Refining Student Thinking through Computational Modeling

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

Abstract: Scientists systematically refine their thinking through theory-building practices. Computational modeling is a key mode of theory building in the 21st century. This study investigates how engagement in the construction and refinement of computational models helps students refine their thinking. Specifically, it presents a microgenetic analysis of an episode during which a student refined their understanding of how disease spreads by building and refining a computational model of an Ebola epidemic. The analysis decomposes the episode into nine theory-building steps and locates the shifts in the student's thinking that occur in each step. It draws connections between these shifts and the theory building activity.

Introduction

In recent years there has been a shift, in science education, toward engaging students in epistemic practices (Ford & Foreman, 2006; Osborne, 2016). This builds on the perspective that students should construct scientific understanding through the kinds of knowledge-building practices scientists use to make sense of the world (Schwarz, Passmore, Reiser, 2017). Theory building is a central epistemic practice of science (Suppe, 1974). We define theory building as a family of practices through which scientists and students systematically refine theoretical knowledge artifacts, including explanations and models (Swanson, 2019). We further posit that as artifacts are refined, thinking is refined. This aligns with empirical work from sociology and cognitive psychology, which shows that scientists build knowledge by refining artifacts and existing ideas (Dunbar, 1997; Latour, 1999), and Einstein's (1936) notion that "the whole of science is nothing more than a refinement of everyday thinking."

Today, many scientists build theory by constructing computational models that, when run, produce outcomes that can be explored and compared with experimental findings (Weintrop et al., 2016; Foster, 2006). A number of research programs have designed and investigated ways of engaging students in theory building through computational modeling. diSessa (1995) used the Boxer computational modeling environment to help high school students articulate and refine their ideas about Newtonian dynamics. Sherin (2001) looked broadly at the possibility of using programming as a language for expressing simple physical ideas. Wilensky and colleagues have investigated student engagement in computational modeling of complex systems phenomena across domains using NetLogo (Wilensky, 1999; Wilensky & Reisman, 2006). More recently, Wilensky and colleagues have examined student model construction using NetTango (Horn, Baker, & Wilensky, 2020), a block-based interface to NetLogo. These studies examine the development of both scientific understanding and computational thinking that result from student engagement in computational modeling (Horn et al., 2014; Wagh & Wilensky, 2017).

Building on this tradition, the present research investigates how computational modeling helps students refine their thinking by driving them to refine a computational model. Specifically, we examine how one student refined her understanding of disease spread as she refined a computational model of the Ebola virus. We identify the smaller shifts in thinking that resulted from her engagement in constructing, testing, debugging, and making sense of the computational model.

Developing an understanding of the complex dynamics of disease spread is of particular relevance today, given the COVID-19 pandemic. At the writing of this paper, COVID-19 has infected approximately 48 million people worldwide and killed 1.2 million ("WHO Coronavirus Disease Dashboard," 2020). In addition to factors such as symptom-free carriers and a long incubation period, the success of the disease can also be attributed to the fact that it is just fatal enough to kill a percentage of those who are infected, but not fatal enough that those people become obviously sick and die before interacting with, and potentially infecting, many other people (Murray & Mearns, 2020). This quality of the coronavirus is at odds with commonsense conceptions about the fatality of a disease, which is that a disease that is very deadly to an individual must be very deadly to a population (Wilensky & Abrahamson, 2006).

This is a good example of what Wilensky and Resnick (1999) refer to as *levels slippage*. Levels slippage is a common error in novice reasoning about complex systems phenomena, whereby a characteristic at the individual level is mapped directly to the aggregate-level. In reasoning about disease, levels slippage occurs when a disease that is deadly to an individual is assumed to be deadly to a population. In reality, as our student learns, a disease that is very deadly to an individual may not be deadly to a population (without additional interacting factors), because it will sicken and kill its carriers before they have the opportunity to infect others. Understanding this complex dynamic between individual and aggregate levels is one of the key understandings that results from our focal student's engagement in computational modeling.

Theoretical Foundations

Our analysis is founded on the basic premise of constructivism, which holds that new knowledge is built on the foundation of prior knowledge (Piaget, 1971). A constructivist epistemology known as knowledge-in-pieces (KiP) (diSessa, 1993), characterizes the process of knowledge construction as a refinement of a complex system of existing knowledge elements. The elements are cued variously for sensemaking, depending on a person's level of expertise. For the novice, the elements are loosely interconnected and employed in a manner that is highly context-dependent. For the expert, the elements are more reliably connected and cued more consistently in contexts where they are productive. Learning (or the movement from novice to expert) occurs through the reorganization and refinement of the networks of elements in the knowledge system. For this reason, the naïve knowledge system is viewed as a resource rich with potentially productive ideas for the construction of more expert knowledge networks (Smith, diSessa & Roschelle, 1994). This stands in contrast with the "misconceptions" perspective which regards naïve knowledge as incorrect knowledge structures that need to be removed and replaced (Chi, 2008).

KiP instruction focuses on eliciting students' ideas and refining those that are productive with respect to the context at hand. The construction of computational models provides an opportunity for students to articulate and refine their ideas. This aligns with constructionism (Papert, 1980), which argues that learning happens best through the construction of public artifacts, such as computational models. We posit that the systematic refinement of artifacts that is accomplished through scientific theory building, and in particular, computational modeling, affords and supports a refinement of thinking. This study seeks to understand how computational modeling supports learning. KiP views learning as the sum over smaller shifts, or refinements, in thinking over time. To understand how the construction and refinement of computational models supports refinement of thinking, we examine a case where learning occurred, looking for shifts in the student's thinking over time and connections between those shifts and the student's theory-building activity.

Methods

This paper presents the results of an analysis of data taken from a larger study. The goal of the larger study is to understand how to engage middle school students in different approaches to scientific theory building, including the construction of computational agent-based models. To lower the threshold for participation in computational modeling, we are designing block-based computational modeling microworlds using the NetTango platform (Horn and Wilensky, 2011). 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. Previously called *semantic blocks (*Wilkerson-Jerde & Wilensky, 2010) and now called *domain blocks* (Wagh, Cook-Whitt & Wilensky, 2017) the blocks are primitive elements of code that represent agents' actions and can be combined to model a specific phenomenon. We are designing domain-block libraries for simulating complex systems phenomena and studying how children use the blocks to build and refine computational models, and how their thinking is refined as a result. The present study seeks to characterize, at a fine grain-size, how theory building supports refinement of thinking and asks the question: *How does building and refining a computational model of the Ebola epidemic refine a student's thinking about the spread of disease?*

To address this question, we designed a NetTango modeling microworld that enables a user to build a model of the spread of disease. We tested the microworld with four students during individual 1.5-hour interviews. During the interviews, each student was seated at a desk that displayed a laptop featuring the computational modeling microworld. The student had full command of the laptop. The interviewer sat at the student's left and guided them through tasks and questions contained in a semi-structured interview protocol. The protocol began with an introduction to the study, its purpose, and the overall timeline of the interview. It then probed for students' background in computing and asked a few basic content-related questions (e.g., Do you know what the word epidemic means?). The protocol then moved on to introduce the student to the modeling microworld interface and specific blocks for programming procedures. It then prompted students to use the existing blocks to model diseases (e.g., Ebola). Data was collected during each interview in three forms: video, screen capture, and audio recordings.

The present report focuses on results from an interview with a student we call Sage. Sage was 13 years old and had just started the 8th grade at the local public middle school in her small Midwestern city. She had been introduced to block-based programming when her 7th grade science class participated in *Hour of Code*, an event hosted by schools all over the world to encourage students' interest in computing. The present report focuses on an excerpt from her exploration of the *Spread of Disease* modeling environment.

The *Spread of Disease* modeling environment is pictured below (Figure 1). The screenshot shows the modeling environment with a model that has been built and initialized. The black box to the left is the *world* which depicts the activity of the agents that are programmed to behave according to the rules specified by the model. The *setup* and *go* buttons are controlled by *setup* and *go* procedures (red blocks) that the user must drag from the block library (far right) into the modeling field (middle) and then define by connecting with blocks (purple, grey, and green) defined by specific commands, such as *move*, *if contact person*, and *infect*.

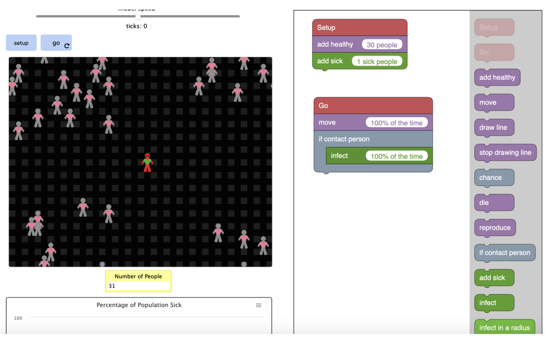

Figure 1. Screenshots of the *Spread of Disease* modeling microworld.

An audio recording of the interview with Sage was transcribed. Both screen capture (with audio recording) and transcript were analyzed to look for evidence of learning. A shift in thinking was identified during an activity in which she attempted to build a model of an Ebola epidemic. A microgenetic analysis (Siegler, 2007) of the episode was conducted to understand the smaller shifts through which her thinking was refined. The episode was ultimately divided into nine steps, each of which featured engagement in aspects of the theory-building process. These included building, testing, and debugging the computational model, and making sense of its simulated outcomes. Having created a temporal decomposition of the episode into these steps, the shifts in thinking that occurred during each were characterized and connected with the theory-building activities.

Findings

We present an episode from Sage's interview to illustrate how she refined her thinking as she refined her model. Ultimately, she developed a sense that a disease that was very deadly to an infected individual can actually be less deadly to a population. In this way, she may have ameliorated a common instance of levels slippage and developed a more sophisticated understanding of the complexity of the phenomenon.

Modeling the Spread of Ebola

Sage is seated at a desk in an office, the interviewer sits at her left. She is looking at a laptop screen on which the *Disease Spread* modeling microworld is open. She has been exploring the microworld for the last 25 minutes, trying out combinations of blocks and watching the resulting activity in the *world*. The interviewer wants to focus Sage on the task of modeling an epidemic and asks: "What if we were to try to make a model of the Ebola epidemic?" The steps below represent a temporal decomposition of Sage's trajectory through the modeling activity, highlighting her shifts in thinking.

Step 1: Building an Initial Model

Sage drags blocks into the programming field, specifying agent rules and constructing her initial model of how Ebola spreads through a population. To aid her design, she browses the internet for information on Ebola, including rates of transmission and death. Turning back to her model, she purposefully selects parameter values that align with what she has found.

> Sage: If sick die 50%, I'll have that, but maybe infect like 10% 'cause like I bet 10% of the time you're like, you accidentally like cough on someone and get their like your saliva in their mouth or

something. And like that's, that's gross to think about it but um, infect, well yeah, infect like 10%. […] And then if you're sick, then you die. Let's say 50% 'cause that about ....

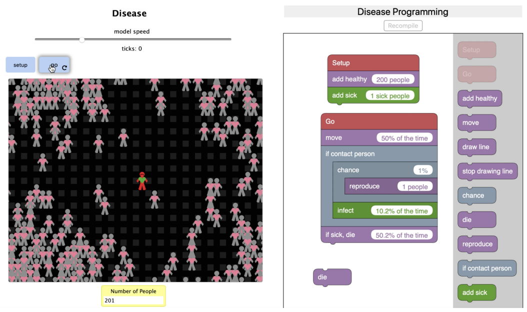

Figure 2. Sage's initial model of Ebola, before she presses "go."

As shown in Figure 2, Sage has set the model to initialize with 200 healthy people (grey bodies, pink hearts) and one sick person (red body with a green heart). With each tick of the clock, each person in the world will move with a 50% probability. If they contact another person, there is a 1% chance they will reproduce, and if they are sick, a 10% chance they will infect the person. If they are sick, there is a 50.2% chance they will die.

Step 2: Testing the Initial Model

Sage runs her model and observes as the single green person in the world disappears almost instantly.

Step 3: Making Sense of the Initial Model

Sage laughs, suggesting she is surprised by the outcome of the model-run. This is evidence that she expects a disease that is very deadly to infected individuals to be deadly to a population. She tries to understand what is happening at the agent level that would have caused the aggregate-level outcome, namely, why the sick person has disappeared, and why the disease did not spread and diminish the population.

Sage: What happened?!

Interviewer: And the number of, what's it look - what happened? Oh the number-

Sage: He died probably.

Interviewer: Oh yeah

Sage: 'Cause, 'cause it's like a 50, 50

Interviewer: Yeah. Yeah. Yeah. And he was already sick. So.

Sage reasons that the single infected person died quickly because they had a 50% chance of dying each tick of the clock.

Step 4: Debugging the Initial Model

Sage's first reaction is that there may indeed not be a way to program the model so that Ebola will spread as she expects. On second thought, she corrects herself, deciding that she could remedy the problem by adding more people.

Sage: Hmm. Maybe, I don't, I don't think there really is a way to like, do, to do that, if that, because no, no, I could have more people. Um, I'm being ridiculous. Let's have 20 sick people, 21.

She attempts to debug the model by changing a parameter value, increasing the initial number of sick people in the world from 1 to 21. It appears she is testing an extreme value, thinking that it will surely help the disease to spread among the population.

Step 5: Testing the Revised Model

Sage runs her revised model to test the effect of her modified parameter and observes as the sick people (the 21 initially infected people and the people they infected) die within six ticks, leaving only healthy people, as before.

> Sage: Then maybe they'll, Oh wait, no they all died.

Sage appears to be surprised by this result. This is evidence that the outcome of the model-run does not match her expectations. It suggests that she may expect a disease that is very deadly to infected individuals to be similarly deadly to a population.

Step 6: Making Sense of the Model Outcome

In an attempt to get Sage to make sense of the relationship between agent- and aggregate-level in this complex-systems phenomenon, the interviewer asks Sage what she thinks is happening.

> Sage: Um, epidemics are hard to start.
>
> Interviewer: Well, yeah. Well, when, what's hard to start about this one, do you think?
>
> Sage: Well, I think because, um, that it has a high fatality rate.
>
> Interviewer: Yeah
>
> Sage: And well maybe because the infectivity is so low, 'cause then if, um-

Sage appears to have drawn an important conclusion about complex-systems dynamics in the context of the spread of disease. This is that a high fatality rate will cause the carriers of a disease to die before they have a chance to spread it, making it in fact *less deadly* to the population. In arriving at this conclusion, Sage is beginning to construct a more accurate understanding of the connection between agent and aggregate levels. At the same time, she may be holding onto an intuition that the disease *should* be more deadly to the population, as she tries to think of another aspect of the model she could modify so that it produces the aggregate-level phenomenon she expects.

Step 7: Debugging the Model

In an attempt to debug the model and cause the disease to be more deadly to the population, Sage purposefully modifies a parameter setting, increasing the rate of infection.

> Interviewer: Do you want to change that and
>
> Sage: Yeah, let's, maybe people in this fictional city really like spitting in each other's mouth.
>
> Interviewer: OK
>
> Sage: Like maybe, maybe, maybe they're just like kissing everyone and that makes them more infectious.

Sage increases the probability of infection from 10.2% to 50.2% and recompiles the code.

Step 8: Testing the Revised Model

Sage runs her revised model to test the effect of her modified parameter. She observes as the sick people (the 21 initially infected people and the people they infected) die within 14 ticks, leaving only healthy people, as before. This outcome matches those of the previous runs, where the sick people die relatively quickly, leaving the world filled with only healthy people.

Sage: Setup. Go. They're still all...

As before, Sage appears to be surprised by this outcome. Again, this suggests that she may expect a disease that is very deadly to infected individuals to be similarly deadly to a population.

Step 9. Making Sense of the Model Outcome

The interviewer engages Sage in making sense of the outcome of the model run. She wants Sage to consider what it would mean if her model were actually not flawed, but rather, a good approximation of a disease that is very deadly to individuals. In reality, the model is not a good approximation of Ebola, as that disease is very deadly to individuals yet has succeeded as an epidemic due to factors that are not representable by the block-based modeling infrastructure (e.g., healthy care-givers tending to their dying and deceased loved ones) (Hewlett & Hewlett, 2008).

Interviewer: Wow. Is that what you expected to happen?

Sage: No

Interviewer: Well, what if you assume that your model is actually a good approximation of Ebola?

Sage: Yeah

Interviewer: What, what conclusion would you

Sage: Well, like highly deadly diseases are hard to start epidemics, but when you do, they like really, really start 'cause, um, like, and like maybe, Hmm. I wish you could have like a, like there's, I don't know, like, 'cause like everyone's touching each other and they're like making people, let's take this out right now. Um, but, um- [...] they're reproducing and um, I don't know. It just, like, it, epidemics are hard to start. Um, even if you have people infecting each other like all the time, like they, like all of those people just died.

Sage compares the outcome of the model-run with her expectations and admits that the two do not align. She reflects on what she can learn from the model outcome and further expresses the notion that it is hard to get an epidemic to start when a disease is very deadly to an infected individual. It appears that she is clinging to an intuition that if such diseases do manage to spread, "they really start." Still trying to find a way to build the model of Ebola so that it wipes out the population (aligning with what she has learned about the disease through the media) Sage removes the "reproduce" command from the "go" procedure.

Discussion

Analysis of this episode shows that Sage refined her understanding as she refined her model. She initially expected Ebola to spread and kill a larger percentage of the population. When her initial model did not produce her expected result, Sage modified parameter values (e.g., the initial number of sick people in the world and the probability of infection on contact) in an attempt to debug her model so the disease would spread. On testing her revised models, she found they still produced the same result. On reflection, she noted that "highly deadly diseases are hard to start epidemics" because their high fatality rates quickly wipe out the infected individuals who would otherwise spread them. In this way, by revising and testing her model, Sage refined her understanding. It is also apparent that her intuitions about disease are appealing and difficult to let go of, as evinced by her statement near the end of the activity "when you do [start an epidemic], they like really, really start."

Her intuitions about Ebola are not unfounded, as she knows that the disease has indeed been an epidemic in Africa, and continues to be a risk. Ebola was able to become an epidemic despite its deadliness, however, because of other factors (e.g., family members caring for their dying and mourning their deceased) that were not represented in the model. Therefore, the important learning that occurred for Sage was a more nuanced understanding of the dynamics of disease spread as a complex-systems phenomenon, where the deadliness of a disease at the individual level does not directly correspond to the deadliness at the level of the population. In the activity that followed this episode, Sage constructed and refined a model of the flu. The activity helped to further refine her understanding, and she arrived at the conclusion "If [a disease] is like super, super fatal then it won't spread because like it will be super, super rare because like, um, then they let it, we'll just kill out all its carriers."

The shift in Sage's thinking over the course of the activity appears to be a correction of a classic case of levels slippage whereby a person determines that a phenomenon occurring at one level is directly caused by the same phenomenon at another level. In this case, a slippage between levels would be attributing the deadliness of a disease at the population level to the deadliness of a disease at the individual level. As Sage learns, however, just because a disease is deadly to an infected individual does not mean that it is deadly to a population.

Taking a step back and looking at what the analysis tells us about Sage's learning, we see that refining the model helped Sage to refine her thinking. We postulate that all kinds of theory building engage students in refining artifacts that represent their knowledge, therefore helping them refine their thinking. It is important to acknowledge that this particular approach to theory building - building a *computational agent-based model* - afforded particular refinements in Sage's thinking.

The fact that Sage was building a *computational* model meant that she could test her theory of how disease spread by clicking "go" and simulating what would happen in a world that was set up according to the initial conditions she selected, and behaved according to the rules she had specified. Had Sage not been able to test her revised models, she would not have seen that her modifications did not yield her intended results. She might have continued to think that a disease that was deadly to an individual was similarly deadly to a population, if one starts with enough sick people, or if the probability of infection is high enough. Testing out these agent-level modifications may have helped Sage see that they in fact had no significant impact on the phenomenon. Revising and testing her model helped her to see that regardless of the number of people initially sick or how infectious the disease was, if it was quite deadly to an individual, it would die out before it could spread, producing the opposite of what many people intuitively expect. The computational nature of her model supported refinement through *debugging*, whereby Sage was able to make small corrections to the model and test them, iteratively refining her model and her thinking.

The fact that the computational modeling microworld was *agent-based* meant that her model would be well-suited to simulating complex-systems phenomena. This, in turn, would help her more clearly see relationships between agent and aggregate levels, potentially clearing up any slippage between levels. Furthermore, the model was built using a block-based interface to the underlying computational agent-based modeling language, NetLogo. The blocks both afforded and constrained Sage's theory building. They afforded her entry into computational agent-based modeling, an activity which would otherwise have been inaccessible without some training in the language. It constrained her theory building because there were only particular commands available. Sage noted a number of times, during the interview, that she wished there had been ways of accomplishing something that was not currently afforded by the block-based modeling infrastructure (e.g., making the person infect a certain number of people before they died).

In addition to making a contribution to work concerned with engaging students in science practices, this work makes an empirical contribution to literature on conceptual change, by offering a microgenetic account of how one student's thinking changed as a result of engagement in theory building. Microgenetic accounts of learning are few and necessary for understanding the finer mechanics of learning (diSessa, 2014). Finally, our work makes a contribution by demonstrating an important connection between knowledge-in-pieces and constructionism: that through the refinement of a physical and publicly shareable artifact, students can refine their thinking.

References

Chi, M.T.H. (2008). Three types of conceptual change: Belief revision, mental model transformation, and categorical shift. In S. Vosniadou (Ed.), *Handbook of research on conceptual change* (pp. 61-82). Hillsdale, NJ: Erlbaum.

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

diSessa, A. A. (1995). Designing Newton's laws: Patterns of social and representational feedback in a learning task. In R.-J. Beun, M. Baker, & M. Reiner (Eds.), *Dialogue and Interaction: Modeling Interaction in Intelligent Tutoring Systems.* Berlin: Springer-Verlag, 105-122.

diSessa, A. A. (2014). The construction of causal schemes: Learning mechanisms at the knowledge level. *Cognitive science*, *38*(5), 795-850.

Dunbar, K. (1997). How scientists think: On-line creativity and conceptual change in science. In *Conceptual structures and processes: Emergence, discovery, and change.* American Psychological Association Press.

Duschl, R. A. (2008). Science education in 3-part harmony: Balancing conceptual, epistemic and social goals. Review of Research in Education, 32, 268 – 291.

Duschl, R. A., Schweingruber, H. A., & Shouse, A. W. (Eds.). (2007). *Taking science to school: Learning and teaching science in grades K-8* (Vol. 500). Washington, DC: National Academies Press.

Einstein, A. (1936). Physics and reality. *Journal of the Franklin Institute*, *221*(3), 349-382.

Ford, M. J., & Foreman, E. (2006). Redefining Disciplinary Learning in Classroom Contexts. *Review of Research in Education*, *30*(1), 1–32.
Foster, I. (2006) 2020 computing: a two-way street to science's future. Nature 440(7083):419
Hewlett, B. S., Hewlett, B. L. (2008). Ebola, culture, and politics: the anthropology of an emerging disease. United States: Cengage Learning.
Horn, M.S., Brady, C., Hjorth, A., Wagh, A., & Wilensky, U. (2014). Frog Pond: A code first learning environment on natural selection and evolution. Proceedings of IDC 2014.
Horn, M.S., Baker, J. & Wilensky, U. (2020). NetTango Web [Computer Software]. Evanston, IL: Center for Connected Learning and Computer Based Modeling, Northwestern University. https://netlogoweb.org/nettango-builder
Latour, B. (1999). *Pandora's hope: essays on the reality of science studies*. Harvard university press.
Osborne, J. (2016). Defining a knowledge base for reasoning in science: the role of procedural and epistemic knowledge. In R. A. Duschl & A. S. Bismack (Eds.) Reconceptualizing STEM education: The central role of practices. New York: Routledge.
Murray, A., & Mearns, E., (2020, April 29). Understanding the disease: Why is COVID-19 so successful? Retrieved from https://newseu.cgtn.com/news/2020-04-29/pandemic-playback-2-understand-disease-PIHbaLQN8Y/index.html
Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc.
Piaget, J., (1971). *Psychology and Epistemology: Towards a Theory of Knowledge*. New York: Grossman.
Schwarz, C. V., Passmore, C., & Reiser, B. J. (2017). *Helping students make sense of the world using next generation science and engineering practices*. NSTA Press.
Sherin, Bruce L. "A Comparison of Programming Languages and Algebraic Notation as Expressive Languages for Physics." *International Journal of Computers for Mathematical Learning* 6, no. 1 (2001): 1–61.
Siegler, R. S. (2007). Microgenetic analyses of learning. In *Handbook of child psychology* (Vol. 2). https://doi.org/10.1002/9780470147658.chpsy0211
Suppe, F. (1974). The search for philosophic understanding of scientific theories: Introduction. In Kuhn, T. S., & Suppe, F. (Eds.), *The structure of scientific theories* (p.244-254). Chicago: University of Illinois Press.
Swanson, H. (2019, April). *Cultivating a theoretical turn-of-mind.* Paper presented at the annual meeting of the American Educational Research Association, Toronto, Canada.
Wagh, A., & Wilensky, U. (2017). EvoBuild: A Quickstart Toolkit for Programming Agent-Based Models of Evolutionary Processes. Journal of Science Education and Technology, 1-16.
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.
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. (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., & Abrahamson, D. (2006). *Is a disease like a lottery? Classroom networked technology that enables student reasoning about complexity*. Paper presented at the annual meeting of the American Educational Research Association, San Francisco, CA.
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.
Wilensky, U., & Resnick, M. (1999). Thinking in levels: A dynamic systems approach to making sense of the world. *Journal of Science Education and technology*, *8*(1), 3-19.
Wilkerson-Jerde, M., & Wilensky, U. (2010). *Restructuring Change, Interpreting Changes: The DeltaTick Modeling and Analysis Toolkit*. Paper presented at the Constructionism 2010 Conference, Paris.
WHO Coronavirus Disease Dashboard (2020, November 5). Retrieved from https://covid19.who.in

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