Stories From the Field: Locating and Cultivating Computational Thinking in Spaces of Learning

Michelle Hoda Wilkerson[1], Cynthia M. D'Angelo[2], and Breanne K. Litts[3]

[1]University of California, Berkeley, CA, USA;

[2]University of Illinois at Urbana-Champaign, IL, USA;

[3]Utah State University, Logan, UT, USA

Correspondence concerning this article should be addressed to: Michelle Wilkerson, Graduate School of Education, University of California, Berkeley, 2121 Berkeley Way West, Berkeley, CA, 94609. email: mwilkers@berkeley.edu

Stories From the Field: Locating and Cultivating Computational Thinking in Spaces of Learning

Abstract. There is considerable debate and ambiguity around what constitutes "computational thinking" (CT). In contrast to Computer Science which is generally treated as a distinct field of study, CT as a construct highlights the integral relationship between computing and other fields. Many recent efforts seek to map computational thinking by making high-level connections to other school disciplines. We argue that while these efforts may help identify specific curricular areas in which computing is likely to take place, they do not sufficiently capture the specificity and dynamism that is characteristic of meaningful computational integration. Worse, they exclude generative examples of computing integration that exist outside of the traditional STEM context or researcher-led efforts. In this special issue, we offer a counterproposal to one-size-fits-all frameworks of CT, exploring in detail how local, emergent definitions of CT develop across a diversity of spaces of learning. Reflecting on these examples can help researchers and educators alike cultivate an awareness of the ways in which learners and educators leverage computing to think, create, and participate across a variety of spaces.

Keywords: computational thinking; situated computing; computational literacy; computational integration; interdisciplinary computing

Stories From the Field: Locating and Cultivating Computational Thinking in Spaces of Learning

There is considerable debate and ambiguity around what constitutes "computational thinking" (CT). In contrast to Computer Science which is generally treated as a distinct field of study, CT as a construct highlights the integral relationships between computing and other fields. While researchers have explored these relationships for decades (diSessa, 2001; Noss, 1987; Papert & Solomon, 1971; Soloway, Lochhead, Clement, 1982; Sutherland, 1994), these efforts have been accelerated by national initiatives (The White House, 2016) and funding opportunities (e.g., the National Science Foundation's STEM+Computing program).

In response to these incentives, many frameworks, taxonomies, and lists identify potential alignments between computational skills and existing school curricula (Repenning, Webb, & Ioannidou, 2010; Weintrop et al., 2015). These early efforts have been helpful for making initial strides toward understanding where and how CT might fit into the traditional K-12 classroom landscape. However, these frameworks often do not offer much in terms of demonstrating how CT practices and skills might be enacted by teachers and students in classrooms, or how they might be enacted and adapted across curricular domains or developmental trajectories. While more specific examples have since been provided in later documents, they are usually hypothetical or limited to a single curricular unit or project context (Barr & Stevenson, 2011; Wilensky, Brady, & Horn 2014; Lee et al., 2011).

In this special issue, we attempt to bring a learning sciences lens to this proliferation of top-down CT frameworks by honoring, and taking seriously, how the complexity of everyday spaces of learning shapes what counts, and what should be counted, as "computational thinking." Now is an appropriate time to begin to explore such complexity. The most recent spate of CT projects have been (necessarily, given the novelty of the field) abstract and agenda-driven — designed to infuse particular visions, tools, and approaches to computational thinking into specific educational spaces. However, the field is now at a point of maturity that behooves us to step back and look at the many examples of computational thinking that have emerged from these efforts, with attention to how they are embedded within and shaped by the daily work of educators and learners. This special issue presents a first step—and a plea to our colleagues—to pause, reflect on, and share how computational thinking unfolds within the day-to-day texts and practices of our collective work; to understand computational thinking as an emergent and dynamic phenomenon rather than a funding-related buzzword or abstract curricular goal.

A Need for Examples of Computational Thinking from the Field

Attending to empirical examples of successful computing integration across projects is needed for a number of reasons. It is well-established that defining CT in terms of static, hypothesized curricular performances risks severely limiting what counts as computational thinking and who is likely to be ratified or identified as participating in computing practices (Turkle & Papert, 1992; Margolis & Fisher, 2002; Richard, 2017). It also limits what kind of theoretical development is possible—consider how studies of learning grounded in seemingly mundane details of shopping (Lave, 1988), family dinners (Ochs, Taylor, Rudolph, & Smith, 1992), the daily work of science laboratories (Pickering, 1995), and moment-to-moment interactions in classrooms (Yackel & Cobb, 1996; Engle & Conant, 2002) have fundamentally changed how we define, recognize, and support mathematical and scientific thinking and learning across the curriculum.

Exploring grounded examples of computational thinking across contexts can also allow us to more seriously attend to computing as a practice*,* motivated by the needs that are most salient to learners themselves in the moment. Existing taxonomies primarily draw from descriptions of how computing is used in professional fields (Weintrop et al., 2014), or enacted on a curricular level (Lee et al., 2011), with less attention to students' emergent pedagogical and epistemic needs. Without sufficient attention to agency and purpose in enacting such practices, curricula risk engaging learners in the reproduction of computational actions with little understanding of why those actions are necessary, why they ought to be valued, or for what other situations those actions might be useful (Berland et al., 2015 and Miller et al., 2019 similarly problematize this with respect to the use of "practices" in the science education literature). Worse, without drawing educators' attention to the complex interplay between computing and other disciplinary values and goals, computational ideas such as "optimization" may be noticed and valued above disciplinary and ethical goals (e.g., optimizing a simulation of predator-prey dynamics with little consideration of actual predator-prey interactions; Smith, Haarer, & Confrey, 1997; optimizing search results without attending to how racist and sexist narratives are reproduced; Noble, 2018).

This call to attend to computing in the daily experience of learners and educators echoes, in some ways, reframing of computational thinking as "computational literacy" (diSessa, 2001), "computational participation" (Kafai, 2016), and "critical computational literacy" (Lee & Soep, 2016). Kafai, Proctor, and Lui (2019) have noted that these reframings mark a diverse, but often hidden or underspecified, collection of theorizations of learning as cognitive, social, and critical. Moving among these theorizations and putting them into dialog, they argued, can allow designers and educators flexibility to study and design for “a diverse patchwork of priorities and stakeholders” (p. 106) in K-12 education. For example, they describe how critical orientations toward CT highlighted gendered crafts and e-textiles as a way to engage girls in computing, whereas cognitive orientations allowed them to examine learning with e-textiles in the formal setting of a computer science classroom.

We agree with Kafai and colleagues that researchers ought to be explicit and flexible in how they are theorizing computational thinking. We further highlight a corollary that has thus far been left implicit: Just as it is important to make explicit one’s position within the *theory space* of computational thinking to inform design and study, so too is it important to explicate the *spaces of learning* within which computational thinking emerges and is sustained. In other words, researchers ought to be explicit, flexible, and *grounded* in how they are theorizing computational thinking. Such a focus can help researchers close the loop between theory, design, and research by highlighting how “locally functional solutions” (Joseph, 2004, p. 238) are necessary to refine theory and adapt design to address the particular needs of different educational environments.

We have seen how attending to and moving across spaces of learning matters for our own research in CT. For example, Michelle began her research career exploring how computational simulation might help students structure their thinking about physical mechanism in science (a commonly-cited dimension of CT as described in e.g. Barr & Stephenson, 2011; Lee et al., 2011; Weintrop et al., 2014; NGSS, 2012). While students did use computing in this way, teachers and students also often adapted computing to serve other immediate needs, such as to cultivate communicative norms (Wilkerson, 2017). These adaptations were too specific, and idiosyncratic, to be captured in *apriori* STEM-CT frameworks, but nevertheless crucial for establishing and sustaining productive classroom engagement. This led us to develop analytic frameworks that drew from the learning sciences (Wilkerson, Shareff, Laina, & Gravel, 2018) and narrative theory (Wilkerson & Gravel, in press) to re-frame performances of computational thinking in light of emergent goals, interactions, and epistemic orientations.

Cynthia's early work included developing and researching the effectiveness of a physics game (SURGE). The game was meant to provide students with experience in gravity-free and frictionless environments to build their conceptual understanding of Newtonian mechanics (e.g., Clark et al., 2011; D'Angelo, 2010). In the game, students are expected to use on-the-fly vector arithmetic in order to successfully navigate the game. A typical physics class would include this kind of computation with static problems, but it is not always supported as an explicit computational activity, especially with a constantly changing (i.e., moving) object. Visual representations of vector arithmetic in the game environment helped students during the game, but also had unanticipated effects later on when students were taking midterms and final exams. In this way, what was initially developed as a computational strategy for students within the game was then used as a computational tool for reasoning through new problems outside of the game (D'Angelo, 2010). The representations unexpectedly became a way of thinking computationally for students to use in these other spaces of learning.

Breanne's work has examined how youth collaborate to design computational artifacts such as with electronic textiles (e-textiles; Litts, Kafai, Dieckmeyer, 2015). In these interdisciplinary, multimodal problem spaces, learners must collaborate and design across disciplines (e.g., art, engineering, and computer science) and modes (e.g., physical and digital spaces) with varying levels of expertise. Initially, the work explored discrete content knowledge, such as circuitry knowledge, but as the work evolved, it became apparent that these simple circuitry tasks (e.g., Peppler & Glosson, 2013) were not fully capturing the situated computational knowledge learners developed, especially in collaborative settings. While helpful, these earlier assessments also did not capture the interdisciplinary nature of the computational knowledge we were teaching. In response to this tension, we developed *codeable circuits* (Litts, Kafai, Lui, Walker & Widman, 2017) as a new construct to understand the interdisciplinary nature of the computational context learners were designing. The new assessment tasks we developed were heavily situated in the computational problem space of e-textiles and provided a more comprehensive understanding of learning.

Contributions of the Special Issue

We suspect that at this point in the development of our field, we can develop more precise and actionable conceptualizations of computational thinking by looking at its diverse manifestations across specific projects, ages, and learning environments. This special issue of *Interactive Learning Environments* puts forth a few such detailed examples of how CT manifests within the curricula, activities, and environments learners engage with on a regular basis. We titled the issue “locating and cultivating computational thinking” for two reasons. First, the language of locating acknowledges that opportunities to engage with computing and computational ideas are not limited to researcher-led CT integration efforts. Instead, we find instances where clear opportunities for engaging youth in computational thinking lie within, for example, existing standards documents (Rich et al., 2020/this issue), and instances where computational thinking is already well rooted in classroom activity but require tools for assessment and reflection (Lui et al., 2020/this issue). Second, the language of cultivation suggests CT as a dynamic and contingent phenomenon which can be seeded, but which is sensitive to external conditions and requires a delicate balance in order to grow and thrive.

The contributions to this special issue can be characterized as addressing three questions related to locating and cultivating computational thinking: *Where are there opportunities for computing to live within day to day functioning of educational spaces? What are examples of computing integration across a variety of learning spaces?* And, *How might educators work to sustain the computational practices and identities developed in activity?* In response to the first question, "Synergies and Differences in Mathematical and Computational Thinking: Implications for Integrated Instruction," (2020/this issue) identifies a number of what Rich and her coauthors describe as *proto-CT* ideas, or core concepts (such as precision, order, and binary ideas) that are identified as important in both computational thinking and that appear within the U. S. nationally-adopted Common Core State Standards for Mathematics (2010). Importantly, they demonstrate the need to move beyond simply identifying similarities in these concepts by articulating points of connection and difference in how these ideas are treated in computing and mathematics. They also attend carefully to the specific needs and constraints of elementary school classrooms, which are most likely to employ spiral curricular approaches that begin with unplugged activities and move toward the use of computer tools in later grades.

A second set of papers respond to the second question by providing examples for how to approach integrating computational thinking across a range of learning contexts. From out-of-school to in-school, this set of papers examines how interdisciplinary approaches to integration support computing practices across contexts. Litts, Lewis, and Mortensen (2020/this issue) in “Computational Making in ARIS” share how a mobile, place-based approach to computational design within a narrative programming environment provided space for students to grow and engage with computation in unique ways. The specificity of location afforded students the ability to test and debug their creations in new ways. For instance, students would make digital adjustments to their program and then go outside and run around to test it. This “embodied debugging” allowed students to engage with the CT concepts in an authentic way. The paper discusses how they designed the activity for transparency rather than embedding or assuming knowledge of computer science principles or privileging certain types of disciplinary practices. This allowed them to afford more flexible computational tools and was a critical consideration in designing for inclusive computing cultures.

The paper by González-Calero, Rodriguez Martinez, and Sáez-López (2020/this issue) offers a more detailed look at the role that programming languages such as Scratch might be playing in students’ learning of integrated content in the classroom context. “Computational thinking and mathematics from Scratch: An experiment with sixth-grade students,” reports a quasi-experimental study in which students completed a module focusing on core computational concepts such as sequences, iterations, and conditions using Scratch. One group of students then used Scratch to complete a number of word problems related to finding a least-common-multiple or greatest-common-denominator, while the other group completed the problems by hand. Analyses revealed that both groups of students performed comparably on measures of computational thinking. Furthermore, while both groups demonstrated gains on measures of mathematics performance, only the gains for the group that used Scratch to complete the mathematics tasks were statistically significant. What is interesting about this finding is that the

Scratch group performed well despite the fact that these students completed fewer such word problems than their peers in the control group—speaking directly to a common concern regarding CT-integrated activities among educators and researchers familiar with the time constraints of the K-12 classroom.

The paper by Clark and Sengupta (2020/this issue) explores the affordances of integrating modeling within what they call “disciplinary-integrated games” from both a science-as-practice and a CT perspective. In “Reconceptualizing Games for Integrating Computational Thinking and Science as Practice: Collaborative Agent-Based Disciplinarily-Integrated Games” they discuss the implications for design and praxis of taking seriously not only the co-incidence of modeling in science and computing, but also each domain’s different perspectives on modeling. This attention to detail allows them to frame computational modeling as mathematization which has implications for classroom learning and how learning environments can be designed. By taking a multi-disciplinary approach to thinking about the affordances of modeling, the location of modeling within the larger curriculum becomes both more nuanced and more expanded, providing new opportunities for computing to grow in multiple places.

Finally, the third question is addressed by a set of papers that explore strategies for sustaining computational thinking for extended periods of time, within the complexities and constraints of everyday learning environments. Lui and colleagues’ (2020/this issue) paper “Communicating Computational Concepts and Practices within High School Students’ Portfolios of Making Electronic Textiles” describes one classroom’s attempt to better capture and assess students’ computational engagement through portfolio assignments that allowed students to document and reflect on their work. Notably, the portfolio assignment was not an original goal of the researchers' work in this classroom. One finding from this collaborative, emergent study was that carefully designed portfolio activities could provide students with a new space to practice more precisely communicating about, and reflecting on, their computational work while also satisfying the assessment needs so common in traditional classrooms.

Another key need to sustain computational thinking is through an equitable approach that supports participation from diverse audiences. Pinkard, Martin, and Erete's paper "Equitable approaches: Opportunities for computational thinking with emphasis on creative production and connections to community" (2020/this issue) share their work implementing a program, Digital Youth Divas, targeting non-dominant youth. Learners engaged in a range of computational making activities over a 20-week program. The authors show how these learners not only developed their content knowledge (e.g., how to make a series circuit), but also highlight shifts in learners computational perceptions. One of the main takeaways from this work is the authors' challenge to consider computational perceptions, alongside concepts and practices, as a key indicator of young people's personal connection with computational making. For example, they found that investigating computational perceptions allowed them more flexibility to track learning, including in content knowledge and computational practices, without reliance on a specific predefined computational vocabulary.

Israel and Lash's "From Classroom Lessons to Exploratory Learning Progressions: Mathematics + Computational Thinking" (2020/this issue) highlight that sustaining computational thinking may also demand a longer-term, developmental approach that frameworks and taxonomies do not readily accommodate. In this paper, the authors describe efforts to codesign and enact integrated mathematics and CT lessons in elementary grades 1-5.

The authors identified a number of systemic and curricular constraints that limited the potential for true integration for teachers. Some of these constraints as well-known and straightforward, such as teachers' perceived need to prioritize mathematical content, and a lack of available integrated curriculum materials. Others, however, included the nonlinear nature of CT concepts (whereby some concepts build on others that might not have yet been introduced within the integrated math+CT materials), and the expectation that students' comfort with both computational and mathematical practices should emerge over multiple encounters experienced across years of instruction.

The findings from this collective work highlight the idea of computational thinking as inherently situative. This is a useful conceptualization for a number of reasons. It accommodates the interdisciplinary nature of computational thinking as a construct, and the complexity of the learning ecosystems in which we expect CT to emerge (e.g., schools, informal environments, curricular materials). It also allows us to be more adaptive to future learning needs and technological innovations. The papers demonstrate the power of locally constructed definitions of CT—in contrast to one-size-fits-all frameworks—that respect and take seriously the contributions, needs, and systemic constraints of the educators and learners whom we, as researchers, seek to serve.

Acknowledgements

This special issue originated from a poster symposium, "Stories from the field: Integrating computational thinking across curricular domains," held at the 2017 Annual Meeting of the American Educational Research Association, and a working session at Cyberlearning

2016. This work was supported by the National Science Foundation under grant DRL-1418163.

Thanks Ciro for bringing joy to our videoconferences.

References

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: what is Involved and what is the role of the computer science education community? *ACM Inroads*, *2*(1), 48-54.

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

Clark, D. B., Nelson, B., Chang, H. Y., Slack, K., Martinez-Garza, M., & D'Angelo, C. M. (2011). Exploring Newtonian mechanics in a conceptually-integrated digital game: Comparison of learning and affective outcomes for students in Taiwan and the United States. *Computers and Education, 57*(3), 2178-2195.

Clark, D. B., & Sengupta, P. (2020/this issue). Reconceptualizing games for integrating computational thinking and science as practice: Collaborative agent-based disciplinarily-integrated games. *Interactive Learning Environments*.

D'Angelo, C. M. (2010). *Scaffolding vector representations for student learning inside a physics game.* Doctoral dissertation. Arizona State University.

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

Engle, R. A., & Conant, F. R. (2002). Guiding principles for fostering productive disciplinary engagement: Explaining an emergent argument in a community of learners classroom. *Cognition and Instruction*, *20*(4), 399-483.

González-Calero, J. A., Rodriguez Martinez, J. A., & Sáez-López, J-M. (2020/this issue). Computational thinking and mathematics from Scratch: An experiment with sixth-grade students. *Interactive Learning Environments*.

Israel, M., Lash, T. & Reese, G. (2020/this issue). From classroom lessons to learning trajectories: Mathematics + computational thinking. *Interactive Learning Environments.*

Joseph, D. (2004). The practice of design-based research: Uncovering the interplay between design, research, and the real-world context. *Educational Psychologist*, *39*(4), 235-242.

Kafai, Y. B., & Burke, Q. (2013, March). The social turn in K-12 programming: moving from computational thinking to computational participation. In *Proceeding of the 44th ACM technical symposium on computer science education* (pp. 603-608). ACM.

Kafai, Y., Proctor, C., & Lui, D. (2019, July). From Theory Bias to Theory Dialogue: Embracing Cognitive, Situated, and Critical Framings of Computational Thinking in K-12 CS Education. In *Proceedings of the 2019 ACM Conference on International Computing Education Research* (pp. 101-109). ACM.

Lave, J. (1988). *Cognition in practice: Mind, mathematics and culture in everyday life*. Cambrid

Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., ... & Werner, L. (2011). Computational thinking for youth in practice. *ACM Inroads*, *2*(1), 32-37.

Lee, C. H., & Soep, E. (2016). None but ourselves can free our minds: Critical computational literacy as a pedagogy of resistance. *Equity & Excellence in Education*, *49*(4), 480-492.

Litts, B.K., Kafai, Y.B., Dieckmeyer, E. (2015). Collaborative electronic textile designs by high school youth: Challenges and opportunities in connecting crafts, circuits, and code. In *Proceedings of the 5th Annual Conference on Creativity and Fabrication in Education*. ACM, New York, NY.

Litts, B. K., Kafai, Y.B., Lui, D.A., Walker, J.T., & Widman, S.A. (2017). Stitching Codeable Circuits: High School Students' Learning about Circuitry and Coding with Electronic Textiles. *Journal of Science Education and Technology,* 26 (5), 494-507.

Litts, B. K., Lewis, W., Mortensen, C. (2020/this issue). Computational Making in ARIS. *Interactive Learning Environments*.

Lui, D., Walker, J., Hanna, S., Kafai, Y., Fields, D., Jayathirtha, G. (2020/this issue). Communicating computational concepts and practices within high school students' portfolios of making electronic textiles. *Interactive Learning Environments*.

Miller, E., Manz, E., Russ, R., Stroupe, D., & Berland, L. (2018). Addressing the epistemic elephant in the room: Epistemic agency and the next generation science standards. *Journal of Research in Science Teaching*, *55*(7), 1053-1075.

Noss, R. (1987). Children's learning of geometrical concepts through Logo. *Journal for Research in Mathematics Education*, 343-362.

Ochs, E., Taylor, C., Rudolph, D., & Smith, R. (1992). Storytelling as a theory-building activity. *Discourse processes*, *15*(1), 37-72.

Papert, S., Solomon, C., Soloway, E., & Spohrer, J. C. (1971). Twenty things to do with a computer. In *Studying the Novice Programmer*. Lawrence Erlbaum Associates, Inc.

Peppler, K., & Glosson, D. (2013). Stitching circuits: Learning about circuitry through e-textile materials. *Journal of Science Education and Technology*, *22*(5), 751-763

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

Pinkard, N., Martin, C. K., & Erete, S. (2020/this issue). Equitable approaches: Computational thinking through design. *Interactive Learning Environments*.

Repenning, A., Webb, D., & Ioannidou, A. (2010, March). Scalable game design and the development of a checklist for getting computational thinking into public schools. In *Proceedings of the 41st ACM technical symposium on Computer science education* (pp. 265-269). ACM.

Rich, K., Strickland, C., Franklin, D., Spaepen, L. (2020/this issue). Synergies and differences in mathematical and computational thinking: Implications for integrated instruction. *Interactive Learning Environments*.

Richard, G. T. (2017). Video games, gender, diversity, and learning as cultural practice: Implications for equitable learning and computing participation through games. *Educational Technology*, 36-43.

Smith, E., Haarer, S., & Confrey, J. (1997). Seeking diversity in mathematics education: Mathematical modeling in the practice of biologists and mathematicians. *Science & Education*, *6*(5), 441-472.

Soloway, E., Lochhead, J., & Clement, J. (1982). Does computer programming enhance problem solving ability? Some positive evidence on algebra word problems. In Robert J. Seidel, Ronald E. Anderson and Beverly Hunter (Eds.), *Computer Literacy* (pp. 171-201). Elsevier.

Sutherland, R. (1994). The role of programming: towards experimental mathematics. In R. Biehler, R. W. Scholz, R. Sträßer, B. Winkelmann (Eds.), *Didactics of Mathematics as a Scientific Discipline–The State of the Art*. Dordrecht, The Netherlands: Kluwer Academic Publishers. pp 177-187.

The White House, Office of the Press Secretary. (2016). *Fact sheet: President Obama announces computer science for all initiative.* [Press release]. Retrieved from http://bit.ly/1oG3OaC

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

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, *25*(1), 127-147.

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

Wilkerson, M. H. (2017). Teachers, students, and after-school professionals as designers of digital tools for learning. In C. DiSalvo, B. DiSalvo, J. Yip, & E. Bonsignore (Eds.), *Participatory Design for Learning*. Taylor & Francis. pp. 127-140.

Wilkerson, M. H., Shareff, R., Laina, V., & Gravel, B. (2018). Epistemic gameplay and discovery in computational model-based inquiry activities. *Instructional Science*, *46*(1), 35-60.

Wilkerson, M. H. & Gravel, B. E. (in press). Storytelling as a support for collaborative constructionist activity. In N. Holbert, M. Berland, & Y. Kafai (Eds.), *Designing Constructionist Futures: The Art, Theory, and Practice of Learning Designs*. Cambridge, MA: MIT Press.

Yackel, E., & Cobb, P. (1996). Sociomathematical norms, argumentation, and autonomy in mathematics. *Journal for Research in Mathematics Education*, 458-477.