Developing Accessible and Sustainable Computational Modeling Tools in Learning Science: What is Next?

Chairs: Tamar Fuhrmann, TC Columbia University; Sharona T Levy, Faculty of Education, University of Haifa
Discussant: Uri Wilensky, Learning Sciences & Computer Science, Northwestern University

Introduction

In this symposium, we address two issues that extend the body of research on students' computational modeling: enabling greater access to computational modeling activities and sustaining such activities within educational settings. Modeling has become an essential part of many scientific disciplines -- a core element in professional scientific practice for formulating and testing hypotheses, justifying new claims, and explaining natural and artificial phenomena (Pluta et al., 2011; Halloun, 2011). The involvement of students in developing a scientific model has been shown to have many pedagogical affordances: models can be used as inquiry tools, creating meaningful learning opportunities (Gilbert et al., 2016; Passmore et al., 2014; Schwarz et al., 2012); additionally, models enhance students' disciplinary knowledge (Passmore et al., 2014) and help develop their understanding of the epistemology behind constructing scientific models (Lehrer et al., 2006; Schwarz et al., 2005). In addition, it has been found that students who engage in designing their models also develop a critical perspective about models and metamodeling knowledge (Wagh et al., 2018; Fuhrmann et al., 2018). The last few years have seen modeling activities gradually take root as part of science education. Incorporating modeling has become a key goal in many countries' national standards in the past decade and a core component of school curricula (Gilbert et al., 2016), as evidenced by the fact that national policy documents have centered modeling activities in the curriculum (National Research Council, 2007; NGSS, 2013) and by the fact that modeling is woven throughout the Next Generation Science Standards (NGSS).

With the realization that constructing models is essential and computational modeling should be learned in schools, new computer modeling environments and designs have proliferated. As a result, today, a wider range of robust, mature, and carefully designed software and hardware tools are available for modeling in K-12. Nevertheless, several issues remain to be addressed before students engage in meaningful modeling activities, create their models based on data, and share them and discuss them with others, thus participating in activities aligned with authentic scientific practices.

The papers in this symposium focus on the big question of what is next in domain-specific computational modeling languages/environments for K-12 students. To this end, we address two relevant issues, namely accessible and sustainable development. Here, we define *accessible* development as development that enables students with different interests, contexts, and abilities to begin interacting with the learning materials. This can be done by lowering the threshold to begin using the tools and/or by catering to different forms of expression to motivate students to begin engaging with the tools. Increasing accessibility to a learning environment is necessary when considering schools with teachers who may or may not have the skills to support a class of students constructing computational models. Students' need for support in constructing a computational model is significant, and learning design needs to consider how to offload some of this responsibility from the teacher to the computational toolkits themselves.

We define *sustainable* development as creating learning materials that can continue generating new activities over time, be it for additional content, for higher ability levels, or for other extensions. Ensuring the sustainability of innovative and expressive activities such as computational modeling in schools is an important endeavor that has been undertaken by several leaders and researchers (Horn et al., 2011; Kelter et al., 2021). Building upon these efforts, there is a need to continue developing and diversifying the strategies for supporting the continued use of computational modeling in science learning. In the process of variation and selection in different contexts, this would advance the broader vision of democratizing people's use of computation as a tool to express their assumptions regarding phenomena in the world and of generating dynamic representations that can be compared with these assumptions.

Paper 1 presents a nuanced framework that differentially integrates divergent data sources with modeling activities and labs. Paper 2 introduces the Much.Matter.in.Motion platform for modeling, which unifies computation across systems in chemistry and physics. It demonstrates the idea with research into learning about electricity through computational modeling. Paper 3 presents research with a novel extension to the Scratch environment that enables topical computational modeling. Further, this paper looks at related research that highlights how students' ideas are expressed visually and evolved through the programs they construct. Paper 4 presents work and research with the StarLogo Nova platform. It looks at how programming blocks might be used for learning a variety of scientific phenomena across the curriculum while enabling the opening of these blocks

 © ISLS

to get a deeper understanding of the computational constructs. Finally, Paper 5 presents a framework for making computational modeling in science a sustainable activity in middle schools.

Each of the papers in the symposium addresses accessibility and sustainability in different ways. Regarding accessibility, common to all the papers is the use of functional computational blocks that are limited to a subset of phenomena and form a library of concepts that students can use to construct their models. By limiting the library of concepts and aligning them with a variety of intuitive concepts, it is made possible for the students to immediately begin constructing with relatively little support. In Paper 4, these blocks can be opened up during the learning process, supporting students as they gradually come to understand the more basic computational constructs the blocks are built from.

As to sustainability, the papers provide an array of strategies. Paper 5 specifically addresses sustainability and presents a framework with several components that could support the continued use of modeling in schools. Papers 1 and 3 both employ Bifocal Modeling, an approach that combines the use of labs and models to facilitate easy comparison. In this way, additional practices common in science education are integrated with modeling, thus strengthening the holding power of those practices.

Similarly, Paper 2 intersperses physical experiments into the learning unit, with the express goal of helping students transition back and forth between experience and representation. Integrating standard science education practices into learning units that engage with modeling can also be seen in Paper 1, where a framework for integrating data practices with modeling is presented. In Paper 4, using the mature StarLogo Nova platform and beginning with a more limited library of blocks enables the gradual transition to a full-blown modeling toolkit by gradually unpacking these blocks, providing for an extended developmental learning process that can continue even over several years. Another strategy used in paper 4 to increase sustainability is creating several libraries, each of which addresses different curriculum topics so that modeling can be integrated into learning throughout the school year. Finally, in Papers 2, 3 and 4, the limited set of blocks enables modeling a broader range of phenomena — systems in chemistry and physics — by capturing the computational similarities among diverse phenomena; this supports the vision of continued use of the same platform as students progress in their studies and learn new topics in these sciences.

The session will open with introductory words. Five paper contributions will then follow. Our discussant, Prof. Uri Wilensky, will draw upon the various strategies developed to increase accessibility and sustainability to interrogate some of their implicit assumptions underlying the learning designs and the related research presented in the papers. The symposium intends to integrate and focus our efforts on furthering scientific computational modeling in schools, streamlining it to the burning issues we have addressed. To summarize, the papers comprising this symposium acknowledge several essential issues regarding computational modeling in science. The more we know about this topic, the more we are driven to further explore and extend our understanding. We hope that the symposium helps democratize this important science learning component by addressing key issues that can make modeling more widely available for all learners and relevant throughout their studies.

Paper 1: Comparing and Contrasting Real-World Data and Models: Cognition, Pedagogy, and Implementation

Engin Bumbacher, Tamar Fuhrmann & Paulo Blikstein

Empirical data plays a crucial role in scientific modeling: scientists use such data to construct, refine or decide between possible models (Passmore et al., 2009), and assess the adequacy of a given model by the degree to which that model is able to explain the data (Passmore & Svoboda, 2012). However, approaches to model-based learning (MBL) have predominantly focused on model-based and data-based practices separately (Bumbacher et al., 2018). While these reductive approaches have helped students improve in the respective practices (Zimmerman, 2000), they have yet to prove successful in helping students develop a deeper grasp of the practices and epistemic understanding of the scientific endeavor (Berland et al., 2016; Chinn & Malhotra, 2002).

Recent technological advances hold the promise to support more integrative approaches that interweave models and data. For example, the Bifocal Modeling framework (Blikstein, 2014, 2016) suggests integrating technologies for real-world data collection with technologies for computational modeling to enable real-time comparisons of simulated and real data. Studies have shown that activities designed according to this framework can develop students' conceptual understanding and meta-modeling competences (Blikstein, 2014; Fuhrmann et al., 2018), but more work is needed to examine the types of learning opportunities that arise from comparing real-world data and computational models, as well as the conditions under which they arise.

Integrating real-world data in the modeling process provides learning opportunities that would not arise if students just focused on the models. For example, real-world data might encourage students to explore more deeply the underlying features of a phenomenon (Schwarz, Akcaoglu, Ke, & Zhan, 2013) in order to decide on criteria of model-fit. Similarly, real-world data might also inform the evaluation of models, when students have to decide how well a model fits the data, and what conclusions are justified by the data (Holmes, Wieman, & Bonn, 2015). But there is not just one type of data source, and there is not just one way of comparing models and data. We argue that the type and quality of learning opportunities arising from comparing models and data are dependent on the type of data source being used, and the way models and data are brought together. For example, juxtaposing a model-based simulation of bacterial growth with time lapse recordings of an actual experiment might encourage students to engage differently in the comparison of models and data than merely contrasting the growth curves of the simulated and real bacteria.

Thus, developing technologies and pedagogical approaches that productively integrate data and models necessitates a better understanding of the types of data sources and types of comparisons of models and data. This is particularly relevant today, as the past decade has seen an explosion of novel empirical data sources and means of collecting data (Lee & Wilkerson, 2018). Novel technologies like probeware, wearable sensor technologies or remote laboratories enable individuals to collect quantitative data continuously and in real-time. The internet provides access to large, publicly-available datasets in the form of spatial data, quantitative data tables, or qualitative data such as videos. Further, computer vision algorithms can even turn non-quantitative data like videos and images into quantified datasets (e.g. Hossain et al., 2016). Depending on the data source, real data might be produced and compared to models in real-time or asynchronously. And depending on the data source, real and simulated data can be compared in side-by-side visualizations or overlaid in the same visualizations; they can be compared across multiple dynamically linked visualizations, or quantitatively by means of analysis tools, and so on. Each of the data sources necessitates different data-related skills, and each of the types of comparisons has different implications for what students might focus on and how they might reason when using models and data. In this presentation, we present a taxonomy of these types of data sources and types of comparisons, and elaborate on how they might impact possible learning opportunities. This work is necessary to guide the development of technologies and activities for rich scientific inquiry through the integration of computational models with real-world data.

Paper 2: Eighth-Grade Students Learn Electricity through Constructing Computational Models with Much.Matter.in.Motion

Janan Saba, Elon Langbeheim, Hagit Hel-Or & Sharona T. Levy

This study focuses on students' learning of the concept of electric current through the lens of complex systems by constructing computational models with an iconic interface. We focus on how students develop mental models that associate properties of a circuit at the macro-level, such as the magnitude and direction of the electric current, with entities at the micro-level, such as electrons and their motion (Eylon, & Ganiel, 1990). We examine the students' development of mental models as they construct computational models with an early widget-based version of the Much.Matter.in.Motion platform (MMM; Levy et al., 2018). One of MMM's main affordances is sustainability, which enables modeling of many physics and chemistry phenomena.

Learners rely on cognitive constructs termed "mental models" when explaining the world around them (Norman, 1983). They develop these mental models through interactions with natural or human-made systems; yet, these models are often partial, unstable, and inaccurate. It is important to support students' expression of these mental models and to follow their transformations through learning. This process is facilitated by having this expression take the visual form of drawing, on the one hand, and become animated through the generative power of computation, on the other hand.

Most computational platforms that were designed to support teaching of electricity are based on a macro-level representation of circuits (e.g. Roll et al., 2018), with only a few environments attempting to introduce electric circuits from a micro-level perspective (Sengupta & Wilensky, 2009). In our study, we developed the MMM modeling platform (created with NetLogo [Wilensky, 1999]) for building and running computational models. MMM enables model construction at the micro-level without coding, by using pre-built model features that can be added and modified. To increase the accessibility of MMM, it is set up in such a way that users first draw the boundaries of their model conductor, and then add particles for which they define size, initial speed, heading, and interactions. After constructing the model, students can run the model and observe the resulting behavior. Figure 1 shows a screenshot of the MMM platform with a constructed model. The brown particles represent the atoms, and the red particles represent electrons.

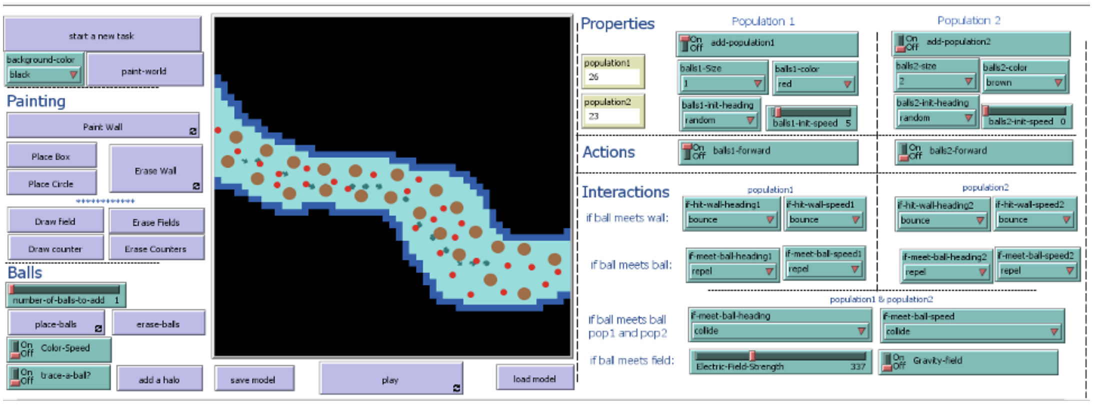

Figure 1. The Much.Matter.in.Motion (MMM) interface

Thirty-three eighth-grade students' learning by constructing models is compared with 23 students' learning with a normative curriculum, during six 1.5-hour sessions. The learning process included physical experiments and modeling activities. Students completed pre-and-post-test questionnaires and four students were interviewed. Activities were logged and screen-captured.

The interaction between time and group ($F(1,54)=6.67$, $p<0.05$) shows the superior learning of the experimental group. The specific component that contributes to this significance is the micro-level reasoning on the systems. A two-step cluster analysis was conducted on the students' responses to the open-ended items in the questionnaires regarding simple electric circuits. Five mental models were revealed that differed across several features: the balance between physical macro-level components and micro-level descriptions and the degree of integration between the micro and macro levels. Comparing the magnitude of the transitions between mental models across the two groups shows a significant transition magnitude for the experimental group between pre- and post-testing compared to the comparison group.

Our study shows that constructing micro-level models of electric conductors advances micro-level conceptualization of the electric circuit, and does not diminish the overall macro-level understanding. In addition, cluster analysis is an important methodological advancement compared to prior studies that examined individual responses without an integrated approach that combines all of the knowledge elements expressed by each student. Results indicate that enabling students to construct computational models using MMM promotes progress in thinking about electric circuits: from simpler mental models that are focused at the macro-level in the beginning of the learning process to more sophisticated ones at the end that include the micro-level entities and interactions. This approach to learning science is now being extended to additional topics. We are now studying the sustainability of these ideas over time and across topics.

Paper 3: Unpacking Students' Reasoning Using Computational Models With Domain-Specific Blocks

Cassia Fernandez, Tamar Fuhrmann, Paulo Blikstein & Roseli de Deus Lopes

Engaging students in modeling practices to make sense of scientific phenomena and the nature of science is the focus of recent reforms in science education (NGSS, 2013; NRC, 2012). Creating a computational model can be a valuable way for students to create and test their own hypotheses and refine their understanding of natural phenomena (Gilbert et al., 2016; Passmore et al., 2014; Schwarz et al., 2012). Although the constructivist idea that students are not "blank slates" but rather possess a wide range of previous knowledge on natural phenomena is well established (diSessa, 2018), there is, still, less research than needed about what students already know in specific domains of science. Thus, often, educators and designers (by no fault of their own) are not completely aware of preconceptions that students might bring to classrooms, what they are confused about, or how their ideas about the world fit with what they are attempting to learn. This paper aims to analyze students' conceptual trajectories while programming a computational model, in order to understand their initial, nonstandard, ideas about diffusion and how those ideas evolve. We are interested in learning how interaction with a constrained model can reveal students' thinking and, thus, inform designers on how to design tools and languages that acknowledge students' ideas and sensemaking (Wilkerson-Jerde et al., 2010; Wagh et al., 2017).

Our preliminary findings are drawn from a pilot study conducted through individual online sessions with seven 5th-grade students. Each session lasted one hour and involved model creating and comparison with data (Fuhrmann et al., 2018). Students observed a diffusion experiment and designed a model to explain the phenomenon at the particle level, using their experimental observations to validate and refine the model. For these sessions, we designed an initial version of nine domain-specific programming blocks integrated into Scratch as an extension. By combining our blocks with the original Scratch blocks, we expected students to test and refine their own ideas about diffusion, using scaffolding tasks and questions from the researchers.

The use of block-based modeling tools has many affordances for young students. For example, it allows them to develop their own scientific models without being overwhelmed by text programming in a dynamic process (Gouvea et al., 2017). By doing so, students can revise their ideas and make changes through many iterations. According to our findings (see Figure 2), designing computational models in this way helped students externalize scientific ideas that would otherwise remain tacit. Through coding their models, students revealed unexpected concepts such as molecule "transmutation," atoms that would simply disappear or "melt," and particles with volition. Being able to express those ideas in code was crucial for students to be able to then reflect on those ideas, understand their possible flaws, and refine their initial ideas.

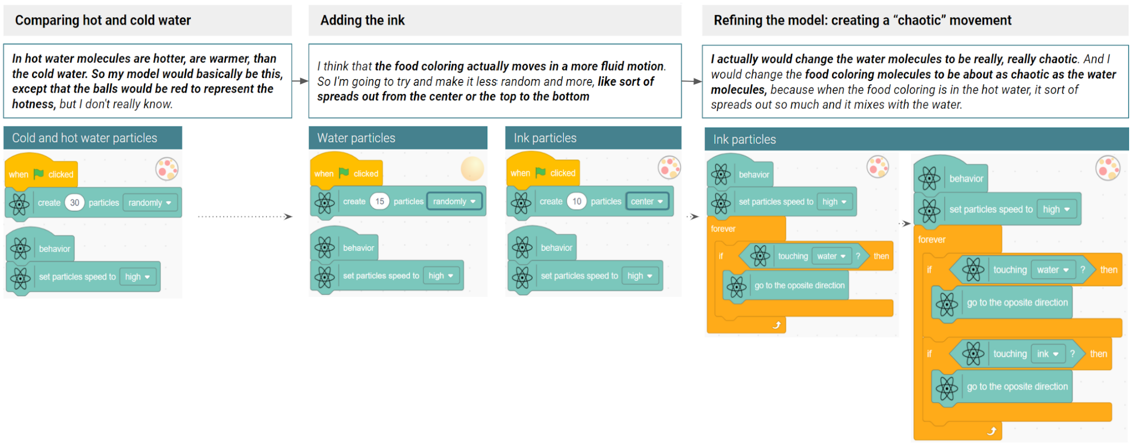

Figure 2. An example of student initial ideas and the evolvement process illustrated with blocks

The use of blocks can be a valuable strategy to explore students' reasoning processes: the translation of ideas into code unpacks students' thinking. It reveals inadequate conceptions since it forces students to make their thinking visible. By being aware of students' prior knowledge and identifying possible sources of confusion, science educators can mediate the scientific knowledge-building process for learners; help them make sense of how scientific models are generated and validated; and scaffold learners' conceptual understandings rather than merely giving them the "correct answer." While learning how students reason about a phenomenon, designers and researchers can create more domain-specific blocks that acknowledge students' initial ideas and sensemaking processes.

Paper 4: Custom Blocks in StarLogo Nova: Lowering Barriers and Providing Pathways to Further Learning

Daniel Wendel & Eric Klopfer

Agent-based computational modeling (ABM) has been shown to help students reason about the parts and interactions in systems that give rise to observed scientific phenomena at the system level (Klopfer, 2003; Resnick, 1994; Sengupta & Wilensky, 2009; Wilensky, 1999). Using models has shown some benefit (Yoon et al., 2016), and modifying and creating models provides a rich opportunity to engage in both the practice of modeling and in computational thinking (Sengupta et al., 2013; Wagh, Cook-Whitt, & Wilensky, 2017; Weintrop et al., 2016), both of which are essential skills for modern scientists (Foster, 2006). However, our experience and the experience of many other researchers in computational modeling is that teachers are often reluctant to incorporate modeling due to its high startup costs, particularly the cost of learning to "code" (Ertmer & Ottenbreit-Leftwich, 2013; Guzdial 1994; Hsiao et al. 2019).

In a current research project, working with our partners at NYU, we found that students — including English language learners — in a 5th grade classroom were able to use our online ABM tool, StarLogo Nova (www.slnova.org), to create models of microbes breaking down solid food into waste products and gas, conserving the overall mass of the system to match their "landfill bottle" physical models. In post interviews, these students demonstrated systems thinking as they explained the process of modeling the phenomena. They were able to reason more effectively about aspects included in their models than about hypothetical scenarios they had not modeled (Haas et al., 2020), which underscores how important having created the model is to students' ability to make sense of the phenomenon. In this full-year curriculum, the costs can be amortized across a whole year, but many teachers need activities that can be integrated into existing curricula within a more constrained time frame.

We have reconciled the need to address particulars of the curriculum with the desire to build student modeling skills across the year by making changes to the StarLogo Nova platform in the form of "custom blocks." We developed StarLogo Nova to be beginner-friendly, using a blocks-based programming language to reduce the barriers of syntax faced by novices (Klopfer et al., 2009). However, the process of modeling concepts or phenomena from a set of domain-general programming primitives remains challenging (Anderson & Wendel, 2020). Recently, domain-specific modeling languages/environments (DSMLs/DSMEs) have attempted to narrow the conceptual gap between programming language semantics and scientific phenomena (Hasan & Biswas, 2017). For example, whereas StarLogo Nova's modeling language may represent a predator's "hunt" behavior as "face toward nearest prey agent; move forward 1 space; on collision, delete prey and increase my energy trait," a DSML for ecosystems may simply use "chase prey; if I catch prey, eat it".

In science education, these DSMLs are typically implemented as sets of "custom blocks" layered on top of existing primitives in a modeling environment. These blocks are carefully designed by educators to be beginner-friendly, but are typically treated as "black boxes" to shield students from the underlying implementation (e.g. Basu et al., 2013; Wilkerson-Jerde, Wagh et al., 2015). In contrast, in computational thinking/computer science education, blocks-based programming environments engage learners themselves in the process of creating and modifying custom blocks as an introduction to procedural abstraction; Snap!'s previous version was even named "BYOB: Build Your Own Blocks" (Harvey, 2020).

StarLogo Nova's new custom blocks feature aims to combine these ideas. Rather than viewing the custom blocks of a DSML as black boxes to hide underlying complexity, we have intentionally designed StarLogo Nova's new custom blocks feature to capitalize on StarLogo Nova's "blocks all the way down" language architecture, hoping to transform a custom block's underlying implementation from a liability into an opportunity. Because custom blocks can themselves contain custom blocks, we envision and are beginning to investigate a layered abstraction approach, with DSML blocks providing not only a seamless introduction to computational modeling, but also a path to deeper understanding by allowing students to "look inside," reading and modifying code at successively more domain-general (and therefore more flexible and powerful) layers of abstraction as their computational modeling skill grows.

Paper 5: Towards Sustained Computational Modeling: Exploring Strategies to Support Extended Investigations in Middle School Classrooms

Aditi Wagh, Tamar Fuhrmann, Michelle Wilkerson, Engin Bumbacher & Paulo Blikstein

There is now a considerable amount of research exploring how, through design and pedagogy, educators can lower the barrier to computational modeling for young learners. For example, modeling environments that make use of accessible media like block-based (e.g., Klopfer, Yoon & Um, 2005) and/or domain-specific (e.g., Repenning & Summer, 2005; Wilkerson, Wagh & Wilensky, 2017) toolkits ease the technical aspects of programming, whilst pedagogical approaches such as embodied (e.g., Wilensky & Reisman, 2006) and bifocal modeling (Blikstein et al., 2016; Fuhrmann et al., 2018) support conceptual aspects of computational modeling. This research has contributed to our understanding of how to make computational modeling an accessible practice in science classrooms. However, less is known about how to make computational modeling a *sustained* practice in classrooms. By "sustained", we mean engaging in extended investigations using computational modeling.

In this paper, we will describe how our research aims to foster sustained engagement in modeling using data from co-design sessions with teachers and classroom implementations. We draw on two design strategies from the broader scientific modeling literature (Lehrer & Schauble, 2012; Manz, 2015) in which extended investigations with modeling and iterative refinement are part of what students do. We use these strategies in the design of the computational modeling toolkit and the curriculum. The first strategy involves building uncertainty into student investigations through experimental data to create resistances for students to consider, tackle, and possibly resolve (Manz, 2015). Tackling such uncertainties together with the use of data can amplify disciplinary ideas while supporting progress along a modeling trajectory. In related work, the authors of this paper have integrated computational modeling with data in order to highlight — rather than dismiss — discrepancies between models and data, bringing noise, uncertainty, and intrinsic differences between them to the forefront (Blikstein et al., 2016). This bifocal modeling approach has been successful in several domains, but for relatively shorter interventions (Fuhrmann et al., 2020). The second strategy we draw on is repeatedly revisiting a small set of disciplinary ideas from multiple perspectives to pursue extended investigations (Lehrer & Schauble, 2012). We envision this as enabling students to construct, deconstruct, and recombine models of related phenomena as well as unpacking blocks and chunking them in ways that can be meaningful for students. We see this as a way to deepen engagement in modeling so as to develop an epistemologically rich perspective on the role of computational modeling in understanding a phenomenon.

In our paper, we report on co-design sessions with three middle school science teachers as they provide feedback on the computational modeling toolkit and refine two curricular activities. Our analysis describes the role that teachers expect the data to play in supporting students' computational modeling activities, and how to support students in coordinating data analysis and model refinement. We also describe how teachers suggest linking across curricular topics, particularly through model construction, deconstruction, and recombination. Finally, we share challenges teachers anticipate in this work.

References:

Anderson, E., & Wendel, D. (2020, June). Learning science through coding: An investigation into the design of a domain-specific modeling experience. In M. Gresalfi and I. Horn. (Eds.). *The 14th International Conference of the Learning Sciences (ICLS) 2020* (Vol. 3, pp. 1253-1260). Nashville, Tennessee: International Society of the Learning Sciences.

Basu, S., Dickes, A., Kinnebrew, J.S., Sengupta, P., & Biswas, G. (2013). CTSiM: A Computational Thinking Environment for Learning Science through Simulation and Modeling. In *Proceedings of the 5th International Conference on Computer Supported Education* (pp. 369-378). Aachen, Germany

Berland, L. K., Schwarz, C. V., Krist, C., Kenyon, L., Lo, A. S., & Reiser, B. J. (2016). Epistemologies in practice: Making scientific practices meaningful for students. *Journal of Research in Science Teaching*, 53(7), 1082–1112.

Blikstein, P., Fuhrmann, T., & Salehi, S. (2016). Using the Bifocal Modeling Framework to Resolve "Discrepant Events" Between Physical Experiments and Virtual Models in Biology. *Journal of Science Education and Technology*, 25(4), 513–526.

Chinn, C. A., & Malhotra, B. A. (2002). Epistemologically authentic inquiry in schools: A theoretical framework for evaluating inquiry tasks. *Science Education*, 86(2), 175–218.

diSessa, A. A. (2018). Computational literacy and "the big picture" concerning computers in mathematics education. *Mathematical thinking and learning*, 20(1), 3-31.

Ertmer, P. A., & Ottenbreit-Leftwich, A. (2013). Removing obstacles to the pedagogical changes required by Jonassen's vision of authentic technology-enabled learning. *Computers & Education*, 64, 175-182.

Eylon, B. S., & Ganiel, U. (1990). Macro-micro relationships: the missing link between electrostatics and electrodynamics in students' reasoning. *International Journal of Science Education*, 12(1), 79-94.

Foster, I. (2006). A two-way street to science's future. *Nature* 440, 419

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

Fuhrmann, T., Schneider, B., & Blikstein, P. (2018). Should students design or interact with models? Using the Bifocal Modelling Framework to investigate model construction in high school science. *International Journal of Science Education,* 40(8), 867-893.

Gouvea, J., & Passmore, C. (2017). 'Models of' versus 'Models for'. *Science & Education*, 26(1-2), 49-63. Gilbert, J. K., & Justi, R. (2016). Models of modelling. *In Modelling-based teaching in science education* (pp. 17-40). Springer, Cham.

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

Haas, A., Grapin, S., Wendel, D., Llosa, L., & Lee, O. (2020). How Fifth-Grade English Learners Engage in Systems Thinking Using Computational Models. Manuscript submitted for publication.

Halloun, I. A. (2011). From modeling schemata to the profiling schema: Modeling across the curricula for profile shaping education. In *Models and modeling* (pp. 77-96). Springer, Dordrecht.

Harvey, B. (2020). You ripped off Scratch! [Discussion forum FAQ post]. Online. https://forum.snap.berkeley.edu/t/you-ripped-off-scratch/2859
Hasan, A., & Biswas, G. (2017). Domain-specific modeling language design to support synergistic learning of STEM and computational thinking. In Kong, S. C., Sheldon, J., & Li, K. Y. (Eds.). Conference *Proceedings of International Conference on Computational Thinking Education 2017*. Hong Kong: The Education University of Hong Kong, 28.
Horn, M. & Wilensky, U. (2011). NetTango [computer software]. Evanston, IL: Center for Connected Learning and Computer-Based Modeling, Northwestern University
Holmes, N. G., Wieman, C. E., & Bonn, D. A. (2015). Teaching critical thinking. *Proceedings of the National Academy of Sciences,* 112(36), 11199–11204.
Hossain, Z., Bumbacher, E. W., Chung, A. M., Kim, H., Litton, C., Walter, A. D., Pradhan, S. N., Jona, K., Blikstein, P., & Riedel-Kruse, I. H. (2016). Interactive and scalable biology cloud experimentation for scientific inquiry and education. *Nature Biotechnology*, 34(12), 1293–1298.
Hsiao, L., Lee, I., & Klopfer, E. (2019). Making sense of models: How teachers use agent-based modeling to advance mechanistic reason. *British Journal of Educational Technology,* 50(5), 2203-2216.
Klopfer, E., Yoon, S. & Um, T. (2005). Teaching Complex Dynamic Systems to Young Students with StarLogo. *Journal of Computers in Mathematics and Science Teaching*, 24(2), 157-178. Norfolk, VA: Association for the Advancement of Computing in Education (AACE).
Lehrer, R., & Schauble, L. (2012). Seeding evolutionary thinking by engaging children in modeling its foundations. *Science Education*, 96(4), 701–724.
Manz, E. (2015). Resistance and the Development of Scientific Practice: Designing the Mangle Into Science Instruction. *Cognition and Instruction*, 33(2), 89–124.
NGSS Lead States. (2013). *Next generation science standards: For states, by states*. Washington, DC: The National Academy Press
Norman, D. A. (1983). Some observations on mental models. *Mental Models*, 7(112), 7–14.
Passmore, C. M., & Svoboda, J. (2012). Exploring opportunities for argumentation in modelling classrooms. *International Journal of Science Education*, 34(10), 1535–1554.
Pluta, W. J., Chinn, C. A., & Duncan, R. G. (2011). Learners' epistemic criteria for good scientific models. *Journal of Research in Science Teaching,* 48(5), 486-511.
Repenning, A., & Sumner, T. (1995). Agentsheets: a medium for creating domain-oriented visual languages. *Computer*, 28(3), 17–25.
Roll, I., Butler, D., Yee, N., Welsh, A., Perez, S., Briseno, A., & Bonn, D. (2018). Understanding the impact of guiding inquiry: The relationship between directive support, student attributes, and transfer of knowledge, attitudes, and behaviors in inquiry learning. *Instructional Science*, 46(1), 77-104.
SageModeler [Computer software]. (2020).
Concord, MA: The Concord Consortium and the CREATE for STEM Institute at Michigan State University.
Schwarz, C., Reiser, B. J., Acher, A., Kenyon, L., & Fortus, D. (2012). MoDeLS: Challenges in defining a learning progression for scientific modeling. In *Learning progressions in science* (pp. 101-137). Brill Sense.
Sengupta, P., & Wilensky, U. (2009). Learning Electricity with NIELS: thinking with electrons and thinking in levels. *International Journal of Computers for Mathematical Learning*, 14(1), 21–50.
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. J*ournal of Science Education and Technology*, 25(1), 127-147.
Wilensky U. (1999) *NetLogo*. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston. http://ccl.northwestern.edu/netlogo.
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., Wagh, A., & Wilensky, U. (2015). Balancing curricular and pedagogical needs in computational construction kits: Lesson from the DeltaTick project. *Science Education*, 99, 465-499.
Yoon, S., Anderson, E., Klopfer, E., Koehler-Yom, J., Sheldon, J., Schoenfeld, I., Wendel, D., Scheintaub, H., Oztok, M., Evans, C., & Goh, S.-E. (2016). Designing Computer-Supported Complex Systems Curricula for the Next Generation Science Standards in High School Science Classrooms. *Systems,* 4(4), 38.
Zimmerman, C. (2007). The development of scientific thinking skills in elementary and middle school. *Developmental Review*, 27(2), 172–223.