Smoke Simulation on the surface of 3-D Manifolds

ALEX KASSIL, UC Berkeley
CATHERINE CANG, UC Berkeley
CHARLES SUN, UC Berkeley
KEVIN ZHANG, UC Berkeley

Additional Key Words and Phrases: manifolds, donuts, smoke simulation

ACM Reference Format:
Alex Kassil, Catherine Cang, Charles Sun, and Kevin Zhang. 2021. Smoke Simulation on the surface of 3-D Manifolds. 1, 1 (April 2021), 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Traditional smoke simulators model smoke on a 2D plane. We plan to explore creating simulating smoke on a torus, klein bottle, and curved space (e.g. hyperbolic space). We have successfully simulated smoke on a 2D plane and a torus. This is useful to explore both how smoke behaves when on a plane, as well as enable games on different manifolds with smoke.

2 PRELIMINARY RESULTS

So far, we have created an interactive fluid simulator system that works on a flat plane and on the surface of a torus and a 2D plane. Users are able to simulate smoke by dragging their left mouse button, and are able to input external velocity and move the smoke around by clicking and dragging their middle mouse button. Currently, we approximate an incompressible fluid using the Navier-Stokes equation. The user inputs are density and velocity, and we calculate the advection and divergence using those parameters.

Our simulator is written in JavaScript with the framework three.js, with user interaction from an HTML site. This simulator works at real-time speeds and contains a few different configurations. The simulator is configurable to either work on a 2-D plane (the screen), where the user can interact with the smoke, or on a torus, where the user can click within the torus to add smoke, and the smoke is confined within the torus. The camera view is from an external view of the torus. Additionally, it is configurable to either have the smoke wrap around or to not wrap around. In the plane, the wrap around option makes it so that once the smoke reaches the top of the screen, it will wrap back around to the bottom. In the torus, the wrap around option should be on so the smoke can flow throughout the torus. If it is off, there will be an absorbing point in the torus where smoke will fully dissipate from the simulation.

Authors' addresses: Alex Kassil, alexkassil@berkeley.edu, webmaster@marysville-ohio.com, UC Berkeley; Catherine Cang, catherinecang@berkeley.edu, UC Berkeley; Charles Sun, charlesjsun@berkeley.edu, UC Berkeley; Kevin Zhang, kevinzhang1@berkeley.edu, UC Berkeley.

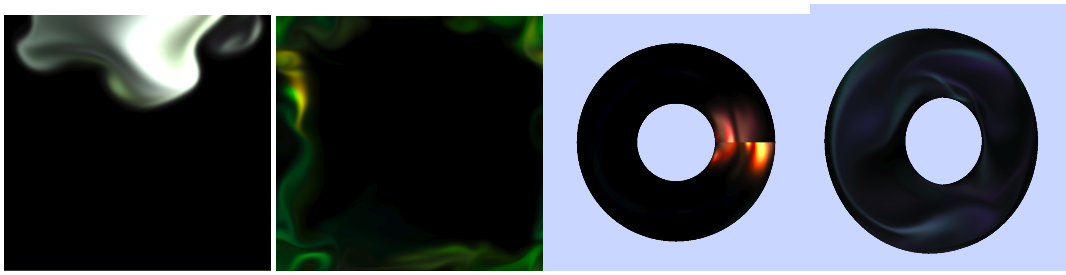

Fig. 1. From left to right: Smoke on Plane with no wrapping, Smoke on Plane with wrapping, Smoke on Torus with no wrap, Smoke on Torus with wrapping

2.1 General Progress and Future Plans

So far, relative to our original plan, we have been making adequate progress. For the milestone report, we planned to have a fluid simulator and have the fluid simulation map to at least one surface in 3D, which we accomplished with the torus.

In the future, we plan to add more manifolds, for example a Klein bottle, a phere, and hyperbolic space of a Poincaré disk model. We also plan on modeling collisions of the smoke with obstacles, and adding GUI options to switch between the different manifolds as well as control the physical parameters of the smoke.

Additionally, we have not fully simulated smoke properly and still need to add in a few adjustable parameters. In particular, we are missing curl and vorticity confinement, temperature and buoyancy, and boundary conditions for both the plane, the torus, and future manifolds. These conditions are important because as it stands, our smoke simulator is currently more of a fluid simulator, and it is missing a few behaviors such as smoke rising with the temperature.

2.2 References

http://developer.download.nvidia.com/books/HTML/gpugems/gpugems_ch38.html
https://rachelbhadra.github.io/smoke_simulator/index.html
https://github.com/mharrys/fluids-2d
https://threejs.org/docs/index.html