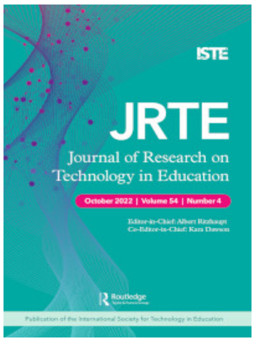


Journal of Research on Technology in Education

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/ujrt20

Coding choreography: Understanding student responses to representational incompatibilities between dance and programming

Selena Steinberg, Melissa Gresalfi, Lauren Vogelstein & Corey Brady

To cite this article: Selena Steinberg, Melissa Gresalfi, Lauren Vogelstein & Corey Brady (2022): Coding choreography: Understanding student responses to representational incompatibilities between dance and programming, Journal of Research on Technology in Education, DOI: 10.1080/15391523.2022.2135144

To link to this article: https://doi.org/10.1080/15391523.2022.2135144

Published online: 27 Oct 2022.

JOURNAL OF RESEARCH ON TECHNOLOGY IN EDUCATION
https://doi.org/10.1080/15391523.2022.2135144

Coding choreography: Understanding student responses to representational incompatibilities between dance and programming

Selena Steinberg[a*], Melissa Gresalfi[a], Lauren Vogelstein[a] and Corey Brady[a]

[a]Teaching and Learning, Vanderbilt University, Nashville, Tennessee, USA

ABSTRACT
This paper considers how a curricular design that integrated computer programming and creative movement shaped students' engagement with computing. We draw on data from a camp for middle schoolers, focusing on an activity in which students used the programming environment NetLogo to re-represent their physical choreography. We analyze the extent to which students noticed *incompatibilities* (mismatches between possibilities in dance and NetLogo), and how encountering them shaped their coding. Our findings suggest that as students attended to incompatibilities, they experienced struggle, but persisted and engaged in iterative cycles of design. Our work suggests that tensions between arts and programming may promote student engagement.

ARTICLE HISTORY
Received 28 March 2022
Revised 8 October 2022
Accepted 9 October 2022

KEYWORDS
Programming; computer science; dance; representation; struggle; design

Programming can be challenging to learn, as students' experiences in computer science are dependent upon interactions among a myriad of factors, including classroom pedagogical practices (Ryoo, 2019), curricular design (Lewis & Shah, 2015), and the affordances of particular tools (Litts et al., 2020). In order to create opportunities for all students to engage in the discipline, it is important to consider how to support computer science students to persist through programming challenges. One strategy is to invite students to engage in computer science as an act of design, capitalizing on the expressivity of coding (Edmonds, 2019) and allowing students to explore spaces of computational possibility (Brady et al., 2020). Another way to support student persistence is to lean into the idea that challenge is an unavoidable part of computer science. Research on productive failure suggests that early failure can support later success (Kapur, 2008). Therefore, understanding the conditions that make challenge productive can help students to persevere.

In this paper, we explore these two possibilities in conjunction, asking how situating a challenging programming task in an open-ended design environment supports student engagement with coding. We look specifically at the context of choreographing a dance and translating it into a computational performance. Computation has been connected with dance because of parallels between the syntactic structures used in programming and the choreographic structures used to describe coordinated movements (Bergner et al., 2021; DesPortes et al., 2016, 2021; Leonard et al., 2021; Leonard & Daily, 2014; Vogelstein et al., 2021). However, analogy is not equivalence: bringing computer science and dance together naturally creates tensions as moving between these representational forms entails processes of translation and adaptation (Greeno & Hall, 1997). Rather than minimize these tensions, we designed an activity that accentuated them,

CONTACT Selena Steinberg ssteinb@iu.edu Learning Sciences, Indiana University, Bloomington, IN, USA.
*Additional affiliation: Learning Sciences, Indiana University, Bloomington, Indiana, USA.

asking students to re-represent their own choreographies in code, shifting the modal representation from ensemble, full-bodied movement to computer-scale procedures. We examine this shift as potentially generative, by looking at how encountering *incompatibilities*, or mismatches between possibilities in creative movement and programming, positioned students to respond in ways that enabled them to persist through challenges.

Literature review

As computer science has become increasingly prominent in K-12 education, researchers have sought to understand how to support younger students to succeed in programming. These considerations are wide ranging, from decisions about how to introduce or sequence programming languages (Resnick et al., 2009; Weintrop & Wilensky, 2019), to considerations about features of curriculum design (Buechley et al., 2008; Garneli et al., 2015; Jayathirtha & Kafai, 2020) and features of pedagogical practices (Franklin et al., 2020; Lytle et al., 2019). This paper contributes to our thinking about curricula and tools, with the goal of better understanding how activities that ask students to think about programming as an act of translation, and position students as designers, influence their learning and engagement.

Research that focuses on the potential of various curricula for fostering computational thinking has demonstrated that settings that have the potential to motivate and enhance computer science learning include game design and development, robotics, and modeling (Garneli et al., 2015). In addition, inviting students explicitly into the framework of design, for example by learning computer science through *electronic textiles,* has been productive in sustaining student engagement (Buechley et al., 2008; Jayathirtha & Kafai, 2020; Kafai et al., 2014). Electronic textiles connect computing with crafting: students engage in design through programming electronic materials such as LED lights and stitching them onto clothing or accessories. This work has been shown to broaden participation in computing, specifically for girls and women, increase interest in computing for K-16 students, and improve students' understanding of circuit design (Jayathirtha & Kafai, 2020). In this paper, we explore how design may support students to persist through programming challenges.

The affordances of design

There are many features of design tasks that make them good candidates for supporting students' persistence. First, design tasks are open-ended, and rely on the personal judgements of the designer to decide when they are satisfactorily completed. Jonassen, in his comprehensive overview of problem solving (2000), makes this observation:

> Design problems are usually among the most complex and ill-structured kinds of problems that are encountered in practice. For many years, researchers (Reitman, 1965; Simon, 1973) have characterized design problems as ill-structured because they have ambiguous specification of goals, no determined solution path, and the need to integrate multiple knowledge domains. (p. 80)

Thus, at their core, design tasks offer many opportunities for students to experience struggle and failure, as well as the capacity to decide what failure and success look like.

Ill-structured problems offer opportunities for expressive potential, as the act of defining and resolving an ill-structured problem requires a learner to bring their experiences and interpretations to the task. Although there are limitless media that can be leveraged for expression, the arts in particular offer a creative, expressive media through which we can make meaning, expose subjectivities, and represent our emotions and ideas (Griffin et al., 2017; Wright, 2015). As Wright (2015) expresses, "through the arts, children not only come to *know* reality, they *create* it" (p. 3), and it is this act of creating one's own reality that positions students working in expressive arts environments as designers. Several researchers have paired coding with artistic practices to broaden participation in computing (Allen-Handy et al., 2020; Bennett & Eglash, 2013; Flesch et al., 2021; Kafai, 2016; Payne et al., 2021) and improve students' computer science learning (Jörg et al., 2014; Owen et al., 2016; Shamir et al., 2019). For example, Bennett et al. (2016) described the opportunities that are advanced when students are invited to engage in programming activities that draw from indigenous expression and to recreate their own expressive representations.

This paper contributes to these literatures by exploring students' designs as they re-represent their own choreographies in the computational environment of NetLogo. Although in our focal activity the subject of students' coding was a dance that students had envisioned and enacted, our analysis focuses on the role of *design*—first through choreography, then through the act of re-representation—as the act of expression. This framing orients us to the data in a different way—rather than focusing on the embodiment of understanding that might be the focus of an analysis of a dance (e.g. Solomon et al., 2021), our focus—on the act of re-representing a design in a computational environment—instead offers a design-thinking lens to explore students' struggle and persistence.

Productive failure in design

Productive failure, or presenting students with a challenging or ill-structured task with the support necessary for them to struggle through it, has been found to support student learning (Kapur, 2008). Such challenge, and time spent exploring a task, seems to be productive in part because it allows "learners to generate conceptions, representations, and understandings, even though such understandings may not be initially correct" (Kapur & Bielaczyc, 2012, p. 48). A related concept, productive struggle, has been investigated in settings including mathematics (Warshauer, 2015), computer supported collaborative learning (Kapur & Kinzer, 2009), and open-ended design contexts (Litts et al., 2016). In computer science, connections have been drawn between productive struggle and debugging (Giri, 2022; Kafai et al., 2019), as encountering and resolving bugs can help students to understand the problem and to reach better solutions.

Settings that integrate STEM and the arts can be inherently challenging, as tensions between representational forms sometimes provoke frustration, even as they are generative. In similar studies in mathematics, tensions and mismatches between representational forms have been understood in terms of the demands they place on discourse and multi-party coordination (Hall et al., 2014; Ma, 2017). Processes of re-representation, or iteratively creating new representations that foreground full body participation can support deep engagement with mathematical concepts (Gerofsky, 2010; Hall, 1996; Hall et al., 2014; Kelton & Ma, 2020; Kremling et al., 2018; Ma, 2016; Vogelstein et al., 2019). Such tasks therefore seem ideal grounds for exploration of productive struggle; indeed, when learners creating expressive movement together experience challenges and disagreements, they tend to persist, expanding perspectives and creating new practices (Vogelstein et al., 2019; Champion, 2018; DesPortes et al., 2016; Ma, 2016).

Building on these bodies of work that have connected arts and computation and demonstrated the potential productivity of struggle and tension, we explore what happens when students encounter challenge (in the form of representational incompatibilities between what it is possible to choreograph for humans and what is possible to show in code) in an open-ended expressive design task.

Study context

Data for this paper comes from an activity that took place during Movement Art & Coding Camp, which was designed by Vogelstein & Brady in collaboration with Rebecca Steinberg & Curtis Thomas, two professional dance artists (Vogelstein et al., 2021). The designs built upon prior work (Brady, 2021; Vogelstein & Brady, 2019; Vogelstein et al., 2019) linking embodied experiences with sense-making around computational agents, beginning from Papert's (1980)

Figure 1. Two focal students, Harrison and Xavier, working with Mylar.

notion of "syntonic learning," and exploring how this approach extends to groups of people reasoning about groups of agents (Brady et al., 2016). Broadly, this work highlights how people's embodied experiences can be generative resources for creating with virtual agents. Although the motivation for the design of this activity comes from questions about embodiment, in this paper we focus instead on the act of *designing* a dance as an expressive, creative art form and its influence on coding.

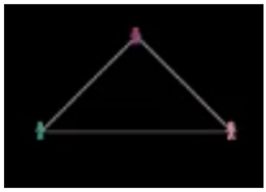
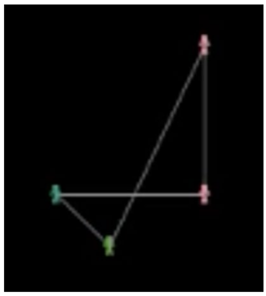
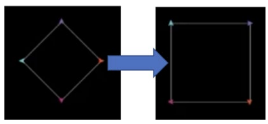
To do so, we analyze the second phase of a designed activity that was called the *Telephone Game*. In the first phase, groups of four students developed choreographies with 7′ x 7′ Mylar squares (Figure 1). In phase two, students re-represented their choreography computationally in NetLogo (Wilensky, 1999), a multi-agent environment that uses text-based programming syntax to orchestrate the behavior of stationary and movable *agents* (called "patches" and "turtles," respectively). In phase three, quartets exchanged the code they had created with another group and performed it as choreography with the Mylar.

In this activity students designed both choreographed movement and a NetLogo model, engaging them in the challenge of re-representing ideas from one mode to another. Asking students to translate their ideas provided opportunities for students to encounter challenge and struggle, as they grappled with the similarities and differences between physical and digital choreographies. The goal of this translation activity was for students to see each expressive mode as powerful and to fuel ideas for what might be possible to create (cf., Payne et al., 2021).

With the goal of better understanding students' coding process, we focus on the second phase of the activity, in which the student quartets translated their physical choreography digitally using NetLogo. NetLogo was used in the camp because, as an agent-based modeling (ABM) environment, it is optimized for capturing the emergent behavior of groups of agents. In addition, its long history of use as a tool in syntonic modeling (Wilensky & Reisman, 2006) and in Participatory Simulations (Wilensky & Stroup, 1999; Brady et al., 2018) suggested it could support students in highlighting group-level patterns. Finally, in prior work, we saw how using agent-based "pseudocode" in NetLogo to describe ensemble movement highlighted this dimension of social coordination in emergent choreography (Vogelstein et al., 2019). The second phase of the *Telephone Game* engaged students for approximately one hour, as they struggled and persisted in translating their choreographies as NetLogo code. The challenge of this effort often provoked multiple cycles of design, implementation, and redesign. Indeed, taken literally, the task to translate their dance into code was technically impossible, as translating from any representational system to another requires choices of selective abstraction (Farris et al., 2016). This "impossible task" provided many opportunities for students to encounter failure and struggle.

In our analysis, we use the term *incompatibilities* to point to mismatches between the representational affordances of Mylar choreography in the physical world and of turtles, links, and patches in NetLogo procedures in the digital world. Students encountered incompatibilities when

Table 1. Examples of incompatibilities.

Incompatibility	NetLogo Model	Dance Choreography
In NetLogo, two people-shaped turtles occupied the same space (top point of triangle), but in dance, it was challenging for two people to occupy the same space.		
In NetLogo, moving a linked turtle caused the links to move as well, making it difficult to move only a person and not the Mylar.		
With the given NetLogo setup, it was challenging to show rotation without the Mylar growing and shrinking because the area enclosed by the links was not fixed.		

their NetLogo code produced phenomena that were foreign to their experiences with Mylar and physical movement; or when they struggled to represent features of their dance in NetLogo (see Table 1 for examples). We focus on how students experienced these incompatibilities between physical and digital representational systems and how incompatibilities provoked students to persist and develop new forms of computational expressivity.

Conceptual framework: engaging the world through the lens of design

As stated, these activities positioned students as designers: first, as designers (choreographers) of a dance for a quartet, and second, as designers of their translated dance in NetLogo. The latter design activity, which is the focus of this paper, offered constraints in the form of being asked to re-represent a previous design in a new medium (coding). Constraints always play a central role in design, and they come in multiple forms: material, cultural, and personal (Bennett & Eglash, 2013; Gravel & Svihla, 2021). Such constraints become the materials of design; as Schön and Wiggins (1992) describe it, designing is "a kind of experimentation that consists in reflective 'conversation' with the materials of a design situation. A designer sees, moves, and sees again" (p. 135).

Many scholars have explored the ways that materials serve as a set of design tools. At its most straightforward, materials are the media of design; in programming, they include the constraints of syntax, the logic and supported structures of a language, and the tools that are available in a programming environment. These materials offer both opportunities and resistance; understanding those material constraints are part of the work of design. In addition, Bennett and Eglash (2013) argue that culture, both in its history and its practices, are incorporated into design, as culture can serve both as an inhibiting form of resistance (e.g., the idea that "that's not what we do here"), and as a facilitating form of resistance (offering meaningful ideas or representations) in the act of design.

Most design also involves some sort of technical skill—not only understanding what the materials make possible, but also, how to manipulate those materials to desired ends. Such forms of human agency (Pickering, 1995) are endemic to the act of programming, such as the tension between creating a program that "works" versus one that is "elegant," developing novel code independently versus "copying" or "remixing" others' code, and the sense of aesthetic that is often involved in determining whether a program is "good enough." Similarly, Schön and Wiggins identified distinct facets of human agency in design. Specifically, they showed how design involves an interplay between *intentions* that a designer has in making design "moves," and the judgments that they form about the effects of those moves, which are influenced by a designer's *appreciative systems*—values, beliefs, and experiences. These appreciative systems are both individual and shared; different designers might reach different conclusions about the rightness of their designs, but overall, their appreciations have significant overlap with others' appreciations.

Central to this idea is that such design work involves struggle, compromise, and the development of new understanding through the back and forth—the "seeing-moving-seeing" (Schön & Wiggins, 1992) between the intentions that a designer sets, their specific decisions or versions of their design, and changes to their appreciative systems over time. In this way, a designer does not simply make a plan and execute the plan, but rather notices new things through the act of design, which serves to change both what they notice and appreciate, and what they might intend in the first place. These cycles of ideation, design, redesign, and re-imagining dovetail with the same processes of productive struggle. However, in a context where one's personal sense of aesthetics becomes a criterion for appreciation, there is a different sense of personal agency than in tasks that ultimately have one right answer.

Our analysis takes as a starting point that design is an interaction between these material and human forms of agency (Pickering, 1995), and draws on the idea that design involves developing relations between intention and appreciation as a means of understanding students' work and their persistence in the face of struggle (see Figure 2). Using this analytic framework, we pose the following research questions that consider students' processes of design in a computational environment with incompatibilities:

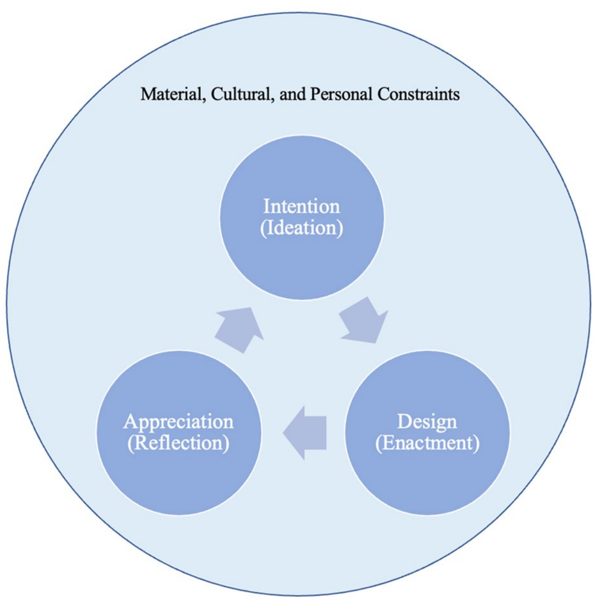


Figure 2. Interactions between human and material agency in design.

1. In the translation process from physical choreography to NetLogo code, to what extent are incompatibilities deemed important by students?
2. How do incompatibilities provoke continued engagement with the activity of design?

Materials and methods

Data sources

The Movement Art & Coding Camp included two classrooms (one with 11 students and one with 13), each led by two teachers and one teaching assistant. Each classroom was additionally supported by members of the research team who talked with students and answered questions when asked. Data for this analysis came from screencaptures of six students independently coding their choreography in NetLogo. Across both classrooms, there were 14 consented students with screencaptures that included audio and video. Eight of these students worked without the constant presence of a teacher, though students were allowed to talk to each other and ask for help, and they often did. We included in our analysis only students that worked independently in this way with the goal of understanding students' coding process without substantive guidance from teachers. Two additional students were removed from the analysis because they did not attempt the task. Therefore, we focus on six students: Chris, Harrison, Jonah, Kyle, Xavier, and Zaair (all names are pseudonyms). All six students identified as male and were entering 6th through 8th grade. Harrison, Jonah and Kyle were 11 years old, Xavier and Zaair were 12, and Chris was 13 years old. Xavier and Zaair identified as African American, and Chris, Kyle, Harrison, and Jonah were white. Xavier and Chris attended the same middle school; the rest of the students in the sample attended different middle schools in the same district. Chris, Jonah, and Zaair were in the same quartet from the first phase of the Telephone Game, and they independently worked on re-representing the same piece of their choreography. Harrison and Xavier were in the same quartet and worked on re-representing different parts of their choreography. Finally, Kyle was from a third quartet.

For this analysis, we treat the six students together as an exploratory and instrumental case study (Yin, 2014)— a case of engaging in translation between representational systems. Doing so allows us to consider variation amongst experiences that together make up the richness of the "case of" translation. Much as in single case studies, where the goal is to gather rich accounts of the experience of a single person, by combining several students into one case we are similarly able to gather rich data about the experiences of working independently on the task of translation. This approach is consistent with the definition of case study offered by MacDonald and Walker (1975), who explain that, "case study is the examination of an instance in action. The choice of the word 'instance' is significant in this definition because it implies a goal of generalisation" (p. 2). The "instance in action," as it were, is the task of translation, one that we seek to explore in sufficient depth to offer a set of "petite generalizations" (Stake, 1995) which might serve as useful conjectures for future design and exploration. This expansive approach to the subject—identifying new possibilities for learning environments and activity structures that use design tasks to foster computational thinking—positions the goals of the study as making the contributions to design science research in computer science education (cf. Hevner et al., 2004).

Nevertheless, by collapsing these students into a single case, we likely have backgrounded some of the unique experiences of each student, blurring individual variations that might be crucial to future activity designers. Our analysis also does not consider the ways the students were positioned in the classrooms, how their visible identities might have mediated their experiences, or whether or what the individual students learned. We see this as a limitation of our analysis, which we describe at the end of the paper.

Analysis

In the first phase of analysis, we watched videos of students' Mylar choreographies and screencaptures of students' NetLogo work and wrote analytic memos (Saldana, 2011) for each student, cataloging moments where students made judgements about their models and set new intentions about how to re-represent their choreography in code. Specifically, we focused on representational decisions: that is, moments when students chose to represent human versus Mylar movement in their models, where they tried to show both and experienced challenges, or where they transitioned between portraying people and Mylar. Our memos suggested that while students implicitly seemed to show the movement of the people *or* the Mylar in creating their virtual representations, this was disrupted when they noticed that the other did something visually unexpected. These noticings of incompatibilities often spurred students to make new decisions in the re-representation process.

In order to explore these initial themes, our second round of analysis used open coding (Strauss & Corbin, 1990) using the video coding software V-Note. The videos were parsed first by marking incompatibilities and noting whether students noticed them. *Noticing* was coded when a student expressed surprise or dissatisfaction (vocally or through facial expressions) with how their model behaved after encountering an incompatibility, or when a student immediately undid their code to return to a previous, incompatibility-free state. The videos were then parsed to mark each new command or tool that the student tried in order to understand the way that incompatibilities influenced their production of code. Finally, periods of time where a teacher or researcher helped the student were coded. There was one primary coder, and coding questions were discussed and resolved with a second coder throughout the process. The second coder also double-coded instances of noticing incompatibilities. Using this coding as a starting point for each student, we wrote additional analytic memos examining the ways that each incompatibility influenced students' subsequent coding process. Altogether these phases of analysis led us to better understand the ways that incompatibilities were relevant to students and provoked continued engagement with design, as presented in the next section, Findings.

Findings

The goal of this analysis was to understand how students experienced representational incompatibilities, or challenges, in an open-ended design context. We focused on incompatibilities because they provoked struggle, frustration, and surprise for students. As students attempted to "fix" each incompatibility in a way that they deemed appropriate, they engaged in repeated cycles of design and struggle.

RQ1: the importance of incompatibilities

Our first research question asked: To what extent are incompatibilities deemed important by students? Our goal was to understand whether and how students responded to the representational incompatibilities that occurred as they attempted to translate their physical choreographies to a computational model. We examined how students exercised appreciation (Schön & Wiggins, 1992) through reacting and responding to potentially surprising NetLogo feedback.

Student reactions to incompatibilities

All six students noticed an incompatibility at least once in the course of their design work. While there was variability in the number of incompatibilities (ranging from three to 26; perhaps because of differences in choreography or in the amount of time students spent working on the task [$Min = 36$ minutes, $Max = 70$ minutes, $M = 55.8$, $SD = 15.6$]), all but one student noticed and addressed at least two thirds of the incompatibilities we identified (Table 2). The student that did not, Xavier, reacted to only 11.11% of his incompatibilities. His behavior differed from the other students in that he seemed more comfortable with a less literal, more abstract mapping between his choreography and his NetLogo representation. For example, in his dance, one student wrapped the Mylar around another student, which Xavier represented by making two turtles spin in place, instead of keeping one turtle stationary and showing the other moving around it.

Table 2. Incompatibilities and notices by student.

Student	# of Incompatibilities Encountered	# of Notices	% of Notices
Xavier	9	1	11.11%
Chris	3	2	66.67%
Jonah	9	6	66.67%
Zaair	11	8	72.73%
Kyle	15	12	73.33%
Harrison	26	24	92.31%

Figure 3. Harrison's facial expressions after encountering incompatibilities.

Taken through the lens of design, noticing these incompatibilities can be understood to be an act of changing appreciation. With their first code, students were offering an initial design; what followed was their subjective judgements about that design. Very often the system produced outcomes that appeared to be unexpected by the students, as noticing these incompatibilities often elicited frustration or surprise. Students had strong negative reactions to their models behaving in a way that did not mirror their choreographies or defied the possibilities for physical choreography. Verbal reactions (including "what?", "huh?", "why isn't it working", and "nooo!") were commonly expressed when students encountered an incompatibility, and these were often accompanied by physical reactions, such as wide eyes or grimaces (see Figure 3 for examples). These negative reactions suggest that students perceived incompatibilities as challenging. Their *appreciative systems* were activated, which in turn drove them to develop a set of new intentions.

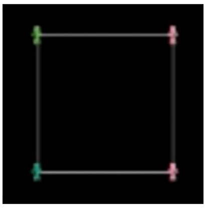
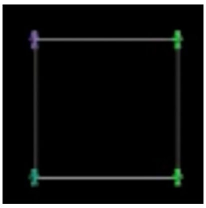
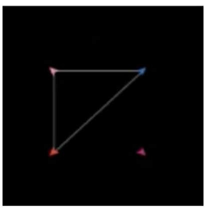
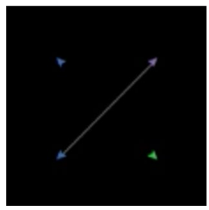
The role of appreciation

Students' appreciative judgements often led to changes in intention; when incompatibilities arose, they generally wanted to adapt their code to improve its representational congruence. For example, the following exchange occurred when Harrison tried to code choreography in which one person held each corner of the Mylar and walked in a circle[1]:

1. Harrison: Oh gosh that's not good!
2. Teacher: Is it spinning?
3. Harrison: No, right now it's not doing too well. So it goes like that, whabam, and then it starts trying to turn.
4. Teacher: That doesn't look… it looks like maybe it's just fast.
5. Harrison: Yeah but it's like getting smaller and then getting big and turning the wrong directions.

Even though the teacher suggested that the model might be a good, but fast, representation of the rotation choreography, Harrison was dissatisfied with the incompatibility of the square changing size as it rotated. This dissatisfaction can be understood as an act of *appreciation,* wherein Harrison was using his own judgment based on his experience as a choreographer of the dance to evaluate the outcome of his model. In seeing the performance run in NetLogo, Harrison changed his intention—a key outcome of the act of design (Schön & Wiggins, 1992). His new intention included the decision that the Mylar's constant side length was important to show in his representation. As will become apparent in other cases, this judgment and shift in

Table 3. Zaair's work addressing an incompatibility.

Row	Initial State	Code	Feedback	Reaction
1		Command Center:*Ask one-of turtles [setxy −1 − 6]*		Zaair: Oh no…. no. Chris, *((leaning over to see Zaair's computer)):* What the heck? Zaair: Yeah, uh… I'm gonna go back to this. *((Clicks "setup2" button))* Zaair: Okay, so it's gonna be… How do I let somebody let go of a link?
2		Code Tab: changes *Ask turtle 0 [create-link-with turtle 1]* *Ask turtle 1 [create-link-with turtle 2]* *Ask turtle 2 [create-link-with turtle 3]* *Ask turtle 3 [create-link-with turtle 0]* to *Ask turtle 0 [create-link-with turtle 2]* *Ask turtle 2 [create-link-with turtle 3]* *Ask turtle 3 [create-link-with turtle 0]*		Zaair *((Hovers cursor over the bottom right turtle)):* So that's turtle 1. *((Inspects turtle 3))* So that's turtle 3.
3		Code Tab: changes to *Ask turtle 0 [create-link-with turtle 2]* *Ask turtle 2 [create-link-with turtle 0]*		Zaair: YES. Ok.

intention came through Harrison's unique appreciations, which are often variable between people. Through his encounter with this incompatibility, Harrison struggled, but also developed a more detailed articulation of his criteria for a good representation, which led him to further refine his intention, and provoked deeper, sustained engagement with the model.

Appreciation in non-action

The few times that students did not react to incompatibilities are also examples of students making appreciative judgements about what constituted an adequate representation of their dance. These incompatibilities were most often mismatches that were created while addressing a different incompatibility; that is, students permitted certain representational inconsistencies to remain in their models in order to avoid others. For example, in Zaair's choreography, two people from either side of the Mylar let go and crossed underneath, trading places. In his first attempt at coding this move, he sent one turtle to the coordinate that he wanted it to cross to. However, while the turtle successfully moved to his intended spot, the links followed the turtle, creating a figure-eight shape (Table 3 row 1). In contrast, in the physical enactment of this movement, the Mylar retained its square shape. This created an incompatibility between Zaair's expectation for how turtles/links would behave and how humans/Mylar behaved. Both Zaair and another member of his quartet, Chris, vocalized their dissatisfaction with his model:

1. Zaair: Oh no… no.
2. Chris: *((leaning over to see Zaair's computer))* What the heck?
3. Zaair: Yeah, uh… I'm gonna go back to this.

Zaair attended to the incompatibility and was satisfied when he learned to edit the code to link only three out of the four turtles, leaving the turtle that he wanted to cross underneath without a link (Table 3 row 2). He continued this process by disconnecting a second turtle, leaving only two turtles connected via links (Table 3 row 3). However, in making these changes, Zaair created another incompatibility that did not bother him— his virtual Mylar snapped to

Table 4. Relationship between noticing incompatibilities and coding productivity.

Student	% of Incompatibilities Noticed	Total # of New Commands (# with Teacher Help)	# (%) of New Commands Tried in Response to Incompatibilities
Xavier	11.11%	9 (0)	0 (0%)
Chris	66.67%	11 (0)	2 (18.2%)
Jonah	66.67%	13 (0)	3 (23.1%)
Zaair	72.73%	16 (2)	7 (43.8%)
Kyle	73.33%	20 (2)	11 (55%)
Harrison	92.31%	21 (2)	8 (38.1%)

become a straight line, even though in the physical world, it would have billowed and moved in a way not captured by this model. In this case, his reaction was the opposite of dissatisfaction. He said, "yes, okay!", and began working on the next part of the choreography. In this case, Zaair resolved an incompatibility that he encountered, and in doing so, he attuned to one aspect of the representation (having people separate from the Mylar) and left out another (showing the movement of the Mylar). Zaair exercised appreciation in deciding that his model was good enough to move on.

Taken as a whole, it appeared that incompatibilities were recognized and acted on as an act of appreciation, which was personal and differed between students. As students evaluated their designs, this also pushed them to, at times, set a new intention for their design, inviting students to adapt their code to address them in some way. Rather than causing students to give up, they appeared to do just the opposite, resulting in repeated cycles of appreciation and intention setting. Incompatibilities that were *not* resolved could also be understood in terms of appreciation, as students' attention was on a different representational element, or they decided that they were satisfied with their model.

RQ2: the influence of incompatibilities

Our second research question asked how incompatibilities might provoke continued engagement with the activity of design. We were curious about whether or how incompatibilities influenced the ways that students engaged with the activity. Did students persist by identifying specific features of their choreography to focus on representing in NetLogo?

Overall, it appeared that encountering incompatibilities led students to make new representational choices: as they attended to incompatibilities, they tried new syntax, commands, and tools available in NetLogo. In other words, although students experienced failure when their initial code produced an incompatibility, resolving it led to deeper engagement with NetLogo. Table 4 includes the total number of new commands/tools that each student utilized during the activity as a measure of their coding productivity, and the number/percentage of new commands/tools that students used directly in response to incompatibilities. Five out of six students tried new commands in NetLogo to address incompatibilities, and for many, it was a substantial part of their coding.

Beyond calling for new commands, incompatibilities required generating new computational ideas. In order to engage with the task, students needed to explore possibilities in NetLogo. In the following section, we present two examples that demonstrate the ways in which students moved through cycles of design as they encountered and addressed incompatibilities. In the first case, we highlight the ways in which students utilized new commands and tools in order to address incompatibilities. In the second, we highlight the ways in which incompatibilities inspired students to move in a new, perhaps unexpected, direction with their design.

Productive struggle in design

Incompatibilities, while initially viewed as challenges, prompted students to explore new computational ideas. For example, when Chris encountered an incompatibility, he generated an alternative setup that allowed him to separate people from Mylar. In Chris's choreography, two

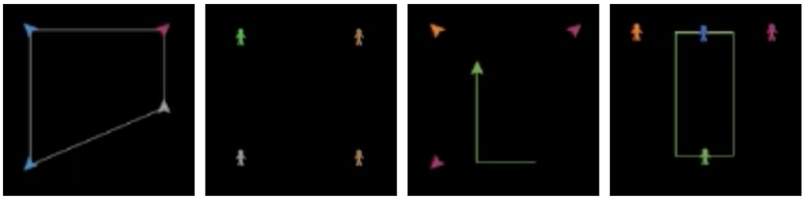


Figure 4. Chris's NetLogo choreography as he encountered and addressed an incompatibility.

people held onto the Mylar, brought it up in the air, and the other two people crossed underneath, switching sides. Chris started his coding process by setting an intention to move one turtle to the center of one of the sides, grappling with calculating the angle that a turtle would need to turn in order to face one of the sides. His next command, ask turtle 1 [fd 3], caused the turtle to move where he wanted it to go, but also had the consequence of distorting the shape of the Mylar, as the turtles were still linked (Figure 4 image 1). Expressing dissatisfaction with this initial failure to achieve his goals, Chris set a new intention: separating people from Mylar. He set the shape of his turtles to be people, suggesting a decision to treat the turtles as people rather than corners of the Mylar and commented out the code that created the links (Figure 4 image 2). In doing so, Chris encountered a second incompatibility, as his model showed the four quartet members but now did not include any representation of the Mylar. In this back and forth between decisions and outcomes, or failure and redesign, Chris explored and revised his own goals and ideas, in iterative conversation with the resistances of NetLogo. These intention-appreciation cycles seem likely to be a mechanism for learning through the act of design, specifically through the development and refinement of appreciative systems. Schön and Wiggins (1992) describe a very similar experience in a close analysis of a student named Petra. They write: "One might say that [Petra's] appreciative system enables her to recognize unintended consequences and qualities of the change she has made. One might also say that her ability to recognize features of the new configuration gives her access to parts of her appreciative system that might not otherwise come into play" (p. 140).

In this case, the repeated cycles prompted Chris to utilize the turtles in a new way— using their built-in pen function to draw the Mylar himself (Figure 4 images 3 and 4). This was completely new code that met his intention of allowing the turtles to let go of the Mylar. In this way, we see Chris learning as his appreciative systems developed over time, and as he needed to use a new set of tools to realize his design intentions. Chris seemed to experience a productive struggle: he encountered struggle in the form of incompatibilities, but ultimately set new intentions and utilized new functionality in NetLogo.

Developing new appreciative systems

After encountering an incompatibility, students engaged in a process of design by setting new intentions. While students primarily set intentions to focus on features of the choreography that they deemed important, there were some cases where students were inspired by the feedback of NetLogo and set new intentions that changed the task and extended their model in new directions artistically. Here, the incompatibilities prompted a positive, rather than negative, judgment and catalyzed students to move in a different direction in their design process. For example, Kyle chose to represent the person/Mylar system using a single turtle with shape set to "square," and as he made his turtle grow and rotate, it eventually became so large that it wrapped (Figure 5). When Kyle encountered this surprising feature of NetLogo, he said, "woah, that's so cool," capitalized on the incompatibility, and changed his intention by adding more square turtles, which created an effect that he found aesthetically pleasing. Kyle appreciated and amplified the incompatibility, deviating from his original choreography and making new, aesthetic choices. His intention changed from representing his choreography to creating a work of art of another form (Bennett & Eglash, 2013).

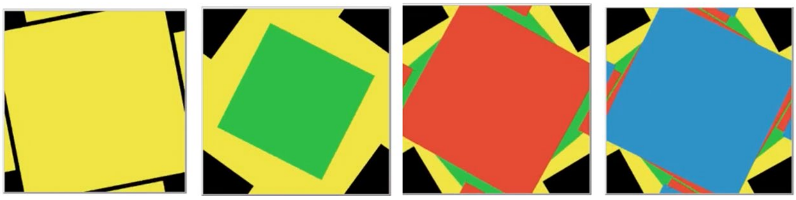


Figure 5. Kyle's new intentions led to an aesthetic expansion of an incompatibility.

Other students also commented on the aesthetic qualities of their models. As Harrison tried to figure out rotating, he expressed, "That's a cool animation but it's not what we need." When Jonah encountered the world wrapping, he said, "Hahah, I ripped it into pieces!" However, in these cases, students' appreciative judgements did not provoke changes in intention, and their aesthetic noticings did not make it into the students' final models. While not all movements that students explored made it into their coded choreography, it seemed that the act of exploring them helped students to broaden what they understood as possible in NetLogo (Brady, 2018). Students' judgements provoked new intentions that reengaged them in the design cycle.

Incompatibilities were generative for students in that they catalyzed the exploration of new syntax in NetLogo. While students sometimes liked the way that incompatibilities looked, they ultimately moved toward creating a NetLogo performance that, to them, represented their dance. These personal decisions and variable appreciative systems led to model diversity, even among students from the same quartet (who had developed the same dance choreography). For example, to show letting go, Zaair deleted the links completely and sacrificed the shape of the Mylar, while Chris drew the Mylar using the pen function. Incompatibilities created opportunities for choice, which led to diversity in design, more expressivity in their work, and the agency to make design decisions.

Discussion and conclusion

In this analysis, we have explored how students responded to challenge in the task of designing computational translations, and how those responses seemed to be mediated by being positioned as designers—both of their choreography and then of their own translations. Through these analyses, we argue that being positioned as a designer was legitimately accomplished with respect to the broader practices of design as outlined by Schön and Wiggins (1992), and more importantly, that such positioning led to both persistence and learning.

As noted, the translation task posed to the students was in some ways impossible: compromises in representational translation had to be made. Indeed, as students ran their code, the computational representation demonstrated its capacity to surprise, and not always in ways that were gratifying. While sometimes frustrating, encountering incompatibilities appeared to inspire new ideas. In this way, students experienced a form of productive failure (Kapur, 2008; Kapur & Bielaczyc, 2012), in which incompatibilities prompted further engagement with NetLogo concepts. Despite the many challenges that students encountered, they persisted until they had a model that was satisfying to them. Brady (2018) suggests that it might be desirable to pose problems where learners encounter limitations to prompt them to move beyond the target concepts and toward building more sophisticated understandings. Likewise, in encountering the differences between digital and physical modalities, our students explored new programming ideas and ultimately, strengthened their understanding of coding possibilities in NetLogo. The openness of the act of translation allowed students to exercise intention and appreciation in ways that were often divergent from other group members who were working on the same translational task. These points of tension were productive for engagement with computer science and should be considered as potential benefits, rather than limitations, of design work.

It is not always that case that encountering challenges leads to persistence—indeed, often challenge and unanticipated outcomes can lead to frustration, dissatisfaction, and even withdrawing from the activity. We wondered what made this different—what about the experience of translating code led students not to drop out, but to lean in? In this activity students were invited to re-represent their own designs, rather than to model systems phenomena found in the world. This opened a different space to exercise appreciation, with the designers themselves having the authority to determine when their models were satisfactory representations (c.f. Jasien & Gresalfi, 2021). Danish and Enyedy (2007) suggest that understanding students' representational practices requires examining how representational practices are seen as legitimate in a student context. In our context, students needed to create a model that others could understand and turn back into a dance; therefore, they often created representations they believed captured their choreography. However, this did not preclude students from exploring aesthetics in their models.

These new kinds of seeing create profound opportunities for learning— in this case about both the language of NetLogo and about the relationship between commands and representational outcomes. Each time students made judgements about the way NetLogo interpreted their code, they also noticed something new or different that they had previously not realized. This prompted shifts in intention, thus prompting further discovery. In doing this, students generated new codes and proposed novel solutions that they found to be personally satisfying. This back-and-forth movement between goals and outcomes is of course a core aspect of coding, similar to the process of debugging, when students make modification to code in order to get it to perform properly (O'Dell, 2017; Rich et al., 2019). Although debugging is a central and valued aspect of programming, we are continuing to explore how to support students to learn how to debug and to persist in doing so. These data offer a contribution to our emergent understanding of the conditions that could support students' sustained engagement in the process.

Echoing the work of Bennett and Eglash (2013), we note that the integration of programming and choreography was not merely an opportunity to invite participation, but to transform and sustain it. Bennett & Eglash write:

> Art in computing education is generally limited to a role as motivator or "content provider;" something to attract student interest. While a project to create an online gallery might be inspiring for some students, it still keeps a conceptual barrier in place between the artistic and technical process. Allowing students to see computational thinking in the artistic works themselves changes the status of both art and computing in the mind of the student; it makes computation itself available as a medium of artistic expression (p. 45).

Here too, we have seen the computational environment become an expressive environment through its interaction with the medium of dance; the representational translation challenge from physical to computational modes invited students to re-interpret their designs, and engage in new ways with underlying ideas—not to circumvent expressivity, but to transform it.

Limitations

This analysis has limitations that create new questions for future work. First, we explored a single programming environment (NetLogo), which has unique affordances. It is unclear whether these same kinds of tensions would support students' engagement in other environments. This is an important question for future research, as the field continues to engage in conversations about students' learning of programming across different platforms (Xu et al., 2019). Additionally, we did not attend to the ways in which the physical embodiment of the dance influenced students' coding practices (c.f. Fadjo, 2012; Kopcha et al., 2021). Future research might consider the question or whether or how the physical embodiment served as a resource for thinking.

In addition, our sample was composed of male-identifying students. We did not consider why that might be or the role that gender might have played in students' experiences. Also, although there was racial diversity in our sample, we did not describe students in relation to that diversity as we had little information to help us to understand the potential role of race. Instead, we treated the cases as a single collective. While useful from the perspective of exploring the overall affordances of the design task, this might have caused us to miss ways that the classroom structure positioned students differently based on their visible identities. Future research should consider variations in students' experiences; for example, Xavier seemed different from the other students for many reasons, and he may be a useful single case for future analysis.

Note

1. Transcription conventions: Throughout this paper, … denote pauses in speech. *((activity descriptions))* appear in italics within double parentheses.

Acknowledgments

The design of the Movement & Coding Camp analyzed in this paper came from the artistic histories and practices of Rebecca Steinberg & Curtis Thomas. The Camp was run by four math teachers—Nikki Newman Higdon, Shannon Reider, Hannah Strickland, and Kate Tarne—whose enthusiasm and pedagogical choices made these activities really dance. This article benefited from feedback by our research group: Amanda Bell, Madison Knowe, and Candice Love. This work was supported by Grant number (DRL-1742257) from the National Science Foundation. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Declaration of interest

We have no conflicts of interests to disclose.

Funding

NSF. Funding Body ID: 20. Grant ID DRL-1742257.

Notes on contributors

Selena Steinberg is a graduate student studying Learning Sciences at Indiana University. She is interested in exploring the potential of the arts and embodiment for supporting students' engagement with and identities in STEM.

Melissa Gresalfi is a professor of Mathematics Education and Learning Sciences at Vanderbilt University. Her research considers how interactions, contexts, and broader narratives organize the identities students develop in STEM, and how learning environments can be transformed to support the development of more productive relationships with STEM disciplines.

Lauren Vogelstein is a Postdoctoral Associate at New York University. She studies how people learn from an embodied and interactionist perspective that leverages the artistic practices of dancers and choreographers in order to better design expansive STEM learning environments for students.

Corey Brady is an Assistant Professor of the Learning Sciences at Vanderbilt University. He studies computational and mathematical modeling, with a focus on supporting and studying learning environments in which groups of learners find they can accomplish things together that they cannot do individually.

ORCID

Selena Steinberg http://orcid.org/0000-0003-0032-8957
Melissa Gresalfi http://orcid.org/0000-0002-3307-3914
Lauren Vogelstein http://orcid.org/0000-0002-2317-2513
Corey Brady http://orcid.org/0000-0002-4086-9638

References

Allen-Handy, A., Ifill, V., Schaar, R. Y., Rogers, M., & Woodard, M. (2020). Black girls STEAMing through dance: Inspiring STEAM literacies, STEAM identities, and positive self-concept. In K. Thomas & D. Huffman (Eds.), *Challenges and opportunities for transforming from STEM to STEAM education* (pp. 198–219). IGI Global.

Bennett, A., & Eglash, R. (2013). cSELF (computer science education from life): Broadening participation through design agency. *International Journal of Web-Based Learning and Teaching Technologies*, *8*(4), 34–49. https://doi.org/10.4018/ijwltt.2013100103

Bennett, A., Eglash, R., Lachney, M., & Babbitt, W. (2016). Design agency: Diversifying computer science at the intersections of creativity and culture. In M. Raisinghani (Ed.), *Revolutionizing education through web-based instruction* (pp. 35–56). IGI Global.

Bergner, Y., Mund, S., Chen, O., & Payne, W. (2021). Leveraging interest-driven embodied practices to build quantitative literacies: A case study using motion and audio capture from dance. *Educational Technology Research and Development*, *69*(4), 2013–2036. https://doi.org/10.1007/s11423-020-09804-2

Brady, C. (2018). Modelling and the representational imagination. *ZDM*, *50*(1–2), 45–59. https://doi.org/10.1007/s11858-018-0926-4

Brady, C. (2021). Patches as an expressive medium for exploratory multi-agent modelling. *British Journal of Educational Technology*, *52*(3), 1024–1042. https://doi.org/10.1111/bjet.13087

Brady, C., Gresalfi, M., Steinberg, S., & Knowe, M. (2020). Debugging for art's sake: Beginning programmers' debugging activity in an expressive coding context. In M. Gresalfi, & I. S. Horn (Eds.), *The interdisciplinarity of the learning sciences, 14th International Conference of the Learning Sciences (ICLS) 2020* (Vol. *3*, pp. 1229–1236). International Society of the Learning Sciences.

Brady, C., Orton, K., Weintrop, D., Anton, G., Rodriguez, S., & Wilensky, U. (2016). All roads lead to computing: Making, participatory simulations, and social computing as pathways to computer science. *IEEE Transactions on Education*, *60*(1), 59–66.

Brady, C., Stroup, W., Petrosino, A., & Wilensky, U. (2018). Group-based simulation and modelling: Technology supports for social constructionism. In *Proceedings of Constructionism*.

Buechley, L., Eisenberg, M., Catchen, J., & Crockett, A. (2008). *The LilyPad Arduino: Using computational textiles to investigate engagement, aesthetics, and diversity in computer science education* [Paper presentation]. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 423–432).

Champion, D. (2018). *The STEAM dance makerspace: A context for integration: An investigation of learning at the intersections of STEM, art, making and embodiment* [PhD dissertation, Northwestern University].

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. https://doi.org/10.1002/sce.20166

DesPortes, K., Payne, W., & Bergner, Y. (2021). danceOn: Exploring embodiment in a dance and computing learning application. Presented at the International Society of the Learning Sciences Annual Meeting 2021.

DesPortes, K., Spells, M., & DiSalvo, B. (2016). *Interdisciplinary computing and the emergence of boundary objects: A case-study of dance and technology*. International Society of the Learning Sciences.

Edmonds, E. (2019). Programming as art. In E. Edmonds & M. Boden (Eds.), *From fingers to digits, an artificial aesthetic* (pp. 209–222). MIT Press.

Fadjo, C. L. (2012). Developing computational thinking through grounded embodied cognition [Doctoral dissertation, Columbia University].

Farris, A. V., Dickes, A. C., & Sengupta, P. (2016). Development of disciplined interpretation using computational modeling in the elementary science classroom. In C. K. Looi, J. L. Polman, U. Cress, & P. Reimann (Eds.), Transforming learning, empowering learners: The International Conference of the Learning Sciences (ICLS) 2016, Volume 1. Singapore: International Society of the Learning Sciences.

Flesch, B., Gabaldón, C., Nabity, M., & Thomas, D. (2021). Choreographing increased understanding and positive attitudes towards coding by integrating dance. *International Journal of Computer Science Education in Schools*, *4*(3), n3.

Franklin, D., Coenraad, M., Palmer, J., Eatinger, D., Zipp, A., Anaya, M., White, M., Pham, H., Gökdemir, O., Weintrop, D. (2020). An analysis of use-modify-create pedagogical approach's success in balancing structure and student agency. In *Proceedings of the 2020 ACM Conference on International Computing Education Research* (pp. 14–24).

Garneli, V., Giannakos, M. N., & Chorianopoulos, K. (2015). *Computing education in K-12 schools: A review of the literature* [Paper presentation]. 2015 IEEE Global Engineering Education Conference (EDUCON) (pp. 543–551). IEEE. https://doi.org/10.1109/EDUCON.2015.7096023

Gerofsky, S. (2010). Mathematical learning and gesture: Character viewpoint and observer viewpoint in students' gestured graphs of functions. *Gesture*, *10*(2–3), 321–343. https://doi.org/10.1075/gest.10.2-3.10ger

Giri, S. (2022, March). Supporting novice learners' coding through productive failure-based debugging activities. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education* (Vol. *2*, pp. 1175–1175).

Gravel, B. E., & Svihla, V. (2021). Fostering heterogeneous engineering through whole-class design work. *Journal of the Learning Sciences*, *30*(2), 279–329. https://doi.org/10.1080/10508406.2020.1843465

Greeno, J. G., & Hall, R. P. (1997). Practicing representation: Learning with and about representational forms. *Phi Delta Kappan*, *78*, 361–367.

Griffin, S. M., Rowsell, J., Winters, K. L., Vietgen, P., McLauchlan, D., & McQueen-Fuentes, G. (2017). A reason to respond: Finding agency through the arts. *International Journal of Education & the Arts*, *18*(25), 1–23.

Hall, R. (1996). Representation as shared activity: Situated cognition and Dewey's cartography of experience. *Journal of the Learning Sciences*, *5*(3), 209–238. https://doi.org/10.1207/s15327809jls0503_3

Hall, R., Ma, J. Y., & Nemirovsky, R. (2014). Rescaling bodies in/as representational instruments in GPS drawing. In V.R. Lee (Ed.), *Learning technologies and the body* (pp. 124–143). Routledge.
Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, *28*(1), 75–105. https://doi.org/10.2307/25148625
Jayathirtha, G., & Kafai, Y. B. (2020). Interactive stitch sampler: A synthesis of a decade of research on using electronic textiles to answer the who, where, how, and what for K–12 computer science education. *ACM Transactions on Computing Education*, *20*(4), 1–29. https://doi.org/10.1145/3418299
Jasien, L., & Gresalfi, M. (2021). The role of participatory identity in learners' hybridization of activity across contexts. *Journal of the Learning Sciences*, *30*(4-5), 676–706.
Jonassen, D. H. (2000). Toward a design theory of problem solving. *Educational Technology Research and Development*, *48*(4), 63–85. https://doi.org/10.1007/BF02300500
Jörg, S., Leonard, A. E., Babu, S., Gundersen, K., Parmar, D., Boggs, K., & Daily, S. B. (2014). *Character animation and embodiment in teaching computational thinking* [Paper presentation]. ACM Siggraph 2014 Posters (pp. 1–1). https://doi.org/10.1145/2614217.2630597
Kafai, Y. B. (2016). From computational thinking to computational participation in K-12 education. *Communications of the ACM*, *59*(8), 26–27. https://doi.org/10.1145/2955114
Kafai, Y. B., DeLiema, D., Fields, D. A., Lewandowski, G., & Lewis, C. (2019, February). *Rethinking debugging as productive failure for CS education* [Paper presentation]. Proceedings of the 50th ACM Technical Symposium on Computer Science Education (pp. 169–170).
Kafai, Y. B., Lee, E., Searle, K., Fields, D., Kaplan, E., & Lui, D. (2014). A crafts-oriented approach to computing in high school: Introducing computational concepts, practices, and perspectives with electronic textiles. *ACM Transactions on Computing Education*, *14*(1), 1–20. https://doi.org/10.1145/2576874
Kapur, M. (2008). Productive failure. *Cognition and Instruction*, *26*(3), 379–424. https://doi.org/10.1080/07370000802212669
Kapur, M., & Bielaczyc, K. (2012). Designing for productive failure. *Journal of the Learning Sciences*, *21*(1), 45–83. https://doi.org/10.1080/10508406.2011.591717
Kapur, M., & Kinzer, C. K. (2009). Productive failure in CSCL groups. *International Journal of Computer-Supported Collaborative Learning*, *4*(1), 21–46. https://doi.org/10.1007/s11412-008-9059-z
Kelton, M. L., & Ma, J. Y. (2020). Assembling a torus: Family mobilities in an immersive mathematics exhibition. *Cognition and Instruction*, *38*(3), 318–347.
Kopcha, T. J., Ocak, C., & Qian, Y. (2021). Analyzing children's computational thinking through embodied interaction with technology: A multimodal perspective. *Educational Technology Research and Development*, *69*(4), 1987–2012. https://doi.org/10.1007/s11423-020-09832-y
Kremling, J., Arndt, N., Roman, A., Zilm, M., & Straehler-Pohl, H. (2018). Dancing on the junction of mathematics and bodily experience. *The Mathematics Enthusiast*, *15*(1–2), 201–227. https://doi.org/10.54870/1551-3440.1424
Leonard, A. E., & Daily, S. B. (2014). The Dancing Alice Project: Computational and embodied arts research in middle school education. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*.
Leonard, A. E., Daily, S. B., Jörg, S., & Babu, S. V. (2021). Coding moves: Design and research of teaching computational thinking through dance choreography and virtual interactions. *Journal of Research on Technology in Education*, *53*(2), 159–177. https://doi.org/10.1080/15391523.2020.1760754
Lewis, C. M., & Shah, N. (2015). *How equity and inequity can emerge in pair programming* [Paper presentation]. Proceedings of the Eleventh Annual International Conference on International Computing Education Research (pp. 41–50).
Litts, B. K., Kafai, Y. B., Searle, K. A., & Dieckmeyer, E. (2016). *Perceptions of productive failure in design projects: High school students' challenges in making electronic textiles*. International Society of the Learning Sciences.
Litts, B. K., Searle, K. A., Brayboy, B. M. J., & Kafai, Y. B. (2020). Computing for all?: Examining critical biases in computational tools for learning. *British Journal of Educational Technology*, 52(2), 842–857.
Lytle, N., Catete, V., Isvik, A., Boulden, D., Dong, Y., Wiebe, E., & Barnes, T. (2019). *From 'use' to 'choose': Scaffolding CT curricula and exploring student choices while programming (Practical Report)* [Paper presentation]. Proceedings of the 14th Workshop in Primary and Secondary Computing Education (pp. 1–6).
Ma, J. Y. (2016). Designing disruptions for productive hybridity: The case of walking scale geometry. *Journal of the Learning Sciences*, *25*(3), 335–371. https://doi.org/10.1080/10508406.2016.1180297
Ma, J. Y. (2017). Multi-party, whole-body interactions in mathematical activity. *Cognition and Instruction*, *35*(2), 141–164. https://doi.org/10.1080/07370008.2017.1282485
MacDonald, B., & Walker, R. (1975). Case-study and the social philosophy of educational research. *Cambridge Journal of Education*, *5*(1), 2–11. https://doi.org/10.1080/0305764750050101
O'Dell, D. H. (2017). The Debugging Mindset: Understanding the psychology of learning strategies leads to effective problem-solving skills. *Queue*, *15*(1), 71–90.
Owen, C. B., Dobbins, A., Lansing, E., Lansing, E., Keppers, N., Levinson, M., & Rhodes, M. (2016). *Dancing computer: Computer literacy through dance* [Paper presentation]. Proceedings of the 14th International Conference on Advances in Mobile Computing and Multi Media (pp. 174–180).
Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books.

Payne, W. C., Bergner, Y., West, M. E., Charp, C., Shapiro, R. B. B., Szafir, D. A., Taylor, E. V., & DesPortes, K. (2021, May). *danceON: Culturally responsive creative computing* [Paper presentation]. Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (pp. 1–16).
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., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., & Kafai, Y. (2009). Scratch: Programming for all. *Communications of the ACM*, *52*(11), 60–67. https://doi.org/10.1145/1592761.1592779
Rich, K. M., Strickland, C., Binkowski, T. A., & Franklin, D. (2019, February). *A K-8 debugging learning trajectory derived from research literature* [Paper presentation]. Proceedings of the 50th ACM Technical Symposium on Computer Science Education (pp. 745–751). https://doi.org/10.1145/3287324.3287396
Ryoo, J. J. (2019). Pedagogy that supports computer science for all. *ACM Transactions on Computing Education*, *)19*(4), 1–23. https://doi.org/10.1145/3322210
Saldana, J. (2011). *Fundamentals of qualitative research*. Oxford university press.
Schön, D. A., & Wiggins, G. (1992). Kinds of seeing and their functions in designing. *Design Studies*, *13*(2), 135–156. https://doi.org/10.1016/0142-694X(92)90268-F
Shamir, M., Kocherovsky, M., & Chung, C. (2019). *A paradigm for teaching math and computer science concepts in k-12 learning environment by integrating coding, animation, dance, music and art* [Paper presentation]. 2019 IEEE Integrated STEM Education Conference (ISEC) (pp. 62–68). IEEE. https://doi.org/10.1109/ISECon.2019.8882015
Solomon, F., Vogelstein, L., Brady, C., Steinberg, R., Thomas, C., Champion, D., Lindberg, L., Enyedy, N., DesPortes, K., Payne, W., Bergner, Y., Taylor, E., & Shapiro, R. B. (2021). Embodying STEM: Learning at the intersection of Dance and STEM. In E. de Vries, Y. Hod, & J. Ahn (Eds.), *Proceedings of the 15th International Conference of the Learning Sciences - ICLS 2021* (pp. 819–826). International Society of the Learning Sciences.
Stake, R. E. (1995). *The art of case study research*. Sage.
Strauss, A., & Corbin, J. (1990). *Basics of qualitative research*. Sage publications.
Vogelstein, L., & Brady, C. (2019). Taking the patch perspective: A comparative analysis of a patch based participatory simulation. *Computer-Supported Collaborative Learning*, *1*, 512–519.
Vogelstein, L., Brady, C., & Hall, R. (2019). Reenacting mathematical concepts found in large-scale dance performance can provide both material and method for ensemble learning. *ZDM*, *51*(2), 331–346.
Vogelstein. L., Brady, C., Steinberg, R., Thomas, C. (2021). Flares in the Soup Game: Improvisational Collective Choreography and Computational Expressivity. In *the Symposium Expansive Modeling: Broadening the scope of modeling in K-12 education in the proceedings of the International Society of the Learning Sciences Annual Meeting*.
Warshauer, H. K. (2015). Productive struggle in middle school mathematics classrooms. *Journal of Mathematics Teacher Education*, *18*(4), 375–400. https://doi.org/10.1007/s10857-014-9286-3
Weintrop, D., & Wilensky, U. (2019). Transitioning from introductory block-based and text-based environments to professional programming languages in high school computer science classrooms. *Computers & Education*, *142*, 103646. https://doi.org/10.1016/j.compedu.2019.103646
Wilensky, U. (1999). *NetLogo*. Center for Connected Learning and Computer-Based Modeling, Northwestern University.
Wilensky, U. J., & Stroup, W. (1999). Learning through participatory simulations: Network-based design for systems learning in classrooms. In C. M. Hoadley & J. Roschelle (Eds.), *Proceedings of the Computer Support for Collaborative Learning (CSCL) 1999 Conference*. International Society of the Learning Sciences.
Wilensky, U., & Reisman, K. (2006). Thinking like a wolf, a sheep, or a firefly: Learning biology through constructing and testing computational theories—An embodied modelling approach. *Cognition and Instruction*, *24*(2), 171–209. https://doi.org/10.1207/s1532690xci2402_1
Wright, S. (2015). *Children, meaning-making and the arts*. Pearson Higher Education AU.
Xu, Z., Ritzhaupt, A. D., Tian, F., & Umapathy, K. (2019). Block-based versus text-based programming environments on novice student learning outcomes: A meta-analysis study. *Computer Science Education*, *29*(2–3), 177–204. https://doi.org/10.1080/08993408.2019.1565233
Yin, R. K. (2014). *Case study research design and methods* (5th ed.). Sage.