Math on a Sphere: Using Public Displays to Support Children's Creativity and Computational Thinking on 3D Surfaces

Sherry Hsi
Lawrence Hall of Science
University of California at Berkeley
Berkeley, CA USA
sherryh@berkeley.edu

Michael Eisenberg
Dept. of Computer Science and Institute of Cognitive Science
University of Colorado
Boulder, CO USA
duck@cs.colorado.edu

ABSTRACT
Math on a Sphere (MoS) is a newly developed Web-based environment that enables children to imagine, program, and share creative designs on a public spherical display, the "Science on a Sphere" system created by the National Oceanic and Atmospheric Administration (NOAA). The MoS software, similar in spirit to the Logo language, was installed at an exhibit located in the Lawrence Hall of Science at the University of California at Berkeley and at the Fiske Planetarium at University of Colorado, Boulder. Twenty-five children ages 8 to 13 in two cohorts tested the MoS software during a half-day workshop held at the Lawrence Hall. In addition to using the MoS software to create beautiful and original works of art, children also engaged in hands-on crafts and inquiry-based math activities to further promote learning of spherical geometry and computational thinking. MoS software workshop had a positive impact on children's engagement, but had mixed results about their understanding of geometry as evidenced by direct observations and results from pre/post-surveys, which are reported here.

Categories and Subject Descriptors
K.3.1 [**Computers and Education**]: Computer Uses in Education – *computer-assisted instruction*

General Terms
Human Factors.

Keywords
Math on a Sphere, public displays, spherical geometry, Logo, computational thinking.

1. INTRODUCTION
One of the most important functions of science centers and museums is their role in hosting innovative educational events that engage and motivate schoolchildren through exploration, creativity, and reflection. This paper describes just such an innovative event–a workshop introducing the newly-developed *Math on a Sphere* (MoS) system for exploring spherical geometry and computational thinking. The workshop was our first experiment in teaching non-Euclidean geometry (generally regarded as relatively advanced) to children.

Figure 1. A rainbow design created by a 10-year old girl participating in the Math on a Sphere workshop. (The audience member discernible toward the bottom right provides a sense of the size of the spherical display.)

Briefly (a bit more detail will be provided shortly), the MoS system makes use of a large spherical display, called "Science on a Sphere". The display was developed by the United States National Oceanic and Atmospheric Administration (NOAA) and consists of a set of high-resolution projectors (and associated software) that show graphics and animations on the surface of a white sphere (six feet in diameter). Our aim in developing MoS was to explore the viability of "giving public displays to the public", much like public arts or community murals in cities. Building on the notion of a "constructivist museum" [4], our goal was to create an expressive digital medium that would encourage young learners to build on their prior knowledge, build intuitive ideas about a challenging and important topic (in this case,

IDC 2012, June 12–15, 2012, Bremen, Germany.

spherical geometry and computational thinking), construct and share new artifacts in a learning community, and reflect upon their understanding.

By way of anticipating the workshop activities reported here, Figure 1 shows an example of a spherical design created by one of the children in our workshop; incidentally, the figure conveys some sense of the size and impressive graphic capabilities of the SOS display. Figure 1 also suggests yet another reason to explore spherical geometry–the sheer aesthetic delight of producing spherical designs. Indeed, while the workshop was intended for children, its purpose was also not lost on parents: when asked why she drove over 75 miles to bring her child to the Math on a Sphere class to a local San Francisco Bay Area science center, one mother replied, "*There are so few opportunities that allow kids to be both technical, mathematical, and artistic at the same time. This workshop offered all three things in one.*"

The MoS programming environment (again, a bit more detail will be given shortly) is based on moving a Logo-like "turtle" on a spherical surface. The language of Logo and turtle geometry has historically been successful in promoting and supporting children's mathematical, procedural, and logical thinking when instructional goals are made explicit, intuitive understandings are "mathematized" and scaffolding is provided by a teacher [1,3, 5,6]. Our workshop assumed no prior experience with Logo (or with any sort of programming), and thus introduced the ideas of spherical geometry and computational thinking through a variety of means. Children built upon their intuitions by first using craft materials to make and decorate a model sphere with Styrofoam balls, paper circles, paper plates, colored tape, string, colored pens, and sewing pins. Next, children as a group used their bodies to role play a "turtle walking on a plane", then later moved to computers to program an online turtle-cursor to draw lines, make shapes, and polygons (Figure 3). We aimed to get across big ideas in both spherical geometry and computational thinking, but couched these as fun secrets to be discovered by playing with spheres. Some of these ideas are as follows: Traveling on a sphere surface is different than traveling on a flat plane. Lines and triangles "behave" differently on spheres than on planes. A triangle doesn't always have angles that add up to 180 degrees. Computational thinking ideas and practices of computer scientists included writing commands, variables, procedures, loops; learning how to structure a computer program to execute properly, commenting a program, and debugging.

The remainder of this paper describes the purpose and results of our workshop in greater detail. The following (second) section briefly explains the MoS system and outlines its technical implementation. The workshop and its evaluation are discussed in the third section; while the fourth section summarizes the results of the workshop and concludes with some final reflections and discussion of ongoing work.

2. THE MATH ON A SPHERE (MoS) SYSTEM

The MoS system has two designed components: 1) the MoS web client interface (written in HTML5 and JavaScript) is where the student interacts with three windows – an editor window for composing programs; a command line interpreter which allows the user to type in lines directly (e.g., procedure calls that draw patterns on the sphere); and an interactive sphere view window allows the user to see a graphical rendering of the sphere display itself on his/her computer screen. Through direct manipulation, students can "grab" and rotate this Sphere by mouse button presses and movements, so that the user can view all portions of the surface. Students can use simple abbreviations such as "forward 60" to tell the turtle to go forward and draw sixty paces on the sphere. (Figure 2 shows the programming interface for the MoS system.)

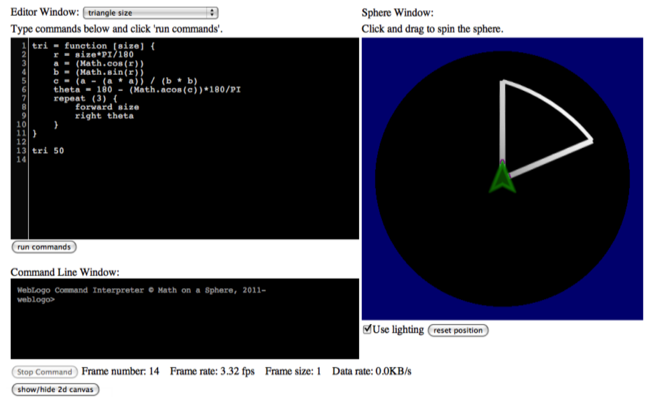

Figure 2. A view of the Web-based programming interface used in our workshop. The window at the upper left is an editor in which programs can be written; the rectangular window toward the bottom left is a "command interface" in which spherical turtle commands can be typed and interpreted directly. At the right, a view of the sphere is shown; here the user can see the result of running a turtle program. The sphere in the right window may be turned "directly" by mouse movements so that the user can look at all regions of the sphere.

The second component of the Math on a Sphere system is the local server that sits between the web client and NOAA's existing server software that drives the large SoS large public display, and communicates programs to the sphere. In order to do its job, the local server must be installed "on site", on the same device as the planetarium's SoS system itself. The job of the local server is to act as the "glue" between the end-user programs written in the language interface, and the SOS system that displays the results of the program commands on the sphere. The local server sends to the SoS a "movie" of constant frame-rate composed of a number of individual frames in a standard image format (JPG, PNG, etc.) When the MoS website is loaded onto a suitably prepared machine at the science center, this button can be used to activate a connection between the MoS web client and the giant spherical display at the museum. (See also Basman *et al*., 2012 for a more extended discussion.)

3. WORKSHOP EVALUATION

We installed the MoS web client onto thirteen wireless PC laptops that were running the Google Chrome web browser. Children from around the Greater San Francisco Bay Area were then invited to participate in the workshop located at the Lawrence Hall of Science; the workshop attracted upper elementary and middle school-aged public school students, home-schooled children, and children from private alternative schools. The half-day workshop began with an overview and orientation in the here, Sphere exhibit; sample designs were shared on the big sphere to provide a context and motivation for programming.

Figure 3. Scenes from the workshop. Intentional design with great circles: with rubber bands; with the Math on Sphere client program; and for the public display on a large sphere.

Then, children moved to a large classroom downstairs. Two different versions of the workshop were tested on two consecutive days and compared. We describe a handful of the different components here. The first activity was a pre-test evaluation that asked both multiple choice questions and open resource questions to help determine the level of prior mathematics knowledge, use of computers, as well as interest in learning math, geometry, and computers. This was followed by a large group discussion about circles, lines, angles, and spheres. In the first day's workshop, children were given a design task – to make a sphere with hands-on craft materials that captured as many of the math ideas discussed and/or with repeating patterns. Some kids were simply using materials to explore (e.g., what would happen if you tried to wrap paper around a sphere to make a shell?), while others had goal-oriented sphere designs (e.g., make a sphere using folded paper plates or wrap a ball using great circles). In the second day's workshop, kids received more computer time. They began in a large group with masking tape and were asked to make small, medium, and large triangles by placing tape onto inflated toy globes, and to then share their observations about the triangle's shape, size, and properties. Kids began to notice that if the side of a triangle is as long as a quarter or half hemisphere, it starts to lose its original definition of a triangle when operating on flat planes. Students were also asked to role-play a turtle cursor. One child played the turtle (while holding a ball of yarn to draw out a trail), while other children instructed the child how to make a square, using the specific language of Logo (FORWARD 20, RIGHT 90, FORWARD 20, RIGHT 90, etc.). After these kinds of embodied exercises, the students were then introduced to the MoS language and given about 1-1.5 hours of programming time on computers. Paper instruction sheets were created for students with additional tutorials about commands and special functions. Two instructors walked around to help students while one main teacher provided periodic, timely instruction using an overhead computer projector.

Below is an example program written by a 13-year old which selects the digital color pen to be yellow, sets the drawing speed of the pen to be fast, and repeats twenty times the drawing of a long bent line.

```
set color 45
setspeed 20
repeat 20
{pendown fd 300 lt 50 fd 200 rt 9
 penup bk 100}
```

At the end of the workshop, children were given a post survey with specific drawing tasks, feedback about the workshop content, and multiple choice questions about their interest in math, geometry, and computers.

4. RESULTS, CONCLUSIONS, AND REFLECTIONS

Future workshops are planned in the next two years for about 200 students. Here, we report upon the first workshop trial with 26 students. All students were able to script a design for public display. Their designs ranged from familiar objects (basketballs, a tree, an eyeball) to more abstract designs (colorful balls made from great circles, arcs, repeating patterns, and lines) (Figure 4). The younger students (8-years old) did not have the prior background in mathematics to understand "degrees" and "angles" which required more remediation. Their designs reflected simple explorations on the sphere with the forward command (lines). Participants were surveyed about their experience and interest in math, geometry and computer programming both before and after the workshop. 38% of the students had never had any programming experience, 43% had just a little experience, and 19% had some kind of prior experience programming. In both groups, there was a significant increase in interest in computer programming (pre=4.24, post=4.62, p=.034, Wilcoxin Signed Ranks Test) even though many participants indicated a strong interest in computer programming before the workshop on the pre-test. However, there was no change in interest in math pre=3.86, post=3.76, p=.157) nor in geometry (pre=3.76, post=3.90, p=.453).

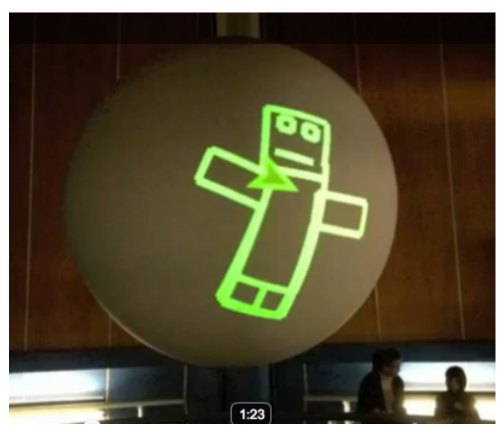

Figure 4. "Robotey Joe" was created by an 11-year old boy who wrote 68 lines of code using the Math on a Sphere program.

A majority of the students (94%) liked seeing their designs on the big sphere and 77% liked showing their designs to everyone else.

Still, not all students were comfortable with this last, "public" aspect of the work–an email follow-up survey to parents revealed this explanation from the parent of a younger student: "*Seeing the work displayed was not important. She was excited until she saw how hers compared to other students*." This suggests that presentation in public spaces requires more preparation and care in managing learner expectations especially when there is a wide range of ages and abilities among the students. In examining changes to their pre-post test gains on assessment items in math, geometry and use of MoS programming language, there was a slight increase overall in both groups, and students in group 2 (the second day) made more gains. This is somewhat expected since they were given an hour more time with computer time using MoS.

When examining specifically students' pre-post test gains on those questions that asked about geometry, students made further gains on their prior knowledge of planar geometry, however, scored lower on questions that aimed to test their knowledge of spherical geometry (Figure 5.) Students clearly had some basic knowledge about planar geometry before taking the workshop and this knowledge was likely reinforced and clarified during workshop instruction as indicated by their post-test responses. However, students were still confused about the representation of a line and triangle when displayed on a spherical surface.

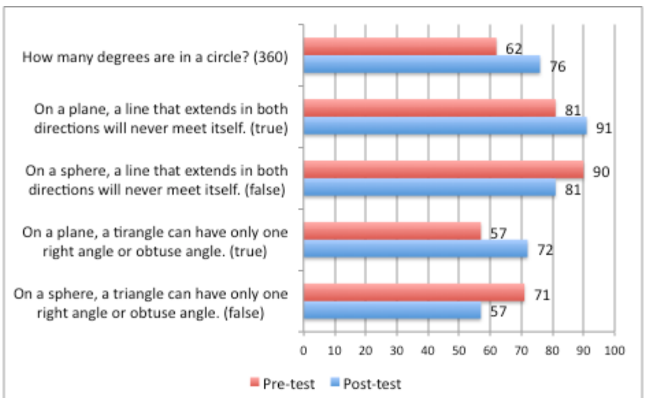

Figure 5. Percent answered correctly on five of the geometry items for students who took the pre and post survey (N=21)

What to make of these results? First, based on this first workshop that tested an early version of the MoS software in a public science center, we are encouraged that students are able to create designs for the public in a few hours with support. However, in this short day, students were still building their intuitions about navigating on a 3D surface and some remained confused about non-Euclidean ideas (e.g. a triangle can have more than one right angle.). We are not discouraged by this result as 3D geometry is an advanced topic in mathematics. We would expect that a multi-day workshop with longer periods should enable children to go still further in exploring spherical geometry. Our plans are to continue improving the MoS programming language and functionality, and to add workshop activities so it can be expanded to a multiple session class to better support children's interest in learning programming, of spherical geometry and computational thinking. In addition, we plan to use children's confusion about non-planar surfaces as a learning opportunity for more discussion, as well as create better assessments to measure the knowledge and attitudes gained from the workshop.

In the medium- to longer term, we plan to create documentary materials available through the website itself, allowing for more extended (or perhaps advanced) explorations of a sort that would be difficult to introduce in a brief museum-based workshop. Indeed, one of the motivating design features of this project is that it weaves together intensive museum experiences (such as our workshop) with more leisurely long-term work at home or in the classroom. Conceivably, the home-based projects can then themselves serve as reasons to revisit the museum at a later time to try out an advanced sphere design created over a longer period.

Finally, it should be noted that the MoS project is itself a first step in what we believe to be novel and fascinating directions. For instance, the project represents an initial step in allowing children to create programs on a non-planar (in this case, spherical) surface; one could imagine extending the idea to other interesting mathematical surfaces such as cylinders, cones, or hyperboloids. Yet another important direction is that children may now write and display their program results in large *public* spaces; one might imagine, in this vein, future endeavors in which programs are displayed (e.g.) on the sides of buildings or large playground floor spaces. Similarly, a customizable MoS-like interfaces could be designed for personal ambient learning spaces such as a basic layout of a child's bedroom or a school classroom. In this way, children may eventually be able to write programs that can display designs on the walls, ceiling, or immobile object surfaces of a specific room. We are unaware of existing applications that can do this. While MoS is, as noted, only an initial step in these sorts of directions, we believe that it suggests the promise of a much more extended and ubiquitous programming culture available to children in the future.

5. ACKNOWLEDGMENTS

The work described in this paper was supported in part by the National Science Foundation under grant DRL1114388. Michael MacFerrin designed the original MoS mathematical software; Antranig Basman and Michelle Redick programmed the current system. Sarah Elovich, Risa Wolfson, and Elizabeth Hager-Barnard co-taught the workshop and Maia Werner-Avidon led the evaluation. Thanks also to the CTI technical team at the Lawrence Hall of Science. Thanks also to Francisco Salas at the Fiske Planetarium and the National Oceanic and Atmospheric Administration (Boulder, CO) for their support.

6. REFERENCES

[1] Abelson, H. and diSessa, A. [1980] *Turtle Geometry*. Cambridge, MA: MIT Press.

[2] Basman, A.; Redick, M.; and Eisenberg, M. [2012] Math on a Sphere: Toward Public Control of Public Displays. (*manuscript in preparation*.)

[3] Clements, D. H., & Sarama, J. (2009) Learning and teaching early math: the learning trajectories approach. Taylor & Francis. Clements, D. H. & Battista, M. T. (1989) Learning of Geometric concepts in a Logo Environment. *Journal of Research in Mathematics Education*, 20, pp. 450-467.

[4] Hein, G. *Learning in the Museum*. Routledge, 1998.

[5] Khasawneh, A. A. (2009): Assessing Logo programming among Jordanian seventh grade students through turtle geometry, *International Journal of Mathematical Education in Science and Technology*, 40:5, 619-639

[6] Papert, S. [1980] *Mindstorms*. NY: Basic Books