Routledge
Taylor & Francis Group

Interactive Learning Environments

ISSN: 1049-4820 (Print) 1744-5191 (Online) Journal homepage: https://www.tandfonline.com/loi/nile20

Generative computing: African-American cosmetology as a link between computing education and community wealth

Michael Lachney, William Babbitt, Audrey Bennett & Ron Eglash

To cite this article: Michael Lachney, William Babbitt, Audrey Bennett & Ron Eglash (2019): Generative computing: African-American cosmetology as a link between computing education and community wealth, Interactive Learning Environments, DOI: 10.1080/10494820.2019.1636087

To link to this article: https://doi.org/10.1080/10494820.2019.1636087

Published online: 03 Jul 2019.

INTERACTIVE LEARNING ENVIRONMENTS
https://doi.org/10.1080/10494820.2019.1636087

Generative computing: African-American cosmetology as a link between computing education and community wealth

Michael Lachney [a], William Babbitt [b], Audrey Bennett [c] and Ron Eglash [d]

[a]Counseling, Educational Psychology, and Special Education, Michigan State University, East Lansing, MI, USA; [b]Science and Technology Studies, Rensselaer Polytechnic Institute, Troy, NY, USA; [c]Penny W Stamps School of Art and Design, University of Michigan, Ann Arbor, MI, USA; [d]School of Information, University of Michigan, Ann Arbor, MI, USA

ABSTRACT

Recent scholarship in computer science (CS) education shifts from a focus on the technical-cognitive skills of *computational thinking* to the socio-cultural goal of *computational participation*, often illustrated as remixing popular media (e.g. music, photos, etc.) in online communities. These activities do enhance the participatory dimensions of CS, but whether they also support broadening the participation of underrepresented youth remains unclear. While online communities that are dedicated to computational participation have existed in the U.S. for over a decade, many communities of color remain underrepresented in CS disciplines. How might CS educators, researchers, and technologists promote culturally responsive forms of computational participation? To answer this question, we propose a culturally responsive framework for computational participation called *generative computing*. Generative computing approaches CS as a means for strengthening relationships between learning environments and local communities, leveraging culturally relevant sources of wealth generation in technology design and implementation. To explore this concept, we conducted a mixed-methods study with a cosmetology high school program that predominantly serves young African-American women. Through a series of computationally and culturally rich cosmetology projects, we tested our hypothesis that generative computing can enhance connections between Black heritage, CS, and cosmetology while supporting students' academic interests and knowledge.

ARTICLE HISTORY
Received 12 December 2018
Accepted 11 May 2019

KEYWORDS
Computer science education; computational thinking; computational participation; cosmetology; broadening participation

Introduction

In efforts to broaden the scope of computer science (CS) education and diversify points of entry for youth, Kafai and Burke (2014) recommend a shift from *computational thinking* to *computational participation,* which they define as "the practices and perspectives that are needed to contribute within wider social networks, including but not limited to schools" (9). However, Kafai (2016) cautions that "It is not possible to [address] all of the difficulties of implementing computational participation by placing students in groups, having them program applications, and encouraging them to remix code" (27). One such difficulty is addressing the significant underrepresentation of many communities of color. While remixing does enhance the participatory dimensions of CS, how such activities can simultaneously support broadening the participation of underrepresented communities remains unclear. Even though online and offline programs dedicated to various technologically and

CONTACT Michael Lachney lachneym@msu.edu

historically situated instantiations of computational participation have existed in the United States for decades (Kafai & Burke, 2014), the representation of African-Americans in CS education and workforce remains significantly low (Zweben & Bizot, 2018). For example, in 2017, African-American women were less than 1% of the total female recipients of CS Masters degrees (Zweben & Bizot, 2018, p. 15).

While there are many economic and political factors that contribute to underrepresentation in CS, there is evidence to suggest that schools themselves play a significant role (Margolis, Estrella, Goode, Holme, & Nao, 2008). In addition to the fact that schools in African-American communities can lack culturally responsive curricula (Emdin, 2016), as early as primary and secondary school African-American girls are often criminalized, expected to conform to White middle-class expectations, and given the impression by some adults – who are supposed to be nourishing their potential – that academic success is out of their reach (Morris, 2016). This is exacerbated by wealth inequalities that African-American communities face due to histories of economic exclusion (Rothstein, 2017), the ongoing privatization of community assets (Lipman, 2011), the lack of representation in science, technology, engineering, and mathematics (STEM) fields – which leads to a dearth of role models – (Farinde & Lewis, 2012), and so on. Thus, because educational equity is so closely coupled with economic access (Anyon, 2014), improving CS education for African-American women is not only a matter of creating more culturally responsive curricula that are self-affirming, but also connecting CS to local sources of wealth generation in students' communities (Eglash, Bennett, Drazan, Lachney, & Babbitt, 2017).

Our guiding research questions to address these issues are: how might CS educators, researchers, and technologists promote culturally responsive forms of computational participation? And, how might these forms of computational participation not only support the educational interest and achievement of young women but also link up to sources of wealth generation that are important to local communities? As partial answers to these questions, we introduce a framework for broadening the participation of underrepresented communities in CS education that we call *generative computing*. Generative computing brings together research and design literature from culturally responsive computing (Bennett, 2016; Eglash, Gilbert, & Foster, 2013; Kafai, Searle, Martinez, & Brayboy, 2014; Lachney, 2017b; Pinkard, 1999; Scott, Sheridan, & Clark, 2015) with the theory of "generative justice" (Eglash, Babbitt, et al., 2017). We hypothesize that when youth participate in generative uses of computing, we can achieve the goals that Kafai (2016) intends with computational participation but in ways that are culturally responsive to local sources of wealth generation.

To explore this hypothesis, we detail a case of generative computing in a multimodal and community-oriented interactive learning environment: a four week after school cosmetology-based CS program that took place at an Upstate New York vocational high school. The program focused on a specific cultural area of computational and mathematical significance: the fantastical geometries of cornrow braiding in African-American traditions. Research shows that West African origins of popular hair braiding styles in the U.S., including cornrows, have mathematical and computational sophistication in their adaptive scaling and iterative patterns (Eglash, 1999). Also, beauticians and braiders have played important roles in political activism, community organizing, and resistance to White supremacy, while simultaneously being a local source of entrepreneurship and wealth generation (Gill, 2010). When the mathematical and computational significance of braiding is made explicit within this context, opportunities arise for CS to become responsive to not only students in school but also to the larger communities where students live and work.

Generative computing: a culturally responsive approach to computational participation

Generative computing can be understood as the computational application of "generative justice": "The universal right to generate unalienated value and directly participate in its benefits; the rights of value generators to create their own conditions of production; and the rights of communities of value generation to nurture self-sustaining paths for its circulation" (Eglash, Bennett, et al., 2017, p. 769).

Generative justice has been applied to a variety of contexts – from energy policy (Dotson & Wilcox, 2016) to bioremediation (Kellogg, 2016) – but it is most frequently associated with the development of culturally responsive STEM education (Cooke, 2016; Eglash, Babbitt, et al., 2017; Lyles, Lachney, Foster, & Zatz, 2016). While these scholars focus on STEM generally, they have taken a particular view of *computational thinking* that grounds it in already existing community assets (Bennett, 2016; Lachney, 2017a).

The ideas and concepts that make up computational thinking (e.g. algorithm, problem abstraction, decomposition, etc.) have a long history in CS dating back to the 1950s and 1960s (Denning, 2009). The term appears in the work of Papert (1980) but isn't popularized, defined, or operationalized until Wing (2006) decades later. While it has many definitions (Shute, Sun, & Asbell-Clarke, 2017), for the purposes of this paper computational thinking is defined as "the thought processes involved in formulating problems and their solutions so that the solutions are represented in the form that can be effectively carried out by an information-processing agent" (Cuny, Snyder, & Wing, 2010, cited in Yadav, Mayfield, Zhou, Hambrusch, & Korb, 2014). Below we will discuss two concepts that are central to computational thinking: *decomposition* – "Breaking a problem down into smaller, more manageable parts" – and *algorithm* – "A list of steps that can be followed to carry out a task" (Krauss, Prottsman, & Yongpradit, 2017, p. 173). As researchers and educators working in the tradition of teaching CS without computers – sometimes known as "CS Unplugged" (Bell & Newton, 2013) – can attest, these concepts do not limit computational thinking to the use of computers (Yadav, Stephenson, & Hong, 2017), but also craft practices, bodily movements, games, and other social and cultural practices that may appear familiar to many children and adults.

Kafai and Burke (2014) expand upon the concept of computational thinking by pointing out that a strict focus on the individual technical-cognitive skills of thinking limits what teaching CS and programing affords people in the twenty-first century, namely participation in online and offline communities. Emphasizing these socio-cultural affordances, they advocate for a shift from computational thinking to *computational participation*. Drawing on examples from online computing communities such as MIT's Scratch and offline computing communities such as the Computer Clubhouse, they contextualize this shift: "When computation is thought of in terms of participation and not just thinking, it becomes clear that there is a tremendous discrepancy in who gets to participate" (Kafai & Burke, 2014, pp. 9–10). Indeed, inequalities are not only reproduced through the digital divide, but also through who does and does not have the knowledge and skills to produce digital media and express themselves in community networks, sometimes known as the "participation gap" (Jenkins, Purushotma, Weigel, Clinton, & Robison, 2009).

Pointing out how technology education, in and out of school, reproduces larger economic and social injustices is important for the actions of social movements and development of social justice scholarship. But only framing the relationship between technology and knowledge as a "divide" or "gap" risks the imposition of *deficit thinking* on the communities in question. Generally understood, deficit thinking in educational contexts takes on an assimilationist logic that assumes young people and the adults in their communities must change and conform to outside institutions (e.g. federal and state government, philanthropic foundations, technology companies, etc.) if they want to obtain academic success (Yosso, 2005). Little focus is placed on the existing assets and sources of wealth generation that already exist within many low-income neighborhoods and communities of color. Educators and researchers have started to challenge such perceptions by drawing on literature from *culturally responsive computing* to show how computational thinking is already part of many community practices and traditions (Babbitt, Lachney, Bulley, & Eglash, 2015; Bennett, 2016; Eglash, Bennett, O'Donnell, Jennings, & Cintorino, 2006; Kafai et al., 2014; Lachney, 2017a).

Building on the goals of culturally responsive teaching to frame students' heritages, families, and personal identities as assets to teaching and learning (Gay, 2010), culturally responsive computing connects these assets with computational technologies and thinking to the benefit of both educational institutions and the communities that they serve (Lachney, 2017b). Culturally responsive computing comes in many forms, including initiatives that use and explore computing for social justice (Scott & White, 2013), democratizing CS (Ryoo, Margolis, Lee, Sandoval, & Goode, 2013), identity exploration (Vogel, Hoadley, Ascenzi-Moreno, & Menken, 2019), among many other applications. Bennett's (2016) approach to culturally responsive computing seeks to show how computational thinking already exists within cultural practices and artifacts of vernacular and indigenous communities; using the term "heritage algorithm" to make the point that algorithms are commonly used in cultural arts such as Native American quilting traditions, West African symbol systems, and African-American braiding (593).

Generative computing builds on Bennett's (2016) approach to culturally responsive computing by making heritage algorithms explicit in their connections to locations of wealth generation (e.g. braiding salons, artisan studios, etc.). Like computational participation, generative computing frames computational thinking in socio-cultural terms, but instead of only seeking connections from the outside it locates computational thinking within existing practices for aiding community goals (education, economy, health, etc.) and showing how those goals can benefit from computing technologies and CS knowledge. We hypothesize that as a framework for broadening participation, generative computing can make CS education more responsive to young people, as well as open up pathways for CS to make contributions to local entrepreneurship, culturally-situated businesses, and other community assets. Generative computing, then, aims to diversify not only who participates in the CS education pipeline, but also diversify the outputs of computational innovation.

Generative computing and cosmetology

In this paper, we focus on a generative computing initiative in which African-American cosmetologists and cosmetology students collaborated in the design and implementation of educational computing activities and tools. Cosmetology is well suited for generative computing because of its political, economic, educational, mathematical, and socio-cultural significance in African-American communities. Gill's (2010) historical research on African-American women in U.S. beauty industries reveals deep legacies of organizing for community change at grassroots levels. Building on the salon as a community asset, Majors (2015) shows how they are important sites of teaching and learning, where cultural literacies are passed down from one generation to the next. Also in relation to the salon, Wingfield (2008) details how ownership is a major entrepreneurial avenue for African-American women to gain financial mobility and generate wealth, while also recognizing that the salon industry is not immune from the gendered racism of larger entrepreneurial patterns and U.S. economic policies. There are many intersectional struggles around class, race, ability, and gender facing those in both the beauty industry and cosmetology education, including a disproportion of non-Black owned companies that make Black hair care products (Byrd & Tharps, 2014).

In the research described below, our team found that computational thinking and other STEM knowledges embedded in cosmetology practices and content were able to be used in an after-school program to the mutual benefit of both cosmetology and computing education. The primary means for making these connections was a suite of educational technologies called "Culturally Situated Design Tools" (CSDTs). The suite includes but is not limited to visual programing environments that help incorporate indigenous and vernacular knowledges into STEM lessons and classrooms (Eglash et al., 2006). This study focuses on one CSDT in particular, Cornrow Curves. The scaling patterns of cornrow braids are part of a larger body of African fractals that include self-similar architecture, recursion in textiles, iterative loops in divination symbol generation, and other computationally significant practices (Eglash, 1999).

It is critical to understand that CSDTs are not imposing Western math on "accidental" fractal patterns. Rather these are a deliberate, intentional body of mathematical and computational knowledge that arose independently of Europe. This does not mean it contains proofs and theorems; African fractals have their own means of communicative practices. The role of the CSDT simulations is to "translate" between different computational traditions. Because students and their teachers have been (mis)educated to think of African cultures as "primitive" societies in which sophisticated math and computing ideas are conspicuously absent, this perspective may not be immediately apparent. But as long as they use the Cornrow Curves learning environment to combine a cultural background of African and African-American hairstyling with the algorithmic aspects of cornrow braiding, the goal of shifting CS interests, skills, and understanding may still be met.

Braiding algorithms exist at several scales. In their original Indigenous context, society-wide coding of distinct patterns signified marital status, kinship, and age-grade initiations. In the U.S. context, one style might sweep the nation, or one salon might be known for a particular cluster of algorithms. Within any one particular rendering, a client might request some symmetry or curvature that organizes all braids across the scalp. Even within a single cornrow braid, it is mathematically significant in part because of the way that it is created with iterative applications of transformational geometry: each plait of the hair grows or diminishes progressively in size, angle, and translation. Cornrow Curves affords users opportunities to experiment with these parameters, reverse-engineering a known pattern to see how hairstylists have been aligning their cultural aesthetics with mathematicians' sense of iterative transforms, and developing a repertoire of what any computer scientist would understand as an algorithm.

Cornrow Curves: computational explorations of African-American heritage algorithms

The graphical user interface (GUI) of Cornrow Curves is a visual programing application – similar to MIT's Scratch where users design and produce media through dragging, dropping, and snapping together code blocks into a script (see Figure 1) – that is framed by the historical, cultural, social, and mathematical significance of cornrow braiding. The software application itself is a fork of Berkley's Snap! blocks based programing language. But unlike traditional Scratch and Snap! applications, parts of Cornrow Curves (e.g. certain blocks, sprites, etc.) were designed through collaborations with cornrow braiding and cosmetology experts (as well as feedback from teachers and students). When connecting cosmetology and CS through cornrow braiding, we aim to neither put a thin ethnic veneer on the same old lessons nor merely mix in more tech and stir.

When users first arrive at the Cornrow Curves landing page, they are greeted with a graphic image of an African American woman with a cornrow hairstyle. Next, background pages take a student

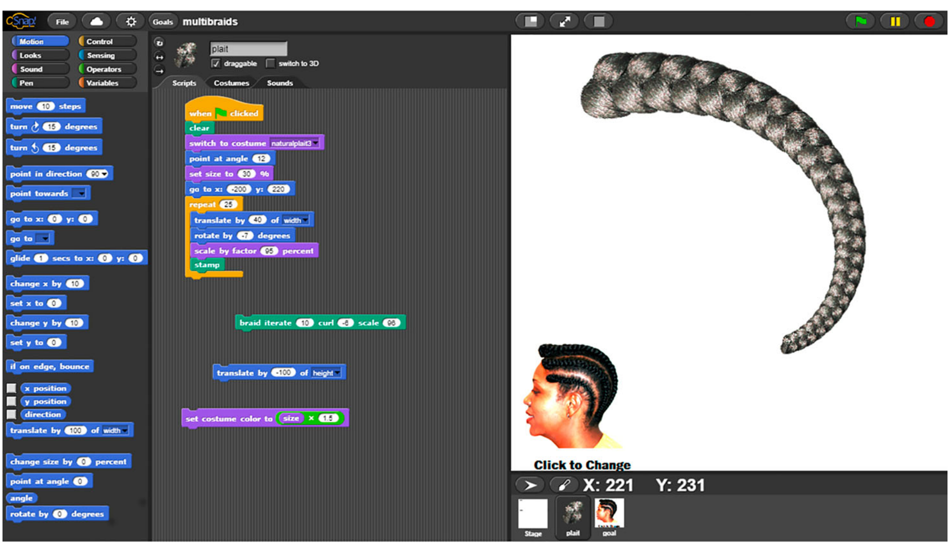


Figure 1. The GUI of Cornrow Curves.

through the history of cornrow braiding to contextualize its computational and mathematical significance. It starts with a page titled "African Origins," which connects the cultural values of braiding historically to how cornrows exemplify larger trends of African mathematics. Next, the page "Middle Passage" highlights the role of braiding and hairstyles in resisting the White supremacist erasure of African culture during the U.S. slave trade. The "Civil Rights" page explores the role of braiding as an African tradition that affirmed Black identity in African-American civil rights struggles of the 1950s and 1960s. The final background page, "Hip Hop" begins in the 1970s and goes through the 1990s to show how braiding has been central to Black cultural expression, with celebrities and artists continuing to innovate the styles for their own purposes and in their own contexts.

The heritage algorithm of cornrow braiding is represented in Cornrow Curves as a specific script of blocks, which students learn to assemble in a tutorial (along with blocks to change hair colors) and then can reverse engineer when they open the software (see Figure 2). When the visual programing application opens, the default script (see Figure 3) loads blocks that first clears the output screen, sets the plait image as active, points the angle of rotation at 12 degrees, sets the initial size of the costume to 30% of the actual size of the PNG file and locates the initial image at the Cartesian coordinates $x = -200$ and $y = 220$. Next, the code enters a loop that repeats 25 times and, in each iteration, translates the graphic image by 40% of the width of the image, rotates the image by -7 degrees, scales the image to 95% of the current size, and then stamps the image. As the loop code iterates, it creates the first cornrow braid on the output screen. The practice of creating this script is one of "translating" localized community knowledge into a formal equivalent that one might find in a mathematics or CS classroom without assimilating it so much that it becomes unrecognizable to braiders themselves.

For example, a math class would see the geometric transformation of translation as length units, whereas Cornrow Curves represents it in the percentage of plait width. This not only better reflects artisans' emic thinking ("Make each new plait consistent with the look of the last one") but also makes the script much easier (you would otherwise have to introduce a variable that scales the translation length so that it is consistent with the scaling in size). Emic goals do not always result in ease of scripting. For example, the addition of a single "braid" block (Figure 4) was created when we found that students interested in complex patterns involving multiple braids were creating scripts of unwieldy length and that it fit with braiders' understanding of each braid as a unit, as opposed to each plait or twist of hair.

The creation of a braid-specific block fits within larger scholarship on the educational uses of domain-specific modeling languages (DSMLs), often for creating "synergistic" lessons between

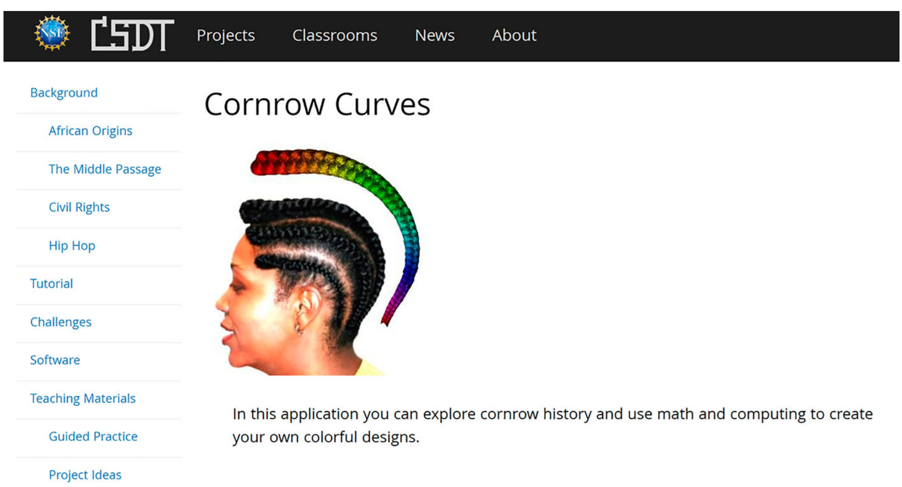


Figure 2. Landing page for Cornrow Curves.

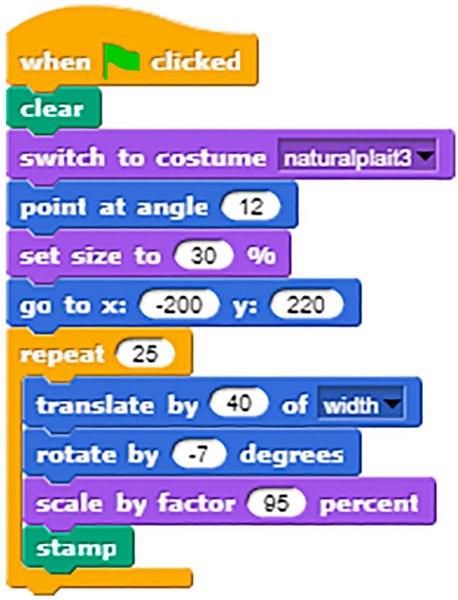


Figure 3. Default script for Cornrow Curves.

computational thinking and STEM disciplines (Hutchins, Biswas, Conlin, et al., 2018). This work draws on the affordances of programing languages to support computational modeling procedures but in specific disciplinary domains. For example, Hutchins, Biswas, Maroti, Ledezci, and Broll (2018) used Snap! to create blocks that help to model motion and force in the context of physics education. The braid block in Cornrow Curves is designed to help bridge the domains of braiding and CS, modeling the geometric features of cornrows and African mathematics. Later work with a high school CS teacher required us to remove the ability of the block to reset size each time it is called (because he wanted to motivate the use of variables). While the change made it better suited for CS, it created a steeper learning curve, not ideal for a cosmetology class, where closer fidelity to the braiding experience is helpful. The next version will likely offer different "editions" for different classrooms.

Context, participants, and methods

Context

Our research on Cornrow Curves in the context of cosmetology education took place at a public vocational school in Upstate New York, over the course of four-weeks during March of 2017. The vocational school is a branch of the local public high school that serves approximately 2,500 students, over 50% of which are Black and/or African-American. The program was designed as an after-school activity in collaboration with the head of the school's cosmetology department, who is an African-American entrepreneur and salon owner with thirteen years of teaching experience. The cosmetology curriculum is largely project-based – using a mix of lessons that are hands-on and theory-based – to prepare students to take the New York State exam to become licensed cosmetologists. The program also places students in salons to get job training and the hours required for gaining

Figure 4. Domain-specific "braid" block.

professional licensure. The program has a strong presence in the local community, organizing spa and hairstyling events at the school that are open to the public.

Building on this interdisciplinary connection between cosmetology and computing, we called the after-school program "Cos-Computing." With the help of the cosmetology teacher, we agreed that in addition to learning math and programing skills through virtually simulating cornrow braids in Cornrow Curves, students would also learn physical cornrow braiding techniques. This presented an interesting opportunity to see if the transfer of knowledge could move in both directions (i.e. between physical and virtual braiding). We hypothesized that connecting this culturally situated knowledge to CS would open up pathways for generative circulations of value – though it was unclear at the time what these would look like – between school and community.

Program design and pacing

Initial discussions about the program took place during October 2016 with the cosmetology teacher. Our goal was for students to learn the transformational geometry that is embedded within cornrow braiding (i.e. angle, rotation, dilation, and translation) and apply that knowledge to virtual and physical braids using the computational thinking concept of *algorithm*. The collaborative planning between the research team and the cosmetology teacher for the after-school program started in January 2017 and extended up until the workshop began. The program took place during March 2017, three days a week (Tuesday-Thursday) after school for one hour. The cosmetology teacher helped with recruitment by advertising the program to her cosmetology students, as well as parents and youth she knew as customers and colleagues from her salon. The workshop was scheduled for three weeks, but due to snow days and inconsistent attendance of some students, we ended up running the program over four weeks. The workshop was designed around three deliverables that would be put on display during a public event held by the cosmetology department at the end of the program: 1) cornrow braided mannequin heads, 2) 2D Cornrow Curves designs, and 3) 3D printed mannequin heads with Cornrow Curves designs.

To explore the cultural background of cornrow braiding, students studied historical and cultural research on the Cornrow Curves website. Next, students worked through a Cornrow Curves tutorial that introduced the blocks needed to create the heritage algorithm of a single braid, plus the "set costume color" block. Once students were familiar with the functionality of Cornrow Curves, they were given a braiding lesson using mannequin heads. Students were then challenged to simulate their own or a peer's braided design in Cornrow Curves. We also gave them the option of choosing a design from an online source or library of goal images. Once students' designs were complete, our research team translated the 2D images into 3D models, finally rendering them as 3D prints in time for the public event. The actual work of turning the 2D designs into 3D designs was done by researchers due to the difficulty of translating between the two spaces. The 3D modeling process that placed the design on top of a mannequin was shared as part of a lesson with the students. The public event was scheduled for the last day of the program and served two purposes: 1) to raise community consciousness about the mathematical and computational significance of African-American heritage and 2) for students to show off their computational work in a public context, which can be an important part of motivating computational modeling and design (Papert, 1980).

Participants

Seven participants attended long enough to learn the software and explore the connections between braiding, computing, and mathematics in a meaningful way. One student did show up for the first two days but never returned thereafter. Of the seven, five identified as Black/African American, two as multiracial, and all as female (see Table 1). They ranged in age from 14 to 17 (see Table 2) and grades from 9th to 11th (see Table 3). Five of the students were in one of the teachers' three cosmetology courses, and two students were not, but had an interest in CS and/or braiding. While

Table 1. Gender and ethnicity of participants.

Gender	*n*	Ethnicity	*n*
Female	7	Black/African American	5
Male	0	Multiracial	2

Table 2. Age of participants.

Age	*n*
14	1
15	1
16	4
17	1

Table 3. Grade level of participants.

Grade Level	*n*
9	1
10	3
11	3

the program was completely voluntary, students were encouraged to show up for as many days as they could to complete their designs for the public event. However, some of the students had obligations beyond school (i.e. work and family) that prevented their consistent attendance.

Five of the participants attended regularly, and two attended sporadically. These two students still learned about the history of cornrow braiding, became proficient with Cornrow Curves software, explored the heritage algorithms both virtually and physically, and one presented at the community event. Because we prioritized these aspects of the program over the creation of their own designs, they started but were unable to complete their final projects in time for them to be 3D printed, though we made sure that the artifacts they did create (physical braids on mannequins and 2D designs) were shown off at the public event. What we learned from this experience was that students, even if they are intrinsically motivated to learn, often have competing interests that, at the moment, will trump educational opportunities. We used this information to inform future work by paying students to attend programs and providing opportunities for multi-generational attendance.

While seven participants cannot be statistically significant, STEM education researchers who focus on issues of equity and racial identity have argued for the importance of "small-n" studies. Slaton and Pawley (2015) argue that academics will need to overcome stigmas associated with small-n studies and learn from individual narratives or small group dynamics to fully understand issues of underrepresentation in STEM education and fields. They argue that ignoring or dismissing small-n studies that focus on already underrepresented identities in STEM risks further marginalization, as they are left out of the conversation or assimilated into more general group identities.

Data collection and analysis

Building on the need for more small-n studies while also seeking to provide valid and strong research findings, we used a mixed-methods approach. Consistent with more qualitative norms in computer science education research (Searle & Kafai, 2015; Tenenberg, 2019), data were triangulated – "seeking the convergence and corroboration of results from different methods and designs studying the same phenomenon" (Biesta, 2017, p. 159) – to construct rich narratives of individuals and a dynamic portrait of the group. Data was collected in the form of pre- and post-surveys from the students, students' written reflections, semi-structured pre- and post-interviews and daily debriefs with the cosmetology teacher, field notes and audio-recordings from each day of the program, video recordings of the public event, and student-created designs. A math teacher who came to observe periodically was also interviewed at the end of the program. Finally, we included in our analyses four interviews from two of the students – who completed their designs to 3D print during the program and later participated in two other programs that aimed to connect computing and cosmetology – that were conducted at later dates. These interviews sought to explore the two students' perceptions about the relationship between mathematics, computing, and braiding.

The pre-survey was administered before students began any of the activities and the post-survey was administered on the day before the public event. The survey consisted of five sections. The first section asked for information on gender, age, school, grade, and race or ethnicity. Section two was made up of ten close-ended statements, designed to measure students' perceptions of school, cosmetology, community, computer science, and the relationships between them on a five-point Likert-type scale: 1 = Strongly Disagree, 2 = Disagree, 3 = Neutral, 4 = Agree, and 5 = Strongly Agree. The third section was made up of five vocabulary statements with a bank of transformational geometry and CS words: angle, iteration, rotation, dilation, and translation. Students were to choose a word from the list and match it with the correct statement.

The fourth section contained two items to measure students' computational thinking by asking them to abstract scaling patterns from cornrow braids and apply them to other physical objects, asking how they would simulate those patterns. The first of these questions showed a picture of a xylophone. We sought to measure how students would transfer or abstract their knowledge of the scaling patterns found in the cornrow heritage algorithms to describe the scaling pattern found in an object unrelated to hair braiding. We chose an image of a xylophone because it still represented a cultural artifact with scaling patterns, but it was different enough from braiding to judge knowledge transfer. The second question sought to measure students' knowledge of loops and iteration by showing them one picture of a triangle on a Cartesian coordinate system with a picture next to it of the triangle repeated four times and rotated around the origin, each by 90 degrees. We asked, "Starting with this triangle, describe the process that creates the pattern of 4 triangles."

Given the geometric features of cornrows, the fifth section contained two items that aimed to measure students' knowledge of transformational geometry; asking them to describe visual changes in shapes on a coordinate plane. The first question in the section presented an "L" shape that was four inches and asked how tall the shape would be if we changed the shape by a scaling factor of 50%. The second question presented two "L" shapes with one rotated ten-degrees and labeled as such. We asked them to apply the same rotation again and write down the new angle in degrees. Due to the small-n of this study, descriptive statistics were used to analyze changes in students' knowledge as a result of participating in the program.

Interviews and daily debriefs were audio recorded and transcribed, while video, pictures, and students' reflections were analyzed in their raw form alongside field notes. Field notes, interviews, debriefs, and video were all analyzed with a "descriptive" coding technique where excerpts from the text or video were summarized or tagged with a word or phrase – mostly nouns – to denote a relevant topic (Saldaña, 2016, p. 292). These codes were then aggregated into organizing themes (i.e. computing, mathematics, and cornrow braiding; computational thinking in braiding and cosmetology; cornrows beyond Cos-Computing) to present as findings below. A video log was created for the public event, which was a minute-by-minute breakdown of students' engagement with their community and their articulations of the connections between computing, mathematics, and braiding to approximately thirty audience members. Video files from two different cameras were used to create the video log, one camera that faced the audience and the other faced the presenters. Sections of interest from these videos were transcribed for a more thorough examination. We used data from the public event as an indicator for students' takeaways from the program and their understandings of math, computing, and braiding connections. Students' 2D designs were analyzed by identifying what and how many blocks were used in the context of what was made available in the original heritage algorithm to judge their command of Cornrow Curves. We compared these findings to images and students' vocal descriptions of their physical braids to see if there was any possible relationship between braiding skills and the scripts they created in Cornrow Curves.

Findings

Computing, mathematics, and cornrow braiding

The role of braiding, including but not limited to cornrow braiding, is a unit in the cosmetology curriculum, one required by New York State for certification programs. But perhaps more important for the Cos-Computing program was students' interest in the cultural and stylistic practices of braiding and their positive associations with the activity. Indeed, when looking at statement two in Table 4, the majority of students consistently indicated that they were passionate about cosmetology. This was supported by students' self-direction and enthusiasm on the third day of the program when they were assigned to apply the math and computing ideas from the Cornrow Curves tutorial to physical braiding on mannequin heads. While students struggled to connect the dense text of the tutorial to the function of specific blocks without the instructors' interventions, the tutorial images still provided them with a way to visually make connections between math, computing, and braiding. As one student explained in a reflection after going through the tutorial, "[(1)] Computing and braiding both deal with where you start your braid (2) What angle the braids are going (3) the color your braids are (4) how small or big your braids are/length and where it stops."

The teacher started the physical braiding lesson by asking who knew how to make cornrows. Four students raised their hands; three did not. The four who knew how to braid were self-directed, while the other three moved closer to the teacher for instruction. She told them they would need to start with a box braid first and then apply that technique to cornrows. A cornrow, she explained, "is just a box braid on the scalp." She instructed them to part three sections of hair. They did while she explained the two different ways to make a box braid,

> We are going to practice the box braid two different ways, it is very important that you get both ways, that you understand both ways, techniques of braiding, so that it is easier to transfer into the cornrow … We know that a braid is three strands of hair. So the first technique I want you to try is alternating from side-to-side, from the left to the right, going over the middle strand.

To emphasize this technique, she repeated "over the middle strand" multiple times before having the students try out the next technique, "under the middle strand." After this short introduction she parted some hair on her mannequin to explain the technique for cornrow braiding:

Table 4. Responses to pre and post- attitudes survey.

	Statement	Pre: # of student who marked agree + strongly agree	Post: # of students who marked agree + strongly agree	Change
1	My future depends on working hard in school.	7	7	0
2	I am passionate about cosmetology.	5	5	0
3	I want to use my cosmetology skills to become a professional cosmetologist.	3	4	+1
4	My future depends on working hard in the cosmetology program.	3	3	0
5	I can use cosmetology to help my community.	6	5	−1
6	I think understanding computer science will make me a better cosmetologist.	3	6	+3
7	I would likely be bored learning computer science.	3	0	−3
8	If I were to go to college I would be interested in study computer science.	1	1	0
9	I can use computer science to help my community.	3	5	+2
10	Computer science is helpful for professional cosmetologists.	3	5	+2

> Now you have your cornrow parted out, right? Okay? So what I normally do, is I'll take maybe a half an inch, okay? Separate it into your three sections, okay? Start your box braid the way you normally would, see that? Start your box braid. Now what I'm going to do, after two rounds, if you want to count it, after two rounds, you want to take hair, the hair beneath it, maybe a quarter of an inch, add it to the middle, the middle strand of hair. Braid two rounds, now move down, the section below it, add to the middle strand. Two rounds, add to your middle strand. Two rounds, add to your middle strand.

Originally, including the braiding part of the program was counterintuitive to our research team – why go back to physical braids when we have been working towards increasing forms of abstraction? But the cosmetology instructor was proved right. We found that this part of the instruction, with its directionality and repetition, was helpful to connect the algorithmic and iterative aspects of physical braiding to the process of creating virtual braids.

Later in the lesson students readily applied the math and computing concepts from the tutorial to explain their own physical braid designs (see Figure 5),

> I took one braid and set it at one angle and my, I'm going to forget this part, iteration? I just iterated it all the way straight back, and then just changed the angle and iterated it straight back three times.

> I used translation because the different plaits they translated along the head.

> I set it at like a 90-degree angle and then like the rotation, it was like, I kept it rotating.

> A little bit of dilation, but not really. And then you would use iteration because you are repeating the same pattern, at least that's the goal.

Students were not only able to remember how each of the terms was applied to braiding in the software but also translated that knowledge to physical designs. The students comfort with the translation process may have been due to their existing familiarity with many of the terms. Indeed, there was little change in students' understanding of the math and computing vocabulary (see Table 5). From school or elsewhere, the majority of students appeared familiar with the transformational geometry vocabulary, and with only one computing term (i.e. iteration), they were either familiar with it or were able to guess through a process of elimination. Still, it would be wrong to assume that this translation process was intuitive just because of their familiarity with braiding or geometry.

When asked if she "buys" the idea that braiding has mathematical significance during an interview the following year, one student responded,

> Ask me a year ago, no, not at all. But after one, being in [the Cos-Computing] program and then being a mentor in this program, definitely. I am, I am your person on board with the mathematics behind braiding, and I might have forgotten some of the terms and things, but definitely, there is a correlation.

The student's reflection indicates that while historically intentional mathematical and computational ideas are embedded within vernacular and indigenous designs, simply engaging them may not be enough to make the translation today. What is more, knowing geometry, for example, and being a

Figure 5. A student uses hand gestures to explain the angles of her cornrow braid design.

Table 5. Mean scores of pre and post vocabulary questions.

Survey	*n*	Mean of points (Maximum = 5)	Change in mean from pre to post
Pre	7	4.43	+0.14
Post	7	4.57	

braider may not be enough to make the connection that braids dilate, translate, rotate, and reflect on the scalp during a transformational geometry lesson. This is not to say that mathematical and computational thinking are absent, but that a cultural/technical dichotomy is often reinforced in and out of school. The knowledge domains of mathematics and braiding, for example, are not often afforded opportunities to explicitly interact, having the translation process between the two become common knowledge. With this understanding, Cornrow Curves acts as a bridge between existing knowledge domains, bringing them together in new ways, shaping the meaning of each without losing site of the original cultural context.

While the questions intended to measure students' abilities to solve transformational geometry problems only saw slight improvement (see Table 6), this should not be an indicator of the program's value. Indeed, the math teacher reinforced the importance of the program as a bridge between different knowledge domains. He told us that the program had positive outcomes in terms of student engagement with mathematical ideas: "… by the end while students were coding I would ask them questions, and so, for example, I would ask, 'how do you dilate in real life?' 'You dilate your hair,' she said, 'Oh you add more hair'." Still, in working with both teachers during and after the program, they continued to struggle to forefront the deeper connections. They tended to favor the connection between cosmetology and business math, with the mathematical design elements of braiding remaining secondary. This is not surprising given that business math is not only a common way for math teachers to highlight "real life" applications, but is also featured prominently in the cosmetology curriculum (i.e. they have explicit, institutionalized opportunities to interact).

Computational thinking in braiding and cosmetology

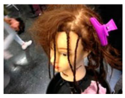
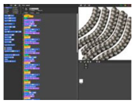
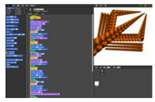
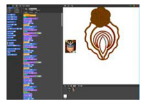
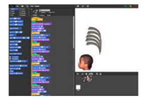
When modeling their own cornrow designs, students struggled to move beyond the basic blocks that were used to represent the heritage algorithm in Cornrow Curves. To a certain extent this makes sense due to the program's emphasis on those blocks and the fact that all of the students were novice programmers. While they were exposed to the library of blocks that were available to them in the software, none of the students chose to experiment with blocks that were not introduced in the tutorial. What is more, only two students used the "braid" block. This resulted in students having extremely long scripts that could have been made more efficient if they were exposed to nested loops or more strongly encouraged to use the domain specific braiding block.

Table 7 is a comparison between the physical braids, 2D designs, and 3D prints for the five students who completed their designs in time for the public event. It is hard to determine how much of a correlation there is between the physical designs and 2D designs in terms of math and computational knowledge. While Student 4, an experienced braider, had the most dynamic 2D design – using the software to turn cornrows into buns – her physical braiding was relatively straight forward, taking the form of a basic pattern that fits the curvature of the head. What does stand out is the fact that Students 2 and 3, who had two of the more creative physical styles – moving

Table 6. Means scores for pre and post-transformational geometry questions.

Survey	*n*	Mean of points (Maximum = 2)	Change in mean from pre to post
Pre	7	0.86	+0.14
Post	7	1.00	

Table 7. A comparison of the physical braids, 2D designs, and 3D prints.

	Student 1	Student 2	Student 3	Student 4	Student 5
Physical Braids					
Cornrow Curves Designs					
3D Prints					

away from following the curve of the head to braid with different angles and rotations – were also the two students who used the "braid" block in their designs. These two novice braiders and programmers not only pushed boundaries physically but also virtually. While more research needs to be conducted to make concrete claims about the mutually beneficial relationship between physical braiding and programing virtual braids in the development of computational thinking, post-survey results indicate that students did improve on those questions intended to measure computational thinking (see Table 8). This coupled with their understanding of physical and virtual algorithms provides grounds for following up on the possibility in future research.

Students' attitudes toward CS appeared to change after the Cos-Computing program (see Table 4). First, CS appeared more relevant to cosmetology. Post-surveys suggest increases in students' attitudes that "computer science will make me a better cosmetologist" and "computer science is helpful for professional cosmetologists." Second, zero students indicated that they would be "bored learning computer science" (down from three). And third, the majority indicated that CS could be a means for "helping" their community. One of the more striking changes from pre to post – one that challenges our initial hypothesis about the generative mutuality of cosmetology and CS – was the decrease in students' attitudes that cosmetology can be used to "help" their community, though the majority continued to think that it could despite this drop. At the same time, the fact that there was an increase in students wanting to use their skills to become professional cosmetologists suggests that the two domains may still be mutually beneficial.

While students had a generally positive view of CS, only a small minority of students indicated that it would be an area of interest in college, with no changes from pre to post. In fact, when looking at other future-oriented academic statements (i.e. "my future depends on working hard in school" and "my future depends on working hard in the cosmetology program"), there were also no changes. Still, their overall understanding of computing appears to have increased, specifically their understanding of the embedded mathematics and algorithms in braiding, as demonstrated qualitatively by their presentations at the public event.

Approximately thirty people attended the public event, which was structured to begin with presentations followed by introductions to three different stations for guests to explore the computational and mathematical significance of cornrow braiding: a physical braiding station, a 2D virtual braiding station, and a 3D printing station. Students presentations and the post-presentation Q&A focused primarily on the African roots of braiding and hairstyling – including but not limited to

Table 8. Mean scores of pre and post computational thinking questions.

Survey	*n*	Mean of points (Maximum = 9)	Change in mean from pre to post
Pre	7	5.43	+1.71
Post	7	7.14	

cornrows – and their roles in African and African-American identities. The presentations included topics that ranged from hairstyles in traditional West African weddings to the assimilationist renaming of cornrow braids as boxer braids by White athletes.

While identity was the primary focus, interspersed throughout the presentations students explained the math and computing of braiding. Perhaps because the Cornrow Curves and 3D printing stations already appeared so directly connected to math and computing, explanations of the connections were most explicit when introducing the physical braiding station. As the student tasked with making this introduction explained,

> Like, for example, in this braiding like you can see that dilation was used because in the beginning, it comes out shorter, like smaller and then it gets bigger. And this is some styles that African-American women do wear.

Later, when prompted by one of the researchers to explain the connection between computing and braiding another student emphasized the versatility of the term algorithm.

Researcher	You guys learned about an algorithm, what is an algorithm?
Student	A set of instructions followed in the computer or anywhere.

The inclusion of "anywhere" speaks to this student's understanding of not only an algorithm as it appears in the Cornrow Curves software, but also Bennett's (2016) take on algorithms as part of existing heritage practices. Playing on the "multiplicity" of algorithms – that is the multiple, but not pluralistic, meanings and shapes of algorithms (Bucher, 2018) – when making connections between computing and culture appears as one of many strategies for broadening what can count as CS education and who has CS-types of knowledge.

Throughout the planning process, during the program, and afterwards our team worked with the cosmetology teacher to explore these conceptual connections. For example, when planning the activity where students reverse engineered and simulated a physical or photographed braided design in Cornrow Curves, the teacher asked: "I noticed that you said the kids they will … actually will design their braid. They will have to pretty much … was it the algorithm? … Algorithm, okay so they will need to pretty much take apart their partner's braid." She also suggested that the skills learned when using Cornrow Curves had direct applicability to working in the salon. Multiple times during debriefs and interviews she framed this potential transfer of knowledge around the notion of planning: " … I actually see this benefiting the salon, especially for planning and prepping, and you know, for some of the braiding services that we provide." One way to interpret the connections between planning a hairstyle and designing in Cornrow Curves is through the computational thinking concepts of *algorithm* and *decomposition*.

The cosmetology teacher used the term algorithm in reference to taking "apart their partner's braids." This was relatively consistent with how we were talking about algorithms in the program, as we engineered and reverse-engineered physical and virtual braids to explore the transformational geometry that is embedded in the designs. In addition, we could have talked about this process as decomposition, fitting in with the trial-and-error activity of exploring what each block does when first opening the Cornrow Curves software. Building on the computational thinking connection to planning and preparing that the teacher alluded to, we might have also interpreted this process in terms of an algorithm. Drawing on the multiplicity of the term to encompass not only design processes but organizing and managing when, how, and where those activities take place fits the computational participation criteria of publicly displaying artifacts and processes in a community network. Online, many computational participation communities have the ability to curate personal profiles and project displays (Kafai & Burke, 2014). The practice of showing off and sharing out youth-created artifacts is common in physical spaces as well, including the Computer Clubhouse (Kafai, Peppler, & Chapman, 2009) and the YouMedia library space (Larson et al., 2013). Moving forward it is worth exploring how the computational thinking that is supported by production and curation activities (Resnick, 2012) can be leveraged as young people network with local entrepreneurs (e.g. natural cosmetic producers) and business owners (e.g. African braiding shop owners) to display artifacts that are computationally and culturally relevant.

Cornrows beyond Cos-Computing

Once the program ended, the 3D printed mannequin heads traveled beyond the school into a professional salon that is owned by the cosmetology teacher. The research team was interested in how these specific 3D prints might not only support students' educational interests and achievements, but also how they might support braiders, cosmetologists, salon owners, and others in the beauty industry. Originally we had thought that the prints could be displayed in the shop's window, hypothesizing that the novelty of having a 3D printed object in the salon might increase foot traffic. However, the teacher's salon was in a basement, without windows. Still, she was excited about showing them off in her salon and agreed to put them on display (see Figure 6). Indeed, the cosmetology teacher was generally willing to try out innovative ways to connect her profession with the work we were doing together. As she explained in one of our debriefing sessions, "I'm open to how we can you know, push into your world. You guys can push into our world." While we did not make a hypothesis about putting the 3D printed Cornrow Curves designs in this particular setting, we did find that they played an unexpected role as a topic of conversation.

Cosmetologists draw on a range of STEM literacies in their daily work, from balancing the pH of different cosmetic products to understanding what products do to hair anatomy and physiology. In our collaborations with cosmetologists, we found that this knowledge can be important for retaining clients by making STEM topics part of broader conversations about hair, health, fashion, and beauty generally. There were multiple indications from the cosmetology teacher that talking about Cos-Computing supplemented conversations about art and STEM already taking place in her salon. During a debriefing session, she explained how cosmetology acts as a bridge for such domains more generally: "I have a chemist … that comes to my salon, she's so into hair and fashion and everything, it's not either or, you know, so I think this is what's great and … these are the things that our customers talk about … " Knowing that cosmetology is already a site of STEM expertise, we found that including the 3D prints in the teacher's salon added to the repertoire of STEM knowledge that is available for conversation.

The following year we interviewed one of the cosmetologists who works in the teacher's salon. She had a daughter in the Cos-Computing program and, at the time of the interview, had collaborated in another workshop we ran a month prior. The interview was conducted in the salon,

Figure 6. 3D printed mannequin heads on display at the teacher's salon.

where the 3D prints were still on display. Having been introduced to Cornrow Curves, the cosmetologist explained how she talks about the 3D prints when clients ask about them:

> I'm just like how do you take like this little algorithm thing and put things here, and it creates this braid over here and then when you expressed to me about … how Europeans didn't start mathematics … I express all of that to them, and they are like "wow!" And see, and then just go on and on, so it sparks like this bigger conversation.

While there was a general goal when working with the seven students to challenge Eurocentric histories of mathematics and computing, the idea that the 3D prints might act as a means to challenge them in the salon was not on our radar. Such unexpected findings support calls for more qualitative measures of success in programs and research on broadening participation in CS (Scott et al., 2015). More broadly, assessing the success of broadening participation cannot be limited to the numbers of people entering the CS pipeline, but must also include the diffusion of CS knowledge across local communities. This may require a transdisciplinary approach to assessment that benefits from not only qualitative and quantitative methods but also humanistic inquiry into cultural arts and designs.

Discussion

Initial results from Cos-Computing suggest increases in students' interest in CS and computational thinking skills, which is consistent with other empirical findings on the CSDTs (Babbitt et al., 2015; Davis, Lachney, Zatz, Babbitt, & Eglash, 2019; Eglash, Krishnamoorthy, Sanchez, & Woodbridge, 2011). So while this aspect of the research is not unique, previous studies took place in traditional formal or informal educational contexts, not explicitly connected to the communities of practice that the CSDTs highlight. In this sense, what stands out about this study is the ability to help answer specific questions about leveraging local sources of wealth generation – that are important to students' communities – in CS education.

How might CS educators, researchers, and technologists promote culturally responsive forms of computational participation?

We have sought to answer this first question by analyzing data from the Cos-Computing high school program. This generative computing effort sought to maintain the socio-cultural framework for computational thinking that is indicative of computational participation but added culturally responsive elements by making explicit connections between computational thinking, local sources of wealth generation, and culturally situated designs. How other educators, researchers, and technologists might also support culturally responsive forms of computational participation largely depends on their willingness to connect with local community experts – found in and beyond the school walls – and identify culturally important community assets that are relevant to computing education.

Connections between educators and local communities are complicated by the fact that many teachers do not live in the communities they serve and the vast majority of teachers in the U.S. are White (Picower, 2009). This has resulted in increased recognition for the need to diversify the teacher workforce, while also developing strategies for White teachers to support students of color in culturally authentic and responsive ways. For example, Emdin's (2016) work on culturally responsive teacher education encourages educators to engage in community events and spend time in neighborhoods where schools are located. This is certainly a strategy for teachers, but in the context of generative computing should also be extended to researchers and technologists.

Finding ways to connect educational technologies and artifacts to local sources of wealth generation provides pathways for the diffusion and traveling of computational artifacts and ideas across community settings. But, like teachers, many technologists and researchers working to broaden participation are not from the communities where they will be working to develop and implement technologies. This is exactly what programs like Cos-Computing seek to change both immediately and in the future. Immediately, a dearth of local technologists or computer scientists can motivate working with community experts to learn what CS-culture connections can be leveraged to make CS an authentically relevant domain for not only young people but also adults. In the future, having greater representation in the field increases the chances that technologists and researchers will have connections to communities where young people live and work, potentially acting as role models and contributing to local forms of wealth generation, such as cosmetology and braiding.

How might these culturally responsive forms of computational participation not only support the educational interests and achievement for young women but also link up to sources of wealth generation that are important to their local communities?

While post-survey results indicate an increase in students wanting to pursue professional cosmetology, to answer this question, it is important to examine direct instances where the possibility for the value generated by the program entered the world of cosmetology. One major indicator of the potential benefit that the Cos-Computing program had for the teacher's local salon was its impetus for adding African mathematics and computing ideas to the repertoire of existing STEM knowledge that exist and are discussed in the salon. In addition to supplementing this existing knowledge-base, the cosmetology teacher also indicated that she was inspired to try and physically braid some of the students' virtual designs, "I mean – I just – you can actually charge a customer for that. So I just saw the dollar signs. The potential dollar signs for having something like that in your beauty salon." In this way, the goal of computational participation to situate programing and computational thinking within socio-cultural contexts is supported not only by making the algorithmic aspects of cornrow braiding explicit in the software but also beyond, using that explicit connection as a foundation for more creative, possibly entrepreneurial, applications of CS in cosmetology.

Additional research is needed to continue to explore the potential of generative computing to support both academics and local wealth generation, but these initial anecdotes suggest there is a strong potential for programs like Cos-Computing to be mutually beneficial. Our argument is that if the CS community is truly committed to broadening the participation of racial and ethnic minorities, it is not enough to reduce metrics of success to classroom demographics. CS educators and researchers need to become involved in local affairs and know how CS and other STEM fields can become useful innovations to the communities where students live and work. This will mean not only re-thinking who has knowledge relevant to CS and computational thinking but also the standard forms of assessment to include a mix of qualitative, quantitative, and humanistic indicators of success.

Conclusion and future work

This paper has introduced the concept of *generative computing* in the context of a cosmetology/CS after-school program, to help researchers, technologists, and educators think more critically about the types of socio-cultural learning environments that educators and researchers should nourish in their support of computational thinking. Generative computing aims to diversify both the entry points and outputs of CS. Our findings suggest that through a set of computationally rich cosmetology projects, pathways opened for students to use computing as a conduit for enhanced value flow between CS and Black cosmetology practices in ways that improved their academic interests and knowledge of CS.

Based on these initial findings, we have two future directions for research on generative computing in the context of cosmetology. The first includes studying the computational thinking of physical braiding practices. While students in the Cos-Computing program were able to translate knowledge between the virtual and physical braids, many questions remain unanswered about whether braiding can act as a type of CS unplugged activity. Running a comparative study between a group of students who learn with both physical braiding and Cornrow Curves and another group who learn the same material but just with physical braiding, may provide some insight into the potential of culturally situated unplugged activities while bringing up important questions about what might be lost without the virtual simulations.

The second future research direction is looking at the impact that programs like Cos-Computing might have on professional settings, like the salon or barbershop. The 3D printed mannequin heads that were based on students' designs attracted a reasonable amount of attention from community members and teachers at the Cos-Computing community event. This is partially due to the novelty of 3D printing but also from the fact that it was a new way to represent popular hairstyles. Our team has started to explore if we can build on these novelty and popularity factors in ways that could benefit the local cosmetology community. One promising avenue is using the mannequin heads as advertisements in salon windows. Many salons already do this with traditional mannequins but adding the computational value of 3D printing might make a salon stand out as technologically unique while also creating relationships between salons and makerspaces or schools.

Acknowledgments

The views and opinions expressed in this work do not necessarily represent those of NSF.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by the National Science Foundation under Grant # DRL-1640014.

Notes on contributors

Dr. Michael Lachney is an assistant professor of educational technology at Michigan State University. His research explores the roles that technologies can play in strengthening school-community relationships. He is currently working on educational technology design strategies and implementation tactics to help teachers enroll community-based expertise in culturally responsive STEM education.

Dr. William Babbitt received his BS in Finance, Mathematics, and Computer Science, and holds a Ph.D. in Multidisciplinary Science. He currently works on educational technology design strategies and the impacts of colonization on society and education.

Professor Audrey Bennett received her BA in Studio Art from Dartmouth College and her MFA in Graphic Design from Yale University. The author of Engendering Interaction with Images, she is a Professor of Art and Design in the Penny W Stamps School of Art and Design at the University of Michigan.

Dr. Ron Eglash received his BS in Cybernetics, MS in Systems Engineering and Ph.D. in History of Consciousness. The author of African Fractals, he is a Professor of Information in the School of Information with a secondary appointment in Art and Design in the Penny W Stamps School of Art and Design at the University of Michigan.

ORCID

Michael Lachney http://orcid.org/0000-0003-3310-8707
William Babbitt http://orcid.org/0000-0002-2684-4901
Audrey Bennett http://orcid.org/0000-0002-6763-2622
Ron Eglash http://orcid.org/0000-0003-1354-1300

References

Anyon, J. (2014). *Radical possibilities: Public policy, urban education, and a new social movement*. New York: Routledge.

Babbitt, W., Lachney, M., Bulley, E., & Eglash, R. (2015). Adinkra mathematics: A study of ethnocomputing in Ghana. *Multidisciplinary Journal of Educational Research*, *5*(2), 110.

Bell, T., & Newton, H. (2013). Unplugging computer science. In D. M. Kadijevich, C. Angeli, & C. Schulte (Eds.), *Improving computer science education* (pp. 66–81). New York: Routledge.

Bennett, A. G. (2016). Ethnocomputational creativity in STEAM education: A cultural framework for generative justice. *Teknokultura*, *13*(2), 587–612. doi:10.5209/rev_TEKN.2016.v13.n2.52843

Biesta, G. (2017). Mixed methods in educational research. In R. Coe, M. Waring, L. V. Hedges, & J. Arthur (Eds.), *Research methods and methodologies in education* (pp. 159–165). Thousand Oaks, CA: SAGE.

Bucher, T. (2018). *If … then: Algorithmic power and politics*. New York: Oxford University Press.

Byrd, A., & Tharps, L. (2014). *Hair story: Untangling the roots of Black hair in America*. New York: St. Martin's Griffin.

Cooke, L. (2016). Metatuning: A pedagogical framework for a generative STEM education in game design-based learning. In *Integrated STEM education conference (ISEC), 2016 IEEE* (pp. 207–214). doi:10.1109/ISECon.2016.7457534

Cuny, J., Snyder, L., & Wing, J. M. (2010). *Demystifying computational thinking for noncomputer scientists*. Work in progress.

Davis, J., Lachney, M., Zatz, Z., Babbitt, W., & Eglash, R. (2019). A cultural computing curriculum. In *Proceedings of the 50th ACM Technical Symposium on Computer Science*.

Denning, P. J. (2009, June). The profession of IT beyond computational thinking. *Communications of the ACM*, *52*(6), 28–30.

Dotson, T. C., & Wilcox, J. E. (2016). Generating community, generating justice? The production and circulation of value in community energy initiatives. *Revista Teknokultura*, *13*(2), 511–540. doi:10.5209/rev_TEKN.2016.v13.n2.52840

Eglash, R. (1999). *African fractals: Modern computing and indigenous design*. New Brunswick, NJ: Rutgers University Press.

Eglash, R., Babbitt, W., Bennett, A., Bennett, K., Callahan, B., Davis, J., … Tully, K. (2017). Culturally situated design tools: Generative justice as a foundation for STEM diversity. In P. Tripathi, Y. Rankin, & J. Thomas (Eds.), *Advances in educational marketing, administration, and leadership. Moving students of color from consumers to producers of technology* (pp. 132–151). Hershey, PA: IGI Global. doi:10.4018/978-1-5225-2005-4.ch007

Eglash, R., Bennett, A., Drazan, J., Lachney, M., & Babbitt, W. (2017). A mathematical tool kit for generative justice. *ETD - Educação Temática Digital*, *19*(3), 761–785. doi:10.20396/etd.v19i3.8648374

Eglash, R., Bennett, A., O'Donnell, C., Jennings, S., & Cintorino, M. (2006). Culturally situated design tools: Ethnocomputing from field site to classroom. *American Anthropologist*, *108*(2), 347–362. doi:10.1525/aa.2006.108.2.347

Eglash, R., Gilbert, J. E., & Foster, E. (2013). Toward culturally responsive computing education. *Communications of the ACM*, *56*(7), 33–36.

Eglash, R., Krishnamoorthy, M., Sanchez, J., & Woodbridge, A. (2011). Fractal simulations of African design in pre-college computing education. *ACM Transactions on Computing Education (TOCE)*, *11*(3), 17. doi:10.1145/2037276.2037281

Emdin, C. (2016). *For White folks who teach in the Hood … and the rest of y'all too: Reality pedagogy and urban education*. Boston, MA: Beacon Press.

Farinde, A. A., & Lewis, C. W. (2012). The underrepresentation of African American female students in STEM fields: Implications for classroom teachers. Online Submission.

Gay, G. (2010). *Culturally responsive teaching: Theory, research, and practice*. New York: Teachers College Press.

Gill, T. M. (2010). *Beauty shop politics: African American women's activism in the beauty industry*. Urbana, IL: University of Illinois Press.

Hutchins, N., Biswas, G., Conlin, L., Emara, M., Grover, S., Basu, S., & McElhaney, K. (2018). Studying synergistic learning of physics and computational thinking in a learning by modeling environment. In *Proceedings of the 26th international conference on computing in education*. Philippines: Asia-Pacific Society for Computers in Education.

Hutchins, N., Biswas, G., Maroti, M., Ledezci, A., & Broll, B. (2018). A design-based approach to a classroom-centered OELE. In *International conference on artificial intelligence in education* (pp. 155–159). Spinger, Cham.

Jenkins, H., Purushotma, R., Weigel, M., Clinton, K., & Robison, A. J. (2009). *Confronting the challenges of participatory culture: Media education for the 21st century. The John D. and Catherine T. MacArthur foundation reports on digital media and learning*. Cambridge, MA: MIT Press.

Kafai, Y. B. (2016). From computational thinking to computational participation in K-12 education. *Communications of the ACM*, *59*(8), 26–27. doi:10.1145/2955114

Kafai, Y. B., & Burke, Q. (2014). *Connected code – Why children need to learn programming*. Cambridge, MA: MIT Press.

Kafai, Y. B., Peppler, K. A., & Chapman, R. N. (Eds.). (2009). *The computer clubhouse: Constructionism and creativity in youth communities*. New York: Teachars College Press.

Kafai, Y., Searle, K., Martinez, C., & Brayboy, B. (2014, March). Ethnocomputing with electronic textiles: Culturally responsive open design to broaden participation in computing in American Indian youth and communities. In *Proceedings of the 45th ACM technical symposium on computer science education* (pp. 241–246). ACM. doi:10.1145/2538862.2538903

Kellogg, S. (2016). Community based bioremediation: Grassroots responses to urban soil contamination. *Revista Teknokultura*, *13*(2), 491–510. doi:10.5209/rev_TEKN.2016.v13.n2.52144

Krauss, J., Prottsman, K., & Yongpradit, P. (2017). *Computational thinking and coding for every student: The teacher's getting-started guide*. Thousand Oaks, CA: Corwin.

Lachney, M. (2017a). Computational communities: African-American cultural capital in computer science education. *Computer Science Education*, *27*(3-4), 175–196. doi:10.1080/08993408.2018.1429062

Lachney, M. (2017b). Culturally responsive computing as brokerage: Toward asset building with education-based social movements. *Learning, Media and Technology*, *42*(4), 420–439. doi:10.1080/17439884.2016.1211679

Larson, K., Ito, M., Brown, E., Hawkins, M., Pinkard, N., & Sebring, P. (2013). *Safe space and shared interests: YOUmedia Chicago as a laboratory for connected learning*. Pennsauken: BookBaby.

Lipman, P. (2011). *The new political economy of urban education: Neoliberalism, race, and the right to the city*. New York: Routledge.

Lyles, D., Lachney, M., Foster, E., & Zatz, Z. (2016). Generative contexts: Generating value between community and educational settings. *Teknokultura*, *13*(2), 613–637.

Majors, Y. J. (2015). *Shoptalk: Lessons in teaching from an African American hair salon*. New York, NY: Teachers College Press.

Margolis, J., Estrella, R., Goode, J., Holme, J., & Nao, K. (2008). *Stuck in the shallow end: Race, education, and computing*. Cambridge: MIT Press.

Morris, M. (2016). *Pushout: The criminalization of Black girls in schools*. New York: The New Press.

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

Picower, B. (2009). The unexamined whiteness of teaching: How White teachers maintain and enact dominant racial ideologies. *Race Ethnicity and Education*, *12*(2), 197–215. doi:10.1080/13613320902995475

Pinkard, N. D. (1999). Lyric reader: An architecture for creating intrinsically motivating and culturally responsive reading environments. *Interactive Learning Environments*, *7*(1), 1–30. doi:10.1076/ilee.7.1.1.3599

Resnick, M. (2012). Mother's day, warrior cats, and digital fluency: Stories from the scratch online community. In *Proceedings of the constructionism conference: Theory, practice and impact* (pp. 52–58).

Rothstein, R. (2017). *The color of law: A forgotten history of how our government segregated America*. New York: Liveright Publishing.

Ryoo, J. J., Margolis, J., Lee, C. H., Sandoval, C. D. M., & Goode, J. (2013). Democratizing computer science knowledge: Transforming the face of computer science through public high school education. *Learning, Media and Technology*, *38*(2), 161–181. doi:10.1080/17439884.2013.756514

Saldaña, J. (2016). *The coding manual for qualitative researchers* (3rd ed.). Los Angeles: Sage Publications.

Scott, K. A., Sheridan, K. M., & Clark, K. (2015). Culturally responsive computing: A theory revisited. *Learning, Media and Technology*, *40*(4), 412–436. doi:10.1080/17439884.2014.924966

Scott, K. A., & White, M. A. (2013). COMPUGIRLS' standpoint culturally responsive computing and its effect on girls of color. *Urban Education*, *48*(5), 657–681.

Searle, K. A., & Kafai, Y. B. (2015). Boys' needlework: Understanding gendered and indigenous perspectives on computing and crafting with electronic textiles. In *ICER* (pp. 31–39). doi:10.1145/2787622.2787724

Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, *22*, 142–158. doi:10.1016/j.edurev.2017.09.003

Slaton, A. E., & Pawley, A. L. (2015, June). *The power and politics of STEM research design: Saving the "small N"*. Paper presented at 2015 ASEE Annual Conference & Exposition, Seattle,Washington. doi:10.18260/p.24901

Tenenberg, J. (2019). Qualitative methods for computing education. In S. A. Fincher & A. V. Robins (Eds.), *The Cambridge handbook of computing education research* (pp. 173–207). Cambridge: Cambridge University Press.

Vogel, S., Hoadley, C., Ascenzi-Moreno, L., & Menken, K. (2019). The role of translanguaging in computational literacies: Documenting middle school bilinguals' practices in computer science integrated units. In *Proceedings of the 50th ACM technical symposium on computer science educations (SIGCSE'19)* (pp. 1164–1170). Retrieved from https://doi.org/10.1145/3287324.3287368

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

Wingfield, A. H. (2008). *Doing business with beauty: Black women, hair salons, and the racial enclave economy*. Lanham, MD: Rowman & Littlefield Publishers.

Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S., & Korb, J. T. (2014). Computational thinking in elementary and secondary teacher education. *ACM Transactions on Computing Education (TOCE)*, *14*(1), 5.

Yadav, A., Stephenson, C., & Hong, A. (2017). Computational thinking for teacher education. *Communications of the ACM*, *60*(4), 55–62.

Yosso, T. J. (2005). Whose culture has capital? A critical race theory discussion of community cultural wealth. *Race Ethnicity and Education*, *8*(1), 69–91.

Zweben, S., & Bizot, B. (2018). 2017 CRA Taulbee survey. Retrieved from https://cra.org/wp-content/uploads/2018/05/2017-Taulbee-Survey-Report.pdf