Journal : Small 11423 | Dispatch : 11-12-2013 | Pages : 23
Article No. : 9327 | ☐ LE | ☐ TYPESET
MS Code : ETRD-D-13-00016 | ☑ CP | ☑ DISK

Education Tech Research Dev
DOI 10.1007/s11423-013-9327-0

DEVELOPMENT ARTICLE

Construction, categorization, and consensus: student generated computational artifacts as a context for disciplinary reflection

Michelle Hoda Wilkerson-Jerde

Abstract There are increasing calls to prepare K-12 students to use computational tools and principles when exploring scientific or mathematical phenomena. The purpose of this paper is to explore whether and how constructionist computer-supported collaborative environments can explicitly engage students in this practice. The Categorizer is a Javascript-based interactive gallery that allows members of a learning community to contribute computational artifacts they have constructed to a shared collection. Learners can then analyze the collection of artifacts, and sort them into user-defined categories. In a formative case study of the Categorizer for a fractal activity in three middle grade (ages 11–14) classrooms, there was evidence that participating students began to evaluate fractals based on structural and mathematical properties, and afterward could create algorithms that would generate fractals with particular area reduction rates. Further analysis revealed that students' construction and categorization experiences could be better integrated by explicitly scaffolding discussion and negotiation of the categorization schemes they develop. This led to the development of a new module that enables teachers and students to explore points of agreement and disagreement across student categorization schemes. I conclude with a description of limitations of the study and environment, implications for the broader community, and future work.

Keywords Computational thinking · Constructionism · Collaborative environments · Middle school · Disciplinary practices · Mathematics education

Introduction

Many collaborative technological environments are beginning to incorporate the construction of computational artifacts—such as simulations, games, or algorithms—as a way

M. H. Wilkerson-Jerde (✉)
Tufts University, Medford, MA, USA
e-mail: michelle.wilkerson@tufts.edu
URL: http://ase.tufts.edu/education/faculty/wilkerson-jerde

to participate in a learning community. As a collection, these artifacts can illustrate important patterns and themes in a domain of study. In this paper I draw from existing theories of learning and pedagogy to argue that such collections of student-generated artifacts hold untapped potential to help learners connect what they learn from constructing individual examples of a topic in math and science to the ways in which they organize and investigate that topic more generally. Specifically, I explore whether encouraging learners to identify patterns within their own collective work is one way to help them to attend to deep structural properties of objects within a domain, rather than only surface features—an important and difficult skill.

The Categorizer is a Javascript-based interactive gallery designed to encourage students to reflect on the themes and patterns evident within collections of their own and other's computational artifacts. It allows members of a learning community to (1) build and contribute artifacts to a shared space, (2) organize those artifacts into meaningful categories learners define themselves, and (3) review similarities and differences across different categorization systems. The Categorizer is designed to engage learners in making sense of connections between *construction* by defining computational rules used to produce an artifact, and *disciplinary reflection* by allowing them to explore and organize those same artifacts according to the themes and properties they find most relevant for the domain they represent.

I report on a case study of the Categorizer in three middle grade (ages 11–14) classrooms during a lesson about the mathematical structure of fractals, an increasingly popular way to explore fractions and functional reasoning in the middle grades (NCTM Illuminations 2003; Romburg and Kaput 1999). Findings suggest participating students identified connections between the computational rules they used to construct the fractals and ways of organizing those fractals mathematically, as evidenced by their use of the environment itself and on pre–post questionnaires. Further analysis highlights classroom interactions that especially encouraged students to explore these connections. These findings led to refinement of the tool itself, and suggested activities and challenges to consider for future implementations of shared collaborative galleries. I describe limitations of the study and environment, implications for the broader community, and future work.

Motivation and background

Educators and policymakers agree that contemporary education should engage students in the practices, skills, and core ideas that underlie a discipline—for example, by engaging in argumentation and supporting claims with evidence in science (NRC 2007, 2012), or finding patterns and moving across representations in mathematics (CCSSI 2010; NCTM 2000). Many technology-mediated collaborative learning environments provide tools and infrastructures for learners to engage in such practices by contributing to a shared collection of knowledge, or working toward a common goal. Examples include CSILE/Knowledge Forum (Scardamalia and Bereiter 1994; Scardamalia 2004), the Collaboratory Notebook (Edelson et al. 1996), the Math Forum (Renninger and Shumar 2002), WISE (Linn et al. 2003), SAIL (Slotta and Aleahmad 2009), and Science Online (Forte and Bruckman 2007). A major goal of such environments is to enable students to aggregate, engage with, and make sense of their collective contributions to a shared knowledge base.

At the same time, creating and using technology-mediated artifacts and tools is in itself a central aspect of what it means to learn, participate, and create knowledge in a discipline (diSessa 2000; NRC 2010, 2012; Papert 1980, 1996; Wing 2006). Simulations, statistical

Journal : Small 11423 Dispatch : 11-12-2013 Pages : 23
Article No. : 9327 LE TYPESET
MS Code : ETRD-D-13-00016 CP DISK

models and data, interactive visualizations, and technology-mediated experimentation all serve important roles in STEM practice (Chandrasekharan 2009; Kress and van Leeuwen 2001; Sabelli 2006). Correspondingly, many learning environments enable students to construct and use computational artifacts to explore ideas in math and science (diSessa and Abelson 1986; Jackson et al. 2000; Kahn 1996; Konold and Miller 2005; Papert 1980; Repenning et al. 2000; Resnick et al. 2009; Wilensky 1999).

Recently, many learning environments have started to integrate both collaborative and constructive approaches: so that *constructing artifacts* is one of the very ways that learners can contribute to *collaborative inquiry*. The WebLabs and Playground environments enable students to construct and share programmed games and mathematical models (Noss and Hoyles 2006) and the Science Created by You (SCY) environment requires students to build executable simulations as part of collaboratively pursuing a problem scenario solution (de Jong et al. 2012). Code Breaker has students collaborate over a network to construct and test cipher algorithms to decrypt a coded message (White 2009), and SAIL Smart Space allows students to assign tags to and sort student solutions for analysis (Tissenbaum et al. 2012). Networked SimCalc aggregates the results of students' algebraic investigations using a multi-representational collective display. Studies suggest that these sorts of integrated environments can facilitate community discourse (Ares et al. 2009), help students connect personal experiences to disciplinary learning (Hegedus and Moreno-Armella 2009) and afford powerful new learning activity structures (Brady et al. 2013).

The current project builds on this work by putting both construction and aggregation/classification of computational artifacts into students' hands to be explored and negotiated. Like constructing knowledge, determining how that knowledge is organized is an important component of reasoning in the STEM disciplines. The purpose is to explore the pedagogical potential of such an approach, and determine how designers and educators can realize that potential.

Theoretical framework

This project adopts the perspective that there is something special about creating and classifying computational artifacts as a way to participate in collaborative mathematical and scientific inquiry: computational ideas provide new and powerful ways of thinking about math and science phenomena (diSessa 2000; Papert 1980).

In describing the role of computation in creating new knowledge through mathematical experimentation, Bailey and Borwein (2011) note that "Never have we had such a cornucopia of ways to generate intuition. The challenge is to learn how to harness them, how to develop and how to transmit the necessary theory and practice" (p. 1419). This highlights the interrelationship of two aspects of integrating computation and disciplinary practice: (1) understanding how to use computational tools to explore a topic in math or science, and (2) identifying how doing so can inform one's exploration of disciplinary phenomena more broadly.

Constructionism: creating public computational artifacts

There is a long tradition of research exploring how programming and computational construction can help students explore STEM phenomena (diSessa 2000; Kafai and Resnick 1996; NRC 2010, 2012; Wilensky and Resnick 1999; Wilensky and Reisman 2006). One goal of such approaches is to connect mathematical and scientific ideas to students' experiences or expectations of how things work by having them make things "work" themselves. For example, the LOGO programming language allows students to generate complex geometric figures by instructing a *turtle* to combine actions like moving and turning in complex ways (Papert 1980). Constructing computational artifacts also encourages students to combine multiple ideas into a cohesive process, organize their understandings in new ways, and 'debug' understandings if their instructions produce something unexpected. The approach has especially been linked to students' learning of underlying structure, causal mechanism, and the epistemological aspects of a domain of study (Blikstein and Wilensky 2009; Harel and Papert 1991; Sherin 2001).

Another important component of Constructionism is that the artifacts students create should be public, so that students feel ownership over their constructions, learn from one another, and receive critique. This is particularly important when thinking about the role that collaborative learning environments can play in supporting computational construction for learning. For the purposes of this study, I will use the term *computational artifacts* to refer specifically to digital objects that students have created using a programming language or computational construction kit and have contributed to a public collaborative environment. Given the importance of the relationship between building and sharing in Constructionism, both a representation of the programmed rules or building blocks used to generate the artifact (the representation of "how things work") as well as a representation of the outcome when those rules are executed, are included as part of the artifacts that are shared among users.

Computational thinking: strategies for complex problems

Computational thinking (NRC 2010; Wing 2006) is often described as a set of ideas, strategies, and habits of mind that are useful for solving problems across curricular domains. It is one aspect of what diSessa (2000) calls computational literacy: the use of computational tools, ideas, and representations in the same way text and language are used in traditional literacy. For example, ideas such as automation, optimization, and recursion are useful for thinking about how to approach complex problems in any domain. This has led educators to explore integrating computational principles and ideas into STEM courses (Clark and Ernst 2008; Hambrusch et al. 2009) and beyond (Dierbach et al. 2011). A considerable amount of this work has focused on K-12 education (Barr et al. 2011; Bers 2010; Repenning et al. 2010), given the increased attention to problem solving and knowledge construction at these levels. While many suggest that programming and computational thinking approaches can increase students' analytical thinking skills (Kurland et al. 1986) and learning of other STEM content, research is still needed (Grover and Pea 2013; NRC 2010). For the purposes of this study, I am interested in whether students connect the *computational ideas* they leverage to construct individual artifacts—that is, the programmed rules and building blocks that are used—to inform what *patterns and themes* they identify in the learning community's collection of work.

The current project builds on the theories of constructionism and computational thinking to posit the following theoretical conjecture: one important part of making sense of a domain is exploring its themes, core ideas, and patterns as a disciplinary community (collaborative disciplinary inquiry). If those themes, core ideas, and patterns are illustrated by categorizing objects that students themselves create as they explore that domain (computational artifacts), students are likely to leverage the shared computational knowledge, experiences, and strategies they used when constructing those objects (computational thinking) in order to do so. To explore this conjecture, the case study I present in

Journal : Small 11423 Dispatch : 11-12-2013 Pages : 23
Article No. : 9327 ☐ LE ☐ TYPESET
MS Code : ETRD-D-13-00016 ☑ CP ☑ DISK

this paper focuses on one particular relationship between computational thinking and disciplinary inquiry skills, in one particular domain of study: that between computational *algorithms/rules*, and the skill of *pattern finding and classification* in the study of fractals and fractal structure.

Thinking about rules and finding patterns

Algorithms are a core idea in computer science and mathematics. Although many definitions of "algorithm" exist, they are generally characterized as a set of rules for how to take some input or starting state and produce a corresponding output or end state (NRC 2010)—for example, algorithms are used to multiply multi-digit numbers, or to tell a computer how to sort a list of numbers. Another important aspect of being able to understand algorithms is being able to predict what the output of a given algorithm will be for a given input (NRC 2012; Wing 2006). In The Categorizer, users must construct their objects by defining some such set of rules that the computer follows to generate the object; and are exposed to a collection of other objects that have been generated by other combinations of those same rules.

Pattern finding and classification involves observing and noting similarities and differences across related specimens that reflect a mathematical or scientific question or phenomenon. Making sense of patterns has been identified as a core crosscutting skill in the recent K-12 Science Framework (NRC 2012), and students' ability to identify and make sense of structure and express regularity has been valued by the mathematical community for years (CCSMI 2010; NCTM 2000). An important part of classification involves evaluating and grouping objects at different levels of analysis or representation, including at the level of microscopic elements, underlying structures, or relational/behavioral processes. The Categorizer is designed to help students to explore whether and how the rules they use to construct objects might produce certain observable patterns in the finished objects, and to experience the often dramatic differences in how objects look at the rules versus output level.

The intersection of *rules* and *pattern finding* is powerful for a number of reasons. There is a large body of literature documenting the importance—and difficulty—that learners face in differentiating between surface and deep structure in science and mathematics (e.g., Chi et al. 1981). More generally, understanding how learners organize a collection of examples can reveal what they understand to be the core ideas and perspectives in a domain. By allowing users to generate their own categorization schemes and make explicit the purpose of each category, the tool emphasizes that there are multiple ways to organize a domain of study and encourages students to make their reasoning explicit. The Categorizer seeks to encourage learners to build connections at deeper levels by using "deep level" rules and algorithms to construct their own computational objects, then providing them access to a shared gallery that allows students to explore both the rules and algorithms of other constructions, as well as their final forms, when deciding how to organize and make sense of the collection as a whole.

Design of the categorizer

The Categorizer is a flexible, web-based Javascript framework that integrates three interfaces representing three related activities: a Construction Interface, Categorization Gallery, and Theme Processor (Fig. 1). Its design is based on the theory and conjecture

Journal : Small 11423 Dispatch : 11-12-2013 Pages : 23
Article No. : 9327 LE TYPESET
MS Code : ETRD-D-13-00016 CP DISK

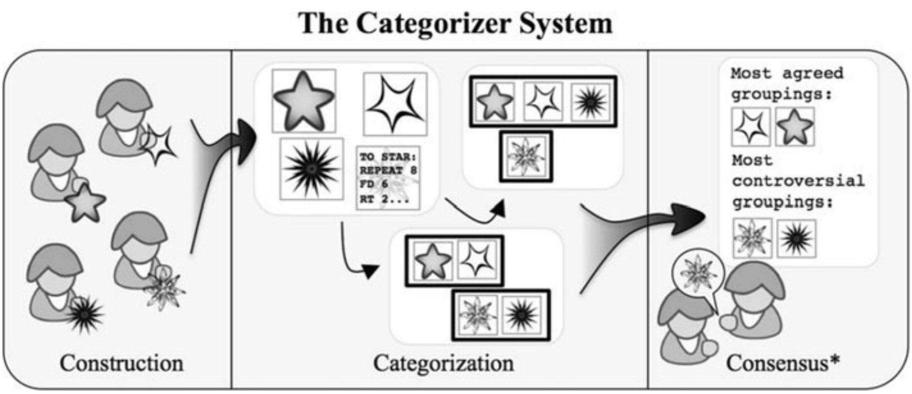


Fig. 1 A schematic of the categorizer system, which aggregates student constructions into a shared gallery, and student categorizations of those constructions into classroom-level themes. The theme processor module to support consensus-building (marked by an *asterisk*) was added as a result of findings from the current study

articulated above, though the Theme Processor was introduced during a second round of development based on findings from the preliminary implementation, as described in the next section.

To work within the Categorizer environment, students create one or more artifacts using the *Construction Interface*, which can be any computational toolkit that allows students to export a visual representation of (1) a set of rules and (2) the resulting artifact to a URL as image files. In this paper, I describe a study that allowed users to upload the rules (set of transformation functions) and resulting recursive patterns for Iterated Function System (IFS) fractals. However, any topic areas characterized by a complex relationship between underlying rules/processes and surface structure would work, such as functions and their resulting graphs (Leinhardt et al. 1990), iteratively generated geometric figures and their generating code (Papert 1980), or emergent visuospatial patterns derived from similar individual interaction rules (Goldstone and Wilensky 2008).

Each contribution from a community member is then uploaded for display to a shared *Categorization Gallery* with all other artifacts constructed by a given learning community. Users visiting the gallery can double click on any object to see its underlying rules. When ready, a user can create one or more category windows to sort the artifacts into meaningful groupings that they deem important, or that have been requested by a facilitator or teacher. The user would enter a name and description for each category window they create. Once all of the objects have been sorted into categories, the user can save the categorization scheme.

The *Theme Processor* was added to the Categorizer as a result of the formative case study described in this paper, and allows a facilitator or teacher to view an aggregated summary of how a particular community of learners is choosing to organize their collection. This summary uses simple matrix decomposition methods to analyze the collection of categorization schemes that users produce. This information can then be used to inform the facilitator of which sets of gallery objects students often group together, and which items are not grouped similarly by different students and hence may reflect borderline or controversial cases for the classroom population as a whole.

These three different modules reflect three underlying broad and interrelated theories of learning guiding the overall development and implementation of the Categorizer:

Journal : Small 11423 Dispatch : 11-12-2013 Pages : 23
Article No. : 9327 LE TYPESET
MS Code : ETRD-D-13-00016 CP DISK

constructivism/constructionism, collaborative knowledge-building, and disciplinary engagement. Drawing from constructivist theories of learning and constructionist theories of pedagogy, students create and obtain feedback about their own fractals using the Construction Interface and when interacting with others in the Categorization Gallery, always have access to the rules used to create those fractals. To support collaborative knowledge-building students' constructions are all accessible to one another, and the Theme Processor helps the class explore other ways of classifying and thematically analyzing those contributions. Finally, as students contribute diverse objects, the nature of their potential classifications may shift or be redefined. Since the categories students use to sort and describe one another's objects are created by students themselves and made visible through the Theme Processor, students' *own* meaning-making processes around the topic of interest, as well as the class' consensus building around those themes, are emphasized, rather than organizations introduced by an outside authority.

Case study: creating and analyzing fractal structure

To explore whether the Categorizer does support students' exploration of the connections between underlying computational rules and ways of classifying objects that represent a particular phenomenon, I conducted one implementation of the Categorizer tool in the context of a lesson on iterated function system fractals (Demko et al. 1985) in three middle school mathematics classrooms. This provided a context to test the first version of the tool, explore the learning theories that underlie its design, and inform subsequent development and refinement. Case studies such as this are one way to conduct research on educational interventions in situ, especially during early phases of development. They are well-suited for research that aims to maintain sensitivity to contextual influences, and can reveal unexpected dimensions that affect how technology designs are used in real settings such as classrooms (Khan 2008). They also allow researchers to collect in-depth data that can speak to students' social interactions and understandings of complex subject matter, and to challenge or extend theory (Yin 2008).

Iterated Function System Fractals, henceforth referred to as IFS fractals or just fractals, are self-similar geometric figures defined by a set of geometric affine transformation functions (the "function system") to be applied to a set of points, in this case to a unit square. The fractal is generated by recursively copying all points in the unit square into each transformation, such that a copy of the unit shape is repeated inside each defined transformation. These repeated transformations reflect the *algorithm* that is the focus of the exploration, the squares that define the transformations are its rules and the resultant fractal the output. For example, Fig. 2a, b feature two IFS fractals: Fig. 2a is a version of the familiar Seirpinski gasket that is defined by three transformations defining scale reductions and translations of the unit square to a non-overlapping triangle arrangement. Figure 2b is the fractal generated when the topmost transformation also includes a 90° rotation. Figure 2c features a number of fractals created by students during the study to illustrate the diversity of forms that can be generated.

As a content area, fractals represent an especially productive context for studying the integration of computational object creation and collaborative inquiry for a number of reasons. By their nature, the space of IFS fractals is rich with a diversity of themes that may emerge as students select different sets of transformation rules. These sets of rules exhibit systematic, mathematically important relationships. For example, IFS fractals often feature geometric, branch-like, or shell/fern-like structures. They are often 'crisp' or

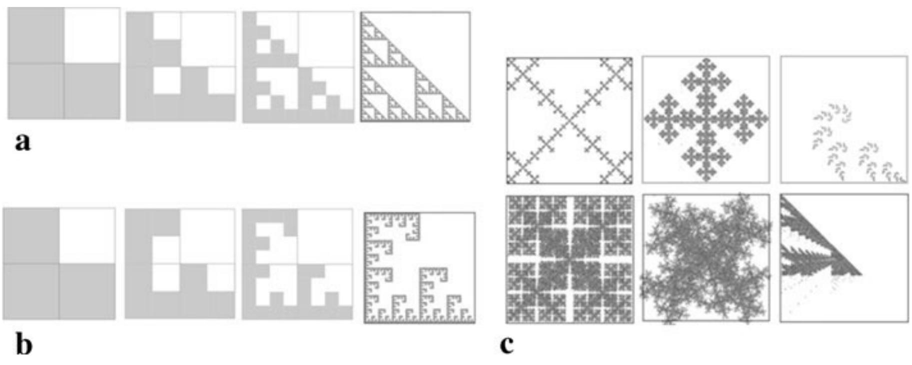


Fig. 2 Examples of IFS fractals and their underlying rules. **a** Sierpinski triangle, **b** Sierpinski w/rotation, **c** Example IFS fractals

well-defined when transformations do not intersect, but cloudy or fuzzy when they do. Certain arrangements of transformations produce figures that do not illustrate fractal structure because they do not appear to repeat within themselves. And, the area occupied by points that undergo recursive transformations reduces measurably during each iteration of the function system. Fractals have been applied to the study of a number of topics, from the study of cancer (Baish and Jain 2000) to the development of computer graphics (Demko et al. 1985). Using fractals to explore fractions and functional reasoning is increasingly popular in middle grades mathematics because they are engaging, cognitively complex, and technology-mediated (NCTM 2001, 2003; Romburg and Kaput 1999).

Constructing public fractals can also provide learners with new lenses into the structure of the content area and new motivations to explore that structure. Complex fractals are often generated from a small set of simple rules requiring deep understanding. There is a high potential for rich diversity of student-produced fractal types to be categorized, and designing fractals is seen by many as a personally expressive/aesthetic pursuit.

Research question

Given (a) the increasing centrality of computational objects in STEM practice and in computer-supported collaborative inquiry environments, (b) the claim that such approaches are useful in particular because they might provide a new way to help students integrate computational thinking into disciplinary inquiry, and (c) a particular focus on connections between *algorithms/rules* and *classification*, the research question driving this study is:

> To what extent do learners who use the Categorizer build connections between the *computational rules/algorithms* used to construct individual fractal objects, and the *organizing themes* they identify within collections of such objects?

Methods

Participants

This study was conducted with three middle grades mathematics classes at a small, sub-urban K-8 school in the Midwestern United States. Six 6th grade, six 7th grade, and eight

8th grade students (total $n = 20$) students consented to participate in the study. The students' teacher assisted in planning the classroom activities, and was present during the implementation. All students had prior exposure to the fractal construction tool through a previous workshop. During that workshop, the classroom teacher and facilitators decided during some class sessions to print out students' fractals and compare them with one another at the end of class. This decision led to the conjecture that more tightly coupling construction and classification activities using a computer-mediated environment might encourage students to draw connections between deep structure and pattern finding. The present study was the first time students interacted with the first version of the Categorizer.

Class activities

Each class session was 1 h long. First, each session started with a short paper-and-pencil warm up activity designed to remind students of their prior lesson on fractals and of the construction interface, as well as to collect baseline data on students' understandings of fractal structure, described in more detail below. Next, students were allowed to freely explore the fractal construction interface (Fig. 3), including the new "save to gallery" feature. We asked students to try and make sense of the connections between certain rules they chose to construct fractals and the resulting features within those fractals. During our previous lesson, students had completed challenges to create fractals with specific qualities (such as "spiral fractals" or "spongy fractals") and were hence familiar with such a request.

After about 10 min, students were introduced to the Categorization interface of the tool (Fig. 4). This introduction included an explicit mention that they could double click a given image to see the rules used to generate it. Students were then given a series of prompts on the board to complete over the class session. There were not assigned times that each task would begin and end, although students were given a warning about 15 min before the end of the activity. Instead the goal was to better understand how students themselves would interact with the environment as a constructive and collaborative tool. The first prompt was to create groupings that reflected any patterns they found interesting, to familiarize themselves with the interface, and to collect their first impressions of what themes make sense to capture from the fractal collection. Next, they were to try and create categories that reflected the mathematical properties about fractals discussed during the last class session: things such as density, "crispness" or "fuzziness;" structure (twisting, branching, sponging); area reduction; and so on. The hope was that since students had spent time during both class sessions exploring the connections between rules and structure, they would explore and cite fractal rules as part of this classification process. At the end of the session, students completed a short follow up questionnaire.

Data collection

Data collected during the study included the written pre and post questionnaires, synchronized video and screen capture of one consenting focal student per class using Camtasia (Techsmith 2004), and Categorizer usage log files. At the beginning and end of each class, students completed a pre questionnaire that asked them to predict the next "step" or iteration for a set of fractal rules ("Item 1"; understanding of algorithm), to predict from a set of four possibilities the fractal figure that would result from those rules ("Item 2", understanding of algorithm), and to generate two sets of rules that would produce two different fractals exhibiting a particular area reduction factor ("Item 3,"

Journal : Small 11423 Dispatch : 11-12-2013 Pages : 23
Article No. : 9327 ☐ LE ☐ TYPESET
MS Code : ETRD-D-13-00016 ☑ CP ☑ DISK

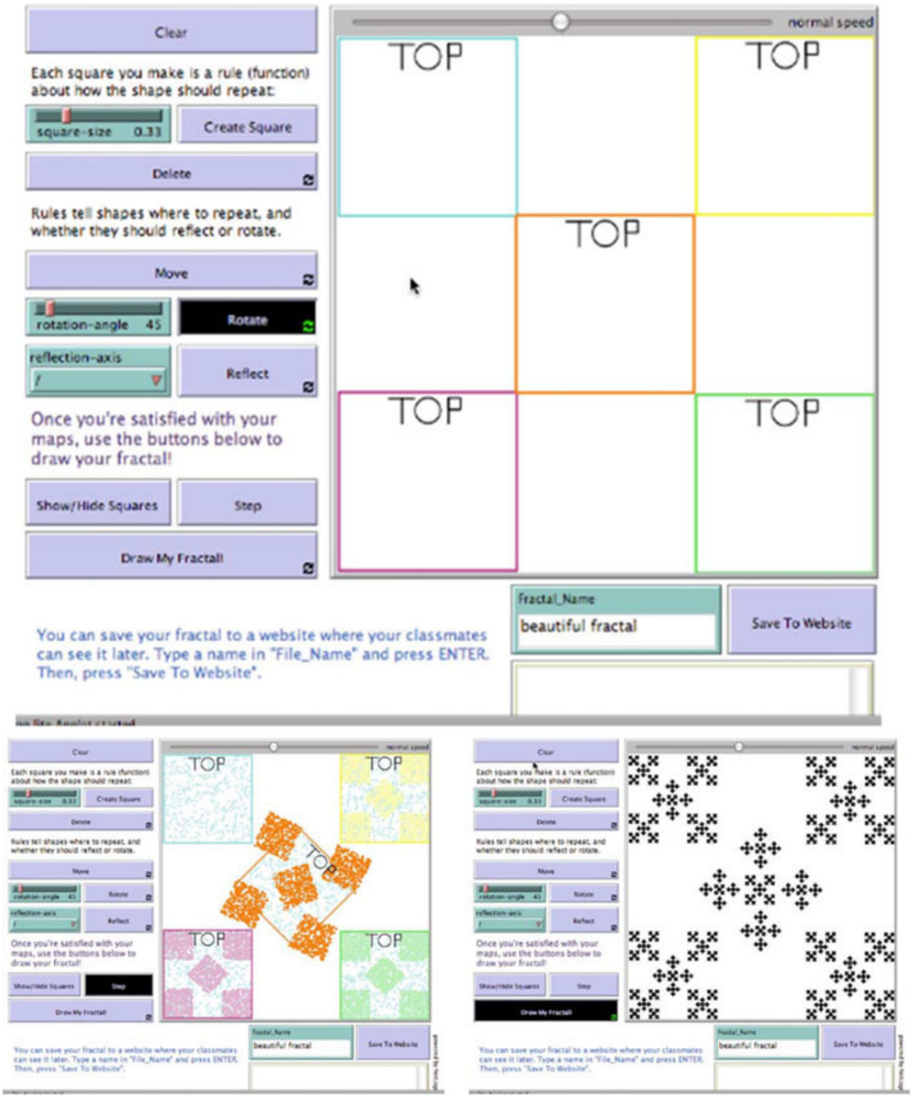


Fig. 3 Fractal construction interface (*top*) and resulting computational objects, illustrating stepwise iterations (*bottom left*) and final 'infinite' product (*bottom right*)

connection between algorithm and theme). Since IFS fractals are a relatively new, exploratory topic area in the middle grades, items were loosely adapted from a higher-level textbook (Alligood et al. 2000) specifically for this study. Items for students in grade 6 were simpler than those in grades 7 and 8 in that items for 7th and 8th graders included not only translation, but also rotation rules.

Camtasia video captured students' discussions with one another and class facilitators, as well as all on-screen activity including the fractals that focal students generated and their ongoing interactions with the Categorizer interface. Finally, the Categorizer log files captured students' categorization schemes created over the course of each class session including time

Journal : Small 11423 Dispatch : 11-12-2013 Pages : 23
Article No. : 9327 LE TYPESET
MS Code : ETRD-D-13-00016 CP DISK

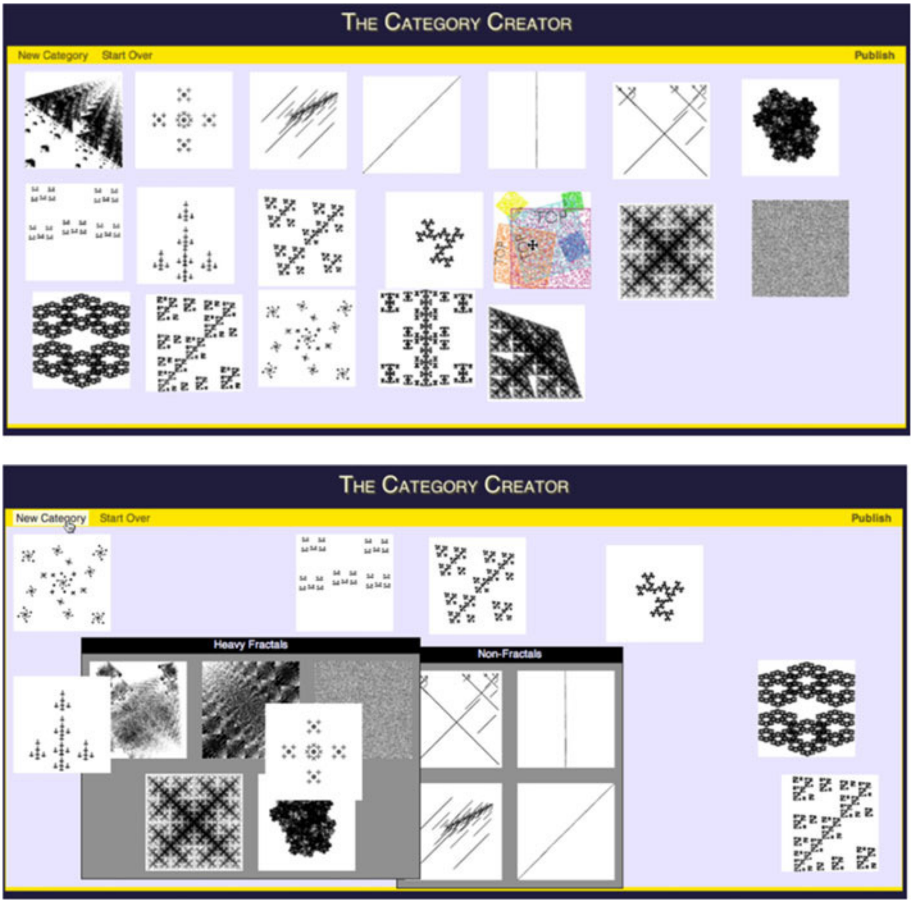


Fig. 4 Categorization gallery featuring fractal objects and their underlying rules, accessed by double-clicking a fractal (*top*). Fractals can be placed into one or more user-generated category windows (*bottom*)

stamps, the text entered by students as titles and descriptions of categories and categorization scheme, and adjacency tables that indicated which objects students grouped together.

Analysis

Categorizer log file data was analyzed using a bottom-up grounded theory approach (Glaser and Strauss 1967) to characterize themes students generated while classifying artifacts within the environment, and identify to what extent those themes included aspects of the deep structure rules they were using to generate those artifacts. Pre and post questionnaires were scored for whether students produced correct or incorrect responses for each item, which would indicate that they had started to understand the algorithms that generate fractals (Items 1 & 2), and build connections between fractal rules and thematic mathematical properties (Item 3) after working with the environment. Finally, video of focal students was coded to identify what activities within the software tool (e.g., construction, categorization, exploration) and within the classroom environment (e.g., discussion) they engaged in over the course of the class session. Detailed descriptions and

Journal : Small 11423 Dispatch : 11-12-2013 Pages : 23
Article No. : 9327 ☐ LE ☐ TYPESET
MS Code : ETRD-D-13-00016 ☑ CP ☑ DISK

examples of coded data for each of these coding schemes are provided in the results section.

To establish reliability, an independent rater also analyzed a representative subset of at least 20 % of each data corpus. Agreement for log file coding was 87 %, questionnaire scoring was 100 %, and video coding was 86 %. Most disagreements in video coding were a result of student multitasking—in particular, *discussing* while also engaging in some other activity with the software. When these discussion disagreements were resolved among coders, reliability of video coding rose to 95 % agreement.

Results

Overall, results indicate that while students generated a number of categorization systems including many that reflected *potential* connections between rules and organizational themes, only a few students explicitly linked these connections in their category descriptions. But although this suggests some students may not have formed such connections, significantly more students did create multiple sets of rules to generate fractals belonging to a requested mathematical category after the intervention than before. In this section, these findings are described in more detail, and further analysis of one focal student's interaction with the Categorizer is provided to shed light on why students may not have made more explicit connections between rules and categories in the environment. The following section describes resulting modifications to the environment and activity structures to address these findings.

Log file data

Over the course of the entire implementation, a total of 68 categorization schemes were uploaded. Of these, 35 % of the schemes either explicitly referred to rules, or identified features of fractals that were directly related to features of their underlying rules. The majority of categorization schemes, however, did not mention rules explicitly, and 65 % could be related to rules only tangentially or not at all. Table 1 includes full descriptions of the categories identified, explicit criteria used to identify each category during analysis, examples of each, and overall results of the analysis. There was no evidence of meaningful differences between grade levels in categorization schemes created. For example, there was not consistent change in the proportion of rules-based to non-rules-based schemes students employed by grade level, and one instance of rule-based categories emerged from a Grade 6 student and the other from a Grade 8 student.

Pre–post questionnaire data

While the categories that students generated while interacting with the Categorizer indicate that only some students explicitly referenced rules when generating categories for their class' fractal collection, there is more evidence that students began to link rules to deep structure in the pre and post questionnaire data. This is especially evident when analyzing student responses to Item 3 of the pre–post questionnaire.[1]

[1] Items 1 and 2 on the post questionnaire were designed to be more difficult than those on the pre questionnaire. Pre–post differences on both items were not significant (Item 1, $W = 12$, $p =$ n.s.; Item 2, $W = 28$, $p =$ n.s.).

Journal : Small 11423 | Dispatch : 11-12-2013 | Pages : 23
Article No. : 9327 | ☐ LE | ☐ TYPESET
MS Code : ETRD-D-13-00016 | ☑ CP | ☑ DISK

Table 1 Student categorization schemes, by presence of connection between theme and underlying fractal rules

	Description	Example	*N* (/68)	%
More evidence of connection	At least one category explicitly cites *rules* in its description	"Simple first steps" (*features fractals made from only a few nonoverlapping transformations*)	2	3
	At least once category refers to a feature *uniquely determined by a rule* (examples include density, rotations, reduction in area)	"These are fractals that look fuzzy" (*features fractals that result from nonoverlapping transformations*)	22	32
	At least one category refers to *fractal structure* such as shape or self-similarity	"They all have triangles" (*features fractals with triangular structure*)	19	28
	At least one category identifies fractals as *recognizable or aesthetically pleasing*	"Each fractal looks like there are little people inside of them" (*features fractals for which a link to systematic rules is difficult to determine*)	12	18
Less evidence of connection	Categories do not satisfy any of the above descriptions	"Mine/not mine"	13	19

Examples drawn from student log file data

Journal : Small 11423 | Dispatch : 11-12-2013 | Pages : 23
Article No. : 9327 | LE | TYPESET
MS Code : ETRD-D-13-00016 | CP | DISK

Table 2 Number of students with correct responses on pre and post questionnaires by item ($N = 20$; Wilcoxon paired signed rank test)

	Pre	Post	W	N_{diff}	p
Item 3 (at least one rule)	5	14	0	9	$p < .005$
Item 3 (two or more rules)	4	11	5	9	$p < .025$

Item 3 dealt explicitly with connecting mathematical themes to construction rules by asking students to create two different sets of rules that would produce fractals that would look different, but that each exhibit the same rate of area reduction (area reduction rates varied between 3/4 and 5/9, all well within target grade levels; NCTM 2000). Table 2 reports the number of correct responses on Item 3 of pre and post versions of the questionnaires administered during the intervention. Student responses are marked correct if there is evidence students included boxes intentionally sized to approximate fractional units of the area reduction rate, positioned in nonoverlapping configurations. For example, a reduction factor of ¾ would be marked correct if it included three boxes, each approximately one-fourth of the total area of the square, positioned so no area of the squares are overlapping. Both responses were marked correct if at least one box was visibly translated, rotated, or reflected in the second picture. A two-tailed Wilcoxon paired signed rank test indicates that significantly more students produced more sets of rules that would generate fractals belonging to a category that exhibits a certain rate of area reduction for Item 3 on the post questionnaire, including a significant number of students who moved from generating no rule sets to two or more.

One explanation for this is that over the course of the session, students began to attend to the ways in which fractal rules mimic the area model of fraction because this was one of many prompts used during the activity. However, like Items 1 and 2, this topic had been covered in our previous session, yet there was marked improvement on this item and not the others. Furthermore, of the students that generated rules for Item 3, some included features specific to the construction interface, or extra transformations of the rule beyond only defining the needed area (see Fig. 5). This suggests that at least some students were actively connecting their constructive experience *within* the environment with normative ways of classifying *multiple different* fractal structures.

But why, if so few students explicitly mentioned rules in their organization of fractal categories, were they so adept at articulating rules to create multiple different fractals exhibiting a particular mathematical theme? And, how could this apparent connection be leveraged and built upon using the Categorizer environment?

In-depth analysis

One way to shed light on this apparent tension is through in-depth analysis of what actually happens when students interact with the Categorizer environment. For the purposes of this paper, analysis focuses on one of the four focal students, 6th grader Carol.[2] I focus on Carol because she never explicitly, or ostensibly implicitly, connected fractal rules to the organizations she created within the Categorizer: her only saved scheme was coded in Table 1 as "Aesthetic". To describe Carol's experience using the tool, I present her activity within the environment, as well as important events completed within it, using a

2FL01 [2] Pseudonym.

Journal : Small 11423 Dispatch : 11-12-2013 Pages : 23
Article No. : 9327 LE TYPESET
MS Code : ETRD-D-13-00016 CP DISK

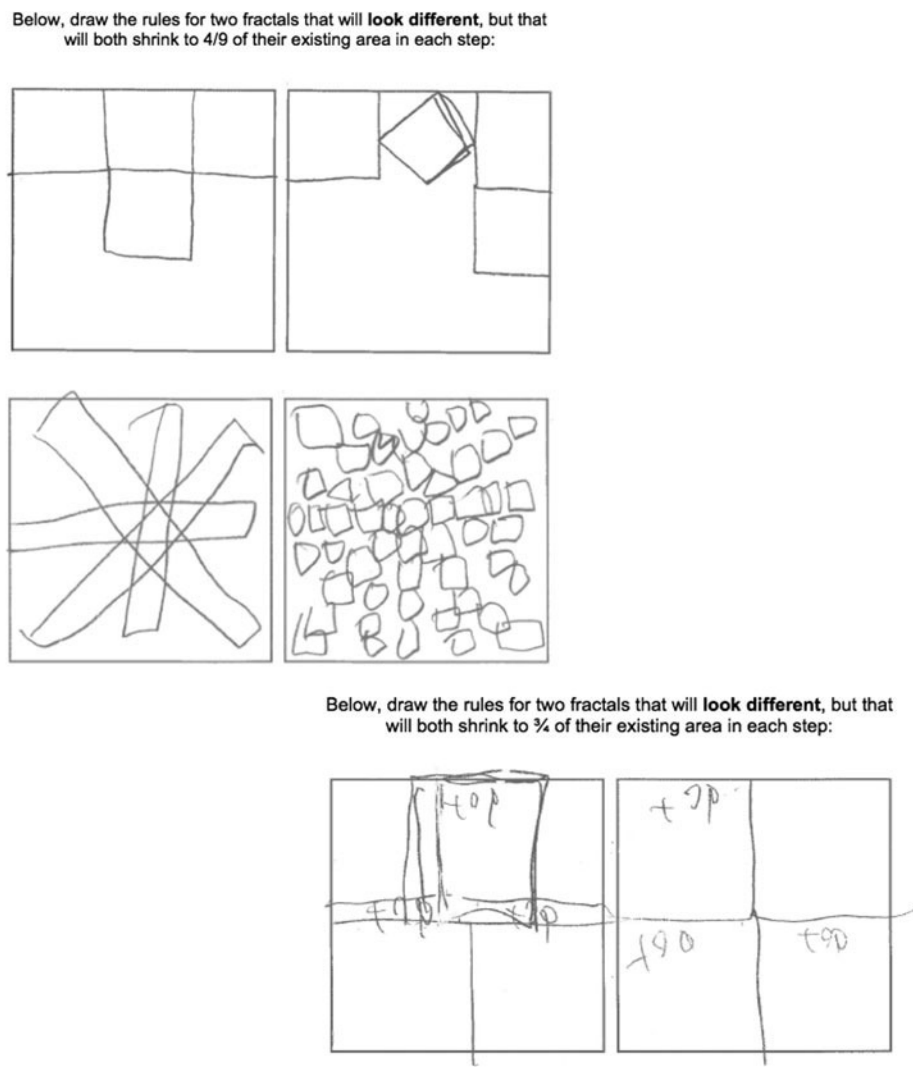


Fig. 5 Examples of student post questionnaire responses measuring students' linking of mathematical and computational properties. The first examples show valid (*top*) and invalid (*middle*) rules for fractals that illustrate an area reduction factor of 4/9 (7th & 8th grade question). The second example (*bottom*) shows valid rules for an area reduction factor of 3/4 (6th grade question)

time series diagram (Fig. 6; similar to, but simpler than, those described in Hmelo-Silvere et al. 2011).

Figure 6 represents Carol's navigation within and across different parts of the Categorizer on a minute-to-minute basis over the course of her class session. The timeline was constructed using screen capture and synchronized student video to identify times when Carol was:

- *constructing* fractals (viewing and interacting with the Categorizer construction screen),

Journal : Small 11423 Dispatch : 11-12-2013 Pages : 23
Article No. : 9327 □ LE □ TYPESET
MS Code : ETRD-D-13-00016 ☑ CP ☑ DISK

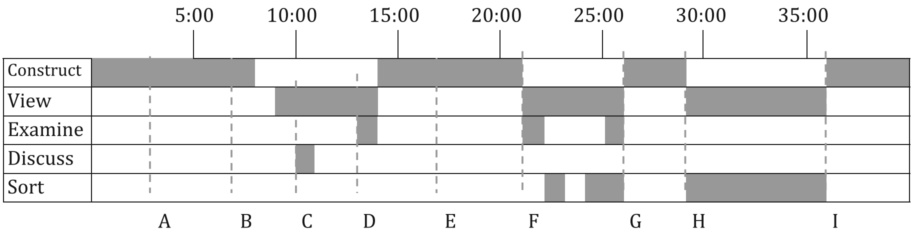


Fig. 6 Timeline of Categorizer use by focal student, Carol

- *viewing* those of her classmates (viewing the categorization screen, including moving fractals within the interface without placing them into categories)
- *examining* the rules of particular fractals in the gallery (double-clicking fractals in the categorization interface to view their rules)
- *discussing* what she was doing with her peers (speaking with other students during the activity as captured on synchronized video)
- *sorting* gallery items (creating categories and placing fractals into them)

Three things are immediately obvious: First, Carol did not apparently spend much time examining her peers' fractals, or discussing them with others. Second, she did not begin to sort the fractals into categories until almost the end of class, even though students were asked to do so sooner. Third, Carol spent most of her time moving between constructing her own fractals and viewing and examining particular fractals within the shared gallery, rather than analyzing the fractals as a group.

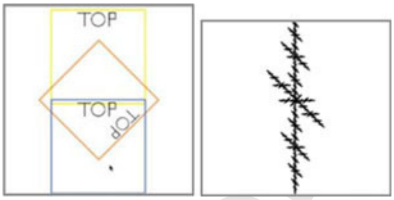
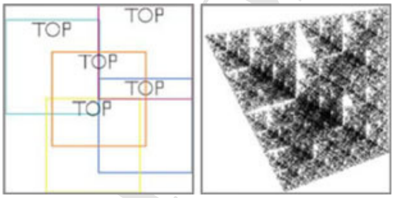
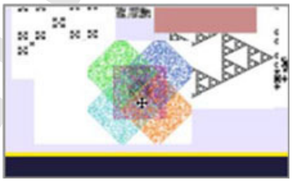
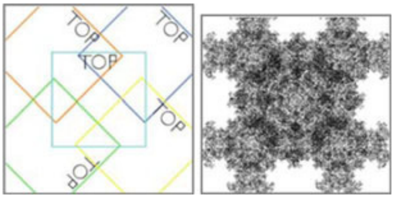
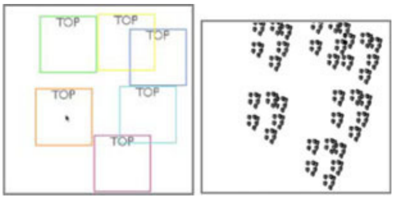
An analysis of exactly what objects Carol constructed, referenced, and examined during this time sheds further light on this pattern. The vertical lines labeled A-I in Fig. 6 correspond to the events listed in Table 3. It appears that Carol was primarily moving between the Categorizer Gallery and the Construction Interface so that she could identify particular patterns she especially liked and reproduce them as her own—for example, she twice returned from examining a particular fractal in the Gallery to reconstruct that fractal herself (or something close to it, see events D, E and G, H). This pattern also seems consistent with the way that Carol did categorize the objects when she engaged in sorting activity (events G and I)—she seemed primarily concerned with which fractals she liked, and which she herself constructed.

This suggests some important strengths and areas for improvement in Carol's experience. First, it is clear that Carol not only sensed ownership for her constructions, but was trying to systematically learn more about how she could build objects she found interesting—by uncovering and reproducing their underlying rules. Second, Carol was exploring a particular 'theme' that connected rules and output—one that reflected her own interest in objects she created or wished to create. This theme was reflected in her sorted categories as well as in the way she identified and attempted to reproduce particular patterns. Third, Carol was interested in identifying, learning more about, and discussing her peers' constructions.

It seems, then, that Carol found construction and her own sense of ownership over artifacts as interesting and rewarding enough that it interrupted (during event G) and played an important role (during event I) in how she classified the larger group of objects. This general pattern was evident in other case studies of focal students, and was noticed more generally by facilitators at the implementation. What was missing from such an

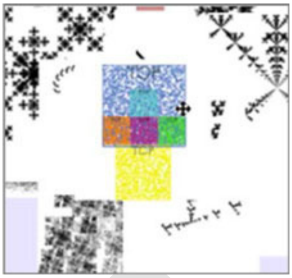
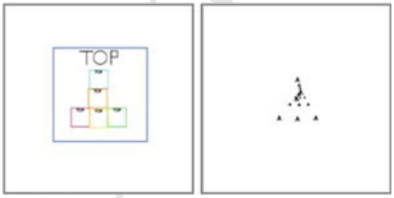
Journal : **Small 11423** Dispatch : **11-12-2013** Pages : **23**
Article No. : **9327** ☐ LE ☐ TYPESET
MS Code : **ETRD-D-13-00016** ☑ CP ☑ DISK

Table 3 Review of events marked on the timeline featured in Fig. 6

A	Carol contributes fractal 1 (right)	
B	Carol contributes fractal 2 (right)	
C	Carol calls out to two classmates that she has located their fractals in the categorization gallery	
D	Carol examines a fractal in the gallery (right)	
E	Carol contributes fractal 3, which appears to be a copy of the rules of the fractal she examined (right)	
F	Carol contributes fractal 4 (right)	

Journal : Small 11423 Dispatch : 11-12-2013 Pages : 23
Article No. : 9327 ☐ LE ☐ TYPESET
MS Code : ETRD-D-13-00016 ☑ CP ☑ DISK

Table 3 continued

G	Carol examines fractal (right) in the midst of categorizing by ownership into categories entitled "mine" and "others", and returns to the construction interface	
H	Carol contributes a new fractal, whose rules mimic but do not replicate features of the rules of the fractal she examined (right)	
I	Carol sorts fractals into aesthetic categories (entitled "iwish" and "other")	

investigation wasn't Carol's sense of connection or an emphasis on the relationship between rules, outcomes, and themes—but rather a motivation to push themes beyond aesthetic interest toward structural or mathematical foci. In the next section, I describe modifications to the tool and to supporting activities that might provide such motivation.

Carol's story is not unique. Of the four focal students for which I have data like Carol's, all of the students spent the most time constructing artifacts (between 60 and 71 % of time spent, versus Carol's 55 %). But unlike Carol, the rest of the students spent less time viewing their peers' fractals without sorting them (between 8 and 10 % of time spent, versus Carol's 40 %). Interestingly, the only focal student to have spent more than 2 % of his time examining the underlying rules of fractals was also the only to have classified fractals by themes that bore explicit mathematical meaning (related to density and self-similarity).

These patterns of use suggest that even though students were encouraged to explore these themes during the session, they did not spend much time doing so. It makes sense that they might not find those themes intrinsically interesting right away. What students *did* engage with was the construction activity and their ability to view and share artifacts and identify those of their peers. There are indications even in student log file data that this may have been a widespread pattern: Students created on average more than 2 fractals for each categorization scheme saved (143 fractals/68 categorization schemes $\approx$2.1).

Discussion

The Categorizer is a specific ongoing project, but reflects a broader goal shared by the educational technology community: to engage students in STEM knowledge construction

Journal : Small 11423 | Dispatch : 11-12-2013 | Pages : 23
Article No. : 9327 | □ LE | □ TYPESET
MS Code : ETRD-D-13-00016 | ☑ CP | ☑ DISK

practices by enabling them to express and test their ideas using a computational medium. Therefore, there are two levels of contribution of this work. One lies in the design and refinement of the tool itself, described in this section, and the other lies in the design principles gleaned from its use and study that might be more generally informative to the educational technology community, described in the next section.

In terms of the design and refinement of the Categorizer tool in particular, these findings suggest that more connection between student construction—something the students are motivated and engaged in doing—and the identification of more mathematically and scientifically relevant themes might be in order. In particular, it seems that one way to engage students in exploring more scientifically or mathematically relevant themes is to tie that investigation explicitly to more opportunities for students to construct and investigate individual fractals.

This finding directly led to one refinement to the Categorizer tool itself, and two specific activity structures that will be integrated into future implementations of the tool. First, the *Theme Processor*, described in the Design section toward the beginning of this paper, was added to the environment as a result of this study. This will highlight points of agreement and disagreement at the category level. The aim is to provide students a more explicit sense of ownership and opportunity for discussion around categorization themes, much in the same way they were already engaging in discussion around the individual objects they constructed. Second, future implementations of the tool will involve activities designed to motivate a connection between fractal rule sets and *categorization themes* (versus only specific *fractal objects*). Two examples of activities that can help foster such integration include (1) a "Build for My Category" challenge—where students must create novel fractals that can be classified as members of existing themes determined by their peers or the classroom as a whole; and (2) a "Recreate My Categories" challenge, where students challenge their peers to uncover and articulate the connective threads that pull together different categories in a given student's scheme.

While the reported study was only a preliminary case-based exploration of the original Categorizer environment, a larger design-based research project (DBR; Brown 1992; Collins 1992) can shed even further light on whether and how students leverage knowledge related to computational thinking to make sense of mathematical and scientific phenomena, and how they can be supported. The DBR approach involves designing and researching theory-based educational interventions in real educational contexts in a way that is iterative and reciprocal. The goal is to develop interventions, learning theories, and design frameworks that are scalable and sensitive to the realities of educational practice. Scholars have highlighted the potential DBR especially holds for the development and study of technology-enhanced learning environments (Wang and Hannafin 2005), especially those that involve new educational content, practices, or approaches (Cobb et al. 2003). This study would have benefitted from returning to the same classroom to investigate how the same students might interact and learn differently given the new design features. However, this was difficult given the realities of the academic school year. This highlights the importance and tension of working with school and site partners on iterative design projects that require repeated, and at times unexpected, rounds of formative study and testing.

Conclusions

The goal of this study was to explore whether and how a computer-based environment—based on constructionist and collaborative learning principles—could support the insight

Journal : Small 11423 | Dispatch : 11-12-2013 | Pages : 23
Article No. : 9327 | LE | TYPESET
MS Code : ETRD-D-13-00016 | CP | DISK

that computational construction activities might contribute to learners' more general STEM inquiry practices. Findings suggest that (1) such an environment can help students begin to explore the connections between rules and the features of their resulting outputs, and that (2) while some students also begin to connect themes they identify to features of rule sets, (3) this exploration might be even better supported by giving students more reason to assume ownership of *classification schemes themselves*, in addition to the objects within.

This study was preliminary, and only took place over the course of 1 day during a short mathematics unit. Despite this, students generated sophisticated fractal structures, began to explore important mathematical properties of those structures, and offered more ways to construct fractals with particular mathematical properties after interacting with the system. These early results should be extended by investigating longer sustained classroom engagement with such an environment, and by investigating how students and teachers may use the environment in other domains.

Although exploratory, this study does suggest that The Categorizer represents one way of leveraging students' sense of creativity and ownership to help them explore relationships between computational exploration and other forms of scientific and/or mathematical inquiry. It also highlights the difficulty and importance of merging these practices in ways that remain authentic to and contribute to mathematical and scientific inquiry. A larger design-based research project that investigates the relationship between all three designed modules, students' interaction with the system and one another, and their learning of mathematical relationships among fractal objects can further highlight how different aspects of students' reasoning are supported by these design elements. On a theoretical level, the findings suggest a new area to explore regarding how computer-based learning environments that attempt to integrate construction and inquiry allow students to explicitly establish ownership and investment in products of *both* of these processes in a connected way: the artifacts they generate, as well as the ways in which they might generate meaning and insight into what those artifacts represent.

Acknowledgments Many thanks to the teachers, students, and school administrators who worked with me on this project. It would not have been possible without the help of Aditi Wagh, Nathan Holbert, Forrest Stonedahl, Susa Stonedahl, Christopher Macrander, and Uri Wilensky. I am also grateful for feedback on earlier versions of this manuscript provided by several anonymous reviewers, J. Michael Spector, Jenna Conversano, and Ben Shapiro.

References

Alligood, K. T., Sauer, T. D., & Yorke, J. A. (2000). *Chaos: An introduction to dynamical systems*. New York: Springer.
Ares, N., Stroup, W. M., & Schademan, A. R. (2009). The power of mediating artifacts in group-level development of mathematical discourses. *Cognition and Instruction, 27*(1), 1–24.
Bailey, D. H., & Borwein, J. M. (2011). Exploratory experimentation and computation. *Notices of the AMS, 58*(10), 1410–1419.
Baish, J. W., & Jain, R. K. (2000). Fractals and cancer. *Cancer Research, 60*(14), 3683–3688.
Barr, D., Harrison, J., & Conery, L. (2011). Computational thinking: A digital age skill for everyone. *Learning & Leading with Technology, 38*(6), 20–23.
Bers, M. U. (2010). The TangibleK robotics program: Applied computational thinking for young children. *Early Childhood Research & Practice*, *12*(2).
Blikstein, P., & Wilensky, U. (2009). An atom is known by the company it keeps: A constructionist learning environment for materials science using agent-based modeling. *International Journal of Computers for Mathematical Learning, 14*(2), 81–119.

Journal : Small 11423 Dispatch : 11-12-2013 Pages : 23
Article No. : 9327 ☐ LE ☐ TYPESET
MS Code : ETRD-D-13-00016 ☑ CP ☑ DISK

Brady, C., White, T., Davis, S., & Hegedus, S. (2013). SimCalc and the networked classroom. In S. J. Hegedus & J. Roschelle (Eds.), *The SimCalc vision and contributions: Democratizing access to important mathematics* (pp. 99–121). Dordrecht: Springer.

Brown, A. L. (1992). Design experiments: Theoretical and methodological challenges in creating complex interventions in classroom settings. *Journal of the Learning Sciences, 2*(22), 141–178.

Chandrasekharan, S. (2009). Building to discover: A common coding model. *Cognitive Science, 33*(6), 1059–1086.

Chi, M. T., Feltovich, P. J., & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science, 5*(2), 121–152.

Clark, A. C., & Ernst, J. V. (2008). STEM-based computational modeling for technology education. *The Journal of Technology Studies, 34*(1), 20–27.

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

Collins, A. (1992). Toward a design science of education. In E. Scanlon & T. O'Shea (Eds.), *New directions in educational technology*. Berlin: Springer.

Common Core State Standards Initiative. (2010). *Common core state standards for mathematics*. Retrieved December 15, 2012, from http://www.corestandards.org/Math.

de Jong, T., Weinberger, A., Girault, I., Kluge, A., Lazonder, A. W., Pedaste, M., et al. (2012). Using scenarios to design complex technology-enhanced learning environments. *Educational Technology Research and Development, 60*(5), 883–901.

Demko, S., Hodges, L., & Naylor, B. (1985). Construction of fractal objects with iterated function systems. *AC SIGGRAPH Computer Graphics, 19*(3), 271–278.

Dierbach, C., Hochheiser, H., Collins, S., Jerome, G., Ariza, C., Kelleher, T., & Kaza, S. (2011). A model for piloting pathways for computational thinking in a general education curriculum. In *Proceedings of the 42nd ACM Technical Symposium on Computer Science Education* (pp. 257–262). ACM.

diSessa, A. (2000). *Changing minds: Computers, learning, and literacy*. Cambridge, MA: MIT Press.

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

Dubinsky, E. (1995). ISETL: A programming language for learning mathematics. *Communications on Pure and Applied Mathematics, 48*(9), 1027–1051.

Edelson, D. C., Pea, R. D., & Gomez, L. M. (1996). The collaboratory notebook. *Communications of the ACM, 39*, 32–33.

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

Forte, A., & Bruckman, A. (2007). Constructing text: Wiki as a toolkit for (collaborative?) learning. In *International Symposium on Wikis: Proceedings of the 2007 International Symposium on Wikis* (Vol. 21, No. 25, pp. 31–42).

Glaser, B. G., & Strauss, A. L. (1967). *Discovery of grounded theory: Strategies for qualitative research*. Mill Valley, CA: Sociology Press.

Goldstone, R. L., & Wilensky, U. (2008). Promoting transfer by grounding complex systems principles. *The Journal of the Learning Sciences, 17*(4), 465–516.

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

Hakkarainen, K. (2003). Emergence of progressive-inquiry culture in computer-supported collaborative learning. *Learning Environments Research, 6*(2), 199–220.

Hambrusch, S., Hoffmann, C., Korb, J. T., Haugan, M., & Hosking, A. L. (2009). A multidisciplinary approach towards computational thinking for science majors. In *ACM SIGCSE Bulletin* (Vol. 41, No. 1, pp. 183–187). New York: ACM.

Harel, I., & Papert, S. (1991). *Constructionism*. New York: Ablex Publishing.

Hegedus, S. J., & Moreno-Armella, L. (2009). Intersecting representation and communication infrastructures. *ZDM, 41*(4), 399–412.

Hmelo-Silver, C. E., Jordan, R., Liu, L., & Chernobilsky, E. (2011). Representational tools for understanding complex computer-supported collaborative learning environments. *Analyzing Interactions in CSCL, 12*, 83–106.

Jackson, S., Krajcik, J., & Soloway, E. (2000). Model-It: A design retrospective. In M. J. Jacobson & R. B. Kozma (Eds.), *Advanced designs for the technologies of learning: Innovations in science and mathematics education* (pp. 77–116). New York: Wiley.

Kafai, Y., & Resnick, M. (1996). *Constructionism in practice: Designing, thinking, and learning in a digital world*. Mahwah, NJ: Lawrence Erlbaum Associates.

Kahn, K. (1996). ToonTalk™: An animated programming environment for children. *Journal of Visual Languages & Computing, 7*(2), 197–217.

Khan, S. (2008). The case in case-based design of educational software: A methodological interrogation. *Educational Technology Research and Development, 56*, 423–447.

Konold, C., & Miller, C. D. (2005). *TinkerPlots: Dynamic data exploration [Computer software]*. Emeryville, CA: Key Curriculum Press.

Kress, G., & Van Leeuwen, T. V. (2001). *Multimodal discourse: The modes and media of contemporary communication*. London: Hodder Arnold.

Kurland, D. M., Pea, R. D., Clement, C., & Mawby, R. (1986). A study of the development of programming ability and thinking skills in high school students. *Journal of Educational Computing Research, 2*, 429–458.

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

Linn, M. C., Clark, D., & Slotta, J. D. (2003). WISE design for knowledge integration. *Science Education, 87*(4), 517–538.

National Council of Teachers of Mathematics. (2000). *Principles and standards for school mathematics*. Reston, VA: NCTM.

National Council of Teachers of Mathematics (2003). Fractal tool [computer software]. NCTM Illuminations Resource Website, Retrieved August 17, 2013, from http://illuminations.nctm.org/ActivityDetail.aspx?ID=17.

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

National Research Council Committee for the Workshops on Computational Thinking. (2010). *Report of a workshop on the scope and nature of computational thinking*. Washington, DC: National Academies Press.

National Research Council Committee on Science Learning, & Kindergarten Through Eighth Grade. (2007). In R. A. Duschl, H. A. Schweingruber, & A. W. Shouse (Eds.), *Taking science to school: Learning and teaching science in grades K-8*. Washington, DC: National Academies Press.

Noss, R., & Hoyles, C. (2006). Exploring mathematics through construction and collaboration. In: *Cambridge handbook of the learning sciences* (pp. 389–405) Cambridge: Cambridge University Press.

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

Papert, S. (1996). An exploration in the space of mathematics educations. *International Journal of Computers for Mathematical Learning, 1*(1), 95–123.

Renninger, K. A., & Shumar, W. (2002). Community building with and for teachers at the Math Forum. In K. A. Renninger & W. Shumar (Eds.), *Building virtual communities: Learning and change in cyberspace* (pp. 60–95). New York: Cambridge University Press.

Repenning, A., Ioannidou, A., & Zola, J. (2000). AgentSheets: End-user programmable simulations. *Journal of Artificial Societies and Social Simulation, 3*(3), 351.

Repenning, A., Webb, D., & Ioannidou, A. (2010, March). Scalable game design and the development of a checklist for getting computational thinking into public schools. In *Proceedings of the 41st ACM Technical Symposium on Computer Science Education* (pp. 265–269). New York: ACM.

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

Sabelli, N. H. (2006). Complexity, technology, science, and education. *Journal of the Learning Sciences, 15*(1), 5.

Scardamalia, M. (2004). CSILE/Knowledge forum®. In: *Education and technology: An encyclopedia* (pp. 183–192). Santa Barbara, CA: ABC-CLIO.

Scardamalia, M., & Bereiter, C. (1994). Computer support for knowledge-building communities. *The Journal of the Learning Sciences, 3*(3), 265–283.

Sherin, B. L. (2001). A comparison of programming languages and algebraic notation as expressive languages for physics. *International Journal of Computers for Mathematical Learning, 6*(1), 1–61.

Slotta, J. D., & Aleahmad, T. (2009). WISE technology lessons: Moving from a local proprietary system to a global open source framework. *Research and Practice in Technology Enhanced Learning, 4*(2), 169–189.

Techsmith Corporation (2004). Camtasia [Computer software].

Tissenbaum, M., Lui, M., & Slotta, J. D. (2012). Co-designing collaborative smart classroom curriculum for secondary school science. *Journal of Universal Computer Science, 18*(3), 327–352.

Wang, F., & Hannafin, M. J. (2005). Design-based research and technology-enhanced learning environments. *Educational Technology Research and Development, 53*(4), 5–23.

Journal : Small 11423 Dispatch : 11-12-2013 Pages : 23
Article No. : 9327 □ LE □ TYPESET
MS Code : ETRD-D-13-00016 ☑ CP ☑ DISK

White, T. (2009). Encrypted objects and decryption processes: Problem-solving with functions in a learning environment based on cryptography. *Educational Studies in Mathematics, 72*(1), 17–37.

Wilensky, U. (1999). *NetLogo [Computer Software]*. Evanston, IL: Center for Connected Learning, Northwestern University.

Wilensky, U., & Reisman, K. (2006). Thinking like a wolf, a sheep, or a firefly: Learning biology through constructing and testing computational theories—an embodied modeling approach. *Cognition and Instruction, 24*(2), 171–209.

Wilensky, U., & Resnick, M. (1999). Thinking in levels: A dynamic systems approach to making sense of the world. *Journal of Science Education and Technology, 8*(1), 3–19.

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

Yin, R. K. (2008). *Case study research: Design and methods*. Thousand Oaks, CS: AGE Publications.

Michelle Hoda Wilkerson-Jerde is an Assistant Professor in the Department of Education at Tufts University. Her research interests include the design and study of computational toolkits to help learners make sense of complex and data-intensive systems in mathematics and science.