MEGL Summer 2016: Virtual Reality

Joseph Frias, Mae Markowski, & Mark Boahen

August 15, 2016

1 Introduction

As the virtual reality team, we are tasked with creating games, visualizations and demonstrations in Unity to aid in mathematical education and research. Our past work includes a klein bottle roller coaster to teach users the concept of orientability, an ordinary differential equations solver which solves chaotic attractors in real time to display the importance of initial conditions, and a rotations demo, made to complement a paper by Daniel Ramras and David Pengelley, which illustrates a nullhomotopy in SO(3). This summer, our work was divided into three areas: perfecting the rotations demo for publication in the journal The Mathematical Intelligencer, creating a visualization for one of MEGL's graduate students, Jack Love, in his thesis project on polygons, and making a 3-manifolds demonstration in virtual reality, modeled after Jeff Weeks' Curved Spaces program. While the former has been completed, the latter two projects are ongoing and will carry over into the fall semester.

2 Perfecting the Rotations Demo

During the spring of 2016, we were contacted by Daniel Ramras of IUPUI, who was seeking a virtual reality version of his html animations showing the double-twist nullhomotopy of the rotation group SO(3) to complement his and David Pengelley's paper "How efficiently can one untangle a double-twist? Waving is believing!" This paper formalizes the well-known Dirac belt trick or Philippine candle dance and presents a series of quaternion rotations simulating this double-twist. Over the course of the spring semester, the virtual reality team created Version 1.0 of the Rotations Demo for Mac and Windows, an interactive version of the rotations described in the paper. The user could control the speed, tilt, and viewpoint of the rotations, all while trying not to throw up. Additionally, the user was able to see a white contrail traced out by one of the rotation axes, a more clean visual of the nullhomotopy.

Over the summer, we were again contacted by Drs. Ramras and Pengelley, who sought a "Version 2.0", which corrected some elements in the first version. Some were simple:

make the starting speed slower, rotate the robot viewpoint from the belly button rather than the feet, add an optional pause time after each loop, and include a credits screen. Some amendments, however, were more complicated: add optional contrails to each axis, and the most challenging of all, rotate the axes to exactly match the html animations on Dr. Ramras' website. We were able to add the r,g,b buttons which turned on/off the red, green, and blue axes' contrails, respectively. Our rotations started off by tilting up, whereas in the animations it showed the axes first tilting towards the floor. After much trial and error, we realized we had to negate the rotation in the z-direction, and rotate the camera angle so that the red axis pointed away from the user, the blue was to the right, and the green was straight up.

The finalized version with all of these changes is up on MEGL's website as Version 2.0, and will be featured in Drs. Ramras' and Pengelley's article in the Mathematical Intelligencer.

3 How I Learned to Love the Polygon

In the Spring of 2016, we contacted Jack Love to see if we could help him with his thesis on the moduli spaces of polygonal linkages. A polygonal linkage in $\mathbb{R}^3$ can be conceptualized by a set of non-zero 3-vectors that sum to zero. The vertices of the polygon are the ordered partial sums of the vectors, and the edges are the vectors based at the corresponding vertex. As an example, consider three vectors A $= e^{\frac{2\pi i}{3}}$, B $= e^{\frac{4\pi i}{3}}$, and C $= 1$ in the complex plane. being roots of unity, they sum to zero. If we start at the origin, we can use C as our first edge and get to the point (1,0). This is our second vertex, with the origin being the first. Now we add A to this, and get $(1+cos(\frac{2\pi}{3}), sin(\frac{2\pi}{3}))$. This is our last vertex, with A being the edge that connected C to the vertex. We finally add B and get back to the origin. This is a planar linkage (living in $\mathbb{R}^2$, and we mostly dealt with planar 4-gons. Using complex numbers and elementary geometry, we made an animation of a 4-gon undergoing bending, and we plotted the diagonal lengths of the 4-gon while it was changing. Our results seemed to square with what Jack Love had, and we will continue to work with him to expand what we can show.

4 Real or Unreal? Visualizing 3-manifolds

At the tail end of updating our demonstration of the rotation project, our team had a conversation with our advisor about what our next main project should be. We discussed porting a game, called Curved Spaces (Curved Spaces), to a VR ready game engine such as Unity or Unreal Engine. In this game, you can choose between a few examples of 3-manifolds to explore as a spaceship. A topological 3-manifold (without boundary) is a second countable Hausdorff topological space that is locally Euclidean in 3 dimensions. It is essentially a "volume" in which squinting enough will "bring you back" to Euclidean space. In particular, we were interested in hyperbolic manifolds, which are modelled after hyperbolic space instead of Euclidean space. For hyperbolic spaces, using the hyperboloid model which is based off the Lorentz group, the game creates a tesselation of 3-space in which you travel. This, the game does by way of inputting a set of isometries of $\mathbb{H}^3$ that generates the tesselation by iteration. While in the early stages, this project will likely spawn side-projects that can be released while the main project is ongoing.