Agent-based Programming and Modeling in Elementary Science: Aesthetic Experience, Disciplined Interpretation, and Heterogeneity

By

Amy Voss Farris

Dissertation

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfilment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

in

Learning, Teaching and Diversity

August 10, 2018

Nashville, Tennessee

Approved:

Pratim Sengupta, Ph.D.

Richard Lehrer, Ph.D.

Corey Brady, Ph.D.

To my beloved parents – I study children's learning and development in STEM fields because you valued these first.

To the teachers and students who gave of themselves in this work – Thank you for opening your classrooms to me. Your inventiveness provokes and improves my own inquiry.

To Daniel and to Ari – You have been endlessly patient in my pursuits.

And to Pratim – You continue to deliver on your promises of intellectual freedom and unfettered support. As you say, "I am with you all the way, and beyond."

ACKNOWLEDGEMENTS

This research was supported by the National Science Foundation under a CAREER Grant awarded to Pratim Sengupta (NSF CAREER OCI #1150230). Pratim made this work possible in many ways—through envisioning the general research niche, securing funding, and providing constant mentorship. Rich Lehrer, Corey Brady, Barbara Stengel, and Kevin Leander generously asked questions and offered insights that deepened my own thinking and the trajectory of the research. My lab mate and friend, Amanda Dickes, contributed in countless ways in conceptualization, data collection, and feedback. I (along with Pratim) gratefully acknowledge Uri Wilensky's work on redefinition of the concrete as the primary inspiration behind Chapter II. Chapter II also benefitted from extensive critical feedback and generous suggestions from Barbara Stengel and Leonard Waks. Chapter III is shaped from feedback from my committee members, Philip Bell, and three anonymous ICLS reviewers. Finally, the teachers and students of the classrooms in which this work was carried out also own this knowledge, and are the generous and lively originators of the collective experience that I am attempting to share here.

TABLE OF CONTENTS

LIST OF TABLES

Chapter 3 Page

Chapter 4

LIST OF FIGURES

Chapter 2 Page

Chapter 3

Chapter 4

CHAPTER I: INTRODUCTION

Philosophers and sociologists of science have long argued that subjective and personal experiences play an important role in the creation of scientific knowledge (Daston & Galison, 2007; Engler, 2002; Lynch & Edgerton, 1996; MacLeod & Nersessian, 2013; Kosso, 2002; Wechsler, 1983). For example, Daston and Galison (2007) argue that new representational technologies can require scientists to adopt a more interpretive stance while designing scientific representations. However, heterogeneous conceptions of scientific practice such as these are rarely legitimized in most K12 settings of science learning, where the roles of interpretive and socially negotiated actions are often minimized or excluded from the development of scientific experience.

A focus on certain forms of subjective aspects of scientific experience can bring to light some of the most important elements that constitute the practice of scientific modeling. These elements include issues such as identifying what kind of external representation might make an explanation more convincing, or what questions are worth pursuing, or how to interpret or explain new data that is incongruent with prior understandings. These elements of practice are commonly recognized as valuable within professional science—they are mature aspects of practice that may be regarded as a kind of tacit "professional vision" (Goodwin, 1994) of scientists.

Educators (e.g., Hammer, Sherin, & Kolpakowski, 1991; Lehrer, 2009; Lehrer, Schauble, and Lucas, 2008; Manz, 2015) have also long valued learners' engagement in similar judgments and tacit sensibilities, such as (a) criteria for what makes a good question, or evidence convincing, and (b) ways of wrangling the material difficulties of scientific investigation. In work on representational literacies in science, diSessa (2002), Sherin (2001), and Danish and

Enyedy (2007) have generated descriptive taxonomies of metarepresentational competence (MRC) that account for *how* children decide what to include in their models. We also know that children's playful engagement with scientific ideas can support them to progressively ignore the need for literal representation and become more fluent with selective in more abstract forms of symbolizing (Enyedy, Danish, Delacruz, & Kumar, 2012). However, as computational modeling becomes an expected form of representational fluency within science classrooms (NRC, 2012; NGSS, 2013), learning scientists and science educators face challenges related to how and why the integration of computational modeling can become most generative.

Within the specific context of agent-based computational modeling for K12 science, I seek to describe episodes of the development of the tacit sensibilities in children's technoscientific practice. I use the word "technoscientific" in the sense that Latour (1987) uses it to emphasize the embeddedness and interdependence of material and technological actions within the production of scientific knowledge. Overall, my work suggests that scientific computation can become a generative pathway for supporting students to form scientific explanations that are rooted and formed in "softer" sensibilities—thereby complementing and contrasting the prevalence of epistemic objectivity that is canonically characteristic of science classrooms. The chapters that follow (Chapters I – IV) each investigate a different form or forms of the experience of computational modeling as well as a different educational context.

In each of my studies, scientific modeling is integrated with computational modeling using an agent-based programming and modeling language called ViMAP (Sengupta, Dickes, Farris, Karan, Martin, & Wright, 2015). I examine the ways in which 4th, 5th, and 6th grade learners (and their teachers) generatively leveraged personal, contextually-driven, and interpretive ways of knowing in scientific modeling practices in studies of motion. I describe learners' modeling and related actions in order to illuminate how their subjective actions become valued (or not) in the life of the classroom. The three papers examine different aspects of students' appropriation and use of computational modeling: Chapter II proposes a theoretical framework for thinking about children's computational modeling work as a form of Deweyan (1934/2005) aesthetic experience, with a focus on initially disengaged learners. Chapters III and IV examine the integration of ViMAP in classroom contexts where teachers with little or no programming experience integrate ViMAP programming into their teaching, and students consistently construct explanations by modeling them.

Theoretical Coherence Across Papers

Each paper of this dissertation describes students' work in computational modeling and illustrates how students generatively employ their own subjective sensibilities and preferences. Each paper also analyzes episodes in which students meaningfully participate in interpretation and negotiated modeling decisions within science learning settings in which computational modeling is a regular form of activity. Across my studies, I continually saw that the integration of computational modeling—used as an ongoing outlet for learners' efforts to represent and explain scientific ideas—created a new form of language and representation in the classroom that opened the door for heterogeneous and subjectively-determined expressions of authentic scientific activity. I therefore take a phenomenological approach to understand how and why this happened. The work is not limited to new *representational* activity: it serves to ultimately reshape *epistemic* actions in ways that supported learner agency. In this sense, all the studies are grounded in the *science as practice* perspective (Pickering, 1995; Lehrer & Schauble, 2006; NRC, 2007), where conceptual development in science happens through a deeply intertwined dance of epistemic and representational work.

As a set, the three papers explore how the integration of computational modeling in the context of scientific modeling in classrooms served to deepen students' conceptual development and supported the refinement of representational work. Their work involved progressive symbolization across investigations, paper-based representations, and agent-based computing and makes visible the heterogeneous nature of scientific modeling, in terms of the varied approaches and interests that come to bear in students' use of computational tools (especially Chapters II and IV) and ownership of interpretations and novel ways of making meaning across forms of abstraction (especially Chapters III and IV). Across the three papers, I articulate learner agency and experience in ways that have been foundational to the constructionist work in science and computing education during the last four decades (e.g., Papert, 1980; Resnick, Berg, & Eisenberg, 2000; Turkle & Papert, 1992; Wilensky & Reisman, 2006), yet remain undertheorized. I demonstrate how different forms of experience of coding in the science classroom can draw together personal narratives, desires, and disciplinary aims and—as a result—make way for a more democratic and inclusive science education.

All three papers are bound by a perennially important and difficult area of content in early physics education: motion. Learning goals in each study included learners' conceptualizations of kinematic phenomena as continuous processes of change in position and speed, and to be able to explain these changes in disciplinarily appropriate ways, including computational modeling.

Although kinematic phenomena are arguably the most well-studied genre of children's modeling and progressive symbolization, the semiotic complexities of representing position as a process of continuous change remains challenging for learners. Many researchers (e.g., Hammer et al., 1991; Dykstra & Sweet, 2009; Elby, 2000; Halloun & Hestenes, 1985; Leinhardt, Zaslavsky, & Stein, 1990; McCloskey, 1983; and McDermott, Rosenquist, & van Zee, 1987) have reported that learners of all ages face conceptual difficulties in: (a) understanding and explaining the formal (mathematical) relationships between distance, speed, time and acceleration, and (b) interpreting and explaining the physical concepts, relationships and phenomena represented by commonly used graphs of speed vs. time or distance vs. time.

In particular, researchers have found that understanding continuous change in motion is challenging for novice learners. For example, when provided with a situation that involves objects moving with uniform acceleration (e.g., during free fall or along an inclined plane), students in elementary and middle grades (4th & 6th grades) find it challenging to differentiate between instantaneous speed and average speed (Elby, 2000); and tend to describe or explain any speed change(s) in terms of comparisons of fast and slow, rather than describing speeding up or slowing down as a continuous process (Halloun & Hestenes, 1985; Dykstra & Sweet, 2009). Previous research shows that given appropriate scaffolding, middle and high school students can effectively use agent-based learning environments that involve programming and modeling in order to develop sophisticated mathematical representations of key kinematic concepts and phenomena (Hammer et al., 1991; Sengupta & Farris, 2012; Sherin, 2000; Sherin, diSessa, & Hammer, 1993).

Enyedy, Danish, Delacruz, and Kumar (2012) extend similar physics learning aims to early elementary grades using and augmented environment that transforms children's (ages 6 – 8) movements in the world into a microworld using motion capture technologies. Like Enyedy and colleagues (2012), we are interested in the children's ability to transform kinematic symbols across semiotic ecologies. We extend the semiotic space of their study to include children's use of programming commands to simulate motion. We describe how agent-based programming, too, can contribute to a coherent cascade (Latour, 1999) across materials and media.

Sequence of the Studies

The research settings in the three studies progressively deepen in their immersion in authentic classroom spaces. The research context of Chapter II is a two-week design study that took place during the summer and was led by myself and two Vanderbilt faculty members. The data presented in Chapter II serve as a worked-out example to illustrate the transformational aspects of *aesthetic experience* (Dewey, 1934/2005) for science learning. Chapter II (Farris & Sengupta, 2016) presents aesthetic experience (Dewey, 1934/2005) as a previously-unarticulated—yet foundational—commitment of constructionism. This paper illustrates how, within different forms of experience of computing and in personal life, learners' desires and disciplinary ideas are brought together in the form of aesthetic experiences. It also presents the theoretical foundation for Chapters III and IV in Deweyean terms: Democratizing computing within science education hinges on designing pedagogies that enable the learner to transform the computer into an expressive medium.

The second and third papers (Chapters III and IV) emerge from and build upon this perspective, describing how learners conduct this transformation in classrooms taught by their regular science teacher. Neither the students nor their teachers had any prior experience with computing. The research contexts for the second and third studies (Chapters III and IV) are public charter school classrooms, and the participants are from racial and socioeconomic groups that are persistently underrepresented in STEM fields. Children's deployment of computational abstractions that they use to explain scientific processes creates new means of dynamic representation (diSessa, 2001; Dickes, Sengupta, Farris, and Basu, 2016), and also creates important representational uncertainty. This uncertainty can help bring the tentative and communicative aspects of scientific explanation to the forefront of children's experience, and, we argue, allows children to participate in the heterogeneous work of scientific meaning-making, in contrast to views of science that simplify the work of science to rote "method" and "facts." As a set, the papers represent an entrée into a new kind of science, one in which the mundane is reimagined, and true aims find their place alongside institutionally mandated aims.

Contributions to the Field

My dissertation makes the following contributions:

1. *A deeper understanding of the experience of computational modeling in the science classroom:* Modeling is the "language" of science (Giere, 1984, p. 80; Lehrer, 2009; NRC, 2007). In the context of science education, a "science-as-practice" (Lehrer & Schauble, 2006) perspective of science learning allows a focus on how modeling re-positions learners as participants in the social negotiation of practices and ideas (e.g., Manz, 2015) in ways similar to professional scientific practice. Following Lehrer and Schauble (2005), Lehrer (2009), diSessa (2001), Sengupta, Dickes, & Farris (2018), and others, I position scientific and computational modeling as an epistemic as well as a representational activity. It is epistemic in the sense that deep engagement with modeling necessarily "entails changes in students' epistemic goals" (Lehrer & Schauble, 2005, p. 383). However, as our early studies revealed, in the context of computational modeling in the science classroom, these epistemic shifts co-occur with shifts in students' use of the representational infrastructure (Sengupta & Farris, 2012; Sengupta, Farris & Wright, 2012; Dickes et al., 2016). These studies demonstrate that the representational infrastructure in use is not limited to the programming language and

different elements of the computational modeling platform, but also involves embodied and physical modeling. However, given the interventionist nature of these studies where the researchers designed the computational platforms, activities and studies, and served in the role of instructors in these studies as well, we had a limited understanding of how teachers and students appropriate such forms of modeling. My dissertation seeks to make a contribution along this dimension by illustrating a) how students who do not conform to the instructional mandates in the classroom and are not interested in computing can, in fact, find their way into computational science, and b) how students and teachers appropriate computational modeling in their science classroom when the teacher is in charge of the classroom instruction.

As Sengupta, Kinnebrew, Basu, Biswas, and Clark (2013) argued, designing computational models by creating and iteratively refining programs corresponds to core scientific practices, such as model construction, refinement, validation and deployment, design-based thinking and verification (Hestenes, 1993; Lehrer & Schauble, 2005; Nersessian, 2008; and Papert, 1980). However, as Duschl (2008) argued, the experience of modeling is rife with uncertainties, which necessitates interpretive and subjective work. These elements of modeling are usually not highlighted in most studies in the field of educational computing, where the focus is on students' production of canonically correct scientific representations (Sengupta et al., 2018). The focus of my dissertation asks how engaging with a particular form of modeling—agent-based modeling (explained in the following section)—can help students develop expertise in scientific modeling by acknowledging and highlighting, rather than de-emphasizing, their interpretive and subjective experiences.

2. *A deeper understanding of how agent-based modeling can be integrated with K-12 science classrooms:* Agent-based modeling (ABM) is a genre of computing that allows users to define individual actions of computational agents in order to simulate processes of change, thereby demonstrating how complex patterns emerge from simple, agent-level actions (Papert, 1980; Wilensky & Resnick, 1999; Sengupta et al., 2013). Agent-based modeling (ABM) is widely used in scientific and sociological research (Jacobson & Wilensky, 2006), and prior studies have demonstrated affordances of ABM for learning in K16 settings. Beginning with LOGO, many researchers have argued for reflexivity between developing scientific expertise and learning computer programming (Harel & Papert, 1990; Guzdial, 1994) and agent-based modeling and programming, in particular (diSessa, 2001; Jacobson & Wilensky, 2006; Papert, 1980). Learners' development of modeling practices using ABM and programming can also support their development of their computational thinking (Sengupta et al, 2013), which is defined by Wing (2006, 2008) as a general, analytic approach to problem solving, designing systems, and understanding human behaviors.

   A key affordance of ABMs is that programming the agent involves thinking like it (Papert, 1980, Wilensky & Reisman, 2006). This means that that the LOGO turtle's objectives, behaviors, and rules of operation can be easily interpreted and understood even by young learners by bootstrapping their intuitive understandings of their own bodies and movement. Researchers have demonstrated that ABM can indeed serve as a powerful pedagogical approach for understanding complex scientific topics (Dickes et al., 2016; Grotzer, Derbiszewska, & Solis, 2017). However, beyond the intuitive nature of agent-based modeling, there are also challenges associated with using ABM in the science classroom, which have not been well studied by researchers. For example, my own previous work (Farris & Sengupta, 2014) illustrated how learners' successful adoption of the agent-perspective is a non-trivial task, which in turn involves being able to flexibly take multiple perspectives within the system being studied. Other researchers have argued how learning both programming and science at the same time can be prohibitive in science classrooms (Sherin et al., 1993), and more recently, some forms of agent-based modeling can present conceptual and representational challenges for middle school students that might require extensive scaffolding (Basu & Biswas, 2016).

My dissertation further extends this body of work by illustrating how agent-based modeling becomes grounded in the experience of students and teachers in science classrooms through highlighting the role of non-canonical resources that learners can bring in the instructional setting, and how the representational infrastructure of ABM itself must be re-imagined beyond the programming language and the computer. In each paper, I identify forms of experience that can help us understand what such representational infrastructures might look like, and how heterogeneous elements of these representational infrastructure can be productively brought into contact with one another by students and teachers. In each paper, I discuss how such phenomenological images of computational modeling in classroom and scientific practice can help us better design computational modeling environments and activity systems for classroom integration in K-12 science.

References

Azevedo, F. S. (2000). Designing representations of terrain: A study in meta-representational competence. *The Journal of Mathematical Behavior*, *19*(4), 443-480.

Basu, S., & Biswas, G. (2016). Providing adaptive scaffolds and measuring their effectiveness in open ended learning environments. Singapore: International Society of the Learning Sciences.

Danish, J. A., & Enyedy, N. (2007). Negotiated representational mediators: How young children decide what to include in their science representations. *Science Education*, *91*(1), 1-35.

Daston, L., & Galison, P. (2007). *Objectivity.* Zone Books.

Dewey, J. (2004). *Democracy and education: an introduction to the philosophy of education.* Courier Dover Publications. (Original work published 1916)

Dewey, J. (2005). *Art as Experience.* Perigree, The Berkley Publishing Group, New York. (Original work published 1934)

Dickes, A. C., Sengupta, P., Farris, A. V., & Basu, S. (2016). Development of mechanistic reasoning and multi-level explanations of ecology in third grade using agent-based models. *Science Education, 100*(4), 734–776.

diSessa, A. A. (2001). *Changing minds: Computers, learning, and literacy*. The MIT Press.

diSessa, A. A. (2002). Students' criteria for representational adequacy. In K. Gravemeijer, R. Lehrer, & B. Van Oers (Eds.), *Symbolizing, modeling and tool use in mathematics education* (pp. 105-129). Springer Netherlands.

diSessa, A. A. (2004). Metarepresentation: Native competence and targets for instruction. *Cognition & Instruction, 22,* 293-331.

Duschl, R. (2008). Science education in three-part harmony: Balancing conceptual, epistemic, and social learning goals. *Review of Research in Education*, *32*(1), 268–291.

Dykstra, D. I., Jr. & Sweet, D. R. (2009). Conceptual development about motion and force in elementary and middle school students. *American Journal of Physics, 77*(5), 468-476.

Elby, A. (2000). What students' learning of representations tells us about constructivism. *Journal of Mathematical Behavior*, 19, 481-502.

Engler, G. (2002). The omniscienter: Beauty and scientific understanding. *International Studies in the Philosophy of Science 16*(1). 27-38.

Enyedy, N. (2005). Inventing mapping: Creating cultural forms to solve collective problems. *Cognition and Instruction*, *23*(4), 427-466.

Enyedy, N., Danish, J. A., Delacruz, G., & Kumar, M. (2012). Learning physics through play in an augmented reality environment. *International journal of computer-supported collaborative learning*, *7*(3), 347-378.

Farris, A. V. & Dickes, A. C., & Sengupta, P. (2016). Development of disciplined interpretation using computational modeling in the elementary science classroom. *Proceedings of the 12th International Conference of the Learning Sciences*. Singapore.

Farris, A. V. & Sengupta, P. (2014). Perspectival computational thinking for learning physics: A case study of collaborative computational doing. *Proceedings of the 11th International Conference of the Learning Sciences*. (pp. 1102-1106). Boulder, CO: International Society of the Learning Sciences.

Farris, A. V. & Sengupta, P. (2016). Democratizing children's computation: learning computational science as aesthetic experience. *Educational Theory, 66(*1-2), 279- 296.

Giere, R. N. (1984). Understanding scientific reasoning (2nd ed.). New York: Holt, Rinehart, and Winston.

Giere, R. N. (1988) Explaining Science: A cognitive approach. Chicago: University of Chicago Press.

Goodwin, C. (1994). Professional vision. *American Anthropologist*, *96*(3), 606-633.

Grotzer, T. A., Derbiszewska, K., & Solis, S. L. (2017). Leveraging Fourth and Sixth Graders' Experiences to Reveal Understanding of the Forms and Features of Distributed Causality. *Cognition and Instruction*, *35*(1), 55-87.

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

Halloun I. A., & Hestenes, D. (1985). The initial knowledge state of college physics students. *American Journal of Physics, 53*(11), 1043-1056.

Hammer, D., Sherin, B., & Kolpakowski, T. (1991). Inventing graphing: Meta-representational expertise in children. *Journal of Mathematical Behavior*, *10*(2), 117-160.

Harel, I. and Papert, S. 1990. Software design as a learning environment. *Interactive Learning Environments*, *1*(1): 1–32.

Hestenes, D. (1993). Modeling is the name of the game. *A presentation at the NSF Modeling Conference* (Feb. 1993).

Higgins, C. (2008). Instrumentalism and the clichés of aesthetic education: A Deweyan corrective. *Education and Culture, 24*(1), 6-19.

Jacobson, M. J., & Wilensky, U. (2006). Complex systems in education: Scientific and educational importance and implications for the learning sciences. *The Journal of the learning sciences*, *15*(1), 11-34.

Keller, E. F. (2003). Models, simulation, and computer experiments. In H. Radder (Ed.), *The philosophy of scientific experimentation* (pp. 198-215). University of Pittsburgh Press Pittsburgh, PA.

Kosso, P. (2002). Einstein and the most beautiful theories in physics. *International Studies in the Philosophy of Science 16*(1). 39-48.

Latour, B. (1987). *Science in action: How to follow scientists and engineers through society*. Harvard University Press.

Latour, B. (1999). *Pandora's Hope*. Cambridge, MA: Harvard University Press.

Lehrer, R. (2009). Designing to develop disciplinary dispositions: Modeling natural systems. *American Psychologist, 64*(8), 759.

Lehrer, R., & Schauble, L. (2005). Cultivating model-based reasoning in science education. In K. Sawyer (Ed.) *The Cambridge handbook of the learning sciences.* Cambridge University Press.

Lehrer, R., & Schauble, L. (2006). Scientific thinking and science literacy. In D. Kuhn and R. Siegler (Eds). *Handbook of child psychology (6th ed) vol. 2: Cognition, perception, and language* (153-196). Hoboken, New Jersey: John Wiley & Sons, Inc.

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

Leinhardt, G., Zaslavsky, O., & Stein, M. K. (1990). Functions, graphs, and graphing: Tasks, learning, and teaching. *Review of educational research*, *60*(1), 1-64.

Lynch, M. & Edgerton, S. Y. (1996). Abstract painting and astronomical image processing. In A. I. Tauber (Ed.), The elusive synthesis: Aesthetics and science (pp. 103 – 124). Dordrecht, Netherlands: Kluwer Academic Publishers.

MacLeod, M., & Nersessian, N. J. (2013). The creative industry of integrative systems biology. *Mind & Society, 12*(1), 35-48.

Manz, E. (2015). Resistance and the development of scientific practice: Designing the mangle into science instruction. *Cognition and Instruction*, *33*(2), 89-124.

McCloskey, M. (1983). Naive theories of motion. In D. Gentner & A. Stevens (Eds.), *Mental Models* (pp. 299-324). Hillsdale, NJ: Lawrence Erlbaum.

McDermott, L. C., Rosenquist, M. L., & van Zee, E. H. (1987). Student difficulties in connecting graphs and physics: Examples from kinematics. *American Journal of Physics, 55*, 505-513.

National Research Council (NRC, 2007). *Taking science to school: Learning and teaching science in grades K-8*. National Academies Press.

National Research Council (NRC, 2012). A framework for K–12 science education: Practices, crosscutting concepts, and core ideas. Washington, DC: National Academies Press.
Nersessian, N. (2008). Creating scientific concepts. MIT Press.

NGSS Lead States. (2013). *Next Generation Science Standards: For states, by states*. Washington, DC: National Academies Press. *www.nextgenscience.org/ next-generation-science-standards*.

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

Pickering, A. (1995). *The mangle of practice: Time, agency, and science*. University of Chicago Press.

Resnick, M., Berg, R., & Eisenberg, M. (2000). Beyond black boxes: Bringing transparency and aesthetics back to scientific investigation. *The Journal of the Learning Sciences*, *9*(1), 7-30.

Schwarz, C. V., & White, B. Y. (2005). Metamodeling knowledge: Developing students' understanding of scientific modeling. *Cognition and instruction*, *23*(2), 165-205.

Sengupta, P., Dickes, A., & Farris, A. V. (2018). Toward a Phenomenology of Computational Thinking in K-12 STEM. In M. S. Khine (Ed)., *Computational thinking in STEM disciplines: Foundations and research highlights.* Springer.

Sengupta, P, & Farris, A. V. (2012). Learning kinematics in elementary grades using agent-based computational modeling: A visual programming approach. *Proceedings of the 11th International Conference on Interaction Design & Children* (pp. 78-87). Bremen, Germany.

Sengupta, P., Dickes, A. C., Farris, A. V., Karan, A., Martin, K., & Wright, M. (2015). Programming in K12 Science Classrooms. *Communications of the ACM, 58*(1), 33- 35.

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, 17*(1-2), 23 - 42.

Sengupta, P., Kinnebrew, J., Basu, S., Biswas, G., and Clark, D. (2013). Integrating Computational Thinking with K12 Science Education Using Agent-Based Modeling: A Theoretical Framework. *Education & Information Technologies*, Vol. 18, 351 – 380.

Sherin, B. L. (2001). How students invent representations of motion: A genetic account. *Journal of Mathematical Behavior* (19) 399-441.

Sherin, B., diSessa, A. A., & Hammer, D. (1993). Dynaturtle revisited: Learning physics through collaborative design of a computer model. *Interactive Learning Environments*, *3*(2), 91-118.

Turkle, S., & Papert, S. (1992). Epistemological pluralism and the revaluation of the concrete. *Journal of Mathematical Behavior, 11*(1), 3-33.

Wechsler, J. (1988). *On aesthetics and science.* MIT Press, Cambridge.

Wilensky, U. (1991). *Abstract meditations on the concrete and concrete implications for mathematics education*. Epistemology and Learning Group, MIT Media Laboratory.

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.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33-35.

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical transactions of the royal society of London, A: mathematical, physical and engineering sciences*, *366*(1881), 3717-3725.

CHAPTER II:

DEMOCRATIZING CHILDREN'S COMPUTATION: LEARNING COMPUTATIONAL SCIENCE AS AESTHETIC EXPERIENCE

Author note: An earlier version of this paper appeared in Farris, A. V. & Sengupta, P. (2016). Democratizing children's computation: Learning computational science as aesthetic experience. *Educational Theory, 66*(1-2), 279 - 296.

Abstract

In this paper, we argue that a democratic approach to children's computing education in a science class must focus on the *aesthetics* of children's experience. In *Democracy and Education*, Dewey links democracy with an even more distinctive understanding of experience. For Dewey (1916), the value of educational experiences lies in "the unity or integrity of experience" (p. 248). In *Art as Experience,* Dewey presents *aesthetic* experience as the *fundamental* form of human experience that undergirds all other forms of experiences, and can also bring together multiple forms of experiences, locating this form of experience in the work of artists. Particularly relevant to our current concern (computational literacy), Dewey (1934) calls the process through which a person transforms a material into an expressive medium an aesthetic experience. We argue here that the kind of experience that is appropriate for a democratic education in the context of children's computational science is essentially *aesthetic* in nature. Given that aesthetics has received relatively little attention in STEM education research, our purpose here is to highlight the power of aesthetic experience in making computational thinking available to and attractive to all children, including those who are disinterested in computing, and especially those who are likely to be discounted by virtue of location, gender or race.

Over the past several years, *computational literacy* (diSessa, 2001) has become an important topic for discussion for K12 STEM education. Developing computational literacy requires developing epistemic and representational practices such as thinking algorithmically, and designing and creating computational artifacts such as programs and simulations. Still, computational literacy does not yet have any noticeable representation in the standard scope and sequence of public schools, especially at the elementary level. Several scholars have argued that increasing access to computational literacy for children in the realm of public education involves integrating computation with existing courses such as science and math that *all* children are required to take, rather than trying to create room for computer science as a new curricular domain (e.g., Wilensky, Brady, & Horn, 2014; Sengupta, Kinnebrew, Basu, Biswas, & Clark, 2013). Some scholars have also argued that broadening access to computation must involve efforts to create computing in the image of children's lives, and *not* vice versa; however such anthropological approaches to children's computing remain largely outside the purview of STEM classrooms (Eisenberg, 2012).

In this paper, we argue — in part by example — that an effective and democratic approach to children's computing education within a science class can and must focus on the *aesthetics* of children's experience. In *Democracy and Education*, Dewey (1916) links 'democracy' with an even more distinctive understanding of 'experience'. For Dewey, the value of educational experiences lies in "the unity or integrity of experience" (p. 248). In *Art as Experience,* Dewey (1934) presents *aesthetic* experience as the *fundamental* form of human experience that undergirds all other forms of experience, that can also bring together multiple forms of experiences, and locates this form of experience in the work of artists. Particularly relevant to our current concern (computational literacy), Dewey (1934) calls the process through which a person transforms a material into an expressive medium an aesthetic experience (pp. 68-69). In this paper, similar to Higgins (2008), we maintain that the kind of experience that is appropriate for a democratic education is essentially *aesthetic* in nature. We extend Higgins' argument by claiming that aesthetic education should not only be exemplified by the arts – it must also bring computing and science education into its fold. Further, we illustrate that in doing so it can fundamentally transform computational science as an *experience* to a more inclusive one, especially for young learners at the fringes of computing.

This paper is structured as follows: First, we explain that a) although aesthetics has been studied as an important aspect of the work of scientists, that framing of aesthetics does not concern itself with a democratic nature of learning; and b) the discourse about children's computing and science education in general fails to account for the aesthetic dimensions of learning. Next, we articulate our own view that *democratizing* children's computing (in a Deweyan sense, that is, grounded in aesthetic experience) is a pathway to worthwhile STEM education. Finally, we demonstrate that this democratized and aesthetic experience is possible by tapping our own research as a working model, that is, a demonstration of the integration of computation as an example of how it became a democratizing force.

On Aesthetics in Professional Science and K12 STEM

Philosophers and historians of science agree that there is an epistemic role of beauty and aesthetics in the development of scientific knowledge, and furthermore, that both beauty and aesthetics often represent deep conceptual understanding in science. For example, Kosso argued that physicists' own admissions about what they find beautiful about their theories are premised on the aesthetic qualities of coherence and interconnection, for example*.,* a deep understanding of the relevant theory, as that may reveal "interconnectedness of facts" (Kosso, 2002, p. 43).

Along similar lines, Engler (2002) argued that the beauty associated with Einstein's theories, which was also widely acknowledged by his peers, can be understood in light of the following aesthetic qualities: simplicity, symmetry (including invariance, equivalence, and covariance), unification (unity) and fundamentality. It has also been argued that choosing theories on aesthetic grounds – that is, what makes a theory *beautiful*—is neither irrational nor a hindrance to progress because the aesthetic properties of theories are, by and large, reliable indicators for the empirical adequacy of theories (Chandrasekhar, 1987) and even in cases where aesthetic qualities are derived from finding analogical relationships between a multitude of phenomena by conducting "mimetic" experiments (Rueger, 2002) the underlying theme of interconnectedness is still evident.

Philosophers of science have also argued for the importance of scientists' interpretive work in the production scientific knowledge, including scientific inscriptions, such as drawings, diagrams, photographic images, and computer visualizations. For example, Nersessian showed the importance of developing fictive representations as explanatory models in Maxwell's work on electricity. More generally, Gooding (2003) noted that scientists make knowledge by "relocating it, moving it from the personal and local context to the larger domain of publicly reproducible phenomena, proofs, or processes" (p. 261). In their critique of objectivity in the sciences, Daston and Galison (2007) argue that the production of inscriptions in science reflect the values and the epistemology of the scientific culture in which they are made. For example, with the rise of photographic technologies in the 19th century, the perceived objectivity of mechanized images led to treating them as "facts"; the use of interpretive representations such as hand-made drawings also declined. This form of mechanical objectivity can be contrasted with the use of images in modern astronomy, where non-imagistic and necessarily interpretive representations (for example, infrared emission data) are often represented alongside and within photographic images, in a manner that is meaningful to a wider audience (Datson & Galison, 1992). The progressive centrality of computation in current scientific practice has further transformed the epistemological nature of science. For example, scientists computationally develop simulations in cases where data is sparse; these simulations, which are essentially fictive and analogous representations of reality, then serve as further sources of data (MacLeod & Nersessian, 2013).

One can therefore conclude that although scientists generally acknowledge that their work has inescapable aesthetic dimensions, they have typically focused on the aesthetics of "final form" science. There is little, if any, understanding of the aesthetic dimensions of scientific practice, as well as the journey of "becoming" a scientist. In the domain of K12 STEM education research, a few scholars have begun recognizing that learning science is itself an aesthetic experience; however, this body of work, despite adopting a Deweyan perspective, proposes a rather thin, and purely discursive conception of both aesthetics and experience.
For example, Girod, Rau and Schepige (2003) adopt the transformative and continuous qualities of Deweyan aesthetic experiences. They argue for including the artful aspects of science for generating interest and an expansion of perception in the sciences. They locate aesthetics in the classroom discourse, and present guidelines for teaching science as an aesthetic experience by proposing forms of questions, that teachers could ask students during the curriculum, with the central goal of helping students establish personally meaningful connections to content. Jakobson and Wickman (2008) also adopted a similar definition of aesthetics and identified the role of qualitative judgments such as "nice" or "disgusting" in students' utterances as indicators of conceptual understanding, and genres of experiences.

While applauding the inclusion of aesthetic experiences in the K12 science classroom, Lemke (2001) critiques this approach by arguing that “the heightened vitality” we associate with a Deweyan “Experience” in professional science can be understood in terms of “accounts of what happened, or experienced from idea to design to data and conclusions” (p. 300 – 301). Lemke argues that central to this account of the transformative nature of Experience in the production of scientific knowledge is “the vital fusion of theory and experiment (or observation) that makes science truly a performance art,” which science education has failed to address in an authentic fashion.

We agree with Lemke’s critique, that in science education research, the transformative nature of experience resulting from fusion of theory and experiment is rarely investigated. The historian of science Pickering (1995) has termed this vital fusion the “mangle of practice,” and argued that this fusion is much deeper than discourse, by showing that at the heart of scientific progress is the “dance of agency” between theories and instrumentation. However, Pickering’s work does not address the dimensions of the scientists’ affective involvement and personal meaningfulness, which are also essential elements of Deweyan aesthetic experiences. Along this line, a further critique of the studies of aesthetics in professional science is that they can be viewed as efforts to identify “beauty at the helm,” as they focus on the aesthetics of the interested—for example, accomplished scientists, who were deeply interested and thoroughly engaged in their professional pursuit. A heightened form of deep engagement occurs when the scientist literally identifies herself or himself with the object of inquiry by conceptually projecting herself or himself on the object, and engages in thinking *like the object of inquiry* (e.g., Keller, 1983; Ochs, Gonzales, & Jacoby, 1996).

In stark contrast to such heightened forms of engagement and experiences lie the learning experiences of the dis-empowered and the dis-interested, who are typically left out of the fold of deep engagement with the curricular content in most classrooms (Delpit, 1988). In the context of educational computing, this population includes women and ethnic minority students as well as students interested in the arts, most of whom do not identify themselves as computing or STEM competent, even at the college level (Margolis & Fisher, 2002). As Dewey (1916) argues, democratizing education necessitates the focus on how to foster conditions in which such students will develop a deep interest in their curricular work, and (in the context of learning science) come to see that work as both scientifically and personally meaningful (for example, pp. 128, 227-239). We therefore posit that the study of aesthetic experience in science learning that does not concern the disinterested or the disenfranchised is fundamentally undemocratic. We therefore ask the following question: What is the nature of aesthetic experience for such students in the context of doing computational science? We address this in the next section.

Democratizing Science and Computing Education: The Role of Aesthetic Experiences

Dewey (1916) argued that since a democratic society repudiates the principle of external authority, it must find a substitute (of authority) in voluntary disposition and interest, and further, that education is the means through which interest could be generated (Chapter 7). Achieving coherence between the learners' interest or what he terms "true aims", and pedagogical aims, would foster continuity of the pedagogical experience with the learners' experiences outside the classroom (Chapter 8). Dewey therefore argues for two forms of continuities—continuity of the curricular experience with the learner's life outside the classroom, and continuity of the pedagogical aims with the "true aim" of the learner. He claims that the latter form of continuity is dependent on the former.

For Dewey (1916), “the measure of the value of an experience lies in the perception of relationships or continuities to which it leads up” (p. 147). The richness of an experience is marked by a variety of interests, but Dewey argues that these interests have been “torn asunder” in schools. Curricular domains of knowledge are institutions that are disconnected from each other (pp. 294-297), and this isolation of curricular experiences “rupture[s]… the intimate association” between domains of knowledge as experienced by the learner in a continuous form in his or her everyday life outside the classroom (p. 295). Dewey considers this a serious breach in the learners’ continuity of mental development, because this makes the curricular experience unreal for the learner, and can therefore, lead to a loss of interest. Dewey then challenges us to think beyond these discontinuities for pedagogical design:

> The point at issue in a theory of educational value is then the unity or integrity of experience. How shall it be full and varied without losing unity of spirit? […] How shall we secure breadth of outlook without sacrificing efficiency of execution? How shall we secure the diversity of interests, without paying the price of isolation? (pp. 238-239)

In contrast to the fragmented experiences that are still common in public educational settings stands a more fundamental form of experience that in his later work, Dewey (1934) termed “[a]esthetic experiences.” (We have adopted the more common modern spelling: “aesthetic,” however Dewey used “esthetic.). Dewey argues that in the case of an aesthetic experience, the traditional divide between domains of knowledge (such as science, art, religion, etc.) do not exist, because such experience is fundamental to *all* domains. He finds the paradigm of such experiences in the artist, and argues that aesthetic experiences arise in the artist’s process of transformation of a *material* into an *expressive medium* (pp. 111-113). The process of expression is necessarily constrained, but not restrained—that is, the conversion of an act of immediate discharge (i.e., a direct representation) into one of expression depends upon the existence of conditions that impede direct manifestation and instead “switch it to a channel where it is coordinated with other impulsions” (p. 102). This modification of the original impulsion by “cooperative” and “collateral tendencies” gives it added meaning – “the meaning of the whole of which it is henceforth a constituent part” (p. 102). The expressiveness of the object therefore represents an interpenetration of the materials of undergoing and of action, and thus, the “complete fusion of what we undergo during the process of expression” (p. 108).

It is this interpenetrative nature that makes aesthetic experiences *fundamental,* in that they transcend domains of knowledge and represent the unity of experience through which the object becomes expressive, and personally meaningful to the artist. Aesthetic experiences thus foreground *experience* over canonical forms of knowledge that typically exist in isolation from one another, both in professional practice and pedagogy. This isolation, Dewey (1950) argued, is the result of “non-experiential” or “anti-experiential” philosophies, which Dewey contrasts with the fundamentally continuous nature of experience.

We find Dewey’s notion of *aesthetic experiences* to be appropriate for our purposes for two reasons. The first reason is tied to the nature of computation (including its practice): domain-generality is a “habitual nature” (p. 109) of computational programming and modeling. The creation of computational programs that underlie any usable software (or application) involves the use of computational abstractions (Wing, 2006), such as representational structures that are domain-general (e.g., algorithms, data structures such as lists and arrays, etc.). That is, the same programming language can be used to create applications in diverse domains such as physics, biology and social sciences. In our own research, we have used the same programming language to develop models in physics, biology, microeconomics and artist networks. The essential nature of the practice of computation is therefore *transformative.* That is, in Dewey's terms, the material of computation—typically, a programming language—gets transformed to an expressive object, a software application that has value because of its usability and meaningfulness in other domains.

Our second argument concerns Dewey's emphasis on the continuity of learning experiences for a democratic education. Herein lies an important affordance of the particular genre of computation we use in our worked example: agent-based computation, that is, a form of computation where a user can simulate a complex phenomenon (e.g., a traffic jam) through programming the behaviors of virtual agents, by assigning them simple, body-syntonic "rules", (e.g., moving forward, slowing down, etc.). The complexity of the overall phenomenon (e.g., the formation and backward propagation of the jam) emerges from the aggregation of simple, agent-level behaviors. Furthermore, because a computational agent is a protean agent, it can take on any form: an image, a word, an object, a mathematical representation (e.g., a graph), etc. This in turn makes agent-based computation a suitable medium for modeling phenomena in domains as diverse as physics, biology, art and engineering.

Over the past three decades, research on making agent-based computation accessible to young learners has identified several activity forms that can potentially support interest-driven computing. These studies extend the range of learning activities beyond the traditional image of programming as writing code to include new forms of activities within which programming is embedded: game design (e.g., Repenning, Smith, Owen, & Repenning, 2002; and Smith, Cypher, & Tesler, 2000), digital narratives (e.g., Resnick et al, 2009), digital animations of sketches and graphic design (e.g., Bollen & van Joolingen, 2013), and integration of programming with physical computing and the use of low-tech objects (e.g., Resnick, Berg, & Eisenberg, 2000.

Using Wilensky's (1991) definition of "concrete", where concretion is defined as the process of the new knowledge "coming into relationship with itself and with prior knowledge" (p. 201), such forms of knowing can be termed "concrete". That is, as the learners (in these studies) engage in the development of multiple, personally meaningful representations of the object of inquiry, they begin to "see" the unknown using experiences that are personally meaningful and familiar.

One can therefore argue that these studies present us several images of learning that allude to some elements of the Deweyan notion of aesthetic experiences. For example, taken together, these studies suggest that computation, and in particular agent-based computation, is indeed a malleable medium that can lend itself to multiple activity forms, and further, that certain forms of computation might even bring together multiple domains within the act of learning. Some of these studies also show that using agent-based computation, learners can appropriate the goals of the assigned activity in order to pursue something rising from their own interests, but without losing focus on the disciplinary learning objectives. Azevedo (2006) termed these forms of learner-generated activities "personal excursions."

To summarize, we argue that the *transformative* and *fundamental* nature of aesthetic experiences can provide us useful guidelines for designing an inclusive and democratic pedagogy for kids' computing in particular. Along the first dimension, we posit that pedagogical experiences should provide learners opportunities to transform a material (e.g., a computational programming language) into an expressive medium. In the context of computing education, this means that the learner should be able to create a personally meaningful artifact. This in turn, requires balancing the leaners' interests or true aims with institutionally mandated aims that instructors have to abide by. With respect to the second dimension, the fundamental nature of aesthetic experiences implies that the learning experience must also be continuous. That is, it should also enable learners to connect the present experience with their lived experiences outside the classroom, and also to bridge different domains that are traditionally taught as ontologically distinct from one another. The example offered here represents a computing experience that is both transformative and fundamental for Matt and Ariana. The experience was inclusive, inviting them in to a domain of practice for which they initially had no interest, and enabled them to participate fully.

A Worked Example

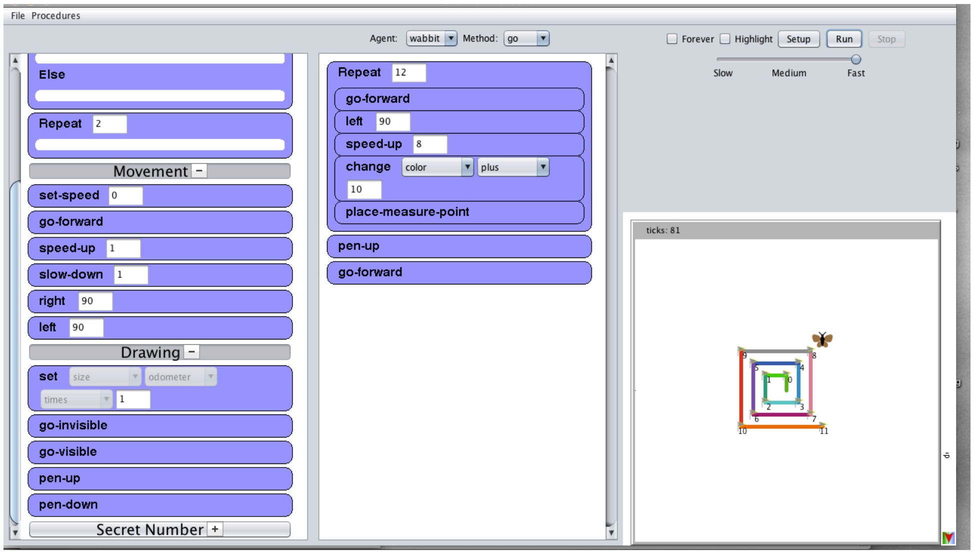
Matt and Ariana, the two 5th grade students considered in this example were enrolled in a two-week long summer course on agent-based computer modeling for learning science that we (the authors) co-taught at Vanderbilt University. During the first couple of days in the course, Ariana and Matt each disclosed to the researchers that they had no interest in computer programming. Ariana was especially interested in history and literature, and Matt was an aspiring actor. Neither saw themselves as people who might enjoy or be good at computer programming, and both of them had joined the course based on their parents' insistence.

From our perspective, as instructors of the course, the central disciplinary learning goals for students in terms of learning programming and physics were: a) to develop fluency with agent-based programming and modeling motion as a process of continuous change; and b) in the process, begin to develop deep conceptual understandings of the relationships among distance, speed, and acceleration. Developing an understanding of motion as a process of continuous change has been shown to challenging for K12 learners, particularly at the elementary and middle school level (Dykstra & Sweet, 2009; Elby, 2000). In the first phase of activities, students were introduced to the ViMAP software (described below) by drawing shapes. In Phases II and

III students generated data about motion by acting as the “agent” in “real-life” situations such as travelling on the building’s elevator and observing the free-fall of a block of ice. Note that our goal was to reframe learning computational science as an aesthetic experience; therefore, we intentionally integrated multiple domains and tools in our pedagogy. For example, in Phase IV, besides ViMAP, students also used a musical programming software called Impromptu TuneBlocks (described below) to build computational models of motion, based on the data they generated during the embodied modeling activities. TuneBlocks enhanced the representational palette of learners to include musical attributes such as pitch and tempo as possible representations for speed and acceleration. For example, an object that is accelerating at a steady pace could be modeled musically in terms of the steadily increasing pitch of a note.

The “Tools”: ViMAP and TuneBlocks

ViMAP (Figure 1) is an agent-based visual programming language and modeling platform (Sengupta, Farris, & Wright, 2012) that uses NetLogo (Wilensky, 1999) as the simulation engine. Instead of typing text-based commands, users use a drag-and-drop interface to select and choose commands from a library of commands in order to control the behavior of a single computational agent—a “turtle”. The ViMAP version used in this study had two components: a construction world, where learners construct their programs by organizing the visual programming blocks; and an enactment world, where a protean computational agent (or a set of agents) carries out users’ commands through movement on the computer screen.



```
Repeat 12
    go-forward
    left 90
    speed-up 8
    change color plus 10
    place-measure-point
pen-up
go-forward
```

Figure 1. The ViMAP interface.

Impromptu (Bamberger & Hernández, n.d.) is a computer-based musical programming environment in which students can learn to compose melodies using their musical intuitions by arranging small blocks that represent musical notes. We used one of Impromptu's five "PlayRooms", called Tuneblocks. In this course, students composed tunes or melodies by editing existing tunes from the TuneBlocks library and used pitch or the duration of the notes in order to represent constant speed, constant acceleration and constant deceleration (see Figure 2).

In our research, we used an illustrative case-study approach (Yin, 2013) grounded in naturalistic inquiry methods. We videotaped in-depth interviews with the students in order to understand students' perspectives on and explanations of their own work. These interviews were transcribed and analyzed inductively using the double-coding method in order to identify salient themes. Here we present two episodes of Matt and Ariana's work, one near the beginning of the course, and a second episode occurring during their final project. In each episode, we identify two themes, which are key criteria of Deweyan aesthetic experiences: the synthesis of multiple domains of knowledge and practice that traditionally remain separate in classroom instruction, and the balancing of true aims and institutional aims, through the realization of the representational properties of the different forms of computation media.

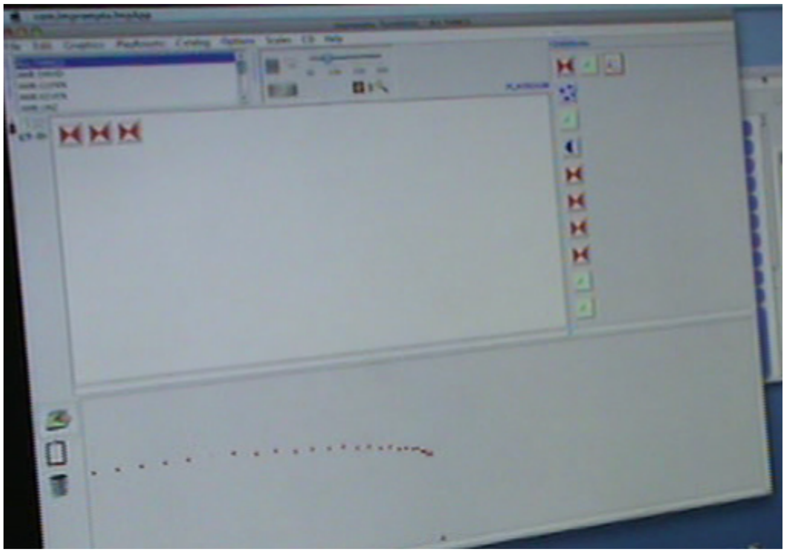


Figure 2. Ariana's TuneBlocks model of acceleration.

Episode 1: Programming "Thomas"

After completing the introductory activity of drawing some simple LOGO-based shapes, Ariana began writing a ViMAP program to make the turtle write "Thomas" (Figure 3). This activity began as a teacher-directed task, in which we asked students to either draw a shape of their choosing, or draw one letter from their name. Ariana's work spanned several days during the first week: she worked on other assigned tasks and kept returning to complete the Thomas program when she found time. Writing "Thomas" became an important side-project for Ariana, one of her own choosing.

After observing her eager work on this project, a researcher interviewed her about the meaning of an inscription of the name "Thomas." In this interview (transcript provided in the appendix), Ariana explains her relationship with Thomas, and her statements provide evidence of the continuity between her biographical experiences outside her classroom and the programming activity. There are five Thomases in Ariana's life, including the newly found ViMAP turtle.

Ariana: The fourth Thomas is my best friend, the third Thomas is from *The Maze Runner*, and the second Thomas is the Maze Runner's dad, and the first Thomas is Thomas Edison, the scientist. And this will be our fifth Thomas, our little turtle here. And he is so good. He is going to preschool and he is knowing how to spell his name.

One way to interpret Ariana's work is that through this activity, she brings together some of her favorite aspects of her personal life—for example, her fondness for her best friend, who is also her neighbor and a classmate and "really, really close" to her, and her favorite fiction character from a young-adult book series (Thomas in *The Maze Runner*)—and merges them with the protean ViMAP (LOGO) turtle. The ViMAP turtle, as Ariana points out in her interview, is the fifth, and youngest Thomas in her life. The turtle has now become an object of affection for her - she positions Thomas the ViMAP turtle as a preschooler, who learning how to spell his name. The turtle, as Papert (1980) pointed out, therefore acts as a *transitional object*—i.e., both as a protean computational object, as well as a representation of the child's favorite aspects from her own biography outside the classroom.

Figure 3. Ariana’s ViMAP model of “Thomas.”

The Thomas narrative also created a space for humor between Matt and Ariana. During her interview, Matt, another 6th grader who sat next to Ariana, jokingly complained about “too many Thomases!” As the interview began, Matt attempts, humorously, to prevent Ariana from going “through the list.”. As the interview proceeded, he exclaimed “it burns! It burns,” covering his ears, and making humorous expressions, playfully communicating that he did not see the importance of the Thomases that Ariana did. As Matt clarified after the interview, his attempts at humor were directed to indicate that he had already been subjected multiple times to listening to the long legacy of the Thomases in Ariana’s life. This further suggests the importance of Thomas in Ariana’s life. Once Ariana completed her excursion, Matt and Ariana, who did not know each other prior to this class, chose each other as programming partners and continued to work together on all subsequent assignments.

In what follows, we highlight the two key criteria of an aesthetic experience that are central to a democratic education, as evident in this episode:

Continuity across Domains. In Ariana's work, geometry and programming were deeply intertwined with one another. Programming involved the successful use of relevant computational abstractions, such as variables and loops. In terms of learning geometry, using turtle graphics to create the shape of a letter involves thinking *like* the turtle, in order to use the egocentric coordinate system in ViMAP, a feature of agent-based modeling. However, note that it was her love and affection for the many "Thomases" in her life that created this context for productive unification of these domains. Simulating the trajectory of the computational agent (the turtle) in the shape of each letter involved significant complexity in terms of figuring out both the turtle's egocentric coordinate system, as well as the Cartesian coordinates of the pixels at the beginning of each letter. On the other hand, the instructor-mandated activity of drawing only a single alphabet letter would have involved a far less extensive exploration of both key geometry and programming.

Balancing Institutional Aims and True Aims. Ariana's project shows that the computational agent (the turtle) truly became a transitional object—i.e., she projected her identity onto the turtle. Her way of learning programming was by making the turtle learn how to write Thomas. This in turn transformed the material (ViMAP) and the activity (learning programming by drawing letters) into a means to talk about her serendipitous encounters with the many Thomases in her life: literary figures, historical figures, and friends. Matt became humorously critical of Ariana's personal attachment to Thomas and her persistence with the Thomas project. This relationship created a space for playful humor between the two students, which was important for Matt, who wanted to be an aspiring actor, one with an expressed interest in comedy. As Matt became familiar with Ariana's project, he progressively developed a deep interest in how Ariana had calculated the size each letter in relationship to the geometry of the

ViMAP world, because it was closely related to a challenge he was facing in his own work. During their collaboration, humor played an important role, and established a comfortable working relationship between the dyad. They decided to work together as partners for the remainder of the course.

Episode 2: A Collaborative, Multimodal Model of Acceleration

During the final phase of the activities, students were asked to represent how the speed of a car on a roller coaster (as shown in a YouTube video) was changing, using either ViMAP or TuneBlocks. Matt and Ariana decided to collaboratively develop both a ViMAP and a TuneBlocks model. Their ViMAP model (Figure 4) represented a period of constant acceleration of the roller coaster using line segments (dot-traces) of different colors to represent distance traveled in each interval of time. Speed was represented by gaps between successive dots: constant speed meant equal gaps between successive dots, and acceleration meant increasing gaps. Ariana also explained the significance of the color changes of the lines: “The color kind of rapidly changes and then spreads out.” When one of the interviewers asked her to explain this more, she explained that:

> Ariana: …It kind of changes because it is kind of slow during HERE (pointing to top portion of the line), then it spreads out and the colored lines get further, so that would be one of the reasons that it is better [than alternative models] and it has…(pause)...This would be acceleration, see here how it is getting, how it’s kind of slow, how they are all crumpled up, and they get bigger and bigger.

From the perspective of learning physics, the learning goal of the activity was for students to begin to distinguish among distance traveled, speed, and acceleration. In Ariana’s explanation, “crumpled up” was a visual metaphor for slow, and “spread out” for fast. Her explanation also makes explicit how she was using a systematic change in color to represent the rate of change in motion. While she did not explicitly identify rate of change, her explanation suggests that she was beginning to identify how fast (or slow) the color was changing as an important and communicative aspect of her representation.

Later in the interview, when Amy asked Matt and Ariana about the regularity of the change in distance per unit time, Ariana pointed to the steady regularity of the placement of measurement flags in the execution of her model: the same commands repeated in a loop: forward (step-size), plant-flag, speed-up (increase of step-size), change-color (amount). The words in parenthesis indicate the parameters associated with the commands that the students also had to specify.

Matt's explanation of steady, however, was somewhat different:

> Matt: Because it's just accelerating like .. [Matt snaps two of his fingers twelve times. The frequency of snaps increases steadily].

In this excerpt, Matt uses a steadily increasing frequency of a particular sound to explain what he means by "steady". The increases in distance occur regularly, because their ViMAP model is incrementing the distance travelled by the turtle in each step by the same amount that is decided using the "speed-up" command. Matt's explanation of a steady pace used a combination of gesture and sound to represent a *steadily increasing* tempo in order to represent a steady acceleration. This in turn was similar to the representation of acceleration in their TuneBlocks model that accompanied their ViMAP model. In their TuneBlocks model (Figure 2), they used two variables—pitch and duration—as representations of change in speed. They used a gradually decreasing pitch to indicate the decreasing altitude of a roller coaster moving down a steep incline, and programmed the duration of each note to represent speed, where increasingly shorter, closer together tones indicated increasing speed.

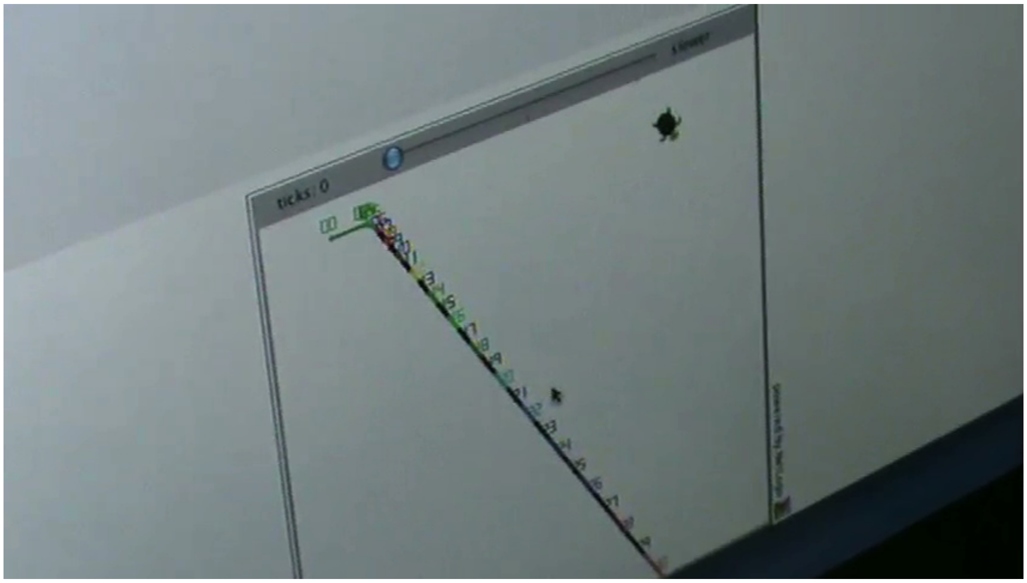


Figure 4. Ariana and Matt's ViMAP model of motion.

After constructing both the models, Matt and Ariana decided to synchronize their models so that both the ViMAP and the TuneBlocks models would each serve as components of one unitary model of the motion phenomenon. This eventually resulted in a "live performance" (as Matt explained to an instructor), where they "played" both their models simultaneously for the instructors. During the final segment of the interview, Matt extended the description of the ViMAP model to include periods of gradual slowing down and of rest in order to illustrate a narrative about accelerating onto a highway, then getting off at an exit, stopping at a red light, and parking—a situation that was familiar to him from his daily life.

Again, we highlight the two key criteria of an aesthetic experience that are central to a democratic education, as evident in this episode:

<u>*Integration of Domains of Knowledge*</u>. In this episode, learning about the physics of motion—i.e., learning to represent motion as a process of continuous change was deeply

intertwined with use of programing and musical notations. Similar to Episode 1, the use of computational abstractions such as variables and loops continue to serve an important role here: color and gaps were used as representations of speed and acceleration. In Ariana's case, her explanation used visual attributes such as color, while Matt used a steadily increasing tempo of finger-snaps to represent rate of change. Mathematically speaking, Ariana and Matt also began to develop representations of *rate* of change of motion and represent changes in speed in terms of continuous change in the distance traveled in each successive increment of time. They each describe a different aspect of the uniformity of increase in motion that is accelerating at a constant rate: the uniformity of the chronology of measurement (Ariana) and the uniformity of the *change* in speed (Matt). Furthermore, as Ariana's verbal explanations make explicit, she was also beginning to distinguish between different rates of acceleration (e.g., "crumpled up" vs. "spread out"). The introduction of TuneBlocks further widened the representational palette for the children; musical attributes such as tempo and pitch were used as representations of speed, and modeling motion was transformed to musical composition.

Balancing Institutional and True Aims. The multi-modality in Ariana and Matt's models illustrate children's agency in interpreting and symbolizing scientific ideas. Contrary to the instructors' advice, the dyad also refused to choose one programming environment over another, and instead, created for themselves the goal of synchronizing two models to be executed at the same time. They therefore created a perceptually enhanced, representational account of motion that consisted of both visual and auditory representations of motion as a process of continuous change. Their final project was therefore a coordinated *performance:* Matt ran the visual simulation designed in ViMAP while Ariana played the audio tune designed in TuneBlocks in a synchronized fashion. To the students, the final project was therefore a work of art, *despite* being a composite model of motion. It was thus a realization of the true aims of the two artistically inclined children—Matt the future actor, and Ariana, the history and literature fan.

Conclusion

Can the disinterested find their voices in the STEM classroom, especially in classrooms where computation is the medium of "doing" science? To answer this question, we have argued that one must reimagine learning computing in the science classroom as an aesthetic experience (in the Deweyan sense). That is, the democratization of computing hinges on designing pedagogies that enable the learner, especially the disinterested, to transform the computer as a material into an expressive medium, in a manner that can create a fundamental and unifying experience for the learner. Grounded in Dewey's (1916) work in *Democracy and Education*, we have further argued that such aesthetic experiences must bring together sanctioned domain knowledge with the leaners' experience and intuitions from their everyday lives.

This bringing together is not an act of recognition – but as Higgins (2008) argues, it is an act of heightened perception, an act of "seeing more" rather than merely seeing. Computational media such as the kinds we report here can concretize this metaphor of "seeing more" by enriching the perceptual engagement of the learner. To this end, Matt and Ariana's work shows that by opening up the representational palette to include multiple modalities of expression such as visual dynamics of agent-based simulations, visual and auditory representations of musical notations, and musical composition, computational media support authentic engagements of children with the analogical, interpretive, and symbol-laden work of science, while accommodating their interests. It can therefore provide children entrée into a new *kind* of science, where the mundane is reimagined as complex, and children's *true aims* find a place alongside the institutionally mandated aims.

Dewey (1934) argues that the habitual nature of art is such that it helps us “see” complexity in the world of our everyday experiences, the mundane. He wrote: "Art throws off the covers that hide the expressiveness of experienced things" (110). We see this re-imagining of the mundane to be at the core of democratic pedagogy, both in general, as well as in the specific context of educational computing, especially in the science classroom. The many Thomases in Ariana’s life and Matt’s experience of his daily car rides as well as his aspiration to be an actor are representations of children’s interests and Deweyan “true aims” that are typically left behind in pedagogical time. The meaning that learners develop in STEM classrooms, in the truly democratic sense, must not be depleted or devoid of these true aims, because the “value” of a democratic education lies in the unity (or integrity) of experience.

Appendix: Ariana and Matt on "Thomas"

Transcript conventions include the following:

[ ]	Brackets are used to show overlapping speech of two speakers
=	Latched speech
..	Pause, less than 2 seconds

Ariana: Well, my best friend's Thomas and I really, really, really, we're really, really close. See, he's like my neighbor [and he's also my classmate

Matt: Don't go through the list!] Don't go through the list!

A: Okay=

M: =too many Thomases, way too many Thomases

A: And, [oh yeah, I made a video about how

M: *(whispering to the camera)* too many, too many]

A: I made a video about an [explanation of the Thomases

M: *(moaning)* I said don't go through with it!]

A: So it's Thomas *(counting on fingers)* named after Thomas who's named after Thomas, who's named after Thomas.

M: *(covering ears)* it buuuuurrrr-urrrr-urrrr-urrrrns!!

A: The fourth Thomas is my best friend.

M: It burns! It bur-ur-urns!

A: The third Thomas is..um..from *The Maze Runner,*

M: *(whispering)* It burns!

A: and the second Thomas is the Maze Runner's dad and the first Thomas is Thomas Edison..the scientist. And this will be our fifth Thomas, our little turtle here and he is so good..He is going to preschool and he is knowing how to spell his name.

Pratim: So the turtle is going to preschool and he is knowing how to [spell his name?

M: Heeelp me, help me! (hands extended to the camera)]

A: Yes, basically.

Pratim: Alright.

References

Azevedo, F. S. (2006). Personal excursions: Investigating the dynamics of student engagement. *International Journal of Computers for Mathematical Learning*, *11*(1), 57 - 98.

Bamberger, J. & Hernández, A. (n.d.)*Impromptu (computer software).* (*Version 2.4.1b*: www.Tuneblocks.com.

Bollen, L., & van Joolingen, W. R. (2013). SimSketch: multiagent simulations based on learner-created sketches for early science education. *IEEE Transactions on Learning Technologies, (3),* 208-216.

Chandrasekhar, S. (1987). *Truth and beauty: Aesthetics and motivations in science*. University of Chicago Press.

Daston, L., & Galison, P. (1992). The image of objectivity. *Representations.* (*40*, Autumn: Special issue: Seeing Science). 81 – 128.

Daston, L., & Galison, P. (2007). *Objectivity.* Zone Books.

Delpit, L. (1988). The silenced dialogue: Power and pedagogy in educating other people's children. *Harvard educational review*, *58*(3), 280-299.

Dewey, J. (1950). Aesthetic experience as a primary phase and as an artistic development. *The Journal of Aesthetics and Art Criticism*, *9*(1), 56-58.

Dewey, J. (2004). *Democracy and education: an introduction to the philosophy of education.* Courier Dover Publications. (Original work published 1916)

Dewey, J. (2005). *Art as Experience.* Perigree, The Berkley Publishing Group, New York. (Original work published 1934)

diSessa, A. A. (2001). *Changing minds: Computers, learning, and literacy*. The MIT Press.

Dykstra, D. I., Jr. & Sweet, D. R. (2009). Conceptual development about motion and force in elementary and middle school students. *American Journal of Physics, 77*(5), 468-476.

Eisenberg, M. (2012). Technology for learning: Moving from a cognitive to the anthropological stance. *Proceedings of the 10th International Conference of the Learning Sciences*, Sydney, 379.

Elby, A. (2000). What students' learning of representations tells us about constructivism. *Journal of Mathematical Behavior, 19,* 481-502.

Engler, G. (2002). The omniscienter: Beauty and scientific understanding. *International Studies in the Philosophy of Science 16*(1). 27-38.

Girod, M., Rau, C., & Schepige, A. (2003). Appreciating the beauty of science ideas: Teaching for aesthetic understanding. *Science Education*, *87*(4), 574-587.

Gooding, D. (2003). *Varying the cognitive span: experimentation, visualization, and computation.* In H. Radder (Ed.), *The philosophy of scientific experimentation* (pp. 198-215). University of Pittsburgh Press Pittsburgh, PA.

Higgins, C. (2008). Instrumentalism and the clichés of aesthetic education: A Deweyan Corrective. *Education and Culture, 24*(1), 7-20.

Jakobson, B., & Wickman, P. O. (2008). The roles of aesthetic experience in elementary school science. *Research in Science Education*, *38*(1), 45-65.

Kosso, P. (2002). Einstein and the most beautiful theories in physics. *International Studies in the Philosophy of Science 16*(1). 39-48.

Lemke, J. L. (2001). Articulating communities: Sociocultural perspectives on science education. *Journal of research in science teaching*, *38*(3), 296-316.

MacLeod, M., & Nersessian, N. J. (2013). The creative industry of integrative systems biology. *Mind & Society, 12*(1), 35-48.

Margolis, J., & Fisher, A. (2003). *Unlocking the clubhouse: Women in computing*. MIT press.

Ochs, E., Gonzales, P., & Jacoby, S. (1996). “When I come down I'm in the domain state”: Grammar and graphic representation in the interpretive activity of physicists. *Studies in Interactional Sociolinguistics, 13*, 328-369.

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

Pickering, A. (1995). *The mangle of practice: Time, agency, and science*. University of Chicago Press.

Repenning, A., Smith, C., Owen, R., & Repenning, N. (2012, June). AgentCubes: Enabling 3D creativity by addressing cognitive and affective programming challenges. In *EdMedia: World Conference on Educational Media and Technology* (pp. 2762-2771). Association for the Advancement of Computing in Education (AACE).

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., ... & Kafai, Y. (2009). Scratch: programming for all. *Communications of the ACM*, *52*(11), 60-67.

Rueger, A. (2002). Aesthetic appreciation of experiments: The case of 18th-century mimetic experiments. *International Studies in the Philosophy of Science 16*(1). 51-59.

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, 17*(1-2), 23 - 42.

Sengupta, P., Kinnebrew, J., Basu, S., Biswas, G., and Clark, D. (2013). Integrating Computational Thinking with K12 Science Education Using Agent-Based Modeling: A Theoretical Framework. *Education & Information Technologies*, Vol. 18, 351 – 38.

Smith, D. C., Cypher, A., & Tesler, L. (2000). Programming by example: novice programming comes of age. *Communications of the ACM*, *43*(3), 75-81.

Wilensky, U. (1991). *Abstract meditations on the concrete and concrete implications for mathematics education*. Epistemology and Learning Group, MIT Media Laboratory.

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

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

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33-35.

CHAPTER III:

COMPUTATIONAL MODELING AND THE DEVELOPMENT OF DISCIPLINED INTERPRETATION IN FOURTH GRADE SCIENCE

Author note: An earlier version of this paper appeared in Farris, A. V. & Dickes, A. C., & Sengupta, P. (2016). Development of disciplined interpretation using computational modeling in the elementary science classroom. In *Proceedings of the 12th International Conference of the Learning Sciences*. Singapore.

Abstract

Studies of scientists building models show that the development of scientific models involves a great deal of subjectivity. However, science as experienced in school settings typically emphasizes an overly objective and rationalistic view. In this paper, we argue for focusing on the development of disciplined interpretation as an epistemic and representational practice that progressively deepens students' computational modeling in science by valuing, rather than deemphasizing, the subjective nature of the experience of modeling. We report results from a study in which fourth grade children engaged in computational modeling throughout the academic year. We present three salient themes that characterize the development of students' disciplined interpretations in terms of their development of computational modeling as a way of seeing and doing science.

Keywords: modeling; agent-based models; disciplined interpretation; epistemology; science education

Modeling, including the generation, evaluation, and test of models, is the key epistemic and representational practice in science (Giere, 1984; Lehrer, 2009). Through modeling, scientists transform the world into shareable representations. The development of correspondence between representations and the things they are intended to describe requires mediations between materials, phenomena, and communities (Latour, 1999). Modeling, however, is not merely the generation of symbolic representations of scientific phenomena. Studies of scientists and science *in action* have revealed that scientists' interpretative moves and embodied and personal experiences constitute a much more nuanced image of scientific practice, one in which the development and refinement of models parallels and is deeply intertwined with the scientists' development of disciplined sensibilities (e.g., Keller, 1983; Watson, 1968; Ochs, Gonzales & Jacoby, 1996).

However, within the science education literature very little attention has been given to the subjective and interpretive judgments that learners must make in the development of scientific models. The emphasis has been overtly on the reproduction of canonical (i.e., disciplinarily accepted) forms of representations, and less so on the interpretive moves that are often necessary for learners. In this "school version" of science, representations of the world are often misconstrued as exact copies of the world (Lehrer, 2009). This results in the reification of an overtly objectivist and rationalistic view of science that is typically presented in school settings, which as Lemke (2001) argued, occurs at the expense of students finding personal entrée into scientific domains or developing identities as emerging scientists (Lemke, 2001). A phenomenological view of design is deeply tied to subjectivities such as learning to see things from the perspectives of others and engaging in reflective conversations with the situation (Schön, 1995). Lehrer and Schauble (2005) and colleagues (e.g., Lehrer, Schauble, & Lucas, 2008), who have studied the long-term development of children's scientific modeling in elementary classrooms, argue that developing scientific expertise involves the development of certain dispositions and practices for modeling that are both epistemic and representational in nature. Some of the key dispositions they highlight are interpretive moves such as deciding what counts as a reasonable question to investigate and formulating defensible arguments. Another important finding from their work is that classroom ecologies—materiality and establishing disciplinarily grounded social norms of scientific participation—are essential for mobilizing new forms of seeing and inscribing in the classroom that can deepen children's disciplinary inquiry.

This interweaving of knowing and action (Schön, 1995) and the progressive journey toward disciplinarily grounded ways of seeing and knowing are a distant reach from common educational uses of computational modeling, where the emphasis is primarily on supporting the development of children's programming competence (Grover & Pea, 2013; Sengupta, Dickes & Farris, 2018). In the science classroom, computational modeling has traditionally followed the grossly linear approximation of teaching correct concepts through guided algorithmic and data-structural refinements (e.g., White & Frederiksen, 1998; Boolen & van Jooligen, 2013). Such images are grounded in technical rationality (Schön, 1995) and leave out the development of necessary subjectivities for modeling, and this is the issue we address in this paper. Building on Daston & Galison's (2007) notion of "trained judgment," we argue for focusing on the development of disciplined interpretation as an epistemic and representational practice that progressively deepens students' computational modeling expertise by valuing, rather than deemphasizing the subjective nature of the experience of modeling. We report results from a study in which fourth grade children engaged in computational modeling by iteratively creating, presenting and evaluating their mathematical measures and computational models of motion and ecology throughout the academic year. In this paper, we will focus on their models of motion and investigate how they develop progressively more mathematically and computationally refined representations of motion as a process of continuous change.

Research Question

We investigate the following research question: What forms of disciplined interpretations about what makes a scientific model "good" can develop in the classroom through long-term computational modeling?

Theoretical Background

A View of Scientific Modeling Beyond Instrumental Clichés

Dewey (1934/2005) contrasted "bare recognition" and "full perception" to describe learning. Bare recognition entails how the initial categories through which one understands the world remain unexamined when those categories are easily recognized. These initial categories act as instruments that guide and shape our cognition, and when learning is limited to these initial categories, education can have the appearance of new knowledge, despite, in reality, relying heavily on instrumental clichés (Higgins, 2008). A richer image of perception—what Dewey termed "full perception" —is an experience that is vastly different. It requires *transformation* of the categories in order to fully perceive an object or process (Dewey, 1934/2005; Higgins, 2008). In what follows, we present a richer image of scientific modeling beyond the image of instrumental clichés. To do so, we draw upon literature from science studies and the philosophy and history of science in order to identify and illustrate how subjective and interpretive moves play important roles in the *experience* of modeling. Our work arises from two concerns: the first is that despite emphasizing modeling as the key practice in science education, an overt emphasis on reproduction of disciplinary forms does not bring to light these relatively more complex and nuanced elements of human experience that are integral to scientific work. And second, when we come to see modeling as a rich and complex experience that is grounded in interpretive moves, it can profoundly deepen and enrich the images of computational science, particularly in K-12 classrooms. The implications, we hope to demonstrate in this paper, are both technological and pedagogical. That is, such imaginations can transform and deepen our own understandings of both what computing can look like in the science classroom and pedagogical moves and practices that may be essential to support such forms of experience.

There is always a gap between scientific representations and reality (Latour, 1999; Giere, 1988). Models in science are representations that *amplify* and *reduce* aspects of experience in order to explain a referent (Latour, 1999). Creating models, therefore, involves making interpretive decisions about which elements of the phenomenon to represent and what to leave out. Representations in science are said to "cascade" from one another, and the phenomenon itself circulates among these varied representational forms (Latour, 1999). Furthermore, because models are for social use, they often provoke new curiosities, thereby feeding into these transformations between model forms. In short, models beget more models.

These transformations from one representation to another often generate new knowledge. Consider Latour's (1999) classic example of scientists' exploratory representations of the boundary between a forest and a savannah in the Amazon Basin: is the forest encroaching or receding? In Latour's (1999) analysis, the boundary between the savannah and the forest is a reference that circulates between many forms of discipline-specific representations used by the botanist, soil scientist, and geographer on the project. Variations in the visual format of the representations of the boundary supported different types of inferential processes that provoked new knowledge, unearthing previously unknown patterns. Cubes of soil samples, for example, arranged according to depth and analyzed by color, suggested that the savanna was advancing upon the forest. However, the botanists' information, when considered alongside the soil patterns, created new questions.

The scientific practice visible in Latour's analysis illustrates the epistemic stance of much of science in the mid-to-late twentieth century, which Daston and Galison (2007) identify as trained judgment. This epistemic virtue stands in contrast to earlier commitments: In the nineteenth century, for example, scientists' tacit commitment to mechanical objectivity demanded "getting out of the way" in order present un-interpreted facts, often using machines to avoid human judgments and interventions. An example of this is found in the introduction of photographic technology in the nineteenth century. The machinic nature of photography reified the impression that scientists could and should step away from the object of inquiry and let the photograph produce bare and objective "facts."

In contrast, beginning in the early to mid-twentieth century, with the advent of the printing press widening the audience for scientific works increasing the need to make sense of scientific photographs the production of scientific images became necessarily more interpretive on the part of the scientist, with a clear goal of enhancing the communicativity of the images (Daston & Galison, 2007). It became more acceptable, even expected, that photographic images be altered to highlight particular details that would not be readily evident to the less-trained eye. Daston and Galison (2007) describe shift in the epistemic moment as an extension of trained judgment called "presentation." Presentation relies on computing to produce images and dynamic simulations that depict scientific explanations. Early examples include particle chambers in the 1940s, which required combining human sensory and pattern-finding capabilities with large datasets (Galison, 1997). Computing continues to serve as an expansion of human sensory processes (Gooding, 2003). In the case of modern astronomical images, for example, celestial features are interpreted from non-visual data and recolored in order to create images that can be interpreted by non-astronomers (Daston & Galison, 2007; Lynch & Edgerton, 1996).

However, some forms of these extensions of human senses can significantly alter the traditional ways of doing science. For example, Chandrashekhar and Nersessian have argued that scientific computing is also unavoidably interdisciplinary, creating opportunities for multiple perspectives to collide as scientists and engineers from multiple disciplines work together to develop computational models of a phenomenon (Chandrashekhar & Nersessian, 2015). Additionally, simulation often requires stepping into analogy, thereby blending (a) computational theory and epistemology with (b) the epistemological commitments and theories of the source domain (Chandrashekhar & Nersessian, 2015). This form of broadening of the representational and epistemic possibilities is a key characteristic of computing technologies in science.

What do these new or broadened possibilities mean for computational science in the K-12 classroom, particularly at the elementary level? At the most general level, this is the question we have set out to answer in this work. We demonstrate how by highlighting, rather than de-emphasizing the need for interpretive moves in the work of using, transforming and manipulating computational models, learners can begin to move beyond simplistic notions of resemblance as criteria for modeling to more nuanced characterizations of what should count as “good” or “acceptable” models of physical motion.

Agent-based Computation and Children's Science

The particular genre of computational programming that learners used in this study is agent-based modeling (ABM). In ABM, users define individual actions of computational agents in order to simulate processes of change. A particular affordance of ABM is that it can demonstrate how complex patterns emerge from simple, agent-level actions, making its use ubiquitous in many fields of scientific research in both the natural and social sciences (Axelrod, 1997; Jacobson & Wilensky, 2006).

Many researchers have long argued for *reflexivity* between developing scientific expertise and learning computer modeling and agent-based modeling and programming (in particular) (diSessa, 2001; Guzdial, 2004; Lehrer & Schauble, 2006; Jacobson & Wilensky, 2006; Papert, 1980). Learners' development of modeling practices using ABM and programming also supports their development of their computational thinking (Sengupta, Kinnebrew, Basu, Biswas and Clark, 2013). Wing (2006, 2008) has described computational thinking as a general, analytic approach to problem solving, designing systems, and understanding human behaviors. NRC and ACM reports also argue for the integration of computational thinking with K12 science curricula (ACM, 2003; NRC, 2010), and we explain this reflexivity in more detail later in this paper. Additionally, agent-based computation can serve as an effective pedagogical approach that can help children bootstrap their own pre-instructional ideas and representational competencies in order to develop scientific expertise through modeling (Papert, 1980; Sherin, diSessa & Hammer, 1993; Sengupta, et al., 2013). Programming the agent involves thinking like it, which enables the learner to engage in embodied reasoning (Papert, 1980, Wilensky & Reisman, 2006). In ABMs, simple, agent-level actions are repeated over time (in the case of generating continuous movement from discrete actions) and/or across multiple agents (e.g., in ecological phenomena). There is ample evidence in the literature that ABMs can support the development of representational competence in children (e.g., Sherin, diSessa & Hammer, 1993; Wilkerson, Wagh, & Wilensky, 2015). Children's computing using agent-based models allows for design and manipulation of discrete representations (e.g., steps) of continuous phenomena (e.g., continuous motion) that are "body-syntonic" (Papert, 1980). That is, children can create discrete units of measurement that are both intuitive and embodied - *mind-sized bytes* – which, when repeated computationally, simulates continuous processes (Wilkerson & Wilensky, 2015 Sengupta et al., 2015). Besides linear processes such as acceleration, ABMs have also been shown to be effective in helping students develop deep understandings of complex systems such as ecological interdependence, where change occurs in both short and long timescales as well as through simultaneous interactions between many agents (Dickes, Sengupta, Farris, & Basu, 2016; Danish, 2014).

As exemplified in Ochs, Gonzales, and Jacoby (1996) and Keller (1983), sense making that is dynamic often deeply involves projections of the self into the system of scientific inquiry. Specific to the context of learning kinematics, previous research shows that given appropriate teacher-led scaffolding, middle and high school students can effectively use agent-based learning environments that involve programming and modeling in order to develop sophisticated mathematical representations of key kinematic concepts and phenomena (Hammer, Sherin, & Kolpakowski, 1991; Sherin, 2000; Sherin, diSessa, & Hammer, 1993; Sengupta & Farris, 2012). In these studies, the notion of computing is limited to coding, and are not concerned with materiality beyond a "material" imagination of the computational agent. Furthermore, little attention has been paid to disciplinary dispositions that are central to modeling science using ABM in learning about kinematics, especially in classroom settings where the teacher is in charge of instructional design and decision making. As our study will show, when students' interpretive moves are not de-emphasized in classroom instruction (or in reports of such classroom studies), the notion of computing itself also greatly expands beyond coding on the computer.

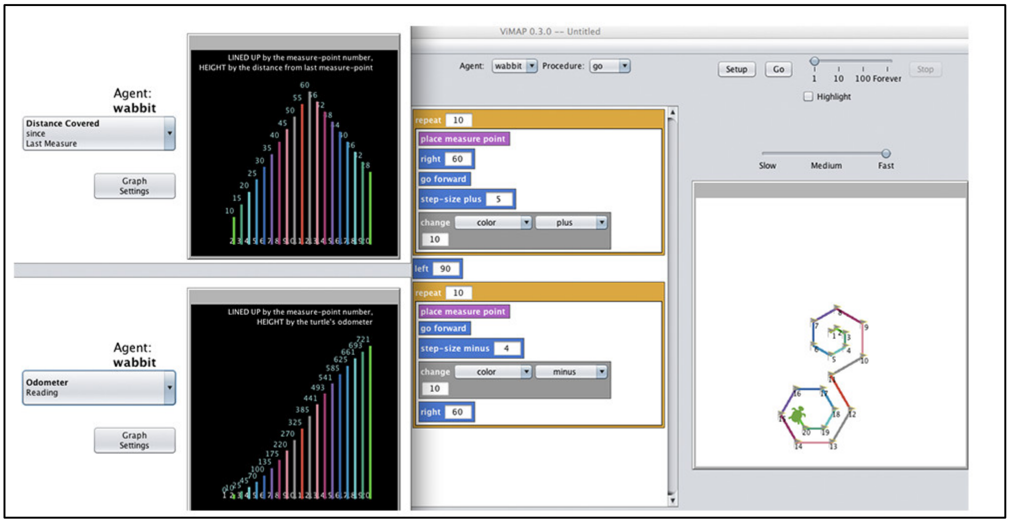
Method

Participants and Setting

The data were collected in a public school fourth grade classroom (most students were ages 9 and 10) in an urban southeastern city. The study is a design study (Cobb et al., 2003) in which we worked in partnership with the classroom teacher to integrate agent-based programming and modeling within the existing math and science curriculum. Students carried out investigations of natural phenomena in kinematics and in ecology in modeling cycles that included modeling in ViMAP. Twenty-one students and their teacher, referred to as Ms. Beck, participated in the classroom work. All students in the class were African-American. The class was comprised of 11 female and 10 male students. Ms. Beck and the research team co-planned the activities based on the students' progress and the Ms. Beck's plans across the curriculum. During class time, the teacher played the primary teaching role, often adjusting the plans to meet the emerging instructional opportunities as plans were enacted. Two graduate-student members of the research team collected data. The data include video of each class, interviews with students and student groups, student work, planning documents and discussions of the teacher and the researchers, detailed field notes from each session, and automated screen captures from the students' computers. Ninety-five percent of students who attend the school are eligible for free and reduced-price lunch. A sequence of the learning activities is shown in Table 1. In this paper, we only report the analysis of motion modeling activity from October 14 until March 31.

The ViMAP Modeling Environment

The modeling platform we used in this study is ViMAP (Sengupta et al., 2015). ViMAP is an agent-based visual programming language that uses NetLogo (Wilensky, 1999) as its simulation engine. In ViMAP, users construct programs using a drag-and-drop interface to control the behaviors of one or more computational agents. ViMAP programming primitives include domain-specific and domain-general commands as well as a “grapher” with multiple graphing windows, which allows users to design mathematical measures and compare across measures of different agent- and class-level variables. Figure 1 shows the programming interface and the graphing interface.



```
repeat 10
    place measure point
    right 60
    go forward
    step-size plus 5
    change color plus
    10
left 90
repeat 10
    place measure point
    go forward
    step-size minus 4
    change color minus
    10
    right 60
```

Figure 1. Screenshot of the ViMAP Modeling Environment (www.vimapk12.net).

Table 1

Sequence of Activities

Unit	Dates	Summary
Observations, pre-assessment, and interviews	Aug. 11 – Sept. 8	Researchers conduct observations, preliminary interviews with all students in the class
Survival Kits Geometry Unit	Sept. 9 – Oct. 2	Intro to ViMAP programming and modeling; Turtle geometry, centered around learning goals in perimeter, area, and angles of polygons; model sharing and revision
"Constant Speed" Robots	Oct. 14 – Nov. 20	Students develop understanding of speed as a rate of the distance traveled in a unit of time, including cycles of model sharing and revision; students used both ViMAP and physical modeling
Constant Acceleration and Gravity	Nov. 25 – Feb. 3	Students find ways to measure and model continuous changes in speed, using acceleration down a ramp and free fall as contexts; students used ViMAP, video analysis and physical modeling
Friction	Feb. 5 – Mar. 31	Students model processes of "slowing down" for Matchbox cars on different surfaces; students used both ViMAP and physical modeling
Interviews	Apr. 7 – Apr. 28	Mid-year interviews with all students
Modeling Ant Colonies	May 6 – May 13	Students model ant foraging, reproduction and predation in ant colonies in an embodied modeling activity, followed by programming in ViMAP-Ants. Students share and refine their models with 8th grade mentors.
Post-Assessment	May 14 – May 19	End-of-year assessment and focus group

Researcher-Teacher Partnership

The teacher, Ms. Beck, reframed programming as a medium for designing mathematical measures (i.e., units of measurement and graphs) of motion. Along with the researchers, the teacher co-designed and implemented learning activities that supported the interpretation and construction of mathematical measures using ViMAP as a way to explain a real-life phenomenon involving motion (e.g., walking and running). In these activities, she maintained an emphasis on connecting modeling in ViMAP to relevant out-of-computer modeling experiences, such as embodied and physical modeling activities. Furthermore, she created a culture for sharing and critiquing peer models, that is, the students' ViMAP programs and graphs. In this process of creation, sharing, critique, and revision, students began developing criteria what makes a representation "good": the emphasis on public model sharing acted as a selective pressure for model improvement (see Enyedy, 2005 for an example). Initially, these criteria originated in teacher-led class discussions as socially defined (voted by popular choice), but over time, became progressively more grounded in students' understanding of aspects of motion or mathematical explanations of their ViMAP simulations. This led students to use progressively more sophisticated computational abstractions, such as loops and conditionals, and to use the ViMAP commands to make less literal, but more illustrative depictions of motion relationships. A particular tension evident in the teacher's instructional approach was the tension between supporting students' exploration of the representational palette and the curricular need for production of canonically correct representations. For example, Ms. Beck explained to us at the beginning of the school year that for her, graphs were the most important "output" of ViMAP models, because it would help her connect students' work on the computer with representational forms that are mandated by the curriculum as well as forms that they would be tested on in standardized tests. Her goal, as was evident during the first several weeks of class, was to help students develop computational models that produced graphs of change over time, with labels that depicted measured data from the phenomenon. In weekly meetings with Ms. Beck, the researchers explained that they wanted the class to have opportunities to consider the communicative value of other features of the computational models—for example, the dynamic nature of the model as it runs in real time, or considering other complementary forms of representing change over time besides graphs. Initially, it seemed to us that this tension would persist throughout the year. However, superimposing our desires as researchers was something we explicitly wanted to avoid, especially because historically, educational computing research in classrooms has typically silenced teacher voices, and have been primarily designed, led and taught by researchers. In researcher-led studies, these tensions that often arise from curricular mandates are non-existent. Our goal instead was to make these tensions explicit in our conversations with teachers when feasible, but work with the teacher's indentions as primarily guiding instructional design.

As our partnership progressed, we noticed that the epistemic and representational diversity that we had been arguing for began to emerge as a result of two instructional commitments that Ms. Beck made: *materiality* and *model matching.* Ms. Beck's approach to introducing new forms of computational representations involved grounding computational representations in physical, tangible experiences with motion (materiality). This resulted in students creating multiple representations of the same phenomenon (e.g., embodied simulations, paper graphs, as well as ViMAP simulations), which in turn created the need for "model matching" (Ms. Beck termed this "making sure your ViMAP models are "accurate"). The cases we highlight in our analysis demonstrate how Ms. Beck's commitment to these issues eventually supported representational and epistemic diversity, by emphasizing, rather than de-emphasizing students' interpretive decision making, and reasoning about tradeoffs about what the models could and should show.

Analytic Approach

The research approach was both microgenetic and sociogenetic, because our goal was to understand changes in student thinking and how these ideas are shared and taken up in the larger class community. Our data collection and planning for instruction included a constant review of student work, daily conversations with the classroom teacher, and constant comparison of field notes with ongoing classroom work (Strauss & Corbin, 1990). As is typical of design research, each day's learning activities were designed in response the events and discussions leading up to them and learners' demonstrated understandings and questions. We identified categories in open coding (Strauss & Corbin, 1990) and from these categories, we eventually developed themes of epistemic and representational actions.

We then conducted a thematic analysis (Braun & Clarke, 2006) in order to identify key representational and epistemic advances that learners developed during the phase of modeling motion. A theme, in Braun & Clarke's (2006) use, "captures something important about the data in relation to the research question, and represents some level of *patterned* response or meaning within the data set." (p. 82; emphasis in the original). We identified key forms of disciplined interpretations that learners developed during the phase of modeling motion. In our study, at the highest level, each theme represents an interpretive judgment. Each theme, in turn, consists of sub-themes, which are sets of relevant *representational moves*, i.e., actions undertaken by the learners that involve the creation, and/or editing of computational programs and other related representations, and *epistemic moves*, i.e., arguments about the validity or significance of certain representations. It is important to note that these moves, in many cases, were deeply intertwined, which was evident in their co-occurrence. Nonetheless, both these dimensions—the epistemic and representational—were key elements of the students' experience of modeling, as well as of the teacher's instructional moves. Over time, these representational and epistemic moves constitute, or lead to the development of an interpretive judgment (e.g., what counts as a "typical" measurement; what counts as a "good video").

We also note that these interpretive judgments developed through progressive refinement of models and moving back and forth across tangible, diagrammatic, and computational models of motion. Therefore, besides the learners' subjectivities, the judgments themselves are inextricably tied to the physical and computational media involved in modeling, in addition to mathematical and physical ideas, and mathematical and computational abstractions.

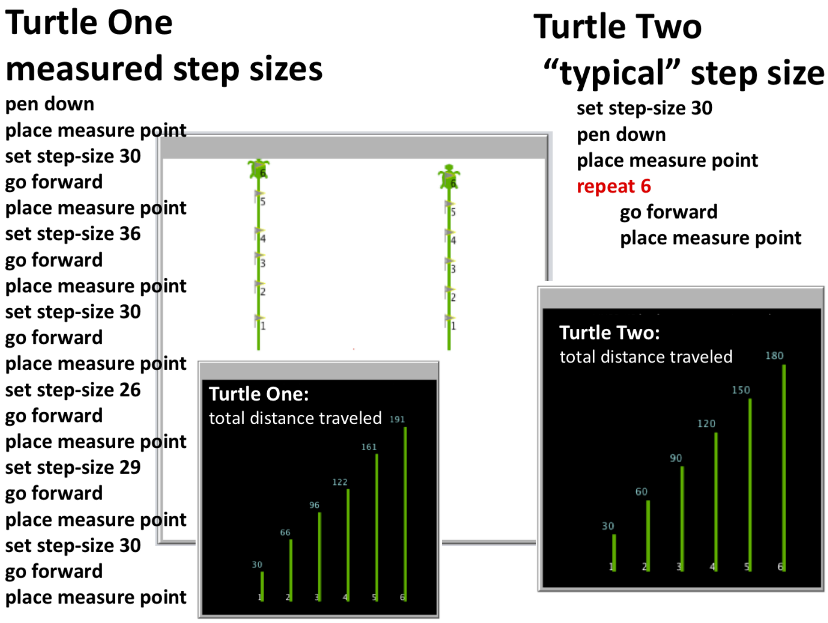
Findings

Interpreting Data as Designed Measures

Analysis. In the first modeling cycle, the teacher wanted to design a context for students to define constant speed motion in terms of distances traveled per unit of time. Ms. B insisted on introducing physical objects, both computational and non-computational, as part of the modeling activity. To use her own words, Ms. Beck's goal was to "make things concrete"— i.e., to transform the modeling activity from a virtual and conceptual one into a lived-in experience for her students. Based on her suggestion, we programmed Lego Mindstorms NXT robots to move at a constant speed. The students' goal was therefore to create measurements of the robots' motion and subsequently, to model that motion.

Figure 2. Measuring constant speed using adhesive paper flags as “measure flags.”

Students used stopwatches, adhesive Post-it flags, and fabric measuring tapes in order to conduct the measurement activity. We asked students to measure the distances traveled in regular intervals of time. To introduce the activity, the graduate students and Ms. B, along with a few student volunteers, demonstrated a possible way to measure the distances: students coordinated the placement of position-marking flags with a stopwatch by placing a flag on the floor to mark the position of the robot at each increment of time, based on the verbal commands of the timekeeper. See Figure 1 for a visual example. Students groups adapted and modified this method in their individual groups. All groups kept the same general approach to measurement: marking the beginning position and continuously marking the positon at the end of each time increment. However, the learners negotiated details such as the length of the increment, the “part” of the robot for which the position was to be marked on the floor, and the distribution of roles. In Figure 2, a student is measuring the distance between flags to find the distance traveled in each three-second interval, based on the flags placed by his classmate. All students’ measurements for the distances traveled in each interval were non-uniform due to challenges with recording the motion of the robot, measuring distance of a curved path, and coordinating a measurement activity across group members.



```
pen down
place measure point
set step-size 30
go forward
place measure point
set step-size 36
go forward
place measure point
set step-size 30
go forward
place measure point
set step-size 26
go forward
place measure point
set step-size 29
go forward
place measure point
set step-size 30
go forward
place measure point
```

```
set step-size 30
pen down
place measure point
repeat 6
        go forward
        place measure point
```

Figure 3. Student model of measured and "typical" step-sizes.

Students later used these non-uniform measurements to create computational models of the motion in ViMAP. In this initial work with robot speed data, none of the students problematized their data by considering, for example, the limitations of their rudimentary measuring devices. This may be a result of the two-day delay between taking the measurements and reflecting on them. Students took the measurements on October 14, but we were not able to spend time talking about the measurement data as a group until our next meeting on October 16. While some of them argued some about issues of fidelity *during* the measurement activity on October 14, for example: "Your flag is too late;" "You're holding the measuring tape wrong", none, to our knowledge, critiqued potential problems in her or his data on October 16. All students' measurement data showed wide variation in distances traveled in uniform increments of time. However, all students made models from the numerical data in their tables without questioning the measurement values further. Upon noticing this, the teacher and researchers designed an activity in which the entire class watched a video from October 14 of one group carrying out their measurement and data collection. As a class, they critiqued their work as shown in the video. The teacher led a discussion during which she replayed the video several times in order for students to notice and reflect upon successes and breakdowns in measurement. The ensuing discussion led to the first student talk about error: "It's not that the robot was moving differently, it's that *we* were making mistakes!" This initial acknowledgement of error was later extended and challenged to include error that humans had no control over, such as the way some robots' wheels slipped, creating a curved path that was difficult to measure with the materials we were using.

Attending to the physical context of the measurement data helped students understand their data as designed measures that were approximations of the world and carry measurement challenges. This was particularly evident in their acknowledgements of the various sources of error as they replayed and re-analyzed the video. For example, some groups noticed that longer measurements of displacement were often coupled with shorter measurements, indicating that the timing of placement of the Post-it flag shared by those measurements was likely early or late. Students problematized the tendency of some of the robots to take a curved path. Some groups also discovered errors due to misreading the measuring tape, and in some cases, due to the sticky flags being unintentionally moved by getting stuck to students' shoes.

We then asked students to review their measurements in order to determine what they believed was a "typical" distance measurement for their robot to travel in three seconds. The teacher welcomed this as an opportunity to connect the measurement activity with learning about measures of central tendency in their math curriculum. The students would later redesign their existing computational models using a second computational agent in the same simulation to represent the motion according to their “typical” values. We did not prescribe a way for students to determine what was typical; however, the teacher did ask all students to find the range of their data and consider reasons for variation. Table 2 shows data sets that the students used for the central tendency activity, which were made in just one run of the activity. The robots used in the activity varied in their designs, and the data reflect variation in robot speed (that is, faster and slower robots). However, students recognized that we should expect internal consistency within each set of measurements for each group. Three of four groups annotated their data set with specific information about measurements. These included information remembered or seen in the video record about specific measurements, with a specific emphasis on reasons for error:

> Group 2: “At this interval [the measurement of 26 inches] the reason our measurement got shorter [is] because this is when [the robot] started curving.”
>
> Group 3: “The time keeper forgot the 3s mark” (preceding the measurement of 13 inches).
>
> Group 4: “The post-it note stuck to someone’s shoe” (associated with the measurement of 12 inches).

Groups 2, 3, and 4 each chose the mode of their datasets. For the data that they were working with, the mode is a defensible choice: it appears multiple times and is near the center of the range. Group 1, however, did not reach consensus: some members of the group wanted to use the mode of the data set, as it was the only number that appeared twice (17). One student advocated for using 20, because 17 was the very bottom of their range (17-24) and 20 was near the mean, but also was a “tens number,” making it easier to perform calculations without need for a calculator.

Table 2. Measurement Data and "Typical" Values Chosen to Represent Central Tendency

	Group 1	Group 2	Group 3	Group 4
	24 inches	30 inches	16 inches	27 inches
	22	36	18	31
	20	30	13	32
	18	26	18	16
	17	39	18	12
	17	30	22	27
"Typical" value(s) selected:	17, 20	30	18	27

After identifying what would count as a typical value of forward movement for each increment, students redesigned their computational models. Figure 3 shows one student's model: her "measure-points" in the first iteration were the following distances apart, measured in inches: 30, 36, 30, 26, 39, 30. Her second iteration data shows six uniform measurements of 30 inches each, as 30 was the value that she and her group determined was the most "typical" measurement. For each agent-based model, students programmed ViMAP to generate two graphs: one showing the value of each speed (not shown), and another producing the total distance traveled by the robot. In the example shown in Figure 3, the graphs of the total distance traveled by the agent show a total distance of 191 inches for the measured data, and 180 inches for the adjusted, or "typical" data. Students also recognized that computationally, the typical model could be expressed as a loop, which the students and teacher appreciated as a more succinct program. A second affordance of the program for Iteration 2 is that the number of repeats could be changed to simulate the robot traveling for a longer period of time at the same speed.

At this point in the year, all students' models looked similar to the one shown in Figure 3. All students' use of the variables built in to ViMAP were minimal: step-size was the prescribed means of communicating speed, or the distance traveled in each successive increment of time. The enactment and the inscription left behind by the agent, without the graphs, is not used, other than to show literal similarity that the robots moved in a (mostly) straight line.

Discussion. In sum, making mathematical meaning of motion as processes of time-based change required the generation of and coordination among different representational moves, involving multiple forms of digital and paper-based, discrete-mathematical representations of the phenomena under study. Annotating video and photographic images in order to communicate and argue for the number of loops needed in their programs became a viable but emergent method for connecting among the representations, and can also be regarded as epistemic moves that grounded these representations within the disciplinary concepts. Students' agency and involvement in creating connections across representations for the purpose of making meaning represents a key practice in model-based reasoning. Epistemologically, the connections among representations were a shared unknown, and it was up to the members of the class to come up with and refine generative ways to see, quantify, and model salient aspects of motion.

As we described above, the inconsistencies in distances between successive flags was problematized by reflecting on the motion using video records of students conducting the investigation. While students agreed that the robots were moving at approximately constant speed, the videos were what eventually pushed them to begin to think about measurement error. In this activity, making mathematical meaning of motion as processes of time-based change required the generation of and coordination among different representational moves, involving multiple forms of digital and paper-based discrete-mathematical representations of the phenomena under study. These representations were grounded in disciplinary concepts of position, time, speed, and uniformity (e.g., chasing the robot and marking time-based positions, static sequences of flags and drawings of the flags, lists of length measurements, and video records of motion and measurement). Epistemologically speaking, students' agency and involvement in creating connections across representations for the purpose of making meaning represents a key practice in model-based reasoning: Weighing the fidelity of measurements and identifying what could "count" as a typical value were key epistemic moves in which students grounded their arguments. Together, and in conversation with one another, these moves allowed students to interpret data in relation to the specific context and challenges underlying the data.

Creating "Good Enough" Representational Re-descriptions

Analysis. The measurement and modeling of robots' motion prompted refinement of their students' descriptions of constant speed. We briefly worked with average speed of accelerated motion events, and finally, began to work on developing descriptions of acceleration, the primary learning context of this theme. Students worked with clear acrylic tracks, marbles, stopwatches, and adhesive paper flags, and Lego bricks to build supports to hold the tracks at various inclines. Students' initial descriptions neglected processes of continuous change: the marble was "slow" at the top of the ramp, and "fast" at the bottom. When asked to measure how speed was changing, students tried to reapply their method with the robots: they attempted to place flags at equal intervals of time, but they soon decided this was too difficult: the motion was *too fast* for the method used when measuring the speed of the (slower) robots. As a potential solution to this problem, the instructors introduced digital video as a new method for collecting and analyzing motion data. We made this design decision partly because it was near to students' existing literacies for making and working with digital video, and the technology was a good match for the kinds of questions students were pursuing: high frame-rate videos afford the possibility of slowing down recordings of motion that are otherwise too fast to measure.

Children's ideas of what counted as a "good" video for measuring acceleration changed dramatically between the first and second iterations of their video recording and subsequent analysis. An example of student work from the second iteration is shown in Figure 4. In the first set of videos, recorded by the students on December 16, many of their videos followed the marble in an action perspective, but this made measurement impossible because there was no frame of reference from which to measure the distances traveled. Students' ideas about coordinating the timing of the start of the video shifted multiple times: at the end of the day on Dec 16, many students advocated for starting the camera at exactly the moment the marble is released (release at frame 1), however, this later changed to an emphasis on making sure the camera is rolling before the marble is released, and identifying the "frame of release" post hoc. This strategy meant that there was a full record that can be interrogated and verified to identify the point of release—missing the exact moment of release was no longer an issue.

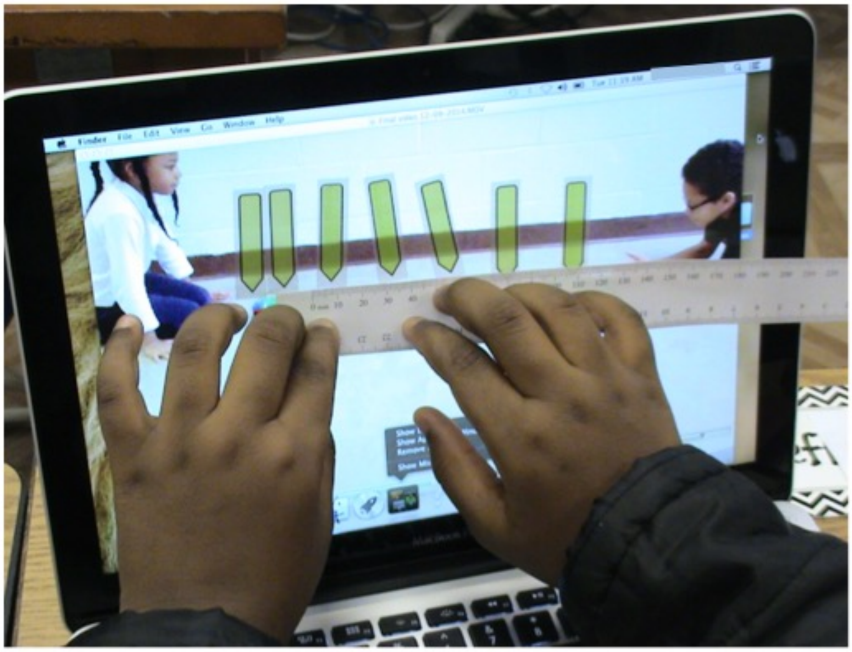


Figure 4. Adhesive flags mark the position of the marble at a 15-frame interval in the students' video.

As a class, we decided to use this set of videos to mark the position of the marbles at 15-frame increments. From a tool-use perspective, jumping the video forward 15 frames and putting a post-it flag was manageable for all students, though many described it as difficult or tedious. Eight of 20 students present during this activity were not able to see or document any pattern of increase in step size (speed during an increment) due to issues with the videos. These problems included the following issues: inconsistent or small field-of-view, shaky videos, shaky ramps, obstruction of the position of marble. However, the visual cue of increasing distance between the flags was salient for 12 of the students, even though the mathematical pattern of *uniformity* of the increase (with each 15 frames) was unclear.

After attempting to measure the acceleration of the marble using the first round of videos, and after classroom-wide discussions, the class developed norms for what counts as a "good" video for measuring motion in a frame-by-frame analysis: the camera has to stay still and the field of view must show the whole motion. Central to this was the realization that the viewer should be able to see the marble and identify the exact frame at which it was released. Student groups developed, shared, and critiqued several videos, and progressively refined their measures, and over time, developed measures using physical and material means that used discrete mathematical representations similar to their ViMAP turtle's "step-size". The dominant form of measure involved placing a flag on their computer screens to mark the position of the marble at regular intervals of video frames. When the teacher asked students what a frame is, a student offered that a frame is "a picture in time." We do not have evidence that any student did not accept the temporal nature of the frames.

December 16 was the first day that students worked with videos to analyze the motion of the marble frame-by-frame. Of the 20 students present that day, five students measured that the ball was speeding up at the beginning, then "maxing out" near the middle of the ramp. Seven students were not able to measure a pattern of acceleration at all. The remaining eight students saw the ball speeding up, but there was no pattern suggesting constant acceleration. Students also planned changes to how to take the videos: While some were non-specific ("I think we should do this whole disaster over again"); other students wrote specific changes they planned to make about the position of the camera, avoiding obtrusions that block the position of the marble, holding the camera still, and issues related to timing the release of the marble and the start of the video.

In the second iteration only a couple of days later, all students made videos that were good enough for creating summary images (using Post-it flags) that shows a continuous process of speeding up. Students made new videos on December 17, one group at a time, according to their new guidelines for creating the videos. Students previewed the videos in order to decide when a video was good enough based on the criteria they had identified. All students had something that worked reasonably well for making visible the pattern of speeding up, with one major fault: the ramp used by all the groups sagged in the middle, creating an inconsistent pattern of acceleration. We did not re-make the videos at this time, but moved on to a different phenomenon: free fall, as a way of avoiding the problems with what came to be known as “the sag”.

Later in the year, students made a third iteration of making videos, this time to compare the effects of forces of friction on two different surfaces. They used matchbox cars pushed on various surfaces (e.g., sandpaper, sidewalk, gym floor, carpet, etc.). The students designed these investigations on their own, and of the 18 students who were present in this day, all 18 created summary images like the in which a pattern of gradual deceleration is readily visible.

Discussion. Unlike constant speed, constant acceleration phenomena were too unwieldy to describe based on the real-time motion. We introduced digital video as an intermediate representation that allowed us to start and stop motion. However, the onus to create usable videos for measurement was on the students. Initial videos neglected to document key aspects of motion: position and distance traveled. Thus, initial videos were worthless for descriptive measurements of speed. Revised videos included attention to position of the marble and made future measurement possible. These videos were then used for a measurement activity in which each student produced a mathematical re-description of the motion. This theme highlights the importance of salient disciplinary ideas in producing usable re-descriptions. The students had to develop aspects of seeing that are, in our view, analogous to professional vision (Goodwin, 1994), in order to recognize what was most important to document regarding changing speed.

The students' epistemic moves were rooted in their attempts, failures, and reattempts to create videos that were useable for measurement. Their realization that position and time were the most important things that needed to be extracted from the video records informed the re-creation of the videos. Likewise, annotating video and photographic images in order to communicate and argue for the length and the number of the "steps" needed in their programs became a viable but emergent method for connecting among the representations, and can also be regarded as epistemic moves that grounded these representations within the disciplinary concepts. Students' agency and involvement in creating connections across representations for the purpose of making meaning represents a key practice in model-based reasoning.

The students' epistemic moves, evident in the form of their explanations and concerns for what makes a video of motion useful for measuring speed represents the students' understanding of important aspects of motion from a disciplinary perspective. These measurements consisted of a set of eminently representational moves were made possible through an innovative use of video Yet, they are deeply interwoven with epistemic moves about measurement and ways of knowing about kinematic phenomena. They were used to make computational models that further communicate the mathematical pattern of change, as discussed the fourth theme.

In representing the world, we amplify certain aspects of it. In the acceleration modeling work, the linked representations required not only setting up a specific case of motion that could be used for investigation, but creating video records of that motion, and carrying out a frame-by-frame analysis in order to document and visualize the distance traveled in each congruent period of time. In the contrast to the initially flat representations, students captured a visually communicative summary of the changing speed of a marble going down a ramp, drawing on concepts they had been working on for about six months, such as speed as a ratio of distance traveled in a unit of time. The development of students' reasoning about representations was deeply tied to the media that are used to investigate phenomena. The learners in this study sought to describe motion events at levels that could not be observed or measured with the naked eye. Developing increasingly detailed accounts of changes in speed of objects required the development of methodologies and languages for making speed discrete and visible. In order to do this the teacher, researchers, and student linked the motion phenomena to a series of representations, and the idea of "the measure point" from the robots in Theme 1 continued as a circulating reference (Latour, 1999) with enactments in both the lived world and in manipulable representations—computational and paper-based—of motion that they had witnessed in the real world. The students learned that representations do work for us—they cascade in the sense that one representation may only be an intermediate form towards a new representation.

The summary images that they made are *haptic images*, meaning that they convey locations of an object across the passage of time. The change in speed is immediately (although qualitatively) visible to all members of the class. These images were later transformed into agent-based models, but those are not as relevant to the re-description of them—the primary extension of what was visible in the visual summaries was that this models had features that Chandrasekharan and Nersessian (2015) have pointed out are affordances of computational models in general: they can be run and re-run, and have a "stop-and-poke" nature that cannot be duplicated in the flat representation.

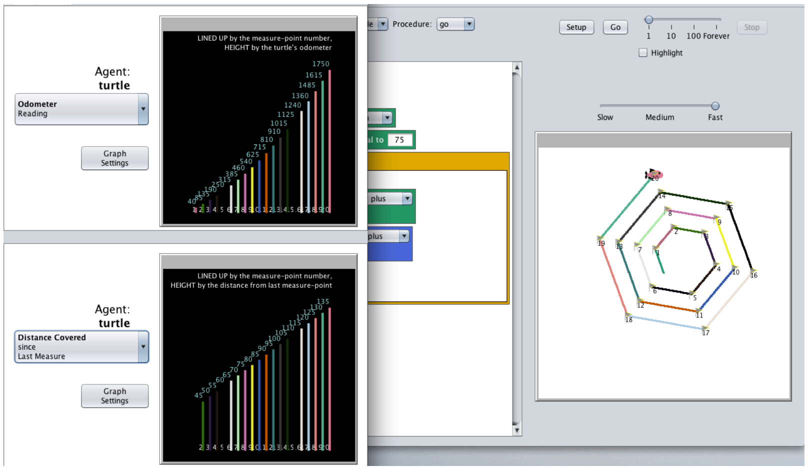
In sum, the movement across media was not an extra or accidental part of this design-based research study. It was essential to developing the kinds of knowledge that are students became proficient in, and the movement between representation are what allowed the progressive deepening to occur. The representational infrastructures continually wrapped back on themselves---representing the passage of time is notoriously difficult, and our very early, life-scale traces of position were an entry point that we continuously reflected back on. It lived in the collective memory of the class. Marking position and the distance traveled in an increment of time became the key, but we believe this was grounded in their computational enactments of placing measure flags.

Expanding Views of "Accuracy" to Create Visually Communicative Models

Analysis. The computational models created in the first two modeling cycles (October – December) were used to create graphs of change, however, the classroom teacher explicitly avoided use of any variables other than step-size to indicate speed, and no students publically proposed alternative ways of representing speed in their computational models at this point. Also in these first two modeling cycles, students' conceptual understanding of the unit of "speed" was essential in the state-mandated science curriculum. The teacher emphasized distance traveled by the ViMAP turtle in one step of the code as the representation for speed. Throughout the year, students normatively evaluated the "goodness" of their ViMAP models as representations of motion based on the match between the speed vs. time graph and the speed data that the model was designed to represent. Given that the graphs made the pattern of change explicit in these models, the students came to see graphs as the primary communicative devices, and the turtle enactment (i.e., the geometric shape generated by the turtle commands) was seen as merely the means to generate the graph. This was evident in multiple student-led presentations of their models and in their written work. Over time, especially in December and beyond, the researchers and teacher began to encourage students to further explore the ViMAP commands library and deepen their use of the programming language. The goal here was to prompt students to re-envision and re-design their models using turtle variables that they had not used before, thus making their turtle graphics less literal and potentially more visually and mathematically communicative. We explained color variables and provided a reference chart. Leading up to that point, the relationships between number assignment in the code and the computational color palette were a conceptual black box to students. That is, they knew that different numbers produced different colors, but the relationships that would allow them to make sense of the mathematics of the color palette had not been explained by the teacher.

As a result of the gradual instructional push that privileged less-literal models, all students began to take a more design-oriented approach to programming in ViMAP during the third modeling cycle. They began to focus on producing models that communicated the features of the target phenomenon that were most important from a physics perspective and also leveraged the semantic possibilities of the programming language. For example, students had been using turtle step size as the only variable to represent speed, however, in their future models, students used pen-width, intensity of color, and geometric features to represent speed. In classroom talk, the teacher and researchers highlighted the diversity of ways of representing motion, even when they were less sophisticated, and oriented class discussions around what key ideas were visible in the dynamic enactment of models. In terms of representational moves, all the students in the class expanded their use of variables by using new commands to mathematically represent the gradual change in speed using one or more of the following variables: rotation, intensity of color, pen-width, or relative size of agents. This, in turn, was motivated by and inextricably related to the epistemic move of making relevant features of the motion more salient to the class during presentations of their ViMAP models. We illustrate this change with the work of three students:

Non-literality of spirals. One form of model that we seeded through instruction is a regular, spiral-shaped polygon as a way to show changes in speed (e.g., Sengupta & Farris, 2012). The rate of change in the length of each line segment is a mathematical expression of acceleration. However, the classroom teacher viewed the spiral form as redundant: If the graphs were being produced to show the distance traveled in each segment, then angles and color changes between segments were unnecessary complications. “Why,” she asked one day, “would you model the forward and straight movement of a car with turns?” We think this is a valuable question, however one that stands in conflict with development of non-literality. Students who later re-discovered spirals often justified their use with more efficient use of the modeling space—a spiral allowed for compact representation of longer total distances, so the turtle doesn’t “wrap” off one side of the microworld and inexplicably appear on the opposite side. Zareen’s transition from straight, green lines inscribed by the turtle to multi-colored spiraling shapes stems from the use of the limited space in the enactment microworld and also the visual appeal of the spiral-like model. Figure 5 shows one of her later models of constant acceleration. Zareen still valued the graphs because they automatically labeled the bars with numerical data, however, she also chose to make geometric figures that represented acceleration as a rate of change.



Procedure: go

Agent: turtle
Odometer Reading

Agent: turtle
Distance Covered since Last Measure

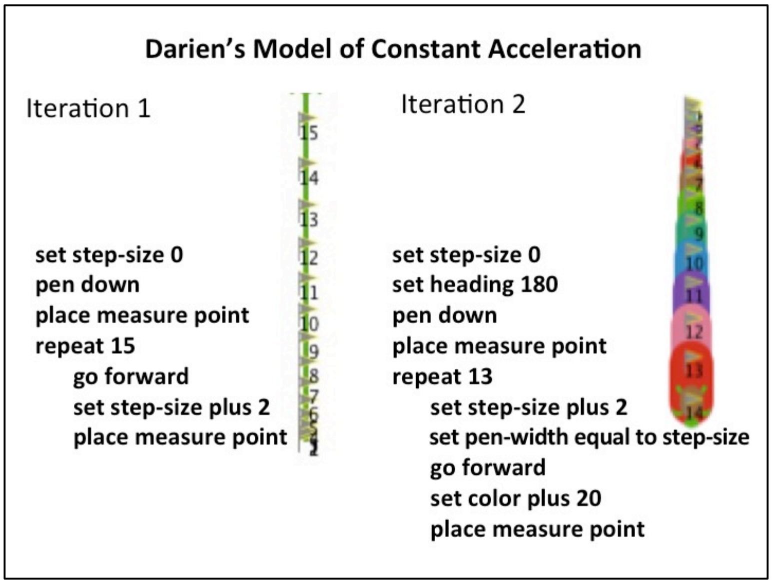
Figure 5. A model of acceleration that uses a spiraling geometric shape, as made by Zareen.

Similar forms of spiral-like of model was made by all students near the beginning of the year, and sixteen of the nineteen students present on this day in late January, while many of these students used it in passing and later changed their model to something else. However, the next day that we worked on these models (February 3), only one of 19 students made a spiral-like model of acceleration.

Graphs vs. dynamic change. Models that captured change by using the size of the agent as the variable for speed spread like wildfire, particularly when students were pressed to find new ways to show acceleration. The result is a screen-filling phenomenon, in which the size of the agent becomes so big that it more than fills the enactment microworld. The image left after the model runs (the end state of the model) is mostly useless, although these models were visually compelling while they were running. Students manipulated the grapher settings so that they were graphing agent size and therefore, they did produce graphs of speed vs. time and appeased their earlier criterion of good models produce “accurate” graphs.

This case illuminates an important distinction between static representations (like graphs) and the dynamic nature of many computational simulations. The students' programs ran each command in turn, so change in the model is dynamic—what we called "watching the model run." Running and re-running the models was important for code visualization and debugging. However, the graphs are static images that provide a historical record of change. Students who made their agent-size change created models in which increases in the agent size (representing speed) were only evident as the model ran, but models these models did not retain that communicativity in the end state—the static image only shows the largest agent size and hides the process of change.

Co-variation. Darien's first model of constant acceleration is shown in Figure 6 (left). The figure shows the inscription made by the agent as it executes the associated commands. The model increases the distance traveled by the agent by two step-size units with each step. However, the Iteration 1enactment is limited in communicating the regularity of the increase—someone interpreting the inscription would need to look at Darien's code or at the graphs to understand the regularity of the change.



Darien’s Model of Constant Acceleration

Iteration 1

```
set step-size 0
pen down
place measure point
repeat 15
    go forward
    set step-size plus 2
    place measure point
```

Iteration 2

```
set step-size 0
set heading 180
pen down
place measure point
repeat 13
    set step-size plus 2
    set pen-width equal to step-size
    go forward
    set color plus 20
    place measure point
```

Figure 6. Darien’s first (left) and final (right) models of acceleration.

In his final model (Figure 6, right), Darien added color changes to visually differentiate the individual steps of the agent, and co-varied the pen-width with the step size using the command *set <pen width> equal to <step-size>*. Darien presented his model to the class, describing that the increasing pen-width of the ViMAP turtle is intended to communicate that the ball is getting faster as it falls. When he was sharing his work with the class, students asked questions about the representational significance of different aspects of his model, such as “Why does it look like a baseball bat?”, “Can we see the data”, and pointed out redundancies in his code (an initial *set <pen-width>* command, that was being overridden by the co-variation command). When Darien described his model to the class, he explained that he had pen width equal to step size so that the pen width would also get bigger with every step of the agent:

Akia: Why do you have *set pen width* equal to *step size*?

> Darien: To actually help the pen width [be] equal to the step size, so that way, the pen, the size will actually get bigger EV-RY STEP. [holds a hands slightly apart to show a space between, gesture beats and enlarges at syllables of ev-’ry step]

Darien wanted to show that pen-width was increasing with every step, and illustrates this with his hand movements as he speaks, beginning with his hands slightly apart, and enlarging the space on beats with the syllables *ev / ry / step*. One could argue that Darien’s initial model was more canonical, because it uses the commonly used representations of dot-traces and graphs. However, in his revised model, Darien’s goal was to make the process of a steady increase in speed explicit *without* the use of graphs. This in turn led him to using an interpretive move that involved using a computationally more sophisticated data representation—co-variation—in order to link the visual appearance of his model (*pen-width*) to a variable that was significant in terms of representing the underlying physics (*step-size,* or speed). Ms. B was interested in this invention and highlighted it, and later invited all students to iterate upon Darien’s code.

Discussion. All students in the class advanced their models with communicative uses of the programming language, either by varying the color inscription with each step-size (so that different colors represent different increments of time), by adding turn angles, dually allowing patterns of uniformity to become more visually salient and using the space of the enactment world more efficiently, by assigning meaning to the size of the agent, or by changing the pen width. Co-variation emerged in one pair of students’ spontaneous work, but was shared with the class when the classroom teacher recognized the importance of their work.

The variation in student models illustrates diversity in values related to the end state of the models. That is, we notice that students’ work differs in terms of final state representations that show change across the passage of time became more valuable to some, but not all, students than end state models in which the enactment world showed no record of change across time. Graphs, which had initially been very privileged by the teacher, remained important, but graph-like records of change made by the agent were also valued. We postulate that the *experience* of observing the simulation unfold serves a communicative function by encapsulating the passage of time. When students shared models with the class, they almost always “played” the models. So, rather than trying to communicate events in a single snapshot (e.g., a graph), many students found it useful to take advantage of the dynamical nature of the simulations as a vehicle to communicate how the relevant events unfolded, despite using analogical representations (e.g., shape drawings) to represent motion.

Conclusion

Our attention to the disciplined and interpretive decisions that learners make during scientific modeling processes stands in contrast to the objectivist image of science that reigns in classrooms. We have sought to describe a more dynamical relationship between knower and knowledge in school science. Within this vision, children’s initial intuitions of their data are recognized and critiqued by teachers and peers. Re-description, carefully steered by the teacher, moved the activity of the class toward productive integration of emerging and personal conceptions with central ideas and values of the discipline. Opportunities to develop disciplined sensibilities about modeling began to take hold in children’s repeated and iterative engagement in the relationships between models and referents, and in some cases, between different representations (models) of the same phenomenon.

How did students come to understand and describe motion? As Latour has argued about professional scientists, they understood the referent by transforming it within a series of inscriptions and symbols. The long-term nature of the students’ work allowed them to connect representational experiences across modalities, including their computational representations with their lived experiences of designing measures in the real world with physical objects. Inextricably related to those representational moves, children came to view data from the world as designed measures, and their epistemic views of what counts as a “good” model deepened as they engaged in cycles of sharing and refining their models to be progressively more communicative. The emphasis on communicativity also led students to make deeper forays into programming and computational thinking. As for the teacher, her vision of what counts as accurate expanded beyond attending to precision (e.g., “accuracy”), to the diverse ways in which phenomena of accelerated motion can be parsed, measured, and communicated about, and class members justified new meanings of accuracy within context-based framings.

Additionally, we have described what happened when computational modeling became part of the life-world of science in an elementary classroom. Our description stands with decades of research on scientific modeling as a practice in science classrooms (e.g., Hestenes, 1992; Sherin, diSessa & Hammer, 1993; Lehrer & Schauble, 2006; Ford, 2003). We do not claim that our integration of computational modeling was revolutionary to opportunities for science learning: for example, the activities could have been designed to leverage an agent-perspective without ABM. Students’ models could have focused on paper-based and diagrammatic forms, without computers. However, we do claim that the integration of computing and programming catalyzed an emphasis on measurement and the mathematics of kinematics in ways that would have been unlikely if computational models were not a part of the design. Using ABM, students’ programs for simple agent-level actions were easily translated into graphs of change, when they programmed them to do so. Additionally, students’ engaged with computational thinking such as loops and proceduralization, seamlessly, in a science classroom.

Agent-based computation bridged the local and syntonic experience of motion in the world with an observer-view measurement perspective. Computing required students to take up a new and more mathematical way of representing a process of time-based change, and in particular, the act of programming in an agent-based language privileged the perspective of the object that was in motion, while our other representations, including flags on the floor and graphs of speed privileged different views of motion events. Disciplined interpretative moves emerged from the nexus of goals for measurement and modeling in kinematic phenomena.

It is also important to note that in our work, computational modeling did not replace the necessity of materiality of science and measurement activities. The ViMAP measurement and graphing tools mirrored the placement of measurement flags in the real world, therefore linking the physical world to computational models and canonical speed-time and acceleration graphs. A key affordance of the programming language was that it connected students' physical and material experiences into their explorations in the computational space, and made new forms of representation accessible while not violating learners' connectedness with the tangible events they set out to model. A deep understanding of the practice of development of scientific models and measures is brazenly incomplete without an understanding of the role that the scientist's interpretation plays in the design of scientific representations. What makes scientific modeling transformative is that it is a fusion of human interpretive and communicative acts made through the reconfiguration of objects and apparatuses, computational tools, and cascading series of representations.

Our work has implications for the praxis of computational modeling in the science classroom. There is now a growing body of literature that argues for the use of multiple and complementary forms of modeling in the classroom (e.g., Danish, 2014; Dickes et al., 2016). In our study too, the students' representational and epistemic work were distributed across a range of computational and non-computational materials, using which they iteratively represented motion as a process continuous change. While modeling with ViMAP enabled the students to connect graphs of change over time to units of change (e.g., step-size), modeling with materials complemented this activity by enabling them to generate the phenomenon being modeled in the "real world", as well as to design the measure of change (e.g., step-size) using video analysis. It is also important to note that "making things concrete" using material forms was an instructional push initiated by the teacher, who co-designed these activities with the research team. We therefore believe that designing complementary forms of computational and non-computational modeling is critical for enabling teacher-adoption and appropriation of computational modeling in the K12 science curricula.

References

Association of Computing Machinery. (ACM, 2003). *A model curriculum for K--12 computer science: Final report of the ACM K-12 Task Force Curriculum Committee.* Task Force of the Pre-College Committee of the Education Board of the ACM.

Axelrod, R. M. (1997). *The complexity of cooperation: Agent-based models of competition and collaboration*. Princeton University Press.

Bollen, L., & van Joolingen, W. R. (2013). SimSketch: multiagent simulations based on learner-created sketches for early science education. *IEEE Transactions on Learning Technologies, (3),* 208-216.

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative research in psychology*, *3*(2), 77-101.

Chandrasekharan, S., & Nersessian, N. J. (2015). Building cognition: The construction of computational representations for scientific discovery. *Cognitive Science, 33*, 267–272.

Cobb, P., Confrey, J., diSessa, A., Lehrer, R., & Schauble, L. (2003). Design experiments in educational research. *Educational Researcher*, *32*(1), 9-13.

Danish, J. A. (2014). Applying an activity theory lens to designing instruction for learning about the structure, behavior, and function of a honeybee system. *Journal of the Learning Sciences*, 23(2), 100-148.

Daston, L., & Galison, P. (2007). *Objectivity.* Zone Books.

Dewey, J. (2005). *Art as Experience.* Perigree, The Berkley Publishing Group, New York. (Original work published 1934)

Dickes, A. C., Sengupta, P., Farris, A. V., & Basu, S. (2016). Development of mechanistic reasoning and multi-level explanations of ecology in third grade using agent-based models. *Science Education, 100*(4), 734–776.

diSessa, A. A. (2001). *Changing minds: Computers, learning, and literacy*. The MIT Press.

Enyedy, N. (2005). Inventing mapping: Creating cultural forms to solve collective problems. *Cognition and Instruction*, *23*(4), 427-466.

Ford, M. J. (2003). Representing and meaning in history and in classrooms: Developing symbols and conceptual organizations of free-fall motion. *Science & Education*, *12*(1), 1-25.

Galison, P. (1997). *Image and logic: A material culture of microphysics*. University of Chicago Press.

Gooding, D. (2003). *Varying the cognitive span: experimentation, visualization, and computation.* In H. Radder (Ed.), *The philosophy of scientific experimentation* (pp. 198-215). University of Pittsburgh Press Pittsburgh, PA.

Greca, I. M., Seoane, E., & Arriassecq, I. (2014). Epistemological issues concerning computer simulations in science and their implications for science education. *Science & Education*, *23*(4), 897-921.

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher*, *42*(1), 38-43.

Guzdial, M. (2004). Programming environments for novices. In S. Fincher and M. Petre (Eds). *Computer Science Education Research*, 127-154.

Hammer, D., Sherin, B., & Kolpakowski, T. (1991). Inventing graphing: Meta-representational expertise in children. *Journal of Mathematical Behavior*, *10*(2), 117-160.

Higgins, C. (2008). Instrumentalism and the clichés of aesthetic education: A Deweyan Corrective. *Education and Culture, 24*(1), 7-20.

Jacobson, M. J., & Wilensky, U. (2006). Complex systems in education: Scientific and educational importance and implications for the learning sciences. *The Journal of the learning sciences*, *15*(1), 11-34.

Keller, E. F. (1984). *A Feeling for the Organism, 10th Anniversary Edition: The Life and Work of Barbara McClintock*. Macmillan.

Keller, E. F. (2003). Models, simulation, and computer experiments. In H. Radder (Ed.), *The philosophy of scientific experimentation* (pp. 198-215). University of Pittsburgh Press Pittsburgh, PA.

Latour, B. (1999). *Pandora's Hope*. Cambridge, MA: Harvard University Press.

Lehrer, R. (2009). Designing to develop disciplinary dispositions: Modeling natural systems. *American Psychologist, 64*(8), 759.

Lehrer, R., & Schauble, L. (2005). Cultivating model-based reasoning in science education. In K. Sawyer (Ed.) *The Cambridge handbook of the learning sciences.* Cambridge University Press.

Lehrer, R., & Schauble, L. (2006). Scientific thinking and science literacy. In D. Kuhn and R. Siegler (Eds). *Handbook of child psychology (6th ed) vol. 2: Cognition, perception, and language* (153-196). Hoboken, New Jersey: John Wiley & Sons, Inc.

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

Lemke, J. L. (2001). Articulating communities: Sociocultural perspectives on science education. *Journal of research in science teaching*, *38*(3), 296-316.

Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook*. Sage.

National Research Council. (NRC, 2010). *Report of a workshop on the scope and nature of computational thinking*. National Academies Press.

NGSS Lead States. (2013). *Next Generation Science Standards: For states, by states*. Washington, DC: National Academies Press. *www.nextgenscience.org/ next-generation-science-standards*.

Ochs, E., Gonzales, P., & Jacoby, S. (1996). “When I come down I'm in the domain state”: Grammar and graphic representation in the interpretive activity of physicists. *Studies in Interactional Sociolinguistics, 13*, 328-369.

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

Pickering, A. (1995). *The mangle of practice: Time, agency, and science*. University of Chicago Press.

Schön, D. A. (1995). Knowing-in-action: The new scholarship requires a new epistemology. *Change: The Magazine of Higher Learning*, *27*(6), 27-34.

Sengupta, P., Dickes, A. C., Farris, A. V., Karan, A., Martin, K., & Wright, M. (2015). Programming in K12 Science Classrooms. *Communications of the Association of Computing Machinery, 58*(11), 33 - 35.

Sengupta, P, & Farris, A. V. (2012). Learning kinematics in elementary grades using agent-based computational modeling: A visual programming approach. *Proceedings of the 11th International Conference on Interaction Design & Children* (pp. 78-87). Bremen, Germany.

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, 17*(1-2), 23 - 42.

Sengupta, P., Kinnebrew, J., Basu, S., Biswas, G., and Clark, D. (2013). Integrating Computational Thinking with K12 Science Education Using Agent-Based Modeling: A Theoretical Framework. *Education & Information Technologies*, Vol. 18, 351 – 380.

Sherin, B. L. (2000). How students invent representations of motion: A genetic account. *Journal of Mathematical Behavior (19),* 399-441.

Sherin, B., diSessa, A. A., & Hammer, D. (1993). Dynaturtle revisited: Learning physics through collaborative design of a computer model. *Interactive Learning Environments*, *3*(2), 91-118.

Strauss, A., & Corbin, J. (1990). *Basics of qualitative research. Grounded theory procedures and techniques*. Newbury Park, CA: Sage.

Watson, J. D. (1968). *The Double Helix: A Personal Account of the Discovery of the Structure of DNA.* New York: Simon and Schuster.

White, B. Y., & Frederiksen, J. R. (1998). Inquiry, modeling, and metacognition: Making science accessible to all students. *Cognition and instruction*, *16*(1), 3-118.

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

Wilkerson, M., Wagh, A., & Wilensky, U. (2015). Balancing curricular and pedagogical needs in computational construction kits: Lessons from the DeltaTick project. *Science Education*, *99*(3), 465-499.

Wilkerson, M. H., & Wilensky, U. J. (2015). Patterns, probabilities, and people: Making sense of quantitative change in complex systems. *Journal of the Learning Sciences*, *24*(2), 204-251.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33-35.

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical transactions of the royal society of London A: Mathematical, physical and engineering sciences*, *366*(1881), 3717 - 3725.

CHAPTER IV:

GROUNDING COMPUTATIONAL ABSTRACTIONS IN SCIENTIFIC EXPERIENCE IN THE ELEMENTARY CLASSROOM

Abstract

The integration of agent-based computational modeling brings about new and productive uncertainties in students' explanations of the relationship of phenomena “in the world” to their models of those phenomena. Computational abstractions that become especially useful in the lifeworld of the classroom are steeped in the history of how those abstractions came to be used and understood. In this manuscript, we first describe how two classrooms used ViMAP in unexpectedly heterogeneous ways. Students developed distinct and varied conceptions about what "counts" as the model, among their code, the agents’ enactments of that code, the graphs, and their verbal explanations of what their model means. Second, we describe how computational abstractions were linked to material enactment, measurements, and paper-based forms of representations. Across both analyses, the heterogeneous nature of computational modeling is illustrated in students’ and teachers’ multiple uses of the ViMAP programming environment and their movement across computational and material media in order to convey meaning. The heterogeneity of computational abstractions—and how they come to enter the shared language within these classrooms—have implications for our understanding of how learners perceive the shared production of scientific explanations among themselves and the computational and non-computational tools they use. Each of these findings has implications for K12 computational modeling, both in regard to science education and computing education.

Science education scholars now predominantly view learning science as the participation in scientific practices and involvement in a “mangle” (Pickering, 1995) among theories, instruments, and investigations (Lehrer & Schauble, 2006; NRC, 2012). In this view of science, the experience of *doing* scientific work is not a linear journey from conceptualizing a hypothesis to verifying it. Instead, scientific work is rife with uncertainties, and given the inherently ill-defined nature of most scientific work, many new scientific explanations and inventions arise from scientists’ efforts of managing such uncertainties (Duschl, 2008). Scholars of computing education also argue that students’ preparedness to cope with complex tasks is a necessary and learned disposition for computational problem solving in professional settings (NRC, 2010; Grover & Pea, 2013).

How can computational modeling help students in managing the complexity and ill-defined nature of scientific modeling in K-12 classrooms? At the broadest level, this is the question we concern ourselves with in this paper. Wing (2006, 2011) defined the phrase “computational thinking” to indicate “thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent” (Wing, 2011, p. 20; Wing attributes this definition to unpublished shared work with colleagues Jan Cuny and Larry Snyder). According to Wing (2008), the “nuts and bolts” (p. 3718) in computational thinking involve dealing with computational *abstractions* in the following ways: a) defining abstractions, b) working with multiple layers of abstraction, and c) understanding the relationships among the different layers (Wing, 2008). Wing (2006) emphasizes the generalizability of computational abstractions as the source of computational power, which in turn give computer scientists the power to scale and deal with complexity. However, a phenomenological interpretation of Wing’s notion of abstractions is incomplete without a deeper understanding of the contextualization that necessitates and grounds computational abstractions in professional practice (Sengupta, Dickes and Farris, 2018; Sengupta, Kinnebrew, Basu, Biswas, & Clark, 2013). For example, as Sengupta et al. (2013) argued, Schmidt (2006) points out that software researchers and developers typically engage in creating abstractions that help them program in terms of their contextualized design goals —e.g., the specific problem that they are solving, which is often in a different field (domain) of professional practice. Therefore, Sengupta et al. (2013) have argued that abstractions in computing are usually grounded in disciplinarily (or contextually) relevant epistemic and representational practices.

Given the current push to integrate computing (e.g., programming and modeling) in K-12 science education (Sengupta et al., 2013; Wilensky, Brady and Horn, 2014), an obvious question then arises in the form of how students and teachers ground (e.g., interpret and use) computational abstractions in the context of modeling science in their classrooms. While a few recent papers have tried to answer this question, their work has predominantly been limited to students' and teachers' work with programming languages and modeling platforms in studies that have been predominantly led by researchers (Sherin, diSessa, & Hammer, 1993; Wilkerson & Wilensky, 2015; Sengupta & Farris, 2012). The goal of this paper is to present a more phenomenological view of how students and teachers take up computational modeling in the science classroom, in the specific context of using both computational and non-computational forms of representations and modeling motion as process of continuous change. A key theme across the two classrooms is the *heterogeneity* inherent in students' modeling, as evident in how they distributed the intended scientific meaning of their work across various forms of modeling.

Our study falls within the design-based research paradigm, where we met with the teacher weekly to discuss curricular activities and provide assistance as needed. Our study is also phenomenological nature, in the following two senses: First, in contrast to most classroom-based studies of programming and computational modeling where researchers typically design both the computational technologies (programming languages) as well as learning activities (e.g., Bers, Flannery, Kazakoff, Sullivan, 2014; Sherin et al., 1993), in our study, the choice of most of these forms of modeling were driven by the teacher and the students. Second, while there is a robust body of literature in statistics education on modeling chance, uncertainty, and variation (Rosebery & Rubin, 1989; Lehrer & Romberg, 1996), our analysis and design approach are not concerned with matching students' work with canonical descriptions of uncertainty in statistics education. In science education, too, scholars routinely regard that learners need to participate in scientific practices which are characterized by moments of uncertainty and tentative knowledge (NRC, 2007; NRC, 2012; Manz & Suárez, 2018). However, very few studies have focused on this issue. Our work extends this goal in the context of educational computing and science education. We focus on how students and teachers manage their *experiences* of uncertainty in the context of modeling science using agent-based modeling and programming. This means that even when participants in our study used terms such as "accuracy" in their own ways, our goal was not to intervene in order to rectify or re-align teachers' and students' work with canonically defined terminology and approaches in statistics and mathematics education. Instead, understanding the meanings constructed by the teachers and students was of critical importance to us.

To this end, we illustrate how computational abstractions are created and re-created in both virtual and physical forms in two teacher-led classrooms in elementary grades in the context of modeling kinematics. We illustrate two forms of heterogeneity that emerged in each classroom. We found two distinct trajectories of modeling, based on how students used graphing and simulation in their models. In addition, we also illustrate how the work of modeling was distributed across computational and physical representations. Attending to these forms of heterogeneity, we contend, is important for understanding of how teachers and students with no prior experience in programming can take up computational modeling as a *language* for doing science. In the final section of the paper, we discuss the implications of our study in terms of the design of computational modeling platforms and activities for classroom integration in K-12 science.

Research Questions

The investigations reported in this paper were conducted as part of extended scientific modeling sequences that integrated agent-based computational modeling in two science classrooms: a 4th and a 5th grade classroom. We seek to answer two research questions:

- How do students and teachers assign and distribute meaning within and across the different computational representational elements in their virtual models?
- How do students and teachers support progressive refinement of models?

Theoretical Background

The Reflexivity Between Computational Thinking, Design-based Learning, and Modeling

Computational thinking is an increasingly ubiquitous epistemic practice in all fields of scientific and engineering research (NRC, 2010). Wing (2006, 2008) has described computational thinking as a general, analytic approach to problem solving, designing systems, and understanding human behaviors. As Wing (2006, 2008), and Sengupta et al. (2013) pointed out, computational thinking draws on concepts that are fundamental to computing and computer science, but also includes practices that are central to a large number of scientific, engineering, and mathematical disciplines. These practices include designing and constructing models and simulations; generating multiple forms of abstraction; reformulation; and verification (diSessa & Abelson, 1986; diSessa, 2001; Guzdial, 1994; NRC, 2010; Papert, 1980; Pea, 1986; Sherin, 2000; Soloway, 1993). At the core of these practices, however, is design-based thinking (NRC, 2010; Wing, 2006).

Design is a form of problem solving in which thinking, tool manipulation, and materials are reflected in the iterative construction and refinement of an artifact (Bucciarelli, 1994; Roth, 1996; Simon, 1969). In pedagogical approaches based on design-based learning, students' construction failures play an important and productive role in learning. By interweaving action and reflection, design-based learning involves students iteratively refining their representations of the target phenomenon (Harel & Papert, 1991; Lehrer & Romberg, 1996; Kafai & Ching, 1998; Kolodner et al., 2003; Papert, 1980; Penner, Lehrer & Schauble, 1998). Both programming and modeling are examples of design-based activities where construction failures and iterative refinement of these failures constitute important aspects of learning, as we explain next.

Papert (1980) noted that learning to be a master programmer requires learning to become proficient at isolating and correcting "bugs", that is, the components of the program that keep the program from working properly or producing the desired outcome (Papert, 1980; see also Khlar & Carver, 1998). Identifying and fixing bugs—also known as debugging (Papert, 1980)—therefore, is an important component of learning programming, and plays an important role in constituting the design-based nature of the practice of programming (Harel & Papert, 1991; Kafai & Ching, 1998; Papert, 1980).

As Sengupta et al. (2013) identified, the characteristics of modeling that contribute to its iterative, design-based nature are reformulation and validation (Kolodner et al., 2003; Penner, Lehrer & Schauble, 1998; Hestenes, 1993). Reformulation involves defining or re-stating the given problem using a formal representational system (e.g., equations, computer programs). Validation involves explicitly stating and justifying the relation between the problem and the proposed solution. As Penner et al. (1998) pointed out, both these processes occur iteratively in design problems. This is also the case in scientific modeling (Giere, 1988; Hestenes, 1993; Lehrer, Schauble, & Lucas, 2008; Nersessian, 2008). In designing scientific models, scientists simplify the world (i.e., the target phenomenon to be modeled) by determining the rules of interaction between constituent elements that they deem important. The process of determination, in turn, is iterative. Modeling, therefore, requires articulating and instantiating appropriate objects and relations in a dialectical manner based on repeated cycles of designing mathematical or computational abstractions, making iterative comparisons of the generated representations and explanations with observations of the target phenomenon, and generating progressively more sophisticated explanations of the phenomenon to be modeled. Therefore, developing a computational model of a physical phenomenon involves key aspects of computational thinking identified by Wing (2008): identifying appropriate abstractions (e.g., underlying mathematical rules or computational methods that govern the behavior of relevant entities or objects), and iteratively refining the model through debugging and validation with corresponding elements in the real world (Sengupta et al., 2013).

Conceptual Difficulties in Learning Kinematics

Studies on naive cognition of kinematics (Elby, 2000; Halloun & Hestenes, 1985; Hestenes, Wells, and Swackhamer, 1992; Larkin, McDermott, Simon, & Simon, 1980;

Leinhardt, Zaslavsky, & Stein, 1990; McCloskey, 1983; McDermott, Rosenquist, & van Zee, 1987; Trowbridge & McDermott, 1980, 1981) show that students experience the following kinds of conceptual difficulties: a) understanding and explaining the qualitative and formal (mathematical) relationships between distance, speed, time and acceleration, and b) understanding and generating graphs and equations that represent these mathematical relationships. Understanding these, in turn, requires being able to conceptualize and represent motion as a process of continuous change, which in turn has been shown to be challenging for novice learners. For example, when provided with a situation that involves objects moving with uniform acceleration (e.g., during a free fall or on an inclined plane), novices find it challenging to differentiate between instantaneous speed and average speed, (Halloun & Hestenes, 1985) and tend to describe or explain any speed change(s) in terms of differences or relative size of the change(s), rather than describing speeding up or slowing down as a continuous process (Dykstra & Sweet, 2009). From the perspective of learning physics, our learning goals in this study were for students to be able to understand and mathematically represent speeding up and slowing down as a continuous process, as well as being able to differentiate between position and instantaneous speed.

Affordances of Agent-based Programming for Learning Kinematics

Agent-based modeling and programming has a long history both in mathematics as well as in physics education (e.g., Abelson & diSessa, 1981; Hammer et al., 1991; Papert, 1980; Roschelle & Teasley, 1995). Papert (1980) and Abelson and diSessa (1981) argued that by commanding the movement of the computational agent (e.g., the LOGO turtle) on the computer screen using body-syntonic programming commands, students can meaningfully explore and develop deep and nuanced understandings of concepts in mathematics that can often be difficult to understand even for more advanced learners. Agent-based modeling has two specific affordances that are particularly relevant to our study.

The first affordance is the intuitive nature of agent-based computational representations. For example, when students use LOGO (Papert, 1980) to build a program or a model, they use two types of representations: programming commands that control the behavior of turtles, and the resultant turtle graphics. The programming commands, as Papert (1980) pointed out, are intuitive because they are body syntonic, that is, they can be easily mapped onto simple physical movements and actions such as moving forward, turning, or repeating actions. Novice learners could use such commands to develop models by generating turtle graphics. Each command is carried out as a discrete step, and therefore the enactment of commands is event-based. For example, let us consider a LOGO turtle moving in a Newtonian world where an applied force leads to a continuous change in a turtle's velocity. The unit of time in LOGO is a "tick", which represents a single iteration (or run) of the program. In LOGO, this process of continuous change of velocity is represented in the form of updating two variables in the turtle's "state" during each tick: the distance it travels, and its heading of the turtle. The result is a rectangular spiral in which every line is slightly longer (or shorter) than the previous line. Each event, in this case, is the movement of the turtle during a single step. This movement, being body-syntonic, can be understood by young learners (Papert, 1980; diSessa, 2001). The spiral as a representation of a process of speeding up or slowing down, can therefore be argued as an intuitive representation of motion for novice learners (Sengupta, Farris & Wright, 2012).

Chi and her colleagues have argued that event-based mental representations can hinder the development of understandings of continuous processes (Chi, 2005; Slotta, Chi & Joram, 1995). They have argued that continuous processes are ontologically distinct from event-based processes (Chi, 2005; Slotta & Chi, 2006). In contrast, our approach here is grounded in the constructivist approach as outlined by Smith, diSessa, and Roschelle (1994), Hammer (1996), diSessa (1993), and Gupta, Hammer, and Redish (2010), where naive conceptual knowledge is not regarded as a hindrance towards the development of expert-like knowledge. In this perspective, the emphasis for educational designers is on trying to identify, recruit, and build upon the productive epistemic resources and representational competencies that learners bring with them to the classroom, rather than discarding their naive knowledge and competencies. Previous research on using agent-based modeling to teach kinematics also supports our conjecture. Several scholars have shown that students' intuitions about motion, that typically discrete and event-based, can be productively leveraged through appropriate scaffolding to generate correct understandings and representations of motion as a process of continuous change (diSessa, 2001, 2004; Ford, 2003; Sherin et al., 1993; Sengupta, Krinks & Clark, 2015).

The second affordance concerns the intertwined nature of development of students' conceptual understandings and representational competencies. Metarepresentational competency or MRC refers to the creation, critique, and refinement of representations of phenomena or data. MRC encompasses knowledge about the purposes of representations in general as well interpretation of pre-determined representations such as graphs and diagrams (diSessa, 2004). Educators in multiple domains—mathematics (Hall, 1996), biology (Lehrer & Schauble, 2006), and physics (diSessa et al., 1991; diSessa, 2004; Ford, 2003)—have shown that students' naive representational competencies can be leveraged and bootstrapped to develop sophisticated mathematical representations such as graphs, computer simulations, and data structures that can successfully represent scientific phenomena by engaging students in modeling-based curricular activities. Of particular relevance to our work are the studies conducted by diSessa and his colleagues. In the first study on developing MRC using agent-based programming, students' hand-drawn graphical representations of motion evolved from representing individual, discrete events, similar to snapshot views as found by Dykstra & Sweet (2009), to representing motion as a process of continuous change by aggregating these events, in ways that were similar to canonical representations—thereby "inventing graphing" (diSessa et al., 1991). Similar results were also found by Sherin (2000) and Ford (2003), who identified that several high school students who undertook with the same BOXER physics curriculum represented continuous change in motion by representing temporal sequences of discrete events.

Our position, similar to Papert (1980) and diSessa (2001, 2004), therefore, is that agent-based representations can leverage the deeply interconnected nature of students' intuitive knowledge about the physical world and their native representational competencies. This is also aligned with the *science-as-practice* perspective that we outlined earlier, in which the development of conceptual knowledge and representational competencies co-occur. This nexus can create what diSessa (2001) has termed regimes of competence—i.e., contexts in which the learner is challenged, yet does not find the challenge insurmountable. Conceptual growth co-occurs as learners develop progressively more sophisticated computational and mathematical representations in order to model the target kinematic phenomena. This, we believe, is at the root of the appropriateness of using agent-based environments for learning kinematics.

Integrating Programming with Physics Curricula: Need for a "Mangled" Approach

Despite the affordances of agent-based programming for learning physics, a central challenge in integrating agent-based programming with K16 classroom physics is that in order to program computer simulations of any physical phenomenon, both teachers and students must have some operational fluency with the programming language being used for instruction. For example, Guzdial (1994) argued that the challenge in using programming to learn other domains is that programming as an activity often requires more skills and knowledge which is disconnected from the domain-specific learning goals (Guzdial, 1994; see also Oren, 1990; Norman, 1993). For example, Pea and his colleagues noted that novice learners found the procedurality of the LOGO language, its control structures that allow very brief recursive programs, and the use of conditional tests to be deeply challenging without adequate scaffolding (Pea & Kurland, 1986; Pea, 1986). Sherin et al (1993) found that even when students had developed some proficiency with a programming language after a few weeks of programming instruction, despite that fairly extensive learning experience, many students still found writing LOGO programs in the BOXER environment (diSessa, Abelson & Ploger, 1991) to be quite challenging. In terms of classroom instruction, this often led to digressions during the class sessions, as the teachers had to clarify details of programming. This in turn reduced the amount of attention focused directly on teaching and learning physics (Sherin et al., 1993).

Therefore, in classroom-wide studies using physics curricula that involve programming, the curricular units have traditionally devoted a significant amount of time on programming instruction that precedes physics instruction. In the studies reported by Sherin et al. (1993) and diSessa et al. (1991) middle and high school students underwent multiple (15 or more) weeks of instruction, out of which the first several weeks of classroom instruction were devoted solely to learning programming taught by a programming expert, and the next few weeks were taught by a physics teacher. As Sherin, diSessa & Hammer (1993) pointed out, given the time constraints already faced by science (in their case, physics) teachers, the additional overhead associated with teaching students to program may “simply be prohibitive” (p. 116).

How can we address these issues? One part of the answer is that domain-specific programming languages can be very helpful in lowering the overhead of learning that Sherin et al. (1993) identified (Sengupta et al., 2013). In such modeling environments, the programming commands are carefully designed to include not only necessary domain-general computational abstractions (e.g., loops and conditionals), but also domain-specific commands (e.g., speed-up and slow-down), which can provide an easier entry point into computational modeling and can be expanded on later in the curriculum through further deconstruction (e.g., creating their own programming commands by modeling how "speed up" or "slow down" should work) (Sengupta et al., 2015; Sengupta et al., 2013). However, we believe that the other part of the answer — perhaps the more complex part—lies in viewing computing in science (including K-12 science education) as a fundamentally heterogeneous practice that is vastly more expansive beyond the more *technocentric* (Papert, 1987) view of learning to do science using programming commands (Sengupta, Dickes and Farris, 2018). This is the part that we are concerned with in this paper.

There is now a growing body of evidence that collectively suggest that computational representations within the programing and modeling environment, as well as other complementary forms of representations outside the computer, need to be considered as relevant and necessary for the integration of agent-based programming and computational modeling in elementary and middle school science classrooms. For example, Dickes, Sengupta, Farris, and Basu (2016) and Danish (2014) showed that embodied modeling as well as creating physical representations of change over time can serve as powerful anchors for grounding multi-agent-based programming and modeling in the context of learning about complex ecological systems. Similarly, our own previous work suggests that physical modeling of motion (e.g., using cars and ramps) can offer 4th and 5th grade students important opportunities for reflection and refinement of their own agent-based computational models by alerting them to the need for properly initializing their models (Sengupta & Farris, 2012). And finally, drawing upon findings from a series of design-based professional learning sessions with 56 teachers in K12 public and charter schools, Sengupta, Brown, Rushton and Shanahan (2018) showed that (a) when teachers, with little or no background in programming, view programming as a way to "mathematize" the world, they can visualize and implement "seamless" integration of programming and modeling with their science curricula; and (b) the use of multiple and complementary forms of programming and modeling (e.g., physical, virtual and embodied) can facilitate such integration.

These studies suggest that a more *mangled* view of practice (Pickering, 1995) is essential for integrating computing and science education in the K-12 science classroom. Specifically, these studies suggest that the programming and modeling activities should be aligned with the forms of investigations and representations that teachers and students already use as learners investigate (in our case) motion. These include representations such as dot traces, velocity and displacement graphs, and word problems, as well as investigations of "real" objects in motion. Pickering (1995) describes scientists' work as a "dance of agency" among theory and instrumentation, between people and things. In order for computational representations to find their place in a "mangle" that was appropriate for young learners and for classrooms, we needed computational representations be able to enter into dialogue with the investigations and non-computational representations that students were using to explain the world. Without continuity (from the *learners'* perspectives) of the computational modeling work with their other forms of modeling and emerging theories, there could be no "dance" of agency. Our goal is to investigate how teachers and students experience this dance of agency in the context of modeling motion using an agent-based programming in their "everyday" science classroom for modeling motion, by using multiple (heterogeneous) representational forms and practices in meaningful dialogue with one another, guided by the teachers' re-framing of programming as a way to help their students model and measure motion as a process of change over time.

Methods

We conducted design-based research studies (Cobb, Confrey, diSessa, Lehrer, & Schauble, 2003) in two different classrooms (Grade 4, in an elementary setting; and Grade 5, in an early middle school setting), with the central aim of understanding how agent-based computational modeling can be integrated in science classrooms. Although the school offered computers for students, to minimize logistical issues for data collection (we collected children's work in the form of log files on the computer), we provided laptops with the software that we were continuing to develop (ViMAP) and met with teachers weekly to discuss plans and future learning activities. Our goal was that plans for instruction would be driven by the teacher, and that continued development of ViMAP would follow the needs and emergent goals in the classrooms. The overall design of curricular activities was guided by our conjecture that teachers and students would adapt the computational media environment (ViMAP) to meet learning goals, and we wanted to understand how this took place and to what ends.

As is the case typically with design-based research studies, we began our investigations with an embodied conjecture (Sandoval, 2004), that programming and computational modeling will be reframed as "mathematizing" (Farris & Sengupta, 2014; Sengupta et al., 2015; Sengupta, Brown, Rushton & Shanahan, 2018) by the teachers and students as part of their everyday science curricula. However, as the studies progressed, the specific forms of appropriation of computing, programming, and computational modeling by teachers and students revealed interesting differences that are essential for understanding the experience of appropriation of computational programming and modeling technologies by K-12 teachers and students. In Classroom A, the learning goals centered on the 5th grade science content and were guided by Ms. Gray's learning objectives, in particular, constructing and interpreting graphs of change over time. In Classroom B, the Ms. Beck also emphasized graphing as a key learning activity. However, what she considered to be productive uses of ViMAP as a modeling tool initially began as a very graph-centric view, and shifted over time to include a greater range of diversity of representational forms. She was always involved in planning, up until the beginning of her leave in February, and only very rarely did the researchers play the role of the "co-teacher" in the classroom when Ms. Beck was present.

Research Settings

The contexts of data collection are two publicly-funded charter school classrooms in a large metropolitan area in the mid-South. In each of these classrooms, the same team of researchers worked in partnership with the respective classroom teachers to integrate agent-based programming and modeling (using ViMAP) within the existing science curricula over a period of approximately 9 months. The study with Ms. Gray was conducted during September 2012 – May 2013 (although work with students did not begin until November), and the study in Classroom B was conducted during late August 2014 – May 2015.

The rationale behind selecting these two classrooms for inclusion in this manuscript is threefold: First, both the classroom teachers were working within similar institutional contexts—the schools were administrated by the same principal, funded publicly, and were operating under curricular mandates within the same state and in the school district in the Southeastern US. Second, neither teacher had any previous experience with programming and both of them approached the researcher-teacher partnership at the beginning of the school year from a similar perspective: their objective was to use computer programming in a way that fit their science curricular needs. More specifically, both teachers approached agent-based programming as a way to mathematize physical phenomena. Third, despite these similarities, as one might expect, the teachers varied in terms of their instructional approach and classroom teaching—in terms of both leading classroom instruction and the design of learning activities.

The differences emerge from several reasons. For example, instruction was differently organized in 4th and 5th grades in the specific school district. In 5th grade, students saw the same teacher only for math and science periods (Classroom A), whereas, in 4th grade, the same teacher was responsible for teaching all subjects to the student (Classroom B). We also noticed that although both teachers, overall, reframed agent-based programming as mathematical modeling in the science classroom, they also came to value different forms of representational and epistemic work as relevant to their science curricula. This in turn was also instrumental in shaping what their students came to view as "good" models. And yet, across these two classrooms, we noticed deep thematic similarities in the form of how the students distributed the *meaning* of their work across different elements within the virtual modeling platform (ViMAP) and between ViMAP and the physical world. Therefore, we believe that paying close attention to the different experiences and trajectories of modeling across the two classrooms can in turn help us develop a deeper understanding of the essential heterogeneity in how computational modeling can be adopted in K12 classrooms as a "*language"* (Giere, 1984, p. 80) of science.

Classroom A was a Grade 5, gender-segregated (all female, as identified by the school) science class with 16 students. The instructional block in which data were collected was specifically during the "science block", that is, instructional time allotted by the school for science education. We entered the classroom in early November were present for 54 days of classroom work between November and May, approximately twice a week. The population of the Classroom A school is predominately Black and low-income, and 95% of students are eligible free or reduced price lunch. All 16 of the participants in the class were African American. The classroom teaching was primarily based on the teacher's (Ms. Gray) objectives of modeling and graphing change over time, although the activities were co-designed by the teacher and the researchers. Throughout the academic year, the teacher and the researchers jointly identified several different phenomena as disciplinary contexts for representing motion as a process of change over time. However, the researchers often initially led the classroom instruction at the request of the teacher. The teacher would participate in leading class discussions as and when she felt comfortable or deemed appropriate. The sequence of activities for Classroom A is shown in Table 1.

Table 1

Sequence of Activities (Classroom A, Grade 5)

Unit	Dates	Summary
Pre-assessment	Nov. 11	Researcher-made pre-assessment
"Constant Speed" Shapes	Nov. 15 – Dec. 4	Intro to ViMAP programming and modeling; Turtle geometry, centered around learning goals in perimeter and angles of polygons
Quilt Stories and Introduction to Enactments as Scientific Models	Dec. 5 – Dec. 18	Storytelling and Modeling, introduction to representing stories through ABM
Gravity 1: Free fall of a dropping ball	Jan. 15 – Feb. 27	Students measure and model continuous changes in speed, using the free fall of a dropping ball (and videos) as contexts; students used paper representations, Lego graphs, and ViMAP to model acceleration
Gravity 2: Marbles on ramps (Introduced force as a reason for acceleration)	Mar. 1 – Mar. 13	Students measure and model continuous changes in speed, using acceleration down a ramp and free fall as contexts; students design experiments and model in ViMAP
Ecology: Energy loss/ gained	Apr. 1 – May 21	Students modeled energy changes for foraging butterflies and their own self-ratings of energy throughout the day.
Post-Assessment	May 21	End-of-year assessment

The data from Classroom B were collected in a mixed-gender 4th grade classroom in an elementary public charter school. The teacher, Ms. Beck, and seven of her students were involved in a year-long ViMAP study during their third grade year. Fourteen additional students were present in Ms. Beck's classroom and were not involved in computational modeling during the previous academic year. The sequence of activities for Classroom B is shown in Table 2.

Unlike Classroom A, the study in Classroom B was conducted in a primary classroom and the teacher considered the instructional time as dually dedicated to math and science content.

Students carried out investigations of natural phenomena in kinematics and in ecology in modeling cycles that included (but was not limited to) modeling in ViMAP. Twenty-one students and their teacher participated in the classroom work. Of 22 students, 19 identified themselves as African-American, one student identified himself as Latino, and one student identified herself as Somali. The class was comprised of 11 females and 10 males. Ms. Beck and Amy co-planned the activities based on the students' progress and Ms. Beck's plans across the curriculum, which often emphasized mathematics aims of number and operation, measurement, and graphing. During class time, the teacher played the primary teaching role, often adjusting the plans to meet the emerging instructional opportunities as she enacted the plans. Every day of the study, one to two members of the research team (Amy and usually an additional person) were present to conduct field observations and collect video data and occasionally co-teach; although unlike Classroom A, the explicit goal was that Ms. Beck remained the primary teacher throughout the study. During February of that academic year, Ms. Beck began a medical leave of absence that continued through the end of the school year. In her absence, Amy taught during the time allotted to our study, and communication with the substitute teachers was sporadic, given school-level barriers to staffing the classroom.

Table 2

Sequence of Activities (Classroom B, Grade 4)

Unit	Dates	Summary
Observations, pre-assessment, and interviews	Aug. 11 – Sept. 8	Researchers conduct observations, preliminary interviews with all students in the class
Survival Kits Geometry Unit	Sept. 9 – Oct. 2	Intro to ViMAP programming and modeling; Turtle geometry, centered around learning goals in perimeter, area, and angles of polygons; model sharing and revision
“Constant Speed” Robots	Oct. 14 – Nov. 20	Students develop understanding of speed as a rate of the distance traveled in a unit of time, including cycles of model sharing and revision; students used both ViMAP and physical modeling
Constant Acceleration and Gravity	Nov. 25 – Feb. 3	Students measure and model continuous changes in speed, using acceleration down a ramp and free fall as contexts; students used ViMAP, video analysis and physical modeling
Friction	Feb. 5 – Mar. 31	Ms. Beck’s leave of absence begins; students model processes of “slowing down” with Matchbox cars on different surfaces; students used both ViMAP and physical modeling
Interviews	Apr. 7 – Apr. 28	Mid-year interviews with all students
Modeling Ant Colonies	May 6 – May 13	Students model ant foraging, reproduction and predation in ant colonies in an embodied modeling activity, followed by programming in ViMAP-Ants. Students share and refine their models with 8th grade mentors.
Post-Assessment	May 14 – May 19	End-of-year assessment and focus group

Forms of Data

The data from Classrooms A and B include video of each class, interviews with students and student groups, student work, planning documents and discussions of the teacher and the researchers, photos of whiteboards and other representations, detailed field notes from each session, and automated screen captures from the students’ computers. The researchers conducted informal interviews while the students were engaged in single, pair or small group work. Informal interviews typically took one of two forms: First, interviews were often conducted when students requested help from one of the researchers. These were recorded as frequently as possible. Second, interviews were conducted in order to ask students to explain his or her thinking and reasoning about the different modeling and representational forms used in the study. Students' ViMAP models were saved and downloaded at the end of each day and transferred to a secure server. Additionally, the computers used in the study ran a script that recorded screen captures every thirty seconds on all student computers. Screen captures were also downloaded at the end of each day from each student computer and transferred to a secure server.

Coding and Analysis

Throughout data collection, we recorded field notes of the events each day and in planning with the teachers, and noted episodes (e.g., from video or student work or conversations with the teacher) that were relevant to the central inquiry of how agent-based computational modeling was becoming reframed in practice. Within the first weeks of the kinematics units, it became apparent to us that the teachers in each class wanted to reframe programming as defining units of measurement. We also had two emerging conjectures: (1) That teachers' reframing of programming as mathematics was the thrust of their local learning aims and learners' understanding of what they were doing, and (2) that computing would be amplified beyond the computer—in lower-tech materials and activities that provided context to the computer models. These conjectures became reframed in activities, which we describe in the data and analysis section.

Our analysis followed a constant comparative approach (Glaser, 1965) of the conjectures as they were iteratively embodied in the design of the study and in the collaborative design of learning activities. Data collection and planning included review of student work, weekly conversations with the classroom teacher, and ongoing review of existing field notes and student artifacts with newer work. As we increasingly noted the centrality of the teachers' and student's heterogeneous and distributed use of computing, our conjectures were reframed as research questions that address (a) how students assign and distribute meaning within the computational models and related materials; and (b) how students and teachers support progressive refinement of models across experiment, other material forms, and computational models.

We traced aspects of experience that were relevant to the emerging research questions, using detailed field notes as the primary data source and tracing backwards to video-recorded classroom instruction, interviews, student work, and other artifacts for deeper analysis. These key data extracts were triangulated with the other sources, across time, and other researchers involved in the study. We reviewed tentative themes via peer debriefing with researchers who were present during the collection and analysis process, and each agreed that the themes and patterns we were seeing were realistic representations of what was unfolding in the classrooms. In order to collect data with potential to reflect diverse contextual factors in students' movement across heterogeneous forms in ViMAP, we followed techniques for persistent observation (Lincoln & Guba, 1985) of the incidents that were relevant to the theme of heterogeneity.

We coded the data and selected cases based on the constant comparative method that emphasizes theoretical sampling (Glaser & Strauss, 1967, pp. 28-52; Miles & Huberman, 1994). The data reported in this paper comes from a constant comparison of incidents in the classroom with our theoretically informed understandings of heterogeneity in students' work. This led us to focus our attention on how students appropriated different features within the ViMAP representational infrastructure (e.g., graphs and graphing functionalities, programming commands, and simulations), as well as outside the computer (e.g., embodied and physical models). In addition to identifying the how meaning gets distributed across these representational forms, we also focus on how students (and teachers) use invented representational forms in order to bring about continuity and coherence across the varied computational and non-computational forms of modeling and representing motion. Finally, where multiple cases were available, our selection was based on how clearly each case communicates the most central aspects of the relevant theoretical perspective. It is also important to remember that both these studies, although led by the teacher in the classrooms, were conducted in the form of researcher-teacher partnerships. Where necessary, we also describe the role of the researcher in shaping the heterogeneity of the work.

The structure of the findings and analysis follows the order of the research questions: (1) How do students assign and distribute meaning within the computational models and related materials? And, (2) How do students and teachers support the progressive refinement of children's models? To answer the first research question, our analysis presents an image of heterogeneity that becomes evident through a comparison across students' computational models. We illustrate two trajectories of scientific modeling that emerged from how students and teachers distribute their intended scientific meanings differently across graphs, programming commands and simulations in their models. Over time, these ways of representing motion as a process of continuous change became two distinct and relatively stable means that students used to explain what parts of their own work "count" as models in Classroom A. Contrastingly, in Classroom B, the teacher first specified students' uses of the programming and modeling environment, and gradually reduced these constraints for students to show scientific meaning in more nuanced ways. To answer the second research question, we describe how classroom participants in both classrooms used heterogeneous forms of models in relation to the computational abstractions they were using and inventing for purposes of re-description. Compared to our analysis for the first research question, this is a different image of heterogeneity that becomes evident in how computing becomes distributed *beyond* programming and modeling on the computer, to include a more diverse range of representations and forms of modeling. Across both studies, for these 4th and 5th grade science learners and their teachers, the integration of a new medium changed the ways that students represented information and ideas. These in turn, created contexts where students stretched to new and ad interim less "certain" forms of explanations.

Findings and Analysis

Part 1: Heterogeneity in Distribution of Meaning across Code, Enactment, and Graphs

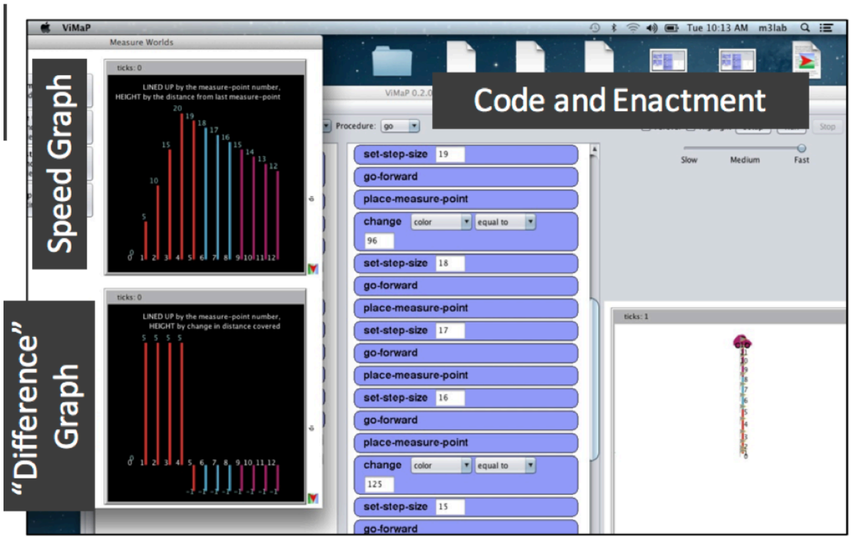
This section is concerned with the ways the that students used the representational features of ViMAP. In order to describe their actions, I must first briefly describe the representational system. ViMAP was designed with three primary modes of representation: First, in the construction world, students select and drag the commands that the agent(s) will carry out when the program is running. Secondly, when students run their programs, they are dynamically played out in the "enactment" world. These displays are always visible are similar to LOGO microworlds (Harel & Papert, 1991) and other modern agent-based programming environments, including Scratch (Resnick et al., 2009). Unlike other environments, we included a graphing display that corresponds to commands for measurement and measure flags in the enactment. For example, if a student uses a command "place measure point," the turtle will drop a visual flag at its position when that command is run. Data from these measures are then automatically graphed in a graphing window, and users select from multiple ways of graphing the data; including "distance traveled since last measure" (speed), and "odometer" (total distance traveled).

Classroom A. Ms. Gray was non-directive in how her students used ViMAP modeling tools. Her desire to have students work on reading and producing graphs of change was a key driver in the development of the ViMAP graphing window, however, she was open to a wide range of possible uses. Therefore, while students in Classroom A used the ViMAP modeling environment in order to communicate explanations of systems of change, there was not a favored or "right" way to do it. However, as the work continued across the academic year, we found that almost all students seemed to favor one of two paths, which we can generally characterize by viewing the turtle geometry shape as the most important feature of the models or the graph as the most important feature.

I summarize these two trajectories with the names "shaping" and "graphing". "Shaping" generally focused on writing code that generated a turtle geometry shape that was representative of some salient feature of the referent. That is, most of the intended meaning was represented in the form of a geometric shape. Likewise, "graphing" focused on the graphical output of their models. The turtle geometry shape, if present, was not used to communicate ideas; instead, it was only used a vehicle to generate the graph, which in turn conveyed the intended meaning. It is important to note that both pathways of modeling led students to similar learning outcomes in terms of computational thinking and kinematics.

Illustrative Examples from Student Work. In what follows, I have selected two students' work as illustrative examples of graph-centric and shape-centric approaches. These episodes are selected in terms of offering succinct characterizations of each approach.

Illustration of graphing approach. Seanna began by using the programming language to create literal, iconic representations of phenomena in the enactment world, using pen-down and movement commands. For example, when modeling a ball in free fall, she generated a program in which the turtle drew the ball, but she had trouble relating the distance traveled by the agent (the turtle) to the distance the ball actually traveled. This confusion caused her to struggle to use the computational toolset to create graph-like representations of the accumulation of distance traveled and speed. Seanna gradually abandoned shape-based (i.e., turtle-geometry) drawings and programmed for the specific purpose of generating graphs, because she saw graphs as a highly communicative convention for communicating processes of change across time. Within her graphs, she invented a mid-level representation, the "period" (a word we did not use). The number of steps taken by the agent was the agent-level representation, and groups of steps she bound together in periods, a mid-level representation (Levy & Wilensky, 2009). Figure 1 shows a characteristic model from Seanna, and an excerpt of talk about that model.



```
set-step-size 19
go-forward
place-measure-point
change color equal to
96
set-step-size 18
go-forward
place-measure-point
set-step-size 17
go-forward
place-measure-point
set-step-size 16
go-forward
place-measure-point
change color equal to
125
set-step-size 15
go-forward
```

Figure 1. Labeled screen capture of Seanna's program and graphs.

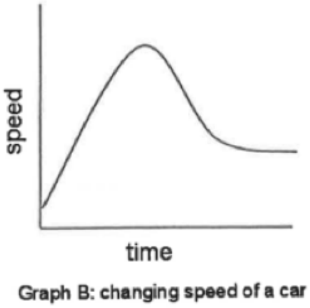


Figure 2. The graph on the activity sheet.

Seanna: …This is just uh, the…this [code and enactment] is just like a model of what this [activity sheet] will probably look like and this [grapher window: speed] explains what this [enactment] means to this [activity sheet] and this [grapher window: "difference" graph] is telling the difference for this [code and enactment]

Researcher (Pratim): Oh, okay, cool. This ["difference" graph] explains the difference…so the difference between the bars?

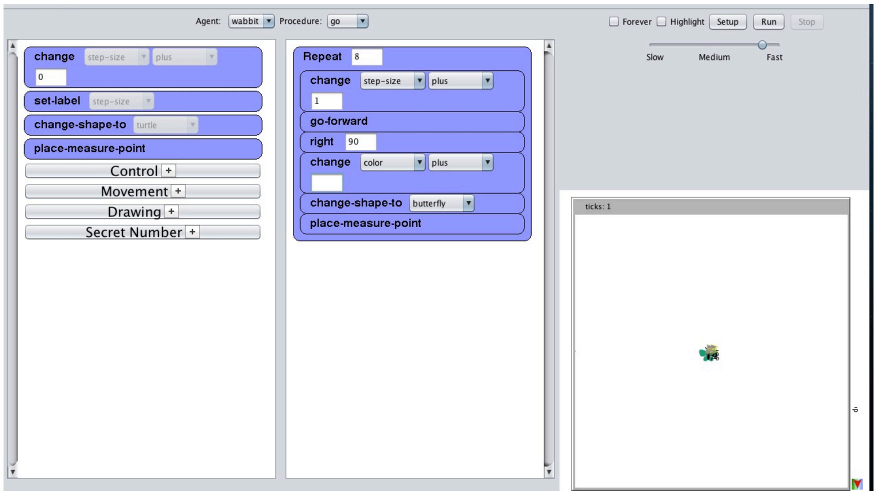
Seanna: Um hmmm, it's sorta, it's cause when I did this I did [toggles to "setup", then "go"..cause every time when I got to a different °hold on°

Pratim: A different step size?

Seanna: No, like in a different period because I tried to show how I did it because like right here, I put a different color, like red, blue, and then purple. I changed the color to show every period I did, it's like, I used it as a period, because at this period it is going up, this period, going down, and then ending.

Seanna's excerpt illustrates an emphasis on graphing, that is, a graph-centric approach. The following excerpt illustrates an example of a shaping, that is, a shape-centric approach.

Illustration of shaping approach. Unlike Seanna's eventually exclusive commitment to the graphs, Shenice described phenomena using relationships evident in the commands and in the shapes left in the enactment space. Throughout the study, she consistently used shapes (drawn by having the agent's pen-down in the enactment space) as a communicative form. In interviews, she talked about multiple representational forms—the code, the enactment, and the graphs as if they all were the model, and each had its own communicative affordances. Her models were distributed across the representational infrastructure, including her own explanations of her work. Her persistence in making shapes led her to take up mathematical scaling and explore the variable space for scaling in ways that Seanna did not. A typical model progression of the type Shenice made is shown in Figures 3 and 4:



```
Repeat 8
  change step-size plus 1
  go-forward
  right 90
  change color plus
  change-shape-to butterfly
  place-measure-point
```

Figure 3. Shenice’s non-scaled (“crumpled up”) model.

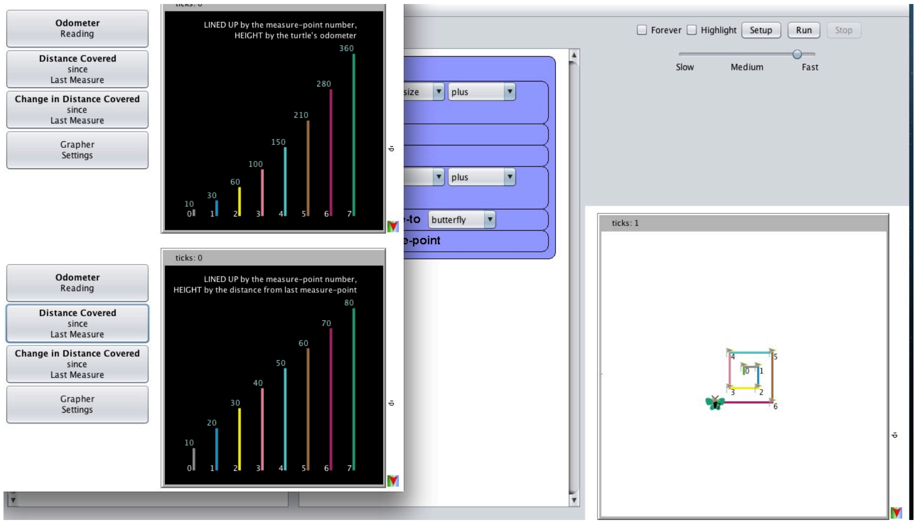


Figure 4. Shenice’s scaled model.

In our interview about the model as Shenice was working on it, Shenice expressed that the non-scaled model was “crumpled up” and “junked up”—although the graphs of the first and final versions were visually equivalent, the shape left behind in the enactment space was the important difference for Shenice.

> Researcher (Amy): Okay, what do you not like?
>
> Shenice: It's crumpled up. Look.

Shenice initially tried scaling up the distance traveled in each "step" in her model by a factor of 100. While she shared that she thought it "look[ed] cool," it was difficult to understand, due to wrapping. We tried multiple different ways of changing the step-size in the code, and discussed the multiplicative correspondence between quantities.

> Amy: *(as Shenice is making a change)* Why do you think 10 is a good idea? You can run it first. *(Shenice runs her code.)*
>
> Amy: Tell me what you're thinking.
>
> Shenice: 1 is 10, 2 is 20, 3 is 30.
>
> Amy: Okay, and this is pretty easy. You could tell somebody who is interpreting your graph that these have been scaled up by 10.
>
> Shenice: And it doesn't look a mess.
>
> Amy: It..it does what?
>
> Shenice: It doesn't look a mess. Like the other one was junked up.
>
> Amy: When we changed step size by 1, it was jumped up *(sic)*. It was all too tiny. Here, it's all easy to see.

The expanding pattern of the length of the lines was important to Seanna, and scaling the length of the lines by a factor of 10 made the enactment change from "junked up" and non-communicative, to something useful for communicating a pattern of change.

Despite the differences in students' use of the modeling toolset, the central learning gains in understanding and representing processes of change across time are common across most of the learners. Of the 13 learners who were present at post-test, 12 were able to interpret one or more speed vs. time graphs correctly to show change over time. This is compared to only 1 correct interpretation at pre-assessment. Ten created paper-based graphs of a novel situation of time-dependent change independently at post. Twelve students wrote about the temporal nature of change in explanations of graphs in their post-assessment, using clock-based times or words like “next” and “then.”

Our primary finding of this analysis are that students distributed meaning and sense-making across the representational infrastructure of computational modeling in diverse ways, based on individual perceptions of what was important and desires about what they wanted to generate. The teacher, as well as the rest of the research team, remained ambivalent. Ultimately, learners adapted the programming environment, along with additional classroom tools, to communicate key disciplinary ideas through combinations of the code, graphs, their talk and gestural explanations, and dynamic enactment of their models. Furthermore, the learning gains were non-specific to particular patterns of use. Students demonstrated two general ways of thinking about what "counts" as the model: (a) those who used the programming language for the purpose of generating graphs (eight students), and (b) those who distributed communicative aspects of their model across the code, the enactment, and their graphs (five students). That is, students’ negotiations about how to utilize the modeling platform in order to communicate scientific ideas varied, but were non-consequential for our learning gains as measured by our post-test.

Classroom B Analysis. In Classroom B, Ms. Beck preferred that students worked with data to generate graphs with measurements that “matched” the measurements that students collected from the world. She referred to this numerical match between data and the labels as “accuracy,” and the students often used this language. The measurements in the graphing interface are generated by asking the agent to “place measure point.” These measure points, along with the agent’s movement around the enactment space, constitute information that is then automatically populated as a unitless numerical value on the graph. While several variables for speed were possible within the command blocks that students selected from, and these other commands could be automatically graphed in the graphing window, she wanted her students to use the distance the agent moved in one “step” of the program as the variable to indicate the speed of an object or agent. Ms. Beck adopted a very graph-centric view of ViMAP, and she orchestrated student activity so that the acceptable forms of work were succinct programs that created graphs. Across the course of the 4th grade year, she expanded her views regarding students’ productive use of ViMAP. I illustrate this change in the following case:

Graphing. In late October, as students constructed ViMAP models based on their measurements of the speed that robots moved across the gym floor (see Chapter III), Ms. Beck’s instructions specified that students must generate graphs. All students produced ViMAP programs that created graphs that corresponded to the total distance traveled by their robots. One student initially included an additional form of representing: Aden used a geometric, square-like shape to show the distance traveled in each consecutive increment of time in his model, in this image, the very small square that appears on the right side of Figure 5, which shows his graphs and enactment world.

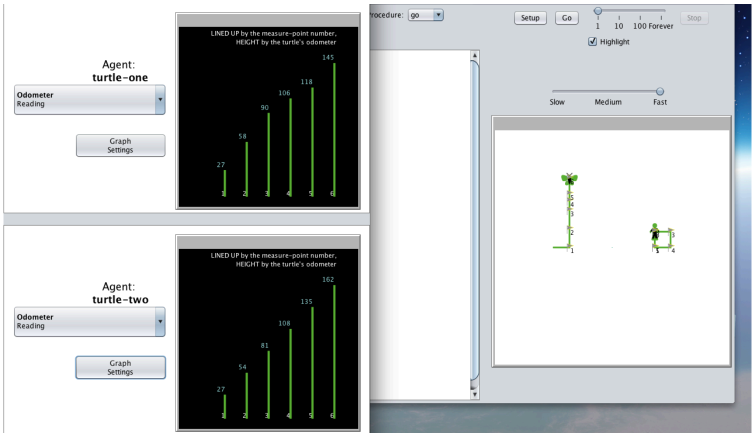


Figure 5. Aden's model. Notice that the turtle inscription on the right uses turns between the steps, despite the referent's non-turning motion.

When I asked Aden about his program, he explained to me that he had used the square to show the "sameness" of the step size in this particular model. That is, Aden's model highlights the *regularity* of his approximated values for the step sizes of the turtle. The square demonstrates uniformity, in contrast to the actual measurements that Aden and his group collected which had variation, ranging from 12 – 32 inches (Chapter III). I wanted him to explain his thinking to his classmates in a research meeting format, because I was curious how other students would think and talk about Aden's decision to show regularity with a square (rather than the straight line) to show this pattern of motion.

In our planning conversation immediately following releasing the students to lunch period, Ms. Beck questioned the value of Aden's program: The robots did not turn, so why should the ViMAP turtle? This reflects the tension regarding the representational appropriateness of abstractions like turns, color change, and scaling. Ms. Beck and I eventually established a working compromise: students could show the same information a different way (using variables such as agent size, pen-width, turns, etc.) only *after* they had completed a ViMAP model that used step-size and graphs to adequately represent the phenomenon.

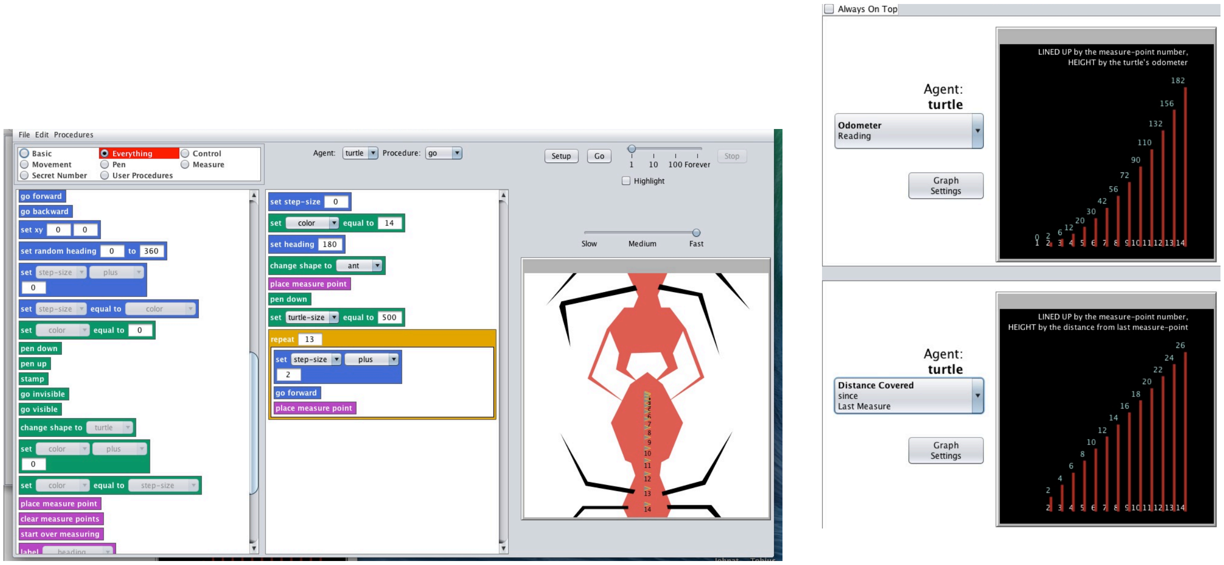
Supporting "Shaping" as an Additional Approach to Representing in ViMAP.

This graphing-centric pattern of use was driven by Ms. Beck's goals and focus on making graphs and using efficient code. With her approval, I later requested to "seed" new forms of representation and assess students' thinking about them. I planned a model evaluation day (Oct 30), based on a students' models that they had generated. In my selections, I intentionally over-represented the models that students had constructed after having made "adequate" graphs (i.e., based on the local expectations of Classroom B). That is, the models that students made when they had exhausted the standard forms and were "messing around" with the toolset. From the students' work, I generated seven sample models. The 21 students were assigned to groups of three, and I asked each group to record what they notice about what the models show. The models each were designed to portray one set of data from the robot task, which described a robot moving a constant speed for 18 seconds, and traveling a total distance of 180 inches. These programs included an exact copy of one student's model, and slight variations from four other students' programs. To those existing programs, I added changes of color (4 programs), changed the agent shape from a turtle to a car (1 program), changed agent size (1 program), and added command for the agent to "stamp" an image of their position and size at the moment that the "stamp" command is run. Additionally, I seeded a new program, which showed how far the robot moved in each second, all the way to 18 seconds. Students had uniformly partitioned the time in 3-second intervals, so this 1-second interval varied in its partitioning of time from every learners' existing programs.

In students' responses and written critiques, they focused on the match between the total distance traveled, as labeled on the last bar of the distance graph, and the total distance traveled as reported by the graphing window. Students also attended to the number of measure points used and how these were displayed in the graphs of change. While students had all used the 3-second increments of time in their constant speed models, some expressed preference for the models that showed 18 steps (for 18 seconds), corresponding to one measurement per second, rather than models in which each step was associated with a multi-second increment of time. Additionally, students identified value in the color change commands to influence the colors of both their enactment and their graphs, making graphs that corresponded to the enactment by the color of the trace of position, made by the command “pen-down”.

As the year progressed, students increasingly used these “seeded” forms and other forms which often served to minimize the exclusive focus on the graphs and allow for increased student talk about the meaning of their code and the relationship to the target phenomenon they were intending to model, which would not have been likely in the graphs-only routine that the class had become accustomed to. An example model is described in what follows:

On January 27, Timothy shared a model that created appropriate graphs to describe a pattern of increasing size of the “gaps” (that is, the distance traveled in each increment of time) of a ball in freefall. Timothy had changed his turtle shape to an ant and made the size of the turtle very large. When Timothy shared the model, the class initially liked it because it was visually shocking—most had not used such large agent size. (Timothy did not suggest scientific relevance; his choice was likely decorative.) Through discussion, we eventually focused on the heading of Timothy’s “dropping ball” which he had represented as a giant ant (transcript follows). Students asked several questions about Timothy’s model and Xander eventually asked about the use of the set heading command: “Why did you use set-heading?”



```
set step-size 0
set color equal to 14
set heading 180
change shape to ant
place measure point
pen down
set turtle-size equal to 500
repeat 13
    set step-size plus 2
    go forward
    place measure point
```

Figure 6. Timothy's model, with the ant falling with a heading of 180.

Ms. Beck: Any other questions about this model?

Xander: Why did you [Timothy] uh, put "set heading"?

Ms. Beck: Yeah, I don't even know that answer, so I have no idea.

Student: I know why.

Students: [*several students speaking at once. One student says* We had that the last time.]

Ms. Beck: Does anybody even know what set heading is? (Hands raise)

Ms. Beck: Aden, what is it?

Aden: It makes the turtle [indecipherable] forward and makes it change its heading.

Ms. Beck: Mmkay. What is the heading?

Aden: Where is it going.

Ms. Beck: Where is it going (inflection rises) Okay. And I don't know this, so this is.. Zareen, what do you think?

Zareen: It changes the way that the turtle is going, like before it was going up and now it is going down.

Ms. Beck: Okay, awesome. So if you didn't know what it was, what is an experiment you could do up on the screen?

Students: Try it!

Ms. Beck: Try it. Meaning.. what? Well, there's a number there, so there's two things we could do. Probably size, is that a good educated guess?

Students: Yes.

Ms. Beck: What would happen if we took "set heading" away?

Students: Try it!

Two students offered explanations for the meaning of "set heading," and collectively, students decided to revise the model by changing the agents' heading. At Ms. Beck's suggestion, they initially removed the "set heading" command to see what happened. They then added it back and varied the parameter (which specifies the magnitude of difference from a zero heading). On the class computer displaying Timothy's model, the class elected to change the model so that the ant's motion was downward, as shown in Figure 6. Timothy's model prompted revisiting heading as a mathematical expression of relationships among angles and how heading should (or should not) be a literal representation of the direction of motion in the phenomenon students were modeling. We returned to a tool we had invented much earlier in the year, what we called a Turtle Protractor, to aide working with headings in ViMAP.

In this episode, Ms. Beck talked openly with her students about "trying things out." She implored the students to try to make sense of new commands by using them, and that the "computer is not going to blow up if the code is broken." She valued students' uncertainity in their approach to their code and their mathematical reasoning to make sense of the relationship between the code and the actions of the agent. This mathematical reasoning was which was supported by use of the turtle compasses from earlier in the year. Ms. Beck focused on two newly emergent goals: mathematical sensemaking about the programming language (beyond graphs and loops) and students' independent, exploratory prototyping in their models. These new goals took the focus off making "correct" graphs and invited more heterogeneous use of the programming language.

Part 1 Summary: Heterogeneity in Students' Distribution of Meaning across Code, Enactment, and Graphs. In Classroom A, all students came to understand and represent motion as processes of change across time, as observed in our post-test and in student work. However, these learners varied in ways of distributing meaning across shapes, code, and graphs in their ViMAP models, and these variations stabilized into two general trajectories. Seanna's focus on graphs lead her to invent the term "period"—it was not a part of instruction—which likely served her understanding of aggregation of steps. Shenice's attention to shapes in terms of making them large enough to be visible lead her to appropriate mathematical scaling.

In Classroom B, the teacher (Ms. Beck) made a pedagogical decision to require students to make graphs and initially limited their use of shapes as expressive models. In Classroom B, Ms. Beck initially viewed ViMAP as a kind of "calculator": that is, that learners' simulations would be used to make calculations of speed and time traveled in order to find total distance and to create graphs of change. In the beginning of the year, Ms. Beck's preferences for literality (such as the "straightness" of the robots' motion) and efficiency of code worked to limit students' use of many ViMAP variables. However, Ms. Beck's understanding and use of the modeling environment progressively developed into a conception that included diversity of mathematical expression and a reduced need for literality in models, and she saw this expansion as meaningful for her goals across the curriculum.

These students and teachers' use of the computational medium followed two general and unexpected trajectories: while some viewed their programming work as efforts to produce a graph of motion, others saw a more distributed view in which the meaningfulness of the overall model was shared among turtle geometry shapes, the code, their verbal explanations, and the graphs. Ms. Beck's case in pertinent because it shows movement along this spectrum, in response to the researcher's requests, her students' growth in graphing, and her ongoing experience teaching with ViMAP.

Part 2: The Emergence of Computational Abstractions from Material Enactments

The second research question seeks to describe how the meanings of computational abstractions were bridged across the experiment, material, and other representational worlds. It is important to note that teaching and the design of instructional activities and environments plays an essential part of the student experience. In Classroom A, we (the researchers) piloted our efforts to ground computational abstractions in material enactments, and these heavily influenced the ongoing design of the graphing tools in ViMAP and our approaches to co-planning in the future, including in Classroom B.

Classroom A

In Classroom A, we realized in the constant speed work that written descriptions of the motion limited the students' ability to measure and "replay" the phenomena for their own investigations. This was not surprising and opened up design space for representing kinematic phenomena in the classroom and supporting students to investigate the phenomena. As we moved on to learning goals aimed as students' descriptions of constant acceleration, we asked students to describe what happens to the speed of a ball after it is dropped from near the ceiling of the classroom (mid-January). Students did this work in small groups, and two dominant stances emerged in the classroom talk: the ball was (1) speeding up or (2) keeping about the same speed as it fell. We also watched a video of a ball falling with a stroboscope flashing on it, so that the positions of the ball at each consecutive interval of time were shown. It is important to note that this video was pulled from YouTube, not made in the classroom, and students needed support to unpack the many unfamiliar features of the video. Since we wanted students to attend to the distances in between successive positions of the ball, Amy used the video to compile all the visible positions of the ball into one image, and printed them on strips of paper (Figure 7). The following day, students attempted to create folded representations the speed of the ball, similar to those shown in Figure 8. We intended this process of *folding* acceleration—that is, taking the total distance traveled and breaking it into segments traveled in same-size increments of time—to support students' understanding of discretized (bar graph) representations of speed. When the paper is positioned to stand on its edge, the product was reminiscent of the shapes in which step-size was constantly changing that we had made in the earlier unit that focused on geometry and shape drawing. Working with those folded strips required students to understand that the small circles in Figure 7 are outlines of multiple photographic images of the ball, taken as the camera flashed and at a regular interval of time. However, the bold horizontal lines were visually very salient in these images, and many students wanted to focus on those lines in relation to the position of the ball. I illustrate these representational challenges with the following episode from Shenice and Imani:

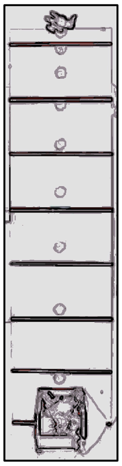


Figure 7. Students were given strips of paper with images of the positions of the ball, as shown.

Figure 8. Shenice's folded representation of the distance traveled in each step.

In this excerpt of interaction, Shenice and Imani were working on their folded representations when Amy asked them to explain their models. Imani first described folding in "every space", creating approximately equally spaced sections. Shenice offered an alternative suggestion: “You could fold it in-between the balls.”

Amy asked Shenice to explain her model:

Researcher (Amy): [Shenice], can you tell me what this means? (laughing) So, like, if I..how am I supposed to look at it?

Shenice: Well, I don't know. I just folded, tried to fold in-between each ball.

Amy: Okay-

Shenice: *(motions to all of the positions of the ball)* All of the balls going down, like this space is just free, because it has nothing. Because the ball could have fell anywhere in this space.

[Some sensemaking about folding between the balls removed for space]

Amy: Okay, so the spaces between your folds, if we look at it like this (arranges paper so that she and Shenice see it from a birds-eye view) I see short, a little bit longer, a little bit longer, a little bit longer. So how is that showing something?

Shenice: It start off short, but it gets longer. It comes closer together and then separates. Cause, the faster it goes, the more it spreads.

Amy: So are you saying that it starts close together, then it gets further, then it gets closer again? Is that what you are saying?

Shenice: No. It's starts closer, but as it, it starts to go faster, as it goes faster it starts to separate.

Analysis. There are several representational challenges in this interaction. The video that we selected included the background of bold black lines, which were retained in the printed inscriptions. Imani and Shenice work to sort out what aspects of this representation are most important to the question about the speed of the ball: Imani wants to fold "in every space," and ignore the positions of the ball. Shenice attends to the space between the bold black lines and the images of the positions of the ball. She describes that she is "tr[ying] to fold in-between the balls." She has also noticed that one space between the black horizontal lines does not have a position, and is uncertain how to fold this spaces: "this space is just free, because it has nothing."

In spite of these representational challenges, Shenice has placed her folds immediately above or beneath each of the images of the balls, and she describes the pattern of change in the distances between the folds near the end of this exchange: "It's starts closer, but as it, it starts to go faster, as it goes faster it starts to separate." However, the word "separate" indicates a possible slippage between the static image and the dynamic process of change that it is intended to represent.

In the next class meeting, we agreed with the teacher to refocus the students on the positions of the ball. As they flattened their folded representations from our previous meeting, they measured the "space between" the positions of each image of the ball and the one immediately beneath it, as shown in Figure 9.

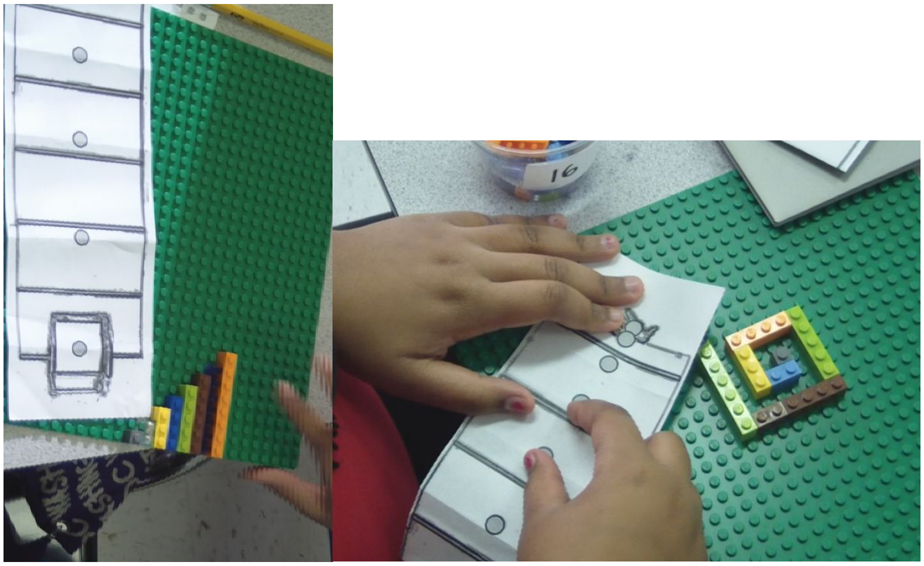


Figure 9. Examples of two students' graph-like Lego representations of changing speed, designed as a way to connect the phenomenon to measurement. For each, students described the correspondence between the positions of the ball and their representations.

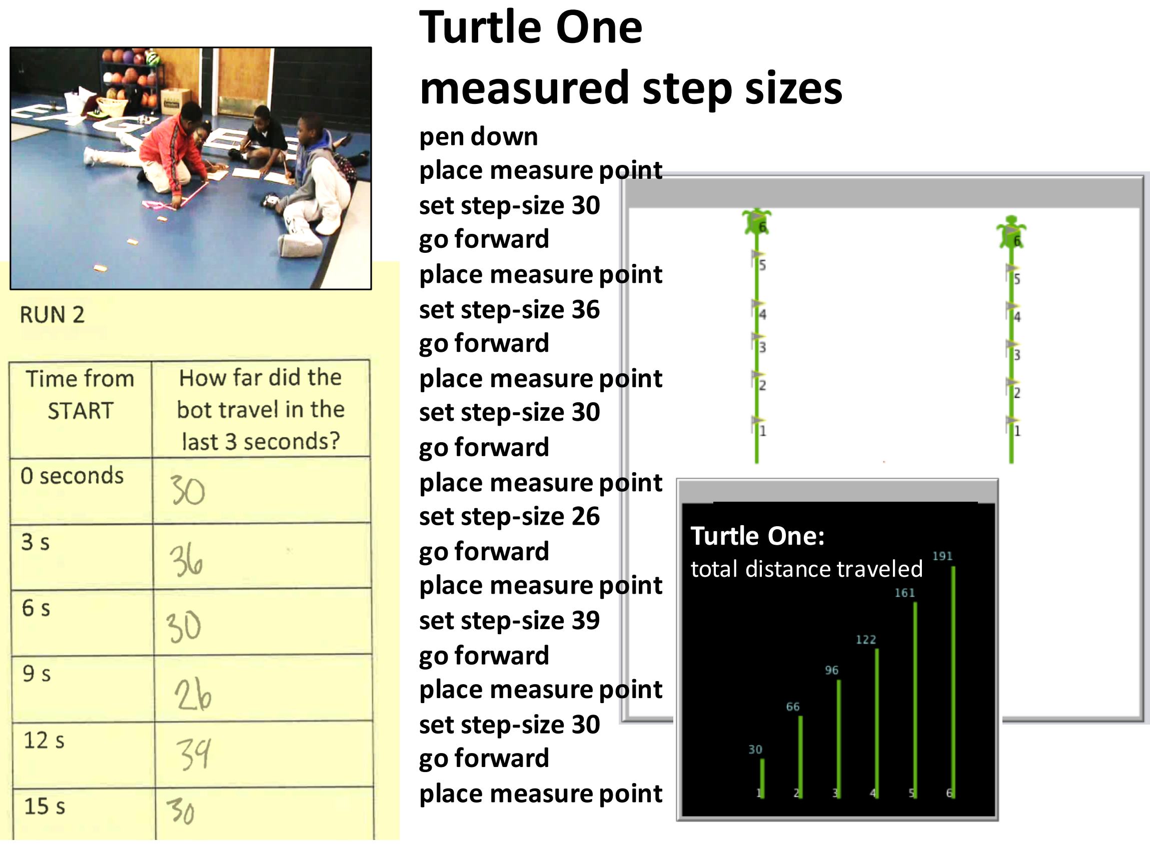
During this activity, students came to use a shared word for the spaces between the positions: “gaps”. Within this sequence, students then translated the gaps to discrete distances, that can be compared and arranged in bar graphs, as is shown by Imani’s right thumb and index finger in Figure 9. This form of symbolization was bridged to the ViMAP representations. In our computational modeling environment, paper folding, and the Lego activities, the speed for each gap was represented with discrete bars (instead of a continuous line graph). Students used the height of the Lego bars (that is, the height of distance traveled in each moment) to describe that the distance between the positions was steadily increasing, our definition of constant acceleration in this classroom.

Classroom B

While Classroom A students invented the word “gap” to describe the spaces between positions, it continued to be useful to the research team in all future iterations of this work. As mentioned in Chapter III, Ms. Beck (Classroom B) insisted that we plan tangible, “concrete” episodes of motion for students to model. The focus of this section is on how these linked forms of representation supported students’ use of abstractions that were grounded in their experiences of the phenomenon. In the constant speed work, we introduced paper flags as a marker of position. Initially, these were 4-inch paper adhesive “flags,” which students placed along the path of the robots as a physical form of the dot-trace representation, a long-supported LOGO-based representation of motion in introductory physics (Hammer, Sherin & Kolpakowski, 1991; Sherin, 2000; Schwarz & White, 2005; Sengupta & Farris, 2012).

In ViMAP, the marks of position were symbolized as "measure points" with the symbol ⚑ and the command "place measure point". Rather than repeat challenges similar to those with the stroboscopic video in Classroom A, we designed instruction so that students created their own videos of motion. Students did not differentiate in their language between these physical and computational symbols, and almost always referred the them as "flags" or "measure flags", in spite of the language in the command "place measure point".

Figure 10 shows one group's work in the material and tangible measurements in the gymnasium. They used the data sheet to record distances between flags, which they had attempted to place at 3-second intervals as a robot moved across the floor. To preserve space, I have typed the command blocks one group member's computational model, which asks the turtle to "go forward" by each measured step size, and "place measure point" as the commands run. In this case, the design of the physical activity and the features of the computational environment are closely linked, supporting students to move between the physical, lived space and their own ViMAP programs.



RUN 2

Time from START	How far did the bot travel in the last 3 seconds?
0 seconds	30
3 s	36
6 s	30
9 s	26
12 s	39
15 s	30

Turtle One measured step sizes

```
pen down
place measure point
set step-size 30
go forward
place measure point
set step-size 36
go forward
place measure point
set step-size 30
go forward
place measure point
set step-size 26
go forward
place measure point
set step-size 39
go forward
place measure point
set step-size 30
go forward
place measure point
```

Turtle One:
total distance traveled

Figure 10. The material enactment, data sheet, and computational models in the constant speed work.

As this work progressed, flags as indicators of a position in time became fluently used an understood among class members, and Ms. Beck and I continued to make “flags” available in all physical investigations of motion. On December 2, student small groups were designing investigations in an inclined plane task. The central problem was, “How can you convince [the principal] that the marble is changing speed? Students’ strategies were varied, but all included making and comparing measurements of time from the beginning position to one or more “flag[s]”, and to the end of the ramp. Our work with ramps and videos called for finer (i.e., smaller scale) measurements, so we used smaller (approximately 1 inch) adhesive flags with arrow-shaped ends.

The following excerpt is taken from Ms. Beck’s questioning of a small group as they set up their “experiment” to demonstrate evidence that the speed of the marble was increasing (Figure 11). They position a tape measure along the ramp. They have a small adhesive measure flag labeled “END,” which they place at the midpoint. Bolded text indicates a deictic gestures (pointing) that co-occur with talk.

Ms. Beck: Uh, real quick set-up question. Why are you guys measuring right now, just out of curiosity?

Aden: to do the half way point

Ms. Beck: Ooooo. I like that. How did you determine what the half way point was?

Nylah: Because..we know that 2 divided into 48 is 24. So when we measure it we know that the..halfway point.

[Ms. Beck questions other groups about their measurement strategy, then re-focuses attention on the presenting group.]

Ms. Beck: Okay. So first, you all tell me how you all are setting up your experiment..um… Aden, tell me a little bit and then I’ll go to each one of you guys. What are you doing today for us today?

Aden: First, when we.. the ball gets right **here..** [points to the midpoint “END” flag] Theo will start the timer, and then when it gets to the edge [points to end of 48” length] he will stop the timer. Then the next one, we are going to do one when [points to the beginning of the ramp] it gets to the **end** [points to the midpoint flag] when it starts from **here** [points to the beginning of the ramp]..he starts the timer and when it gets right **here** [midpoint flag] he stops it.

Ms. Beck: So you are going to take two different measurements of time, correct?

[Members of group nod.]

Ms. Beck: Mmkay. Anything you want to add, Nyla? I mean what are you guys trying to prove? Are you trying to prove the same thing?

Nylah: We're trying to prove that the ball will accelerate or increase its speed.

Figure 11. Experimental setup. Small red flag is at the midpoint labeled "END."

As the students collect their measurements, Ms. Beck organized these in a diagram on the whiteboard (Figure 12). The students release the marble from the top of the ramp, and start the timer as the marble reaches the midpoint, and report the time for this segment as 1.4 seconds. They next release the marble from the top of the ramp and stop the timer at the midpoint, and the report the time of 2.4 seconds.

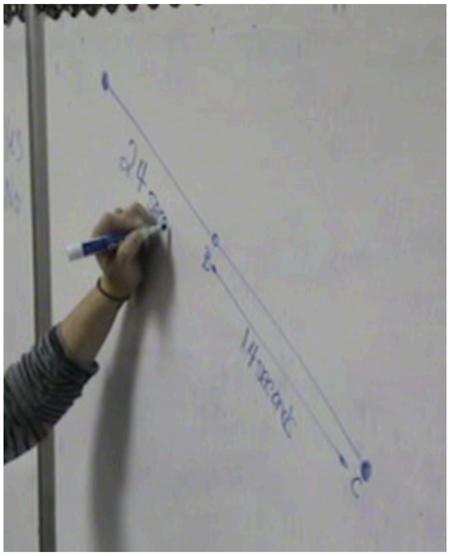
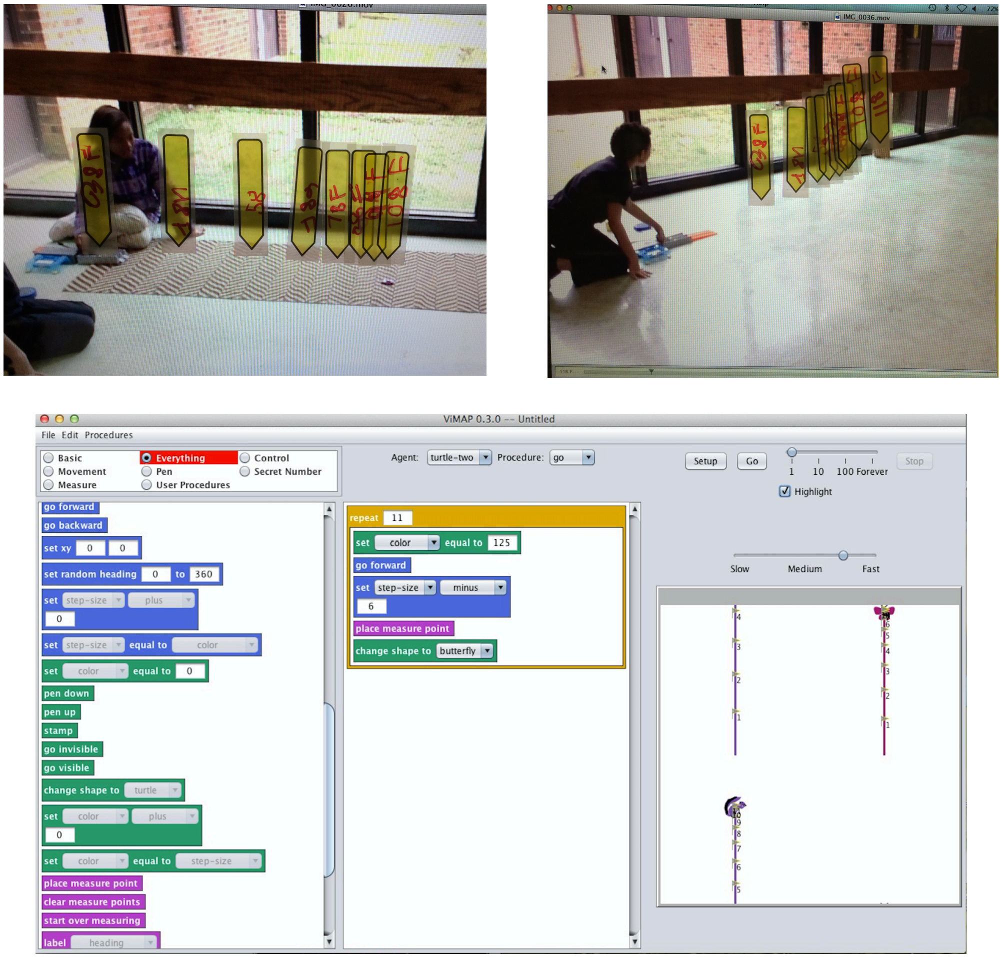


Figure 12. Ms. Beck helps to organize features the students' investigation on the whiteboard with a diagram.

Ms. Beck's diagram uses points to indicate the places that the group members have marked as significant, and she marks up this diagram as they conduct their experiment. She represents the beginning, end, and the "end" flag at the midpoint as points, and labels them A (not yet labeled in Figure 12), B, and C.

As students' kinematic work continued, measure flags eventually also became labeled with frames of video—therefore associating a position with an instant in time. In the friction unit (Figure 12), students made comparisons of total time traveled based on these marked flags. This comparison created a contentious discussion: students were comparing the time in which a toy car slows down on two surfaces: on a rug (Figure 13, left) and on the tile floor (Figure 13, right). Due to differences in the camera angle of their setups, students perceived the total distance traveled by the car on the rug to be *further* than the car on the tile—that is, from the still images, the total distance from (a) the flag that is furthest to the left and (b) furthest to the right was a greater distance on the rug in comparison to the tile. This did not match their experience when running their experiments and collecting the video, nor what they expected to happen.



```
repeat 11
    set color equal to 125
    go forward
    set step-size minus 6
    place measure point
    change shape to butterfly
```

Figure 13. Left: Car path from launcher on a rug with measurements every 10 frames. Right: Car path from launcher on tile hallway, measurements every 10 frames. Bottom: ViMAP model in which a student describes the car on the tile floor (squirrel) and the car on the on the rug (butterfly).

The evidence that eventually resolved the contentious issue within class discussion came in the form of subtracting the time required for the motion to stop, assuming that the force initially applied is approximately equal. These numbers came from the frame numbers that students had marked on the flags, but a researcher first suggested that these could be used as a form of evidence. Students then used the frame numbers to calculate the total time of motion on each surface: 118 frames – 38 frames = 80 frames on hard tile floor, versus 108 frames – 38 frames = 70 frames on the rug. While the difference of ten frames is quite minor, the students were satisfied that they had produced evidence that it takes more time to slow down to a stop on the tile.

Figure 13 (bottom) also shows a student's model of the comparison, in which computational measure flags are used to symbolize physical flags marking position in video-recorded motion, and the variable for decreasing step size (set step-size minus *x*) computationally represents the coefficient of friction on that surface.

Part 2 Summary. Symbolizing positions, time, and "gaps" computationally. In Classroom A, students first invented the term "gaps" to refer the distance between successive positions of an object. In ViMAP, this concept was represented with "step-size," —the distance the programmable agent moved forward with each execution of the "go forward" command. Students were able to become fluent in the intended meaning of step-size through material investigations of free fall, folding activities, and the creation of Lego graphs to discretize the total distance into gaps, which supported students' computational work.

In Classroom B, the researcher (Amy) and the classroom teacher (Ms. Beck) agreed that material enactments were essential for students' learning about motion, initially expressed by the teacher as the need to "make things concrete." In this classroom, we iteratively designed episodes of motion for students to enact or to design investigations of, including constant motion of robots, acceleration down an inclined place, and comparisons of forces of friction on different surfaces. In each activity, students used physical "measurement flags." The meanings of these flags became enmeshed and indistinguishable from the command for placing "measure points" in ViMAP, and we believe that this was a key support for students' level of expression in their own descriptions of the motion events “in-the-world”, in drawings and diagrams, and computationally as commands to be carried out by programmable agents.

Discussion

Papert’s initial critique of technocentrism (1987) is nearing its fourth decade, and many learning scientists are now thinking carefully about the relations among learning, classrooms, and technologies in ways that avoid essentializing learning in terms of the technologies themselves. However, as Sengupta, Dickes, and Farris (2018) have argued, technocentrism continues to be a persistent concern in the domain of children’s educational computing, where researchers have focused less on the complexity of the experience of computing in different contexts, and instead have focused more on assessment of computational thinking.

In this paper, we have put forward a different view of children’s computing in science classrooms, where the focus is on how computing involves not only programming and using computational abstractions within the programming language, but also grappling with the physical and material world in order to create and re-create scientific models. Central to this image of computing is the notion of heterogeneity, as evident in how students came to distribute their intended meanings across different elements of the representational infrastructure. Our analysis illustrates some trajectories of students’ and teachers’ modeling in such classrooms, and furthermore, highlights how the heterogeneity in the representational infrastructure supported development of both conceptual understanding of motion as a process of change over time, as well as offered productive pathways for interpreting and deploying computational abstractions.

So, one might then ask the following question: What value does computing add to science “as usual”? Latour’s (1999) studies of scientists engaged in “doing” science offers an unusual insight here. He showed that the creation of scientific knowledge involves a long cascade of representational transformations where scientists iteratively create, share, and modify mathematical representations of the relevant phenomena. At each stage of this representational cascade, he argued, the scientific ideas (or objects) become "durable". That is, in science, a thing (or an idea) "can remain more durable and can be transported farther and more quickly if it continues to undergo transformation at each stage of this long cascade" (Latour, 1999; p. 58). Similarly, in our work, children's creation of durable and transportable descriptions of the speed of moving objects requires them to think about and inscribe motion in terms of relationships of displacement and time. Specific ways of describing (and inscribing) properties of kinematic phenomena emerged as key ideas that were durable and frequently reused as children moved across representational means—"gaps" and "measure flags." These conceptualizations moved back and forth across the children's computational, physical, and paper-based representations. Similarly, in Enyedy and colleagues' motion learning environments (2012), motion sensing technologies were used to transform learners' movements around a room to a microworld. In this present study, children measured motion and re-described it in a way that the computer can understand though programming.

We see these kinds of transformations in scientific reference as deeply related to Pickering's (1995) "mangle". The representations do not merely point to something beyond themselves, they make that reference available for further manipulation and prodding, and the representations themselves then become subject to further re-description and specification. At each stage, references to the target are changed, but retain an intact and fundamental meaning across the heterogeneous forms of models. These forms were also durable to heterogeneity in the target phenomena (in our case, descriptions of different kinds of motion phenomena) indicating their centrality to ideas that are of disciplinary importance, rather than superficial features.

Our work has implications for designing computational modeling platforms for K-12 science classrooms in terms of the expressivity and heterogeneity of the representational infrastructure in such platforms. Our work offers insights both in terms of how we can design better software systems, as well as for the design of learning activities, as we explain next.

Along the first dimension, our findings suggest that in order to support computational modeling in the context of kinematics, agent-based programming should be complemented by Cartesian graphing functionalities. Commanding the agent's behavior on screen offers an opportunity to "dive in", whereas graphing offers an opportunity to "step out" (Ackermann, 2012). Furthermore, our analysis of the 5th grade students' work also suggests that students can use these functionalities in different ways in terms of representing relevant variables more explicitly either using graphs or simulations or both. We found that heterogeneous use of this representational infrastructure afforded important opportunities for students to participate in the modeling processes of selection, design, and critique.

Along the dimension of designing learning activities, throughout the analysis, we have described pedagogical decisions alongside descriptions of learning. We found that agent-based programming became reframed by teachers as mathematically modeling the relevant scientific phenomenon. This corroborates other studies where we have also found that elementary, middle and school teachers prefer to reframe computational programming, in particular, agent-based programming as modeling and mathematization, with a particular emphasis on *designing units of measures* (Dickes et al., 2016; Sengupta, Brown, Rushton & Shanahan, 2018). These phenomenological re-framings are essential for grounding computing in the science classroom in absence of researchers, and can help us understand how computational modeling is taken up in a manner that is also relevant to scientific practices, both epistemic and representational.

Overall, we believe that our work illustrates a phenomenological view of re-imagining computing for K12 science requires viewing computing and scientific modeling as complex, heterogeneous, and grounded in practice. For the 4th and 5th grade science learners in our study, the integration of an agent-based programming and modeling environment for learning kinematics helped them grapple productively with the complexity and uncertainty of their *experience* of scientific phenomena, which in turn increased the demand for computational abstractions, and at the same time, grounded these abstractions meaningfully in the children's embodied and physical modeling experiences. This in turn created contexts where students did more than learn programming: they learned about the inseparable interdependence of modelers and their materials for making meaning of the world, akin to Pickering's (1995) notion of the scientific mangle of practice.

References

Abelson, H., & diSessa, A. A. (1981). *Turtle geometry: The computer as a medium for exploring mathematics*. MIT press.

Ackermann, E. (2012). Perspective-taking and object construction: Two keys to learning. In *Constructionism in Practice* (pp. 39-50). Routledge.

Bers, M. U., Flannery, L., Kazakoff, E. R., & Sullivan, A. (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers & Education*, *72*, 145-157.

Bucciarelli, L. L. (1994). *Designing engineers*. MIT Press.

Chi, M. T. (2005). Commonsense conceptions of emergent processes: Why some misconceptions are robust. *The journal of the learning sciences*, *14*(2), 161-199.

Cobb, P., Confrey, J., diSessa, A., Lehrer, R., & Schauble, L. (2003). Design experiments in educational research. *Educational Researcher*, *32*(1), 9-13.

Danish, J. A. (2014). Applying an activity theory lens to designing instruction for learning about the structure, behavior, and function of a honeybee system. *Journal of the Learning Sciences*, *23*(2), 100-148.

Dickes, A. C., Sengupta, P., Farris, A. V., & Basu, S. (2016). Development of mechanistic reasoning and multi-level explanations of ecology in third grade using agent-based models. *Science Education, 100*(4), 734–776.

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

diSessa, A. A. (2001). *Changing minds: Computers, learning, and literacy*. The MIT Press.

diSessa, A. A. (2004). Metarepresentation: Native competence and targets for instruction. *Cognition & Instruction, 22,* 293-331.

diSessa, A. A., & Abelson, H. (1986). Boxer: A reconstructible computational medium. *Communications of the ACM*, *29*(9), 859-868.

Duschl, R. (2008). Science education in three-part harmony: Balancing conceptual, epistemic, and social learning goals. *Review of Research in Education*, *32*(1), 268–291.

Dykstra, D. I., Jr. & Sweet, D. R. (2009). Conceptual development about motion and force in elementary and middle school students. *American Journal of Physics, 77*(5), 468-476.

Elby, A. (2000). What students' learning of representations tells us about constructivism. *Journal of Mathematical Behavior*, 19, 481-502.

Enyedy, N., Danish, J. A., Delacruz, G., & Kumar, M. (2012). Learning physics through play in an augmented reality environment. *International journal of computer-supported collaborative learning*, *7*(3), 347-378.

Farris, A. V. & Sengupta, P. (2014). Perspectival computational thinking for learning physics: A case study of collaborative computational doing. *Proceedings of the 11th International Conference of the Learning Sciences*. (pp. 1102-1106). Boulder, CO: International Society of the Learning Sciences.

Ford, M. J. (2003). Representing and meaning in history and in classrooms: Developing symbols and conceptual organizations of free-fall motion. *Science & Education*, *12*(1), 1-25.

Giere, R. N. (1984). *Understanding scientific reasoning (2nd ed.).* New York: Holt, Rinehart, and Winston.

Giere, R. N. (1988) *Explaining Science: A cognitive approach.* Chicago: University of Chicago Press.

Glaser, B. G. (1965). The constant comparative method of qualitative analysis. *Social Problems, 12*(4), 436-445. http://dx.doi.org/10.1525/sp.1965.12.4.03a00070

Glaser, B., & Strauss, A. (1967). The discovery of grounded theory: Strategies for qualitative research. London, UK: Weidenfeld & Nicholson.

Greca, I. M., Seoane, E., & Arriassecq, I. (2014). Epistemological issues concerning computer simulations in science and their implications for science education. *Science & Education*, *23*(4), 897-921.

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher*, *42*(1), 38-43.

Gupta, A., Hammer, D., & Redish, E. F. (2010). The case for dynamic models of learners' ontologies in physics. *The Journal of the Learning Sciences*, *19*(3), 285-321.

Guzdial, M. (1994). Software-Realized Scaffolding to Facilitate Programming for Science Learning. *Interactive Learning Environments. 4*(1). 1-44.

Hall, R. (1996). Representation as shared activity: Situated cognition and Dewey's cartography of experience. *The Journal of the Learning Sciences*, *5*(3), 209-238.

Halloun I. A., & Hestenes, D. (1985). The initial knowledge state of college physics students. *American Journal of Physics, 53*(11), 1043-1056.

Hammer, D. (1996). More than misconceptions: Multiple perspectives on student knowledge and reasoning, and an appropriate role for education research. *American Journal of Physics*, *64*(10), 1316-1325.

Hammer, D., Sherin, B., & Kolpakowski, T. (1991). Inventing graphing: Meta-representational expertise in children. *Journal of Mathematical Behavior*, *10*(2), 117-160.

Harel, I. and Papert, S. 1990. Software design as a learning environment. *Interactive Learning Environments*, *1*(1): 1–32.

Hestenes, D. (1993). Modeling is the name of the game. *A presentation at the NSF Modeling Conference* (Feb. 1993).

Hestenes, D., Wells, M., & Swackhamer, G. (1992). Force concept inventory. *The Physics Teacher*, *30*(3), 141-158.

Kafai, Y., & Ching, C. C. (1998, December). Talking science through design: Children's science discourse within software design activities. In *International Conference of the Learning Sciences, Georgia Tech University, Atlanta, GA*.

Klahr, D., & Carver, S. M. (1988). Cognitive objectives in a LOGO debugging curriculum: Instruction, learning, and transfer. *Cognitive Psychology, 20*(3), 362–404. https://doi.org/10.1016/0010- 0285(88)90004-7.

Kolodner, J. L., Camp, P. J., Crismond, D., Fasse, B., Gray, J., Holbrook, J., ... & Ryan, M. (2003). Problem-based learning meets case-based reasoning in the middle-school science classroom: Putting learning by design (tm) into practice. *The Journal of the Learning Sciences*, *12*(4), 495-547.

Larkin, J., McDermott, J., Simon, D. P., & Simon, H. A. (1980). Expert and novice performance in solving physics problems. *Science*, *208*(4450), 1335-1342.

Latour, B. (1999). *Pandora's Hope*. Cambridge, MA: Harvard University Press.

Lehrer, R., & Romberg, T. (1996). Exploring children's data modeling. *Cognition and Instruction*, *14*(1), 69-108.

Lehrer, R., & Schauble, L. (2006). Scientific thinking and science literacy. In D. Kuhn and R. Siegler (Eds). *Handbook of child psychology (6th ed.) vol. 2: Cognition, perception, and language* (153-196). Hoboken, New Jersey: John Wiley & Sons, Inc.

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

Lehrer, R., Schauble, L., Carpenter, S., & Penner, D. (2000). The inter-related development of inscriptions and conceptual understanding. *Symbolizing and communicating in* *mathematics classrooms: Perspectives on discourse, tools, and instructional design*, 325-360.

Leinhardt, G., Zaslavsky, O., & Stein, M. M. (1990). Functions, graphs, and graphing: Tasks, Learning and Teaching. *Review of Educational Research*, 60, 1-64.

Levy, S. T., & Wilensky, U. (2009). Students' learning with the Connected Chemistry (CC1) curriculum: navigating the complexities of the particulate world. *Journal of Science Education and Technology*, *18*(3), 243-254.

Lincoln, Y. S., & Guba, E. G. (1985). *Naturalistic inquiry*. Beverly Hills, CA: Sage.

Manz, E., & Suárez, E. (2018). Supporting teachers to negotiate uncertainty for science, students, and teaching. *Science Education*. Advance online publication, 1-25. doi:10.1002/sce.21343

McCloskey, M. (1983). Naive theories of motion. In D. Gentner & A. Stevens (Eds.), *Mental Models* (pp. 299-324). Hillsdale, NJ: Lawrence Erlbaum.

McDermott, L. C., Rosenquist, M. L., & van Zee, E. H. (1987). Student difficulties in connecting graphs and physics: Examples from kinematics. *American Journal of Physics, 55*, 505-513.

Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook.* Sage.

National Research Council (NRC, 2007). *Taking science to school: Learning and teaching science in grades K-8*. National Academies Press.

National Research Council (NRC, 2012). A framework for K–12 science education: Practices, crosscutting concepts, and core ideas. Washington, DC: National Academies Press.

National Research Council. (NRC, 2010). *Report of a Workshop on the Scope and Nature of Computational Thinking*. Washington, DC: The National Academies Press. https://doi.org/10.17226/12840.

Nersessian, N. (2008). Creating scientific concepts. MIT Press.

Norman, D.A. 1993. *Things that make us smart: Defending human attributes in the age of the machine*, Reading, MA: Addison-Wesley.

Oren, T. 1990. "Designing a new medium". In B. Laurel (Ed.) *The art of human-computer interface design*. Reading, MA: Addison-Wesley, pp. 467–479.

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

Papert, S. (1987). Information technology and education: Computer criticism vs. technocentric thinking. *Educational researcher*, *16*(1), 22-30.

Pea, R. D. (1986). Language-independent conceptual "bugs" in novice programming. *Journal of Educational Computing Research*, *2*(1), 25-36.

Pea, R.D. and Kurland, D.M. 1986. "On the cognitive effects of learning computer programming". In *Mirrors of minds*, Edited by: Pea, R.D. and Sheingold, K. Norwood, NJ: Ablex.

Penner, D. E., Lehrer, R., & Schauble, L. (1998). From physical models to biomechanics: A design-based modeling approach. *Journal of the Learning Sciences*, *7*(3-4), 429-449.

Pickering, A. (1995). *The mangle of practice: Time, agency, and science*. University of Chicago Press.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., ... & Kafai, Y. (2009). Scratch: programming for all. *Communications of the ACM*, *52*(11), 60-67.

Roschelle, J., & Teasley, S. D. (1995). The construction of shared knowledge in collaborative problem solving. In *Computer supported collaborative learning* (pp. 69-97). Springer, Berlin, Heidelberg.

Rosebery, A. S., & Rubin, A. (1989). Reasoning under uncertainty: Developing statistical reasoning. Journal of Mathematical Behavior, 8, 205 – 219.

Roth, W.-M. (1996). Art and artifact of children's designing: A situated cognition perspective. *Journal of the Learning Sciences, 5*, 129–166.

Sandoval, W. (2014). Conjecture mapping: An approach to systematic educational design research. *Journal of the Learning Sciences*, *23*(1), 18-36.

Schmidt, D. C. (2006). Guest editor's introduction: Model-driven engineering. Computer, 39(2), 25–31.

Schwarz, C. V., & White, B. Y. (2005). Metamodeling knowledge: Developing students' understanding of scientific modeling. *Cognition and Instruction*, *23*(2), 165-205.

Sengupta, P, & Farris, A. V. (2012). Learning kinematics in elementary grades using agent-based computational modeling: A visual programming approach. Proceedings of the 11th International Conference on Interaction Design & Children (pp. 78-87). Bremen, Germany.

Sengupta, P., Brown, B., Rushton, K., & Shanahan, M. C. (2018) Reframing Coding as "Mathematization" in the K12 Classroom: Views from Teacher Professional Learning.

Sengupta, P., Dickes, A. C., Farris, A. V., Karan, A., Martin, K., & Wright, M. (2015). Programming in K12 Science Classrooms. *Communications of the Association of Computing Machinery, 58*(11), 33 - 35.

Sengupta, P., Dickes, A., & Farris, A. V. (2018). Toward a Phenomenology of Computational Thinking in K-12 STEM. In M. S. Khine (Ed)., *Computational thinking in STEM disciplines: Foundations and research highlights.* Springer.

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, 17*(1-2), 23 - 42.

Sengupta, P., Kinnebrew, J., Basu, S., Biswas, G., and Clark, D. (2013). Integrating Computational Thinking with K12 Science Education Using Agent-Based Modeling: A Theoretical Framework. *Education & Information Technologies*, Vol. 18, 351 – 380.

Sengupta, P., Krinks, K. D., & Clark, D. B. (2015). Learning to deflect: Conceptual change in physics during digital game play. *Journal of the Learning Sciences*, *24*(4), 638-674.

Sherin, B. L. (2001). How students invent representations of motion: A genetic account. *Journal of Mathematical Behavior* (19) 399-441.

Sherin, B., diSessa, A. A., & Hammer, D. (1993). Dynaturtle revisited: Learning physics through collaborative design of a computer model. *Interactive Learning Environments*, *3*(2), 91-118.

Simon, H. A. (1969). *The sciences of the artificial*. MIT Press.

Slotta, J. D., & Chi, M. T. (2006). Helping students understand challenging topics in science through ontology training. *Cognition and instruction*, *24*(2), 261-289.

Slotta, J. D., Chi, M. T., & Joram, E. (1995). Assessing students' misclassifications of physics concepts: An ontological basis for conceptual change. *Cognition and Instruction*, *13*(3), 373-400.

Smith III, J. P., disessa, A. A., & Roschelle, J. (1994). Misconceptions reconceived: A constructivist analysis of knowledge in transition. *The journal of the learning sciences*, *3*(2), 115-163.

Soloway, E. (1993). Should We Teach Students to Program? *Communications of the ACM, 36*(10): 21-24.

Trowbridge, D. E., & McDermott, L. C. (1980). Investigation of student understanding of the concept of velocity in one dimension. *American journal of Physics*, *48*(12), 1020-1028.

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

Wilkerson, M. H., & Wilensky, U. J. (2015). Patterns, probabilities, and people: Making sense of quantitative change in complex systems. *Journal of the Learning Sciences*, *24*(2), 204-251.

Wing, J. (2011). Computational thinking—What and why?. *The Link: News from the School of Computer Science. 6,* 20 - 23. Retrieved from https://www.cs.cmu.edu/sites/default/files/11-399_The_Link_Newsletter-3.pdf

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33-35.

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical transactions of the royal society of London, A: mathematical, physical and engineering sciences*, *366*(1881), 3717-3725.