

Revisiting vorticity : pushing fluid solvers to the next level

Bridson, Robert
Autodesk

<https://hdl.handle.net/2324/1546879>

出版情報 : MI lecture note series. 64, pp.51-53, 2015-09-18. 九州大学マス・フォア・インダストリ
研究所
バージョン :
権利関係 :



Revisiting vorticity: pushing fluid solvers to the next level

Robert Bridson*
Autodesk

Abstract Although some of the earliest work in fluid simulation for computer graphics exploited vorticity (e.g. Yaeger et al.’s work on Jupiter for the film 2010 [4]), by and large practical work over the last decade or two has focused on velocity-pressure formulations. This talk looks at why vortex methods are worth coming back to, the troubles that have steered practitioners away from them, and how we might overcome them.

Keywords: Fluid simulation, physics-based animation, vortex methods

1 Background

From vast oceans to air puffing dust around a footfall, fluids have become a staple of visual effects work, and direct simulation of the underlying physics has emerged as the most attractive approach to generate detailed and natural-looking fluid animation. Current research in graphics on this field can loosely be divided into two parts: work extending the range of phenomena that can be achieved (such as new materials and new means of artist control), and work improving the quality and/or efficiency of standard solves. We focus on the latter in this talk, and mostly on improving smoke simulation in particular.

Measuring the “quality” of a simulation is in of itself a tough, open problem. Accurately solving chaotic, high Reynolds number flow is essentially infeasible, so errors are a given: the question is which sorts of error are acceptable? Backed by experience, I will argue that accurately tracking vorticity and vortex structures is a good goal for graphics.

Vorticity $\vec{\omega}$ is the curl of velocity \vec{u} ,

$$\vec{\omega} = \nabla \times \vec{u}$$

which measures locally how the flow is rotating (as opposed to shearing). In rigidly rotation regions, vorticity is exactly twice the angular velocity of the region. Velocity can be reconstructed from vorticity via the Biot-Savart law, in a sense integrating to undo the differentiation, as long as boundaries are also known: vorticity can serve as the fundamental state of a solver just as well as velocity.

Why vorticity might be so important is best illustrated mathematically with the equations of motion for a constant density fluid in two dimensions with negligible viscosity. Recall the material derivative $D/Dt = \partial/\partial t + \vec{u} \cdot \nabla$ tracks how a quantity attached to the fluid changes in time while it flows

*robert.bridson@autodesk.com

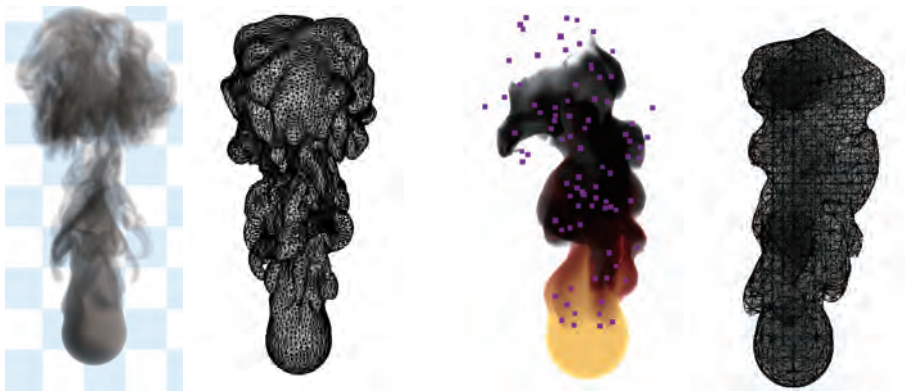


Figure 1: Left: Brochu et al. tracked vorticity just on a triangle mesh dividing space between smoky and clear air, producing highly detailed results [1]. Right: Goldade et al. demonstrated a real-time smoke simulator using a very small number of vortex particles (in purple) to nonetheless capture detailed and lively fluid motion [3].

with the fluid. The usual velocity-pressures equations are

$$\begin{aligned}\frac{D\vec{u}}{Dt} + \frac{1}{\rho}\nabla p &= 0, \\ \nabla \cdot \vec{u} &= 0,\end{aligned}$$

which show velocity is always being changed by the pressure gradient, while pressure is computed so as to keep the velocity incompressible. On the other hand, the vorticity equation which describes exactly the same motion is this simple:

$$\frac{D\vec{\omega}}{Dt} = 0.$$

In other words, vorticity just moves with the flow without otherwise changing, which is both striking mathematically and far easier to numerically solve.

Even better, in many flow scenarios vorticity is highly concentrated in small structured regions, and basically is zero throughout most of the domain. This *sparsity* in the representation can also be exploited numerically by tracking vorticity with a small and sparse set of particles or other Lagrangian elements. Figure 1 shows recent smoke results using relatively lightweight vortex triangle meshes and particles.

The fly in the ointment, which I believe has steered people away from vortex methods in the past, is the reconstruction of velocity from vorticity and boundaries. Without boundaries, finding the velocity at a single point in space with the Biot-Savart law requires integrating vorticity with a kernel over the entire fluid domain; with boundaries additional integrals or PDEs are required to calculate their effect. Even just formulating solid and free surface boundary conditions in terms of vorticity can be very tricky indeed. On top of all this, in three dimensions there is an additional term in the vorticity equation, for “vortex stretching,” which can be difficult to stably approximate.

Velocity reconstruction needn’t be so difficult, however. While a purely Lagrangian approach using the Biot-Savart law may require complex algorithms such as the Fast Multipole Method to scale well, great results can be obtained with the Vortex-in-Cell (VIC) method [2], where vorticity is splatted to a background grid and velocity is reconstructed there via solving the Poisson problem. Even greater detail can be achieved with the Particle-Particle Particle-Mesh (PPPM) approach, without too much more complication for graphics [5].



Figure 2: Left: frame from a velocity-pressure smoke simulation using FLIP. Right: the same with the IVOCK correction to more accurately solve the vorticity equation. [6]

While there is still much more to say about boundary conditions for vortex methods, recent work by Zhang et al. shows that it's possible to augment a traditional velocity-pressure solver with a correction to track vorticity like a vortex method, while handling the boundary conditions directly with velocity and pressure [6]. Figure 2 demonstrates the improvement in quality this correction gives with only a small overhead.

There is still much left to do, especially in transferring these techniques to water simulations; I hope this talk will provide a useful view forward on next steps in research.

References

- [1] BROCHU, T., KEELER, T., AND BRIDSON, R. Linear-time smoke animation with vortex sheet meshes. In *Proc. ACM SIGGRAPH/Eurographics Symp. Comp. Anim.* (2012), SCA '12, pp. 87–95.
- [2] CHRISTIANSEN, I. P. Numerical simulation of hydrodynamics by the method of point vortices. *J. Comp. Phys.* 13, 3 (1973), 363–379.
- [3] GOLDADE, R., KEELER, T., AND BRIDSON, R. Real-time vorticity-based fire and smoke simulation using marching tetrahedra. In *ACM SIGGRAPH/Eurographics Symp. Comp. Anim. (Posters)* (2013).
- [4] YAEGER, L., UPSON, C., AND MYERS, R. Combining physical and visual simulation—creation of the planet jupiter for the film 2010. In *Proc. ACM SIGGRAPH* (1986), pp. 85–93.
- [5] ZHANG, X., AND BRIDSON, R. A ppm fast summation method for fluids and beyond. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* 33, 6 (2014), 206:1–206:11.
- [6] ZHANG, X., BRIDSON, R., AND GREIF, C. Restoring the missing vorticity in advection-projection fluid solvers. *ACM Trans. Graph. (Proc. SIGGRAPH)* (2015), to appear.