A Science Learning Environment using a Computational Thinking Approach

Satabdi BASU[a*], John S. KINNEBREW[a], Amanda DICKES[b], Amy Voss FARRIS[b], Pratim SENGUPTA[b], Jaymes WINGER[a] & Gautam BISWAS[a]
[a]*Department of EECS/ISIS, Vanderbilt University, U.S.A.*
[b]*Mind, Matter & Media Lab, Vanderbilt University, U.S.A.*

Abstract: Computational Thinking (CT) defines a domain-general, analytic approach to problem solving that combines concepts fundamental to computing, with systematic representations for concepts and problem-solving approaches in scientific and mathematical domains. We exploit this trade-off between domain-specificity and domain-generality to develop CTSiM (Computational Thinking in Simulation and Modeling), a cross-domain, visual programming and agent-based learning environment for middle school science. CTSiM promotes inquiry learning by providing students with an environment for constructing computational models of scientific phenomena, executing their models using simulation tools, and conducting experiments to compare the simulation behavior generated by their models against that of an expert model. In a preliminary study, sixth-grade students used CTSiM to learn about distance-speed-time relations in a kinematics unit and then about the ecological process relations between fish, duckweed, and bacteria occurring in a fish tank system. Results show learning gains in both science units, but this required a set of scaffolds to help students learn in this environment.

Keywords: Computational thinking, agent-based modeling, visual programming, simulation, scaffolding, inquiry learning

1. Introduction

Computational thinking (CT) describes a general analytic approach to problem solving, designing systems, and understanding human behavior [10,15]. It draws on fundamental concepts in computing and computer science to support practices (e.g., problem representation, abstraction, decomposition, verification) that are central to modeling, reasoning, and problem solving in scientific and mathematical disciplines [10,11].

Developing scientific practices and problem-solving skills requires sustained, immersive educational experiences, which can be implemented as learning progressions for science in K-12 classrooms [7,9]. CT can support such progressions in the K-12 science curricula by integrating its inherent domain generality with support for domain specific representations, reasoning, and analysis of real world problems across multiple domains [10,11,15]. Balancing and exploiting this trade-off between domain-generality and domain-specificity, however, presents an important educational design challenge.

Previous studies on integrating programming with K-12 science have pointed out a variety of similar challenges [3,4]. Curricula that have effectively addressed these challenges take advantage of *reflexivity*, which hypothesizes that learning programming in concert with concepts in another domain can be easier than learning each separately [6]. Several researchers have shown that programming and computational modeling can serve as effective vehicles for learning challenging science and math concepts [2,5,8]. Further, many programming and CT concepts parallel important aspects of STEM (Science, Technology, Engineering, & Mathematics) learning. For example, the creation of coherent, formal representations of scientific phenomena and mathematical representations of scientific laws are similar to object-oriented programming concepts of encapsulation, abstraction, and generalization. Conversely, the biological concepts of taxonomy and inheritance are the inspiration for class inheritance concepts in programming.

Leveraging the synergy between CT, science, and math learning, we have designed the Computational Thinking in Simulation and Modeling (CTSiM) learning environment and are implementing it using a learning-by-design progression. The learning environment combines visual programming and simulations to allow for flexible iterations between initial instruction in the science topic; modeling the appropriate entities and processes using a visual, agent-based computational framework; simulating and studying the behavior of the model; using explanation and argumentation skills to understand and verify the model; and, applying the developed model and science understanding to problem-solving tasks. The learning progression can be implemented over a progression of topics.

This paper presents the rationale for the learning environment design and a two-unit science sequence (kinematics and ecology) to demonstrate our computational thinking approach across domains. We describe an initial CTSiM study with 6th-grade students in a middle Tennessee public school. The results demonstrate the effectiveness of our approach, supporting the premise that students' conceptual understanding of science topics improves after engaging in CT-based curricular units. Finally, we present the categories of scaffolds used in the study and discuss the role these scaffolds likely played in the students' learning.

2. The CTSiM learning environment and curricular units

To support learning-by-design activities, CTSiM comprises three primary components [11]: (1) the Construction (C) world, (2) the Enactment (E) world, and (3) the Envisionment (V) world. The C world provides a visual programming interface where students build computational models for the science topics they are learning. It includes a library of visual primitives corresponding to agent actions, sensing conditions qualifying agent actions, and controls for regulating the program's flow of execution (e.g., conditionals and loops). Each visual primitive is defined in terms of an underlying domain-independent computational primitive. Students drag and drop these primitives and arrange them spatially to generate their computational models, as illustrated in Figure 1 for an ecology unit.

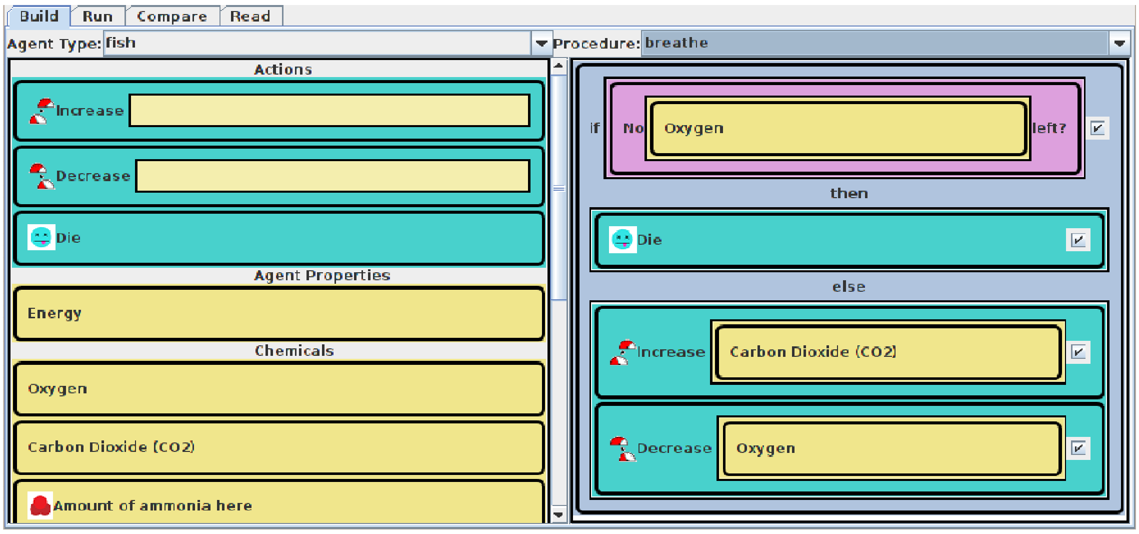

Figure 1: The Ecology unit Construction world with a 'breathe' procedure for 'fish' agents

In the E world, illustrated in Figure 2, students set initial parameter values and observe the Netlogo-based simulations corresponding to their models. NetLogo visualizations and plotting functions [15] provide students with a dynamic, real-time display of how their agents operate in the microworld simulation, thus making explicit the emergence of aggregate system behavior (*e.g.*, from graphs of a species population over time).
The V world provides students the opportunity to perform systematic experiments to compare their models' behavior against behavior generated by an "expert" model. This side-by-side comparison of plots and microworld visualizations for the two models makes it easier for students to investigate and revise their models. With proper scaffolding, we believe that the overall process of model construction, analysis, comparison, and refinement will help students gain a better understanding of science phenomena, mathematical concepts (e.g., rates), and computational constructs and methods.

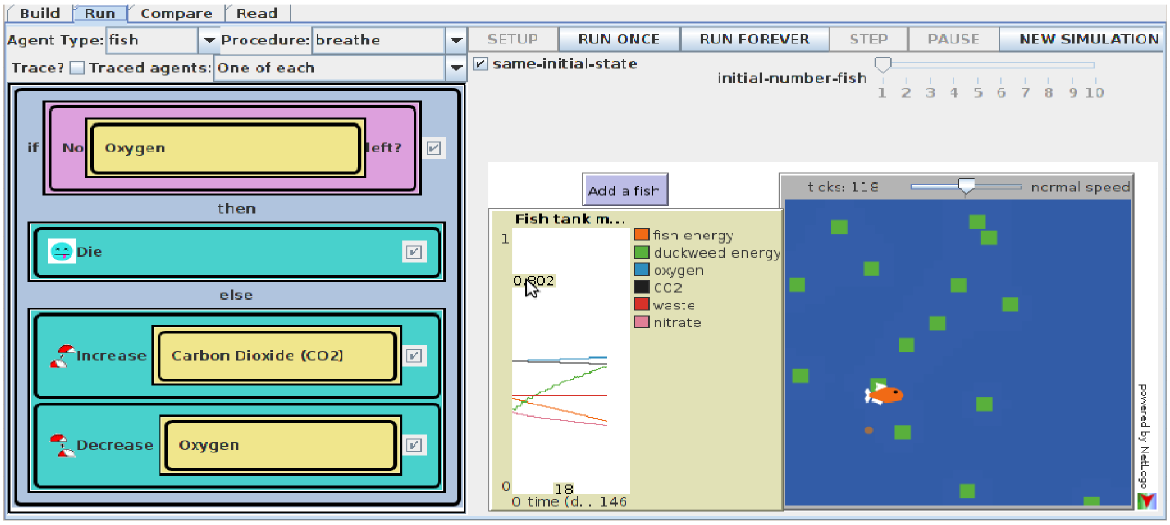

Figure 2: A screenshot of the Enactment world for the Ecology macro-unit

2.1 Kinematics Unit

Activities in the Kinematics unit were divided into three phases [12, 13].

Phase 1: Turtle Graphics for Constant Speed and Constant Acceleration – We introduced students to programming commands by showing them how to manipulate different elements in the user interface. Then, we asked them to generate algorithms to draw simple shapes (squares, triangles and circles) to familiarize them with programming primitives like "forward", "right turn", "left turn", "pen down", "pen up" and "repeat". Next, we asked students to modify their algorithms and generate spiraling shapes in which each line segment is longer (or shorter) than the previous one. This exercise introduced students to the "speed-up" and "slow-down" commands, and it gave them a chance to explore the relationship between speed, acceleration, and distance.

Phase II: Conceptualizing and re-representing a speed-time graph - In this activity, students generated shapes such that the length of segments in the shapes were proportional to the speed in a given speed-time graph. Figure 3 depicts the speed-time graph provided to all students, along with a sample student output where the initial spurt of acceleration is represented by a small growing triangular spiral, the gradual deceleration by a large shrinking square spiral, and constant speed by a triangle. The focus was on developing mathematical measures from meaningful estimation and mechanistic interpretations of the graph, and thereby gaining a deeper understanding of concepts like speed and acceleration.

Phase III: Modeling motion of an agent to match behavior of an expert model - For this activity, students modeled the behavior of a roller coaster as it moved on different segments of a track: up (pulled by a motor), down, flat , and then up again. Students were first shown the simulation results produced by an ‘expert’ roller coaster model in the V world. Then, they were asked to conceptualize and build their own agent model to match the observed expert roller coaster behavior for all of the segments.

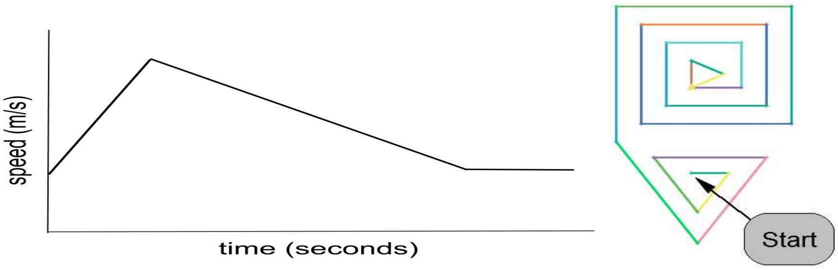

Figure 3: Acceleration represented in a speed-time graph and sample student output

2.2 Ecology Unit

For the Ecology unit students modeled a closed fish tank system in two steps: (1) a macro-level semi-stable model of the behavior of fish and duckweed; and (2) a micro-level model of the waste cycle with bacteria. The macro model included several key relations: (1) the food chain and respiration cycles of the fish and duckweed, (2) the macro-level elements of the waste cycle (fish produce waste, duckweed consume nitrates), and (3) the reproduction of duckweed.

The non-sustainability of the macro-level model (the fish and the duckweed gradually died off), which students built first, helped them reflect, which, in turn provided the transition to the micro model. When prompted to think about why the system was not self-sustaining, students could identify the continuously increasing fish waste as the culprit, and this provided the trigger to introduce the role of bacteria in the system.

At the micro level, students modeled the waste cycle with bacteria and the related chemical processes that converted the toxic ammonia in the fish waste to nitrites, and then nitrates, which sustained the duckweed. The graphs generated from the expert simulation helped students understand the producer-consumer relations: (1) *Nitrosomonas* bacteria consume ammonia and produce nitrites; (2) nitrites are consumed by *Nitrobacter* bacteria to produce nitrates which provide food for the duckweed.

2.3 Sequencing of units and activities

Within each unit, the learning activities were designed to introduce students to (1) *the agent-based program structure, i.e.*, agents and their actions, and (2) *the general computational constructs, i.e.*, conditionals to model situation-based interactions, loops to capture repeated agent behavior, and mathematical operations. The units provided a natural sequencing in which students first learned to model and reason with a single agent in kinematics and then went on to model multiple agents and their interactions in ecology. The focus in the kinematics unit was on modeling real-world phenomena by generating computational abstractions. In the ecology unit, students had the more complex task of modeling multiple agent types (e.g., fish and duckweed at the macro level) and composing several procedures to define the behavior associated with each agent type (e.g., swim, eat, and breathe procedures for fish agents).

3. Method

The study was conducted with 6th-grade students from an ethnically diverse middle school in middle Tennessee. 15 students worked on the system outside the classroom with one-on-one guidance from members of our research team (Scaffolded or S-Group), while the remaining 9 students in the class worked on the system in the classroom (Classroom or C-Group) with some instruction from the researchers and the classroom teacher. The C group also received individual help from the researchers if they raised their hand and asked for help. The students were assigned to the groups by their classroom teacher. During the intervention, five interviewers worked one-on-one with the S-Group students and provided verbal scaffolds. In the C-Group, students received minimal one-on-one scaffolding. We formulated two research hypotheses:

1. The intervention will help both groups improve their understanding of science concepts as demonstrated by their pre-to-post-test learning gains.
2. The one-on-one scaffolding will help the S-Group learn more than the C-group.

As part of the design-based research, we collected and characterized the scaffolds provided by the researchers to inform future system development.

All students worked on the three phases of the kinematics units before the ecology macro and micro units. After completing the ecology micro model, the S group received an additional scaffold: they discussed the combined micro-macro model with their assigned researcher and were shown how the two models were causally linked to support sustainability. Students were given the paper-and-pencil task of building a causal model of the cycles, and then prompted to use this representation to explain the effects of removing one agent on the stability of the cycle.

Students worked on the two science units in hour long sessions for three days each. On day 1 of the study, we administered pre-tests for both units. Students worked on the kinematics unit from day 2 to 4, and then took the kinematics post-test on day 5. This was followed by work on the ecology unit from day 6 to 8, and the ecology post-test on day 9.

4. Results and Discussion

4.1 Learning Gains in Kinematics and Ecology

The Kinematics pre/post-test assessed students' abilities to reason causally about mathematical representations of motion and determined whether agent-based modeling improved their abilities to generate and explain these representations. Specifically, the questions on the test required interpretation of speed versus time graphs and generating diagrammatic representations to explain motion in a constant acceleration field. For the Ecology unit, the pre- and post-tests focused on students' understanding of roles of species in the ecosystem, interdependence among the species, the waste and respiration cycles, and how a specific change in one species affected the others. Some of the questions checked students' declarative knowledge about the fish tank system (Declarative Knowledge Check or DKC), others required performing Causal Reasoning about entities using the Declarative Knowledge (CRDK), and a Transfer Question (TQ) required students to reason about the carbon cycle. An example DKC question asked was *"For each of the following species in the fish tank,: a) Goldfish, b) Duckweed, c) Nitrosomonas, d) Nitrobacter, mention the chemical(s) it directly needs to stay alive"*, while an example CRDK question asked was "*Your fish tank is currently healthy and in a stable state. Now, you decide to remove all traces of nitrobacter bacteria from your fish tank. Would this affect a) Duckweed, b) Goldfish, c) Nitrosomonas bacteria? Explain your answers."*

Students in both groups had higher percentage scores on the kinematics pre-test than the ecology pre-test, indicating that they had a better initial understanding of the kinematics domain. However, pre-test scores and the mean TCAP (Tennessee Comprehensive Assessment Program) science scores suggested differences in prior knowledge and abilities of the S and C groups [significant differences (t=3.15, $p<0.005$) in mean TCAP science scores between the two groups]. Hence we computed a repeated measures ANCOVA with TCAP science scores as a covariate to study the interaction between time and condition. There was a significant effect of condition on pre-post learning gains in ecology ($F(1,21)=37.012$, $p<0.001$), and a similar trend was seen in kinematics ($F(1,21)=4.101$, $p<0.06$). The plots in Figure 4 show that the S group's adjusted gains were higher than the C group in both units.

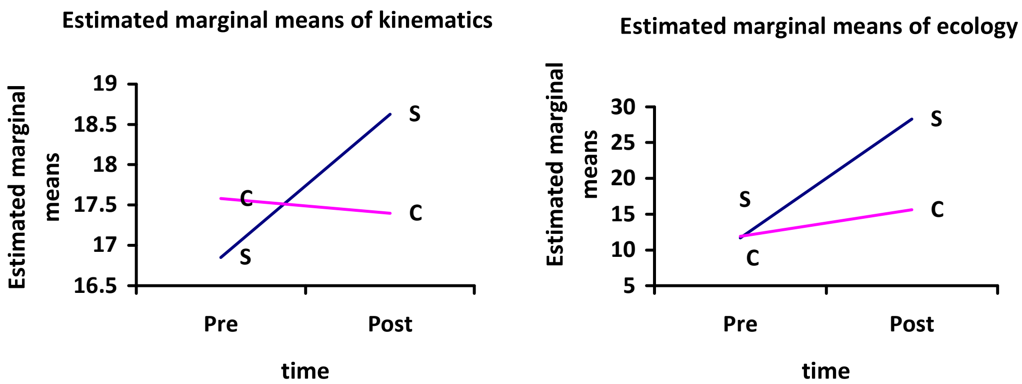

Figure 4: Comparison of gains between groups using TCAP scores as a covariate

Given the significant interaction between time and condition, we performed paired t-tests on pre-to-post gains for each condition. Table 1 shows that the intervention produced statistically significant gains for ecology unit, but not for the kinematics unit. However, for both units, the S group, which received direct one-on-one scaffolding, showed higher learning gains than the C group.

Table 1: Paired t-test results for Kinematics and Ecology pre and post test scores

	Kinematics				Ecology			
	PRE (S.D.) (max=24)	POST (S.D.) (max=24)	t-value	P-value (2-tailed)	PRE (S.D.) (max=35.5)	POST (S.D.) (max=35.5)	t-value	P-value (2-tailed)
S-Group (n=15)	18.07 (2.05)	19.6 (2.29)	2.699	0.017	13.03(5.35)	29.4(4.99)	8.664	<0.001
C-Group (n=9)	15.56 (4.1)	15.78 (4.41)	0.512	0.622	9.61(3.14)	13.78(4.37)	3.402	<0.01

The lack of statistical significance in the kinematics unit may be attributed to a ceiling effect in the students' scores. One exception was a question that asked students to diagrammatically represent the time trajectory of a ball dropped from the same height on the earth and the moon. The students were asked to explain their drawings and generate graphs of speed versus time for the two scenarios. The S group showed significant gains ($p<0.0001$) on this question, while the C group showed an increasing trend, although it was not significant ($p=0.16$).

For the ecology unit, the S-Group students gained on all categories of questions, though all of the gains were not statistically significant. Table 2 reports normalized learning gains (gain/maximum possible gain) by question category for both the groups. Significant gains were observed on the DKC and CRDK questions, which can be attributed to an increased awareness of the entities in the fish tank and their relations with other species. For example, pre-test results indicated that the students did not initially know about the bacteria and their roles. Though students in both groups were told about the role of bacteria during the intervention, the supplementary causal-reasoning activity helped the S-group students gain a better understanding of the interdependence among the species. The S group's gains on the TQ were not significant due to a ceiling effect (most students had strong prior knowledge about the carbon cycle). On the contrary, the C-Group gained only on the CRDK questions, though less than the S-Group ($F(1,21)=21.06$, $p<0.001$). This can be explained by the C group's minimal scaffolding and, especially, the absence of scaffolds targeted towards causal reasoning.

Table 2: Normalized learning gains on categories of Ecology questions

S-Group normalized gains (P-value)			C-Group normalized gains (P-value)		
DKC	CRDK	TQ	DKC	CRDK	TQ
.865 (<0.0001)	.725 (<0.0001)	0.495 (.11)	0 (NA)	.192 (<0.01)	0 (NA)

4.2 Types of scaffolds provided and their effectiveness

A preliminary post hoc analysis of the recorded interviews was used to categorize the different scaffolds provided to the S group, as summarized in Table 3. The SS helped students become familiar with the different modeling primitives, the interface elements, and the modeling task. The MS helped with specifying correct parameter types for the modeling blocks, prompting reflection about how the commands in the model corresponded to the observed simulation behavior, and describing how different functionalities of an agent needed to be separated into different procedures. The TS were specific to the unit the students were working on, and helped clarify the task for the student. The DS suggested that students slow down the simulations to study differences between their model's behavior and that of the expert model using the simulations and plots as guides. The students had to explain possible reasons for the observed differences, and then identify procedures and primitives to modify in order to match expert behavior. The CRS prompted students to reason in causal chains about different entities of the system to help understand global system behavior. For example, as mentioned in Section 3, the S group investigated a combined model of the fish tank and then re-represented the fish-duckweed cycle on paper to reason about the roles of different species in the fish tank.

Table 3: Categories of scaffolds provided to the S-Group of students

Scaffold Category	**Description**
System-use Scaffolds (SS)	Familiarizes students with the UI and use of modeling primitives
Model-building Scaffolds (MS)	Helps students correctly parameterize modeling primitives, modularize code, and correlate models with the resultant simulations
Task-based Scaffolds (TS)	Helps clarify the particular activity being worked on in a unit
Debugging Scaffolds (DS)	Helps students identify differences between their model-generated simulations and expert simulations, elicits explanations for the reason(s) behind the differences and the methods for rectifying them
Causal Reasoning Scaffolds (CRS)	Encourages reasoning about system entities in causal chains to understand global system behavior

Although the results in Section 4.1 illustrate the overall effectiveness of the scaffolding, they do not identify the utility of particular scaffolds. However, some scaffolds likely had a larger impact than others. For example, after adjusting for variations in TCAP scores, there was a significant difference in performance between the S and C groups in the

Ecology Unit on the CRDK questions ($F(1,21)=21.06$, $p<0.001$), which may be a result of the CRS provided only to the S-Group. Reasoning causally through multiple agent interactions in the fish tank provided a global view of the ecosystem dynamics [1], which was essential for understanding concepts of balance and interdependence covered by the CRDK questions.

5. Conclusion

We have presented a learning environment which integrates computational thinking, visual programming, and agent-based modeling and simulations to help middle school students learn science across multiple domains. Our results indicate that the learning environment helped produce significant learning gains, as measured by pre- and post-test scores, for both the Kinematics and Ecology units. We also demonstrated the necessity of scaffolding and some types of scaffolds required in such an environment. As next steps, we will integrate such scaffolds into the CTSiM environment by building scaffolding tools and providing feedback via a virtual mentor agent. Also, now that we have developed common computational constructs for modeling in different domains, one of our design goals is to help students realize and exploit these commonalities.

References

[1] Basu, S., & Biswas, G. (2011). Multiple Representations to Support Learning of Complex Ecological Processes in Simulation Environments. In *Proceedings of the 19th International Conference on Computers in Education. Chiang Mai, Thailand.*

[2] Blikstein P. & Wilensky, U. (2009). An Atom is Known by the Company it Keeps: A Constructionist Learning Environment for Materials Science Using Agent-Based Modeling. *Int Jounal of Computers for Math Learning*, *14*:81–119.

[3] diSessa, A., Hammer, D., Sherin, B., and Kolpakowski, T. (1991). Inventing Graphing: Meta Representational Expertise in Children. *Journal of Mathematical Behavior*, 10, 117-160.

[4] Guzdial, M. (1994). Software-realized scaffolding to facilitate programming for science learning. *Interactive Learning Environments*, 4(1), 1-44.

[5] Hambrusch, S., Hoffmann, C., Korb, J.T., Haugan, M., and Hosking, A.L. (2009). A multidisciplinary approach towards computational thinking for science majors. In *Proceedings of the 40th ACM technical symposium on Computer science education . ACM, New York, NY, USA, 183-187*

[6] Harel, I. & Papert, S. (1991). Software design as a learning environment. *Constructionism*, 51–52.

[7] Krajcik, J., McNeill, K. L. & Reiser, B. (2008). Learning-goals-driven design model: Curriculum materials that align with national standards and incorporate project-based pedagogy. *Science Education*, 92(1), 1-32.

[8] Kynigos, C. (2007). Using half-baked microworlds to challenge teacher educators' knowing, *Journal of Computers for Math Learning*, *12*(2), 87-111.

[9] Lehrer, R., Schauble, L., & Lucas, D. (2008). Supporting development of the epistemology of inquiry. *Cognitive Development*, 23 (4), 512-529.

[10] NRC (2010). Report of a Workshop on The Scope and Nature of Computational Thinking.

[11] Sengupta, P., Kinnebrew, J., Biswas, G., and Clark, D. (2012). Integrating Computational Thinking with K-12 Science Education: A Theoretical Framework. *Proceedings of the 4th International Conference on Computer Supported Education*

[12] Sengupta, P., & Farris, A.V. (2012). Learning Kinematics in Elementary Grades Using Agent-based Computational Modeling: A Visual Programming Based Approach. In *Proceedings of the 11th International Conference on Interaction Design & Children*, pp 78 – 87.

[13] Sengupta, P., Farris, A.V., & Wright, M. (2012). From Agents to Continuous Change via Aesthetics: Learning Mechanics with Visual Agent-based Computational Modeling. *Technology, Knowledge & Learning.* Vol. 17 (1 - 2), pp 23 - 42.

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

[15] Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society*, vol. 366, pp.