# Volcanic Eruption Simulation

Guillaume Cordonnier, Yannis Kedadry

**Abstract**

When thinking about physical phenomenon, lava flows often come first in mind being both visually impressive and a real natural threat for its surrounding areas. Although being popular, fast and physically accurate eruption simulations can be tricky because of the unusual laws the flow follows and the hard solver needed to resolve them. This paper present a new Lagrangian technique to efficiently simulate lava flows using shallow water equations.

## Introduction

Being able to simulate lava flows in a physically accurate manner can lead to both visually stunning results and a useful prediction tool to prevent causalities of real eruptions. During eruptions, the flow of lava can be seen as a pretty thin layer of fluid; this observation leads us to the idea of using shallow water equations[1] to simulate the flow.

## State of the art

Simulation of fluids in 3D are well studied in the literature either for Lagrangian solvers or Eulerian methods. Usually these simulations come at high cost, requiring millions of particles to reproduce small scale effects. A way to reduce this cost, as used in *insert citation*, is to solve only two components in the 3D space hence reducing the number of operations and the internal size of the simulation.

- SWE adding horizontal velocity field [1]

- SWE for non-uniform terrain [2] (but uniform particles, i.e. same height everywhere)

## Method

### First approach: Shallow-water

- 2D sph -> sphere over height map
- Each particle represent a column of lava of a certain height

- Each column same velocity + parallel to the terrain (for now, later add horizontal velocity ?)

$$\frac{Dh}{Dt} = -h\nabla.u \quad (1)$$

$$u = -\frac{g}{k}\nabla H \quad (2)$$

$H$: surface -> $H(x,z) = y$ $h$: height of a particle representing the column of lava, $h$ is calculated using the formula: $h\rho_0 = \rho$, $\rho_0 = 2500kg.m^{-3}$ (lava rest density [3]), $\rho$: the current density of the particle

difficulties:
- neighbourhood
-> first approximation using basic grid search:
-> 2D grid (y = 0) ($cell_{width} = 4 * W_{radius}$), get the particles within the cell in which is the current particle and then check if $dist(p_i, p_j) < W_{radius}$
- multiple branching [4]

terrain gradient:
- euler method

$$z = H(x,y)\frac{\partial H}{\partial x}(x,y) = \frac{H(x+1,y) - H(x,y)}{x+1-x} = H(x+1,y) - z\frac{\partial H}{\partial y}(x,y) = \frac{H(x,y+1) - H(x,y)}{y+1-y} = H(x,y+1)$$

- sobel operator[5]

$\nabla H(x,y) = \sqrt{(\frac{\partial H}{\partial x}(x,y))^2 + (\frac{\partial H}{\partial y}(x,y))^2}$

### Better approach: Stokes problem

### Rendering

- rendu sur texture

## Results

## Conclusion

## References


[1] Barbara Solenthaler, Peter Bucher, Nuttapong Chentanez, Matthias Müller, and Markus H. Gross. Sph based shallow water simulation. In *Workshop on Virtual Reality Interactions and Physical Simulations*, 2011.

[2] Miguel X. Rodríguez-Paz and Javier Bonet. A corrected smooth particle hydrodynamics formulation of the shallow-water equations. *Computers & Structures*, 83:1396–1410, 2005.

[3] Ross Griffiths. The dynamics of lava flows. *Annual Review of Fluid Mechanics*, 32:477–518, 01 2000.

[4] Kao Hua Chang, Tsang-Jung Chang, and Yen-Ming Chiang. A novel sph-swes approach for modeling subcritical and supercritical flows at open channel junctions. *Journal of Hydro-environment Research*, 13:76–88, 2016.

[5] Victor Bogdan, Cosmin Bonchiş, and Ciprian Orhei. Custom extended sobel filters, 2019.

[6] Jonathan Gagnon, Julián Guzman, Valentin Vervondel, François Dagenais, David Mould, and Eric Paquette. Distribution update of deformable patches for texture synthesis on the free surface of fluids. *Computer Graphics Forum*, 38, 2019.

[7] Maud Lastic, Damien Rohmer, Guillaume Cordonnier, Claude Jaupart, Fabrice Neyret, and Marie-Paule Cani. Interactive simulation of plume and pyroclastic volcanic ejections. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 5(1):1–15, May 2022.

[8] Alexey Stomakhin, Craig A. Schroeder, Chenfanfu Jiang, Lawrence Chai, Joseph Teran, and Andrew Selle. Augmented mpm for phase-change and varied materials. *ACM Transactions on Graphics (TOG)*, 33:1 – 11, 2014.

[9] Shenfan Zhang, Fanlong Kong, Chen Li, Changbo Wang, and Hong Qin. Hybrid modeling of multiphysical processes for particle-based volcano animation. *Computer Animation and Virtual Worlds*, 28, 2017.

[10] Noé Bernabeu, Pierre Saramito, and Claude Smutek. Modelling lava flow advance using a shallow-depth approximation for three-dimensional cooling of viscoplastic flows. *Geological Society, London, Special Publications*, 426, 03 2016.

[11] Mathieu Desbrun and Marie-Paule Cani. Smoothed Particles: A new paradigm for animating highly deformable bodies. In Ronan Boulic and Gerard Hegron, editors, *Eurographics Workshop on Computer Animation and Simulation (EGCAS)*, pages 61–76, Poitiers, France, August 1996. Springer-Verlag. Published under the name Marie-Paule Gascuel.

[12] Jos Stam. Real-time fluid dynamics for games. 2003.

[13] Andreas Kolb, Lutz Latta, and Christof Rezk-Salama. Hardware-based simulation and collision detection for large particle systems. *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware*, 2004.

[14] Jos Stam. Stable fluids. In *International Conference on Computer Graphics and Interactive Techniques*, 1999.

[15] Dan Stora, Pierre-Olivier Agliati, Marie-Paule Cani, Fabrice Neyret, and Jean-Dominique Gascuel. Animating lava flows. *Graphics Interface*, 12 2000.

[16] Markus Ihmsen, Jens Cornelis, Barbara Solenthaler, Christopher Horvath, and Matthias Teschner. Implicit incompressible sph. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):426–435, 2014.

[17] Jan Bender and Dan Koschier. Divergence-free smoothed particle hydrodynamics. 08 2015.

[18] B. Solenthaler and R. Pajarola. Predictive-corrective incompressible sph. *ACM Trans. Graph.*, 28(3), jul 2009.

[19] Jonathan Gagnon, François Dagenais, and Eric Paquette. Dynamic lapped texture for fluid simulations. *The Visual Computer*, 32:901 – 909, 2016.