Computational Thinking in the Science Classroom

Hillary Swanson, Gabriella Anton, Connor Bain, Michael Horn, Uri Wilensky
Northwestern University
hillary.swanson@northwestern.edu, gabby.anton@gmail.com, ConnorBain2015@u.northwestern.edu, michael-horn@northwestern.edu, uri@northwestern.edu

ABSTRACT
The importance of Computational Thinking (CT) as a goal of science education is increasingly acknowledged. This study investigates the effect of computationally-enriched science curriculum on students' development of CT practices. Over the course of one school year, biology lessons featuring the exploration of NetLogo models were implemented in the classrooms of three 9th grade biology teachers at an urban public secondary school in the United States. One-hundred thirty-three biology students took both pre- and post-tests that were administered at the beginning and end of the school year. The students' responses to relevant assessment items were coded and scored using rubrics designed to evaluate their mastery of two learning objectives relating to modeling and simulation practices. The first learning objective was to *explore the relationship between a system's parameters and its behavior*. The second learning objective was to *identify the simplifications made by a model*. Each item's pre- and post-test scores were compared using a Wilcoxon signed-rank test. Results indicate a statistically significant improvement with respect to the second of the two learning objectives, suggesting that the computationally-enriched biology curriculum enhanced students' ability to identify the simplifications made by a model.

KEYWORDS
Computational Thinking, STEM Education, Learning Objectives, Curriculum, Assessment.

1. INTRODUCTION
The importance of Computational Thinking (CT) as a goal of science education is increasingly acknowledged (Quinn, Schweingruber, Keller, 2012; Wilensky, Brady & Horn, 2014). Teaching CT in the context of science not only presents students with a more authentic image of science as it is practiced today, it also increases access to powerful modes of thinking and marketable skills for many careers (Levy & Murname, 2004). It is estimated that by 2020, one out of every two STEM jobs will be in computing (ACM Pathways Report 2013). However, students from groups that have been historically underrepresented in STEM fields (such as women and racial minorities) are less likely to enroll in computer science classes (Margolis, 2008; Margolis & Fisher, 2003) and thus are not traditionally exposed to CT practices. We believe we can improve access for all students, especially those underrepresented in CS, by embedding CT practices in subjects such as biology, chemistry, and physics, which all high school students are expected to take. While this does not ensure that these students will be personally motivated to engage in our CT curriculum, it ensures that they will at least be exposed to CT practices and given the opportunity to learn about them.

For the reasons given above, we believe that developing CT practices in the context of science subjects is a productive endeavor. However, the character of CT practices in the science disciplines is not yet well understood, nor is how to create curriculum and assessments that develop and measure these practices (Grover & Pea, 2013). To address this gap, our group has worked to explicitly characterize core CT practices as specific learning objectives and used these to guide our development of science curriculum and assessment. We developed our learning objectives upon a theoretical taxonomy of CT in STEM that our group previously proposed (Weintrop et al., 2016). The taxonomy consists of four strands of CT practices: *Data Practices*, *Modeling and Simulation Practices, Computational Problem Solving Practices,* and *Systems Thinking Practices*. We translated elements from each strand of the taxonomy into learning objectives through a process involving interviews with computational scientists and feedback from high school science teachers.

The general aim of our larger research agenda is to address the question: "Can engaging in computationally-enriched science curriculum help students develop CT practices?" In the present study, we address a more focused version of this question and investigate whether engaging in three computationally-enriched biology units over the course of the school year helped participant students develop CT practices, specifically two practices within the *Modeling and Simulations* strand of our taxonomy. Below, we describe our study design and analytical approach, then present results from a comparison of students' scores for pre- and post-assessments. Our results provide support for our claim that computationally-enriched science curriculum can foster students' development of particular CT practices.

2. STUDY DESIGN
We investigated our research question by analyzing data from the fourth iteration of a design-based research cycle (Collins, Joseph, Bielaczyc, 2004). The implementation spanned the 2015-2016 school year and was tested in three 9th grade biology classrooms at our partner school. Students were given a CT practices pre-test at the beginning of the school year and a CT practices post-test at the end of the school year. Over the course of the school year they

participated in three CT science units, each unit approximately four days long. We investigated the role of the CT science units in students' development of particular CT practices by looking for statistically significant gains in scores for particular items from pre- to post-test.

2.1. Participants

We partnered with a public secondary school (serving grades 7 – 12) in an economically depressed neighborhood in a large city in the Midwestern region of the United States. The school was selected on the basis of the willingness of its teachers and students to participate in our study. The size of the school was typical for an urban public secondary school, with approximately twelve hundred students enrolled. The majority of the students at the school are considered to be of racial minority within the United States (71.1% Black, 24.5% Hispanic, 1.6% Asian, .3% American Indian, .2% Pacific Islander, .9% Bi-Racial, 1.4% White), with sixty-two percent from low income households. The school is characterized as selective-enrollment, meaning that the student population is academically advanced and highly motivated. We addressed our research questions by analyzing a selection of the pre- and post-test responses given by participating 9th grade biology students. A total of 133 of these students, distributed across three biology teachers, took both tests. Due to time constraints, a number of these students did not complete the entire assessment. Ten students did not complete the assessment item measuring learning objective 1 and 24 did not complete the assessment item measuring learning objective 2; these students' responses were therefore not included in the analyzed datasets.

2.2. CT Science Lessons

The biology students participated in three computationally-enriched biology units over the course of the school year. Each unit took approximately four school days and emphasized the exploration and manipulation of computational models of scientific phenomena or concepts. The first unit was on predator-prey dynamics and ecosystem stability. For this unit, students explored population dynamics in a simulation of an ecosystem consisting of three organisms (grass, sheep, and wolves) (Wilensky, 1997b). Students investigated the population-level effects of parameters for individual organisms (such as initial population and reproduction rate) by running the simulation with different values for each organism. Through their exploration, the students learned about the complex population dynamics that emerge from the interactions between individual organisms. The second unit was on AIDS. For this unit, students explored a model that simulated the diffusion of the infectious disease through a population (Wilensky, 1997c). Students investigated the effects of parameters for individual interactions (such as the probability of individuals to form a couple, and the probability of the disease transfer between partners) on the rate of spread of the disease. The third unit was on genetics. For this unit students explored a model that allowed them to change mating rules in a population of fish. Students investigated how changing parameters such as life span and mating choice could bring about changes in the overall allele frequencies in a population of fish. All units were meant to help students develop expertise regarding learning objectives for *Modeling and Simulations Practices* by engaging in science content through the exploration of NetLogo (Wilensky, 1999) simulations. NetLogo simulations were chosen because the agent-based modeling environments make complex systems phenomena (such as those featured in the biology lessons) more intuitively accessible (Wilensky, 2001). Additionally, the NetLogo user interface makes transparent the relationship between a model's code and the phenomenon it simulates. This makes NetLogo a powerful tool for scaffolding students' transition from consumers, to designers and builders of computational models. In order to help students develop a flexible set of CT practices, other CT-STEM units feature simulations built in modeling environments such as Molecular Workbench (Concord Consortium, 2010) and PhET (Perkins et al., 2006) and introduce students to a range of computational tools for data analysis and problem solving.

2.3. CT Assessments

The pre- and post-tests were designed to evaluate students' mastery of CT practices. In this report, we present results concerned with two particular learning objectives within our *Modeling and Simulations Practices* strand. The first learning objective falls under the sub-strand element *Using Computational Models* and states that a student should be able to "explore a model by changing parameters in the interface or code." This is a very basic skill but it plays an important role in students' (and scientists') abilities to learn about the relationship between particular parameters and system behavior at the macro-level. The second learning objective falls under the sub-strand element *Assessing Computational Models* and states that a student should be able to "identify the simplifications made by a model." This learning objective is important to students' epistemological development, as it relates to their understanding of a computational model as a tool that is both powerful and limited with regards to the construction of new knowledge.

Both pre- and post-tests required students to interact with computational simulations. For the pre-test, students interacted with a simulation (shown in Figure 1, below) that modeled climate change and showed the relationship between temperature and amount of CO2 in the atmosphere (Tinker & Wilensky, 2007). For the post-test, students explored a simulation (shown in Figure 2, below) that modeled the relationship between the pressure of a gas and its volume and number of particles in a sealed environment (Wilensky, 1997a; 2005).

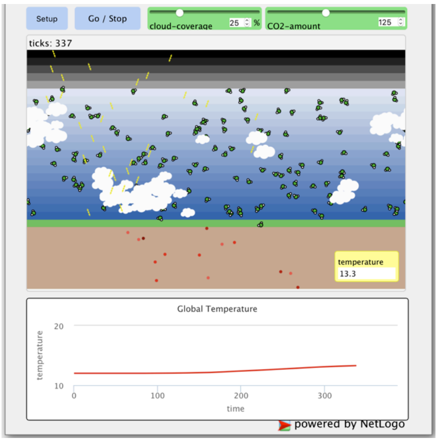

Figure 1. Screenshot of pre-test simulation modeling the relationship between temperature and atmospheric CO2 levels.

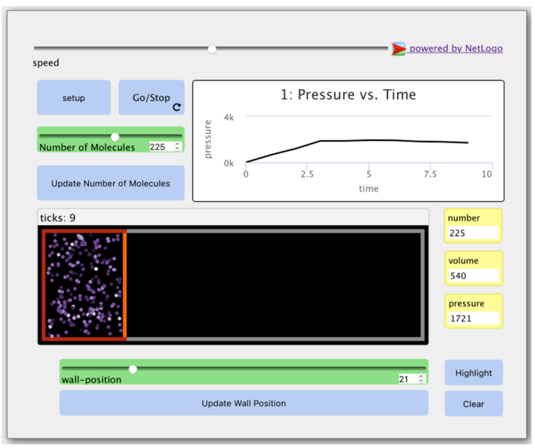

Figure 2. Screenshot of post-test simulation modeling the relationship between the pressure of a gas and its volume and number of particles.

To assess students' abilities to explore a model by changing parameters in the interface or code, we analyzed their responses to test items (quoted below) that asked them to attend to the relationships between adjustable parameters and system-level characteristics. In order to assess students' abilities to identify simplifications made by a model, we analyzed their responses to test items that asked them for the ways in which the simulations differed from the real-world. These assessment items were selected to investigate students' mastery of the same learning objectives across two very different computationally modeled phenomena.

2.4. Data Analysis

We used a combined top-down (learning objective driven) bottom-up (data driven) approach to create rubrics for evaluating students' responses to pre- and post-test questions and characterizing their mastery of both learning objectives.

2.4.1. Learning Objective 1

For the pre-test, in the context of the greenhouse gas simulation, students were asked to *explore the relationship between a system's parameters and its behavior* by changing a particular parameter and reporting on the resulting system-level behavior. In particular, they responded to the prompt: "Set cloud coverage to 0%. Take some time to experiment with different settings for the 'CO2-amount' slider. What happens to the temperature if you increase the amount of the CO2 in the model?" For the post-test, in the context of the gas-law simulation, students were asked to explore the relationship between a system's parameters and behavior by changing parameters to get a specific result. In particular, they responded to the question: "What values for container size and number of particles will result in the lowest pressure in the container? What steps did you take to come up with these values?"[i]

We examined students' pre- and post-test responses, sorting responses into categories based on similarities that were relevant to our focal learning objective. Four categories emerged that characterized response types across both pre- and post-test responses. These categories are Noticing Parameter-System Relationships, Including Explanatory Factors, Comparing Across Trials, and Correctness.

These categories are outlined, described and illustrated with examples from the data in Table 1, below. We scored students' responses by awarding one point for each category included in their response and taking the sum of these points. This resulted in scores ranging from 0-3.

Table 1. Pre- and post-test rubric for analyzing students' responses and characterizing their ability to explore a model by changing parameters in the interface or code.

	Student Example
Relationships Response describes relationship between system parameters and macro-level patterns.	
Pre-Test	"The temperature increases."
Post-Test	"I slid the wall-position to its maximum and the number of particles to its minimum."
Explanatory Factors Response provides some explanation for relationship between system parameters and macro-level patterns.	
Pre-Test	"IR light does not get a chance to go into the sky because it is blocked by CO2."
Post-Test	"A bigger area and less particles shouldn't produce a large amount of pressure since it's a lot of space for the particles."

Comparison Response compares data across multiple simulation trials.	
Pre-Test	"When I increase the CO2 amount there seem to be IR light flying all over the place. But when there are smaller amounts of CO2 molecules the IR light have a better chance of going straight into the sky."
Post-Test	"To come up with these values I first tried putting the number of particles and the container size at its max. After that, I tried the number of particles at its minimum and the container size at its maximum."
Correctness Response correctly addresses the assessment prompt.	
Pre-Test	"The temperature increases."
Post-Test	"Number of particles: 25 Wall position: 96"

2.4.2. Learning Objective 2

As part of the pre-test, students were asked to *identify the simplifications* made by the greenhouse simulation. As part of the post-test, students were asked to identify the simplifications made by the gas-law simulation. For both tests, they responded to the question: "All computational simulations are only approximations of reality. What are some of the simplifications of this simulation that make it different from the real world?"

We examined students' pre- and post-test responses, sorting responses into categories based on similarities that were relevant to the learning objective we were analyzing. Six categories emerged that characterized response types across both pre- and post-test responses. These categories are General Issues, Representational Issues, Controllability, Completeness, Procedural Limitations, and Off-Task. They are arranged in order of increasing sophistication[ii], described and illustrated with examples from the data in Table 2, below. We scored students' responses by awarding them the point-value of the highest category included. "Off-Task" (of point-value zero) was given to responses that did not address the assessment item, or consisted of "I don't know." Scores ranged from 0-3.

Two researchers analyzed students' responses to the two assessment items for both pre-and post-tests. They coded responses (identifying the categories presented in the rubrics) and then scored them. The researchers' inter-rater reliability for the pre-test was at 97% for the item measuring the first learning objective and 90% for the item measuring the second learning objective. Inter-rater reliability for the post-test was at 95% and 80%, respectively.

Table 2. Pre- and post-test rubric for analyzing students' responses and characterizing their ability to identify simplifications made by a model.

	Student Example
General Issues – Score: 1 Response refers to general, as opposed to specific, inaccuracies or missing factors.	
Pre-Test	"In reality, other factors could come into play rather than just CO2 and clouds."
Post-Test	"Inaccuracy in particles and wall position can make it different from the real world."
Representation Issues – Score: 1 Response refers to representational limitations of the model.	
Pre-Test	"Obviously, sunlight is not a bunch of little sticks raining down."
Post-Test	"It's not actually life size."
Controllability – Score: 2 Response refers to the existence of control over factors in the model that one does not have control over in real life.	
Pre-Test	"Because you can control how much CO2 and cloud coverage there is."
Post-Test	"In real life, you cannot add or subtract molecules nor can you adjust the wall positioning."
Completeness – Score: 2 Response refers to specific elements or factors that are missing from, or extraneous to, the model.	
Pre-Test	"There are humans on earth and humans also can add to the amount of heat."
Post-Test	"The real world, does not have this many boundaries and an infinite number of particles."
Procedural Limitations – Score: 3 Response refers to interactions, behaviors, or relationships within the model that differ from real life.	
Pre-Test	CO2 might not speed up that much when it absorbs IR light.
Post-Test	Particles don't travel in and out of room in this simulation, when in real life they do.

To test whether the intervention played a role in their development of CT practices, students' scores for each item on both pre- and post-tests were compared using a Wilcoxon signed-rank test. The findings of this analysis are reported below.

3. Findings

3.1. Learning Objective 1

Students' average score for the pre-test item measuring their ability to explore a model by changing parameters in the interface or code was 2.03. Their average post-test score was 2.19. The p-value obtained using the Wilcoxon signed-rank test was 0.23 (V = 1486). The difference in student scores is therefore not statistically significant and we cannot make the claim that engagement in our curriculum helped students improve their CT skills with regard to this learning objective.

In addition to comparing students' pre- and post-test scores for this learning objective, we compared the frequencies of categories of ideas that appeared in students' pre- and post-test responses. Examination of the bar chart below reveals that during the pre-test, many students were concerned with macro-level effects of changing parameters, while at the time of the post-test, many more students referred to explanatory factors in their responses. This suggests they looked more closely at the model and tried to understand the interactions at the micro-level that explained the macro-level phenomenon. While the comparison of pre- and post-test scores indicates that students are not necessarily developing sophistication regarding their ability to explore a model, the changing frequency of categories gives us insight into one specific way students may in fact be developing expertise.

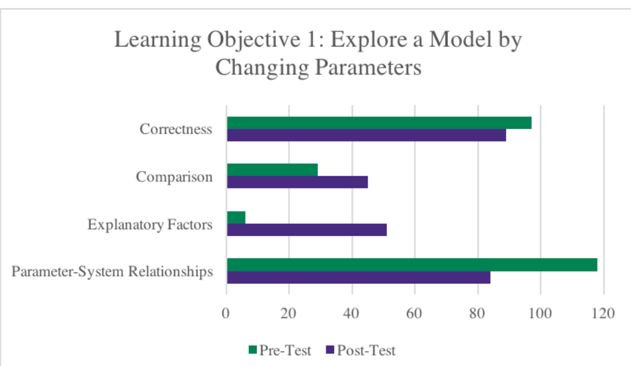

Figure 3. Frequencies of categories included in students' responses to the pre- and post-test items assessing their mastery of learning objective 1.

3.2. Learning Objective 2

Students' average score for the pre-test item measuring their ability to identify simplifications made by a model was 1.39. Their average post-test score was 1.63. The p-value obtained using the Wilcoxon signed-rank test was 0.02 (V = 647.5). The difference in student scores is therefore statistically significant (at the 5% significance level) and this supports our claim that engagement in our curriculum helped students improve their CT skills with regard to this learning objective.

In addition to comparing students' pre- and post-test scores for this learning objective, we compared the frequencies of categories of ideas that appeared in students' pre- and post-test responses. For ease of coding, we combined categories of the same score. This is reflected in the categories shown in the bar chart below. Examination of this bar chart reveals that during the pre-test, many students reported general or representational simplifications, whereas at the time of the post-test, this number decreased and the number of students reporting controllability or completeness as a limitation increased.[iii] The number of students reporting procedural simplifications also increased. While the comparison of pre- and post-test scores indicates that students are developing sophistication regarding their ability to identify simplifications within a model, the changing frequency of categories gives us insight into the specific ways in which students are becoming more sophisticated.

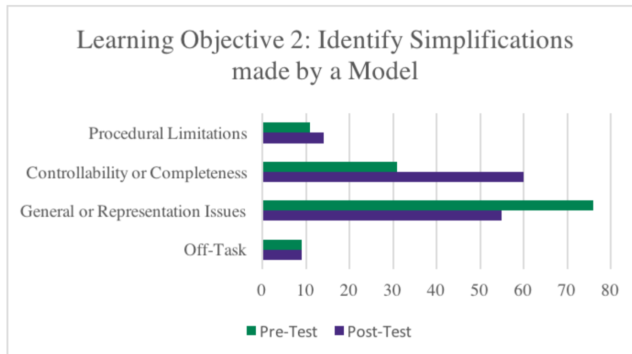

Figure 4. Frequencies of categories included in students' responses to the pre- and post-test items assessing their mastery of learning objective 2.

4. Discussion

This study extends our group's previous work by translating our theoretical taxonomy into learning objectives that can be used to guide the design of curriculum and assessment. The study makes an empirical contribution by presenting evidence that engagement in our CT-STEM curriculum helped participating students develop their ability to identify simplifications made by computational models. Our data also gives us insight into how students might develop their ability to explore a computational model. Toward this, we will conduct qualitative analysis of particular students and examine individual developmental trajectories. Our next steps also include refining our pre- and post- assessment items so that they are more closely aligned with each other, and with our learning objectives. As well, we are refining our curriculum (across the science subjects) so that it is more closely aligned with our learning objectives and assessment items. This refinement includes creating more opportunities for students to explicitly reflect on and discuss their individual ways of exploring models, as well as the simplifications they notice in different models.

5. References

Collins, A., Joseph, D., & Bielaczyc, K. (2004). Design research: Theoretical and methodological issues. *The Journal of the learning sciences*, *13*(1), 15-42.

Concord Consortium. (2010). Molecular workbench. *Java simulations and modeling tools), (2004–2013).*

Grover, S., & Pea, R. (2013). Computational Thinking in K-12: A Review of the State of the Field. Educational Researcher, 42(1), 38–43.

Levy, F. & Murname, R. (2004). *The new division of labor: How computers are creating the new job market.* Princeton, NJ: Princeton University Press.

Margolis J (2008) Stuck in the shallow end: education, race, and computing. The MIT Press, Cambridge

Margolis J, Fisher A (2003) Unlocking the clubhouse: women in computing. The MIT Press, Cambridge

Perkins, K., Adams, W., Dubson, M., Finkelstein, N., Reid, S., Wieman, C., & LeMaster, R. (2006). PhET: Interactive simulations for teaching and learning physics. *The Physics Teacher*, *44*(1), 18-23.

Quinn, H., Schweingruber, H., & Keller, T. (Eds.). (2012). *A framework for K-12 science education: Practices, crosscutting concepts, and core ideas*. National Academies Press.

Tinker, R. & Wilensky, U. (2007). NetLogo Climate Change model. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.

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.

Wilensky, U. (1997a). NetLogo GasLab Gas in a Box model. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL. http://ccl.northwestern.edu/netlogo/models/GasLabGasinaBox.

Wilensky, U. (1997b). NetLogo Wolf Sheep Predation model. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL. http://ccl.northwestern.edu/netlogo/models/WolfSheepPredation.

Wilensky, U. (1997c). NetLogo AIDS model. http://ccl.northwestern.edu/netlogo/models/AIDS. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.

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

Wilensky, U. (2001). Modeling nature's emergent patterns with multi-agent languages. In *Proceedings of EuroLogo* (pp. 1-6).

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

Wilensky, U., Novak, M. & Levy S.T. (2005). NetLogo Connected Chemistry 6 Volume and Pressure model. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.

---

[i] It is important to note that while both items are concerned with students' abilities to learn about a parameter's influence on a system's behavior, they are inversely structured. While the pre-test item instructs students to change a parameter and report its effect on the system, the post-test item instructs students to change parameters until they achieve a specified system behavior. We argue that while they are different in this way, both items are concerned with the causal relationship between parameter values and system-level behavior and are therefore comparable assessments of students' abilities to explore a model by changing parameters in the interface or code.

[ii] General comments about accuracy and representational limitations seemed to be the easiest to make with attention to mere surface-features. These simplifications were therefore awarded the lowest score (one point). The completeness of the model and control given to its various parameters seemed to require more careful consideration of the interface and comparison with the real-world. These simplifications were therefore awarded a slightly higher score (two points). Finally, comments about the procedural correctness of behavior and interactions within the model required students to run the model and track cause and effect relationships between elements at the micro-level and comparison of this with scientific laws or theories. These simplifications were therefore awarded the highest score (three points).

[iii] This point is especially interesting given that the gas-law simulation is just as unrealistic, regarding the visual representation of the system, as the greenhouse effect model.