Characterizing Student Theory Building in the Context of Block-Based Agent-Based Modeling Microworlds

Hillary Swanson, Utah State University, hillary.swanson@usu.edu
Kit Martin, Northwestern University, kitmartin@u.northwestern.edu
Bruce Sherin, Northwestern University, bsherin@northwestern.edu
Uri Wilensky, Northwestern University, uri@northwestern.edu

Abstract: Engaging students in authentic science practices is critical to science education. This paper presents the design of a block-based agent-based modeling microworld meant to support students' engagement in the scientific practice of theory building. We characterize the theory building of one student, Sage, in the context of her construction of a model of a zombie apocalypse. Using a fine-grained analysis, we identify critical elements of Sage's theory building, including moves pertaining to her initial articulation, testing, refinement and application of the model, as well as meta-knowledge about the nature of the model. We present these elements and then illustrate each using data from Sage's construction of the zombie apocalypse model.

Introduction

It is widely acknowledged that engagement in science practices is critical to science education. Theory building is central to the work of scientists. The Next Generation Science Standards include the theory-building practice *modeling* among eight critical science and engineering practices (NGSS Lead States, 2013). Work has been done to characterize professional modeling practices (Svoboda & Passmore, 2011) and imagine how instruction could be designed to support students in meaningful engagement in those practices in the science classroom (Passmore, Stewart & Cartier, 2009). Research has examined students' engagement in the invention, critique, and refinement of models of specific phenomena (Lehrer and Schauble, 2012; Wilensky & Reisman, 2006). Work in this area has also sought to understand how engagement in modeling develops skills for modeling practice (Manz, 2012). The present work builds on this tradition of research by examining the nature of student theory building through a mode that is highly relevant to contemporary scientific practice: computational modeling. More specifically, it characterizes, at a fine grain-size, elements of student theory building enabled and supported by block-based agent-based modeling microworlds.

Theoretical Foundations

We understand the power of these microworlds through the lens of constructionism (Papert, 1980). Like constructivism, constructionism views new knowledge as learner constructed. In addition, it posits that learning happens best through the construction of public artifacts. Constructionism can be seen in examples like programming a robot to draw shapes on the floor or programming a virtual robot to draw shapes on a computer screen. It naturally includes developing computational models to simulate scientific phenomena. We deepen our theoretical view of learning by adding the lens of the knowledge-in-pieces perspective (KiP) (diSessa, 1993). KiP views knowledge as a complex system of smaller elements that are cued variously for making sense of phenomena. Learning occurs through the reorganization and refinement of the networks of elements in the knowledge system. The naïve knowledge system is therefore viewed as a resource rich with potentially productive ideas for the construction of more expert knowledge. KiP instruction focuses on eliciting students' ideas and refining those that are productive with respect to the context at hand. KiP is synergistic with constructionism, as the construction of computational models provides students with a medium for articulating their ideas, which they can then debug, thus refining their ideas to be more scientific.

Methods

Here we present our analysis of a narrow slice of data taken from a larger study. The goal of the larger study is to understand how to scaffold student engagement in different approaches to scientific theory building, including the construction of agent-based computational models. Toward this, we are iteratively refining our design of microworlds built using the NetTango interface to Netlogo (Horn and Wilensky, 2011; Horn et al., 2014). NetTango integrates the computational power of NetLogo (Wilensky, 1999) with the accessibility of block-based modeling languages. NetTango blocks are not a full programming language, but domain-specific blocks relevant to a domain that is modeled. The *domain blocks* (Wagh, Cook-Whitt & Wilensky, 2017) are primitive elements of code that represent agents' actions and can be combined to model a specific phenomenon. We are designing domain-block agent-based modeling microworlds for simulating complex systems phenomena and studying how they support children's engagement in scientific theory building. We are currently testing our microworlds with middle school students through one-on-one 1.5-hour task-based interviews.

In this report, we seek to characterize the nature of one student's theory building in the context of a microworld designed to model the *spread of disease*. The microworld is shown in Figure 1 (Wilensky, 1998). The black box to the left is the *world* that depicts the activity of the agents that are programmed to behave according to the rules specified by the model, which the student builds using available domain blocks. The *setup* and *go* buttons are controlled by *setup* and *go* procedures that must be dragged from the block library (far right) into the modeling field (middle) and then defined by connecting with command blocks, such as *move*, *if contact person*, and *infect*.

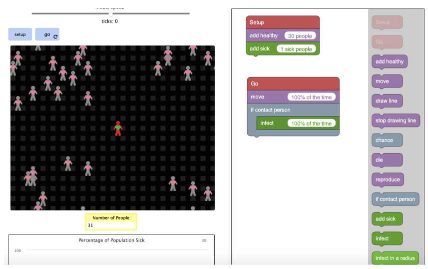

Figure 1. The *Spread of Disease* microworld featuring a student-built model.

We focus on a protocol taken from an interview with a student we call Sage. Sage was 13 years old and had just started 8th grade at a local public middle school in her small Midwestern city. During the interview, she was seated in front of a laptop featuring the microworld. She had full command of the laptop. The interviewer sat at her left and guided her through the tasks and questions of a semi-structured interview protocol. The protocol probed for Sage's background in computing and relevant science content. It then introduced her to the microworld and specific blocks for programming procedures, finally prompting her to use the existing blocks to model the spread of various diseases. Data was collected in two forms: audio and screen capture. The audio recording was transcribed. Both screen capture (with audio recording) and transcript were analyzed to identify the smaller moves through which Sage built, tested, and refined a model of a zombie apocalypse. We present these moves, next.

Findings

We found that Sage engaged in theory-building moves that pertained not only to the initial articulation, testing, and refinement of her theory, but also moves where she applied the model to make sense of zombie apocalypse phenomena. She also demonstrated a kind of modeling meta-knowledge. Below, we present a narrative of Sage's construction of the zombie apocalypse model to illustrate her theory-building moves in context.

Modeling the Zombie Apocalypse

Sage is seated at a desk in an office. She faces a laptop screen featuring the *Spread of Disease* microworld. She has been exploring the microworld for the last 30 minutes, trying out combinations of blocks and watching the resulting activity in the *world*. The microworld at this moment in time is shown in Figure 2. Sage and the interviewer have been joking about modifying the microworld to simulate a zombie apocalypse. The interviewer asks: "Is it possible with the commands we have to make a zombie apocalypse model?"

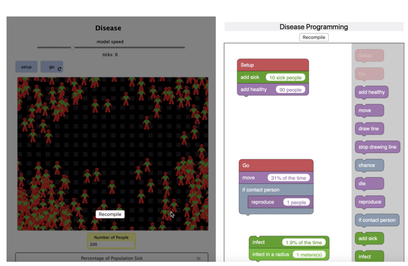

Figure 2. The microworld preceding Sage's construction of a zombie apocalypse model.

Episode 1: Building the Initial Model

Determining which elements of code are relevant to the model. Sage hesitates: "Maybe." She moves the mouse over blocks in the code library titled *draw line* and *stop drawing line*. "I don't want to draw lines." Sage is considering which blocks might be relevant to the new model, considering blocks in the library that tell the agents whether or not to draw lines. She determines that the line-drawing blocks are irrelevant to her modeling needs.

Specifying the rules of agent behavior and interaction. Sage drags the *go* procedure to just below her previously defined *setup* procedure. "But um, okay, so you have *go* and if you touch someone…" She removes the block titled *reproduce* from the if/else block *if contact person*, that is just under *move* in her *go* procedure. "Then we'll say, well then we'll say that you have infected them." She drags the *infect* block into the open jaw of the *if contact person* block. Sage is specifying rules for agent behavior and interaction in her model.

Purposefully selecting parameter values. Sage selects the *add healthy* block, which specifies how many healthy people the model begins with. She opens the parameter slider and drags it from 90 to 5. "Um, let's have five people, oh 10 people total, but five people." She selects the *add sick* block and slides the parameter to 5, then changes her mind and slides the *add sick* parameter to 1. "We'll have one. One zombie and nine people will make 10, total." She selects *add healthy* again and slides the parameter from 5 to 9. Sage appears to be selecting values for the initial population strategically, to simplify mathematical comparisons between initial and subsequent ratios of sick to healthy people as the model is run.

Identifying model limitations. Sage moves the mouse over the blocks in the code library and selects the block in the *go* procedure titled *move*, opening the parameter (which is set at 31%). "I wish we could have the zombies move slower. That's what zombies do, they're slow…" Sage is demonstrating a kind of modeling meta-knowledge by recognizing a limitation of the behaviors available in the code belonging to this particular microworld.

Distinguishing critical components from non-critical components. The interviewer responds to Sage's desire to make the zombies move more slowly, wondering out loud if there is a way to modify the model. Sage responds "It's okay. It's cosmetic […] Like it just changes how it looks. I mean it will probably change how many total people will get infected, but it doesn't really matter that much." Sage again demonstrates modeling meta-knowledge by recognizing that some aspects of the program are cosmetic, while others are functional.

Predicting model behavior based on the computational program. Sage moves the mouse over the *go* procedure and says "Wow. Um, so they move around and then it will spread." She is thinking aloud, predicting how the model will run in response to the program she has built when she presses go.

Episode 2: Testing and Refining the Model

Comparing results of model testing with predictions. Sage clicks the *setup* button in the interface and the world resets with one sick person and nine healthy people. "Oh, it did sort of end up that way." Sage is commenting on the new distribution of people and zombies in the world, where, by random chance, the people appear to be positioned as though they are fleeing the one zombie. This is something Sage had wanted to build into her program, however, as the microworld was not equipped with the necessary blocks, she had predicted the world would *not* end up looking this way. Her comment suggests that she is comparing what she sees as the resulting world behavior with how she had predicted it would behave.

Observing model behavior. Sage clicks the *go* button in the interface and for nine seconds, silently watches the random movement of the people in the world. She is observing the behavior of the agents and the spread of disease simulated by the model she has built.

Evaluating model outcomes. Sage breaks the silence, commenting "It's sort of boring 'cause they're not touching anyone." Sage is evaluating her model, characterizing it as boring, because people are not coming into contact with each other, and the disease is therefore not spreading.

Debugging the model. Sage clicks the *go* button, pausing the model run. "Maybe…" She moves the mouse over to the modeling field and selects *add sick* in the *setup* procedure, sliding the parameter from one to two. "Let's have two sick people and um…" She selects *add healthy* and slides the parameter from nine to 18. She clicks the *setup* button in the interface, refreshing the model so that there are now 20 people in the world. She clicks the *go* button in the interface. Here Sage engages in some quick debugging, increasing the number of people in the world to increase contact between zombies and people so the model will no longer be "boring."

Noticing how the model implements the code. Sage watches the model run and says to the interviewer: "I've noticed that for contact only their hearts have to touch." Through careful observation, Sage has noticed the subtleties of how the model implements the code (here, that agents must be on the same patch to infect each other).

Episode 3: Applying the Model

Describing the aggregate phenomenon. Sage continues to watch the model run and notes: “But yeah, it's sort of stopped.” She points at the graph under the world that documents the percentage of population that is sick (or zombies). “Like, it’s plateauing.” Sage describes the aggregate phenomenon in terms of the graph. The graph is plateauing, indicating that the zombie population may have reached a maximum.

Explaining the aggregate phenomenon as a result of agent interactions. The interviewer asks Sage why she thinks the graph looks the way it does. She responds: “Well, [...] if you bumped into someone, the chances that you were bumping into a healthy person were greater than the chances that you were bumping into a zombie. […] But now there are twice as many people. Twice as many zombies so the chance that you're bumping into a zombie when you bumped into a random person is doubled.” Sage uses probabilistic reasoning to explain how the early part of the curve is, as she says on further questioning, “exponential.”

Conclusions

The present study found that one student demonstrated critical elements of scientific theory building in the context of a block-based agent-based modeling microworld. These elements included moves pertaining to the initial articulation, testing, refinement, and application of a computational model, and meta-knowledge regarding the nature of the model. These moves and meta-knowledge were unpacked and illustrated in the context of the student’s construction of a computational model of a zombie apocalypse. These findings contribute to literature concerned with the design of learning environments for engaging students in scientific theory building through computational modeling, and literature concerned with characterizing student theory building as located on a continuum with scientist theory building.

References

diSessa, A. A. (1993). Toward an epistemology of physics. *Cognition and instruction*, *10*(2-3), 105-225.

Horn, M., S., Brady, C., Hjorth, A., Wagh, & Wilensky, U. (2014). *Frog pond: a codefirst learning environment on evolution and natural selection.* Paper presented at the 2014 conference on Interaction design and children.

Horn, M., & Wilensky, U. (2011). NetTango [Computer Software]. Evanston, IL: Center for Connected Learning and Computer Based Modeling, Northwestern University.

Lehrer, R., & Schauble, L. (2012). Seeding evolutionary thinking by engaging children in modeling its foundations. *Science Education*, *96*(4), 701-724.

Manz, E. (2012). Understanding the codevelopment of modeling practice and ecological knowledge. *Science Education*, *96*(6), 1071-1105.

NGSS Lead States (2013) Next generation science standards: for states, by states. The National Academies Press, Washington, DC

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc.

Passmore, C., Stewart, J., & Cartier, J. (2009). Model-based inquiry and school science: Creating connections. *School Science and Mathematics*, *109*(7), 394-402.

Svoboda, J., & Passmore, C. (2013). The strategies of modeling in biology education. *Science & Education*, *22*(1), 119-142.

Wagh, A., Cook-Whitt, K., & Wilensky, U. (2017). Bridging inquiry-based science and constructionism: Exploring the alignment between students tinkering with code of computational models and goals of inquiry. *Journal of Research in Science Teaching*, *54*(5), 615-641.

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

Wilensky, U. (1998). NetLogo Virus model. http://ccl.northwestern.edu/netlogo/models/Virus

Wilensky, U., & Reisman, K. (2006). Thinking like a wolf, a sheep, or a firefly: Learning biology through constructing and testing computational theories—an embodied modeling approach. *Cognition and instruction*, *24*(2), 171-209.

Acknowledgements

This work was supported by the National Science Foundation (1842375).