IBIS or CBIS performed significantly better on the assessments for specific units. Higher level of CBIS was related to better performance on unit 1 (human-computer interaction) after accounting for the student- and classroom level covariates. Higher level IBIS was related to significantly better performance on unit 3 (web design).

In addition to the implementation levels, teacher attributes such as experience and the learning context such as CS presence appeared to have an impact on student learning. Specifically, students whose teachers reported more experience teaching CS courses performed significantly worse on the assessments across all units; whereas students whose teachers reported more experience teaching ECS courses performed better on unit 2 assessment (problem solving) but not the other three units. In terms of learning context, we found a positive relationship between the number of CS courses offered at school and student assessment performance for Unit 1.

The analysis results indicate the varying relationship between instructional strategies and student performance across units. Moreover, experience teaching ECS appears to be more important than prior experience teaching CS when it comes to student performance, particularly for Unit 2. The positive effect of having

a well-established CS program at a school on student performance does not appear to last past Unit 1.

## 5. Discussion

Overall, the study did not reach strong conclusions that were consistent across the units. Some of this may be due to the reliance on teacher self-report, which were not externally validated. Often the results we found differed by unit. For example, while higher level of belief that ECS benefits all was related to better performance on Unit 2, this result was not found for the other units. While this may in part be due to the small sample size, it also may be an indicator that the factors that promote or impede success in teaching CS may differ depending on the content that is being taught.

One interesting result we found was that students of teachers who reported more experience teaching Computer Science courses performed significantly worse on the ECS unit assessments. This was not found to be the case for teachers who had more experience teaching ECS though. This may be because the focus of ECS is not the same as traditional CS courses, and to effectively teach ECS there are instructionally different strategies needed than those required for more traditional CS courses.

Related to this is the fact that students whose teachers reported higher inquiry and collaboration-based strategies performed significantly better on the assessments for two of the units. Higher reporting of collaboration-based strategies was related to higher performance on Unit 1, while higher inquiry-based strategies was related to higher performance on Unit 3.

Another finding that was expected is that teachers who had difficulty accessing technology tended to modify or skip lessons more than others, as they needed to make curricular adaptations to accommodate the issues with technology. Modifications of Units 1 and 3 was tied to lower performance of the students, while skipping of topics in Units 2 and 4 was tied to lower student performance. This indicates that care should be taken when deciding to modify or skip a lesson, as students may then not have the opportunity to achieve the learning goals of those lessons.

## 6. Conclusion

This study provides an overview of factors that can be considered when examining CS curriculum, and the development of measures for these factors. Overall, this study indicates that the factors for success in an introductory computer science course may not be the same across all CS concepts or CS classrooms. Further refinement of the factors that affect teaching quality and curriculum enactment may help tease out the relationship between teacher/classroom characteristics, teaching quality, curriculum enactment, and student success. In addition, future studies may examine other measures of success, such as future course-taking patterns.

## ACKNOWLEDGMENTS

This work was conducted under NSF Grant-#1418149. We gratefully acknowledge all of the teachers and students who participated in the research.

## REFERENCES

- connect homes and classrooms. *Theory Into Practice* 31, 2: 132–141. <https://doi.org/10.1080/00405849209543534>
- [16]. Robert P. Moses and Cobb, Charles E. 2001. *Radical Equations: Civil Rights from Mississippi to the Algebra Project*. Beacon Press, Boston, MA.
- [17]. C. L. O'Donnell. 2008. Defining, Conceptualizing, and Measuring Fidelity of Implementation and Its Relationship to Outcomes in K-12 Curriculum Intervention Research. *Review of Educational Research* 78, 1: 33–84. <https://doi.org/10.3102/0034654307313793>
- [18]. Outlier Research and Evaluation. n.d. *Barriers and Supports to Teaching Introductory Computer Science (with Exploring Computer Science Materials)*. Retrieved September 1, 2018 from <http://outlier.uchicago.edu/basics/>
- [19]. William R. PenueI, Min Sun, Kenneth A. Frank, and H. Alix Gallagher. 2012. Using Social Network Analysis to Study How Collegial Interactions Can Augment Teacher Learning from External Professional Development. *American Journal of Education* 119, 1: 103–136. <https://doi.org/10.1086/667756>
- [20]. Janine T. Remillard and Daniel J. Heck. 2014. Conceptualizing the curriculum enactment process in mathematics education. *ZDM* 46, 5: 705–718. <https://doi.org/10.1007/s11858-014-0600-4>
- [21]. Pratim Sengupta, John S. Kinnebrew, Satabdi Basu, Gautam Biswas, and Douglas Clark. 2013. Integrating computational thinking with K-12 science education using agent-based computation: A theoretical framework. *Education and Information Technologies* 18, 2: 351–380. <https://doi.org/10.1007/s10639-012-9240-x>
- [22]. Eric Snow, Daisy Rutstein, Marie Bienkowski, and Yuning Xu. 2017. Principled Assessment of Student Learning in High School Computer Science. 209–216. <https://doi.org/10.1145/3105726.3106186>
- [23]. Carol Tate, Julie Remold, and Marie Bienkowski. 2018. Pursuing the Vision of CS for All: Views from the Front Lines. *ACM Inroads* 9, 3: 48–52.
- [1]. Michal Armoni. 2011. Looking at Secondary Teacher Preparation Through the Lens of Computer Science. *ACM Transactions on Computing Education* 11, 4: 1–38. <https://doi.org/10.1145/2048931.2048934>
- [2]. Owen Astrachan and Amy Briggs. 2012. The CS Principles Project. *ACM Inroads* 3, 2: 38–42. <https://doi.org/10.1145/2189835.2189849>
- [3]. Andrea Arpaci-Dusseau, Owen Astrachan, Dwight Barnett, Matthew Bauer, Marilyn Carrell, Rebecca Dovi, Baker Franke, Christina Gardner, Jeff Gray, Jean Griffin, Richard Kick, Andy Kuemmel, Ralph Morelli, Deepa Muralidhar, Rebecca Brook Osborne, and Chinma Uche. 2013. Computer science principles: analysis of a proposed advanced placement course. In *Proceeding of the 44th ACM technical symposium on Computer science education (SIGCSE)*, 251–256. <https://doi.org/10.1145/2445196.2445273>
- [4]. Marie Bienkowski, Nicole Shechtman, Julie Remold, and Jennifer Knudsen. 2014. Supporting STEM Teachers to Inspire through Everyday Innovation. *Teacher Education and Practice* 27: 356–375.
- [5]. Alec M. Bodzin and Karen M. Beerer. 2003. Promoting Inquiry-Based Science Instruction: The Validation of Science Teacher Inquiry Rubric. *Journal of Elementary Science Education* 15, 39: 1–11. Retrieved from <https://doi.org/10.1007/BF03173842>
- [6]. Jeanne Century and Amy Cassata. 2014. Conceptual foundations for measuring the implementation of educational innovations. In *Treatment integrity: A foundation for evidence-based practice in applied psychology.*, Lisa M. Hagermoser Sanetti and Thomas R. Kratochwill (eds.). American Psychological Association, Washington, 81–108. Retrieved May 6, 2016 from <http://content.apa.org/books/14275-006>
- [7]. Jeanne Century, H. King, S. Rand, C. Heppner, Baker Franke, and Jill Westrick. 2013. *Computer Science - Teacher Capacity Study - The Center for Elementary Mathematics and Science Education*. Outlier. Retrieved September 1, 2018 from <http://outlier.uchicago.edu/computerscience/OS4CS/teacher-capacity/>
- [8]. Linda Darling-Hammond. 2000. Teacher quality and student achievement. *Education policy analysis archives* 8: 1. Retrieved April 1, 2015 from <http://epaa.asu.edu/ojs/index.php/epaa/article/view/392>
- [9]. Drew H. Gitomer and Courtney A. Bell. 2013. Evaluating teaching and teachers. In *APA handbook of testing and assessment in psychology, Vol. 3: Testing and assessment in school psychology and education*. American Psychological Association, Washington, DC, US, 415–444. <https://doi.org/10.1037/14049-020>
- [10]. Joanna Goode and Jane Margolis. 2011. Exploring Computer Science: A Case Study of School Reform. *ACM Transactions on Computing Education* 11, 2: 1–16. <https://doi.org/10.1145/1993069.1993076>
- [11]. Dan Leyzberg and Christopher Moretti. 2017. Teaching CS to CS Teachers: Addressing the Need for Advanced Content in K-12 Professional Development. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education (SIGCSE '17)*, 369–374. <https://doi.org/10.1145/3017680.3017798>
- [12]. Jane Margolis, Jean J. Ryoo, Cueponcaxochitl D. M. Sandoval, Clifford Lee, Joanna Goode, and Gail Chapman. 2012. Beyond Access: Broadening Participation in High School Computer Science. *ACM Inroads* 3, 4: 72–78. <https://doi.org/10.1145/2381083.2381102>
- [13]. Kata Mihaly, Daniel F. McCaffrey, Douglas O. Staiger, and J. R. Lockwood. 2013. *A composite estimator of effective teaching*. Seattle, WA: Bill & Melinda Gates Foundation. Retrieved May 9, 2015 from [http://www.nbxcellence.org/cms\\_files/resources/Jan%202013%20%20A%20Composite%20Estimator%20of%20Effective%20Teaching%20Research%20Paper.pdf](http://www.nbxcellence.org/cms_files/resources/Jan%202013%20%20A%20Composite%20Estimator%20of%20Effective%20Teaching%20Research%20Paper.pdf)
- [14]. K-12 CS Framework Committee. 2016. *K-12 Computer Science Framework*. Retrieved April 11, 2017 from <http://www.k12cs.org>
- [15]. Luis C. Moll, Cathy Amanti, Deborah Neff, and Norma Gonzalez. 1992. Funds of knowledge for teaching: Using a qualitative approach to