# An agent-based simulation of the wildfire

Milad Hakimshafaei
*Department of Computational Media*
*University of California Santa Cruz*
Santa Cruz, USA
mhakimsh@ucsc.edu

***Abstract*—This project simulates the probability of the wildfire spread at the UCSC campus with the agent behaviors. The Monte Carlo Physarum Machine (MCPM) computational model, which was previously applied to reconstruct the large-scale distribution of gas and dark matter in the Universe, known as the Cosmic Web, will be implemented to simulate the fire spread. The focus of this visualization is to make it aesthetically pleasing and more engaging for the mass audience with different backgrounds by implementing the computation and graphic capabilities of the visual effect programs.**

***Index Terms*—Agent behavior, MCPM, Simulation, Visual effects, Wildfire**

## I. Introduction

Visual effect programs are becoming increasingly influential in terms of computation and graphics nowadays. They can be used not only to produce cinematic effects but also for reliable visualization of the scientific data. Traditional data visualization is mostly used by experts to analyze data and define exciting areas, and communicate with other scientists by presenting the results in academic journals. The drawback of these visualizations is the necessity of preknowledge to understand the results; however, on the other hand, scientific visualization with visual effect programs will focus not only on the reliability of the visualization but also on the art and, specifically, aesthetics. In other words, it focuses on making it more understandable and engaging for the audience with less knowledge about the field. SideFX Houdini is a 3D procedural program suitable for modeling, animation, rendering, and lighting, which is significantly used for games, TV and film, motion graphics, and virtual reality. Compatibility with various programming languages such as Python and VEX makes these programs efficient for the simulation and scientific visualization; moreover, different rendering styles and effects result in a richer visual that this software provides. The computational power of Houdini both in geometry and shader level makes it an ideal choice for complex data visualization and simulation. In addition to that, it has a powerful capability for volume rendering. One of the highlights of this program is the ability to animate and processing millions of points in 3D space that makes it suitable for particle systems and agent behaviors. According to the mentioned potentials of the VFX programs, in this case, Houdini, this software has been defined as a powerful tool to investigate the agent behaviors for scientific visualization research. Houdini has its “Crowd” built-in system that is prepared to simulate the crowd and movement of people for visual effects. However, this research aims to define the agent systems from scratch based on the lower level of control over the particles that Houdini provides with its procedural and scripting capabilities. Agent systems have been used in many cases, from robotics to biology and computer engineering; however, the focus of this research is the Monte Carlo Physarum Machine (MCPM) computational model, which was previously applied to reconstruct the large-scale distribution of gas and dark matter in the Universe, known as the Cosmic Web [1] [2] [3]. This model has been used to visualize the astronomical data and investigate the aesthetics of the data with a bio-inspired organism called Slime mold. It has been implemented in a different platform as interactivity had a priority for the users. However, the objective of this research is to focus on the simulation and visualization with the Houdini program that provides different cinematic features for the visualization to be more engaging.

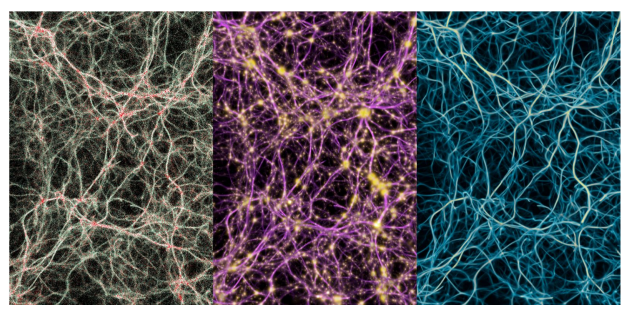

Fig. 1. Main components of MCPM [2]

Houdini has been widely used to improve the communication between the visualized data and audience; In this case, It has applied to the large Astronomical dataset, which required developing a new Python API that enhances importing and manipulating astrophysical data into the three-dimensional spaces [4] [5] [6].

The main idea of this project is to implement the MCPM model inside Houdini software, which provides powerful computational and graphical capabilities. The main focus is on the implementation of the agent behaviors through the procedural approaches of this program. However, the application case is the current concern in California which is the wildfire visual-

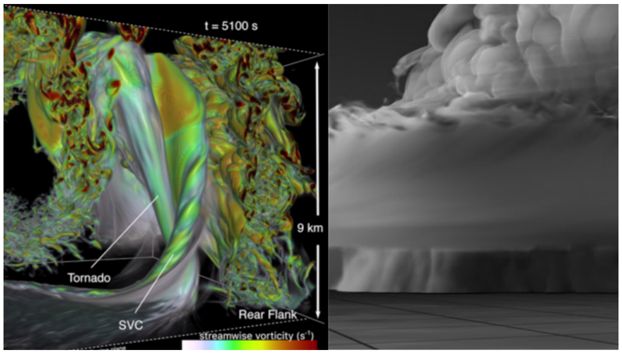


Fig. 2. Scientific vs cinematic visualization by Kalina Borkiewicz

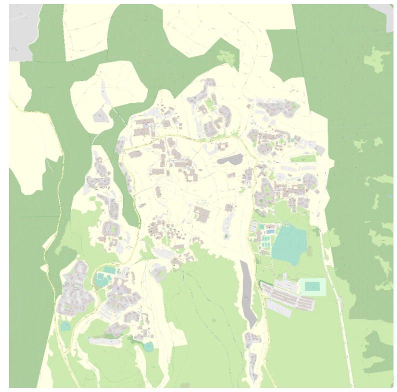

Fig. 4. open street map

ization and prediction. Since MCPM as an algorithmic growth system was a successful computational model to find the most probable path between points in the case of astronomical data, this research implements the same model to predict the growth and spread of the fire. Although many researchers have been working on this issue by proposing dynamic data-driven predictions [7] or focusing on novel visualization tools in 3D [**?**] and with new virtual reality technologies [**?**], this research concentrates on a novel agent-based approach by applying VFX features which have not been applied yet. The main objective is to define how this computational model can be successful in simulating the fire's spread.

## II. Simulation procedure

### *A. Data preparation*

This project deals with the different types of data, especially GIS data, which derive from multiple sources. Qgis software and OpenStreetMaps are the leading resources in order to extract map data. Since the simulation considers different map layers, such as roads, forests, and buildings, it isn't easy to extract all of these data from the same source because each resource can provide specific information. For instance, data for the buildings and roads can be received from OpenStreetMaps. However, forests and vegetations could be extracted from the latest google map source. These features must be derived separately and grouped in multiple layers to later integrate with the agent systems.

Fig. 3. google earth satellite map

### *B. MCPM implementation algorithm*

At this stage of the research, which can be considered the most important part of the implementation step, the Monte Carlo Physarum Machine model should be implemented in the Houdini program where the simulation runs. Since this model was developed within another platform and a different programming language, it is challenging to write the algorithm in a disparate program because it has to follow another procedure, which makes this step hard enough to require extra time to study and experiment.

### *C. Data integration and simulation*

The written algorithm in Houdini, which represents agent behaviors, must be applied to the prepared separated map layers in this research step. The most critical outline of this step is to examine how agents can react to each layer and whether the particles that represent fire can act differently to each layer, such as forests, roads, and buildings.

### *D. Rendering and visualization*

Depending on the required visualization style, the simulated particles can be rendered in different formats. They can contain a gradient of colors with density to represent the pressure of the fire or can visualize the actual fire based on the visual effects capabilities of the Houdini.

## References

[1] Elek, Oskar, et al. "Polyphorm: Structural Analysis of Cosmological Datasets via Interactive Physarum Polycephalum Visualization." IEEE Transactions on Visualization and Computer Graphics, vol. 27, no. 2, Feb. 2021, pp. 806–16.

[2] Elek, Oskar, et al. "Monte Carlo Physarum Machine: An Agent-Based Model for Reconstructing Complex 3D Transport Networks." The 2020 Conference on Artificial Life, MIT Press, 2020, pp. 263–65.

[3] Burchett, Joseph N., et al. "Revealing the Dark Threads of the Cosmic Web." The Astrophysical Journal, vol. 891, no. 2, Mar. 2020, p. L35.

[4] Naiman, J. P., et al. "Houdini for Astrophysical Visualization." Publications of the Astronomical Society of the Pacific, vol. 129, no. 975, May 2017, p. 058008.

[5] Borkiewicz, Kalina, et al. "Cinematic Visualization of Multiresolution Data: Ytini for Adaptive Mesh Refinement in Houdini." The Astronomical Journal, vol. 158, no. 1, June 2019, p.

[6] Borkiewicz, Kalina, et al. "Cinematic Scientific Visualization: The Art of Communicating Science." SIGGRAPH Asia 2019 Courses, ACM, 2019, pp. 1–313.

[7] Crawl, Daniel, et al. “Firemap: A Dynamic Data-Driven Predictive Wildfire Modeling and Visualization Environment.” Procedia Computer Science, vol. 108, 2017, pp. 2230–39.