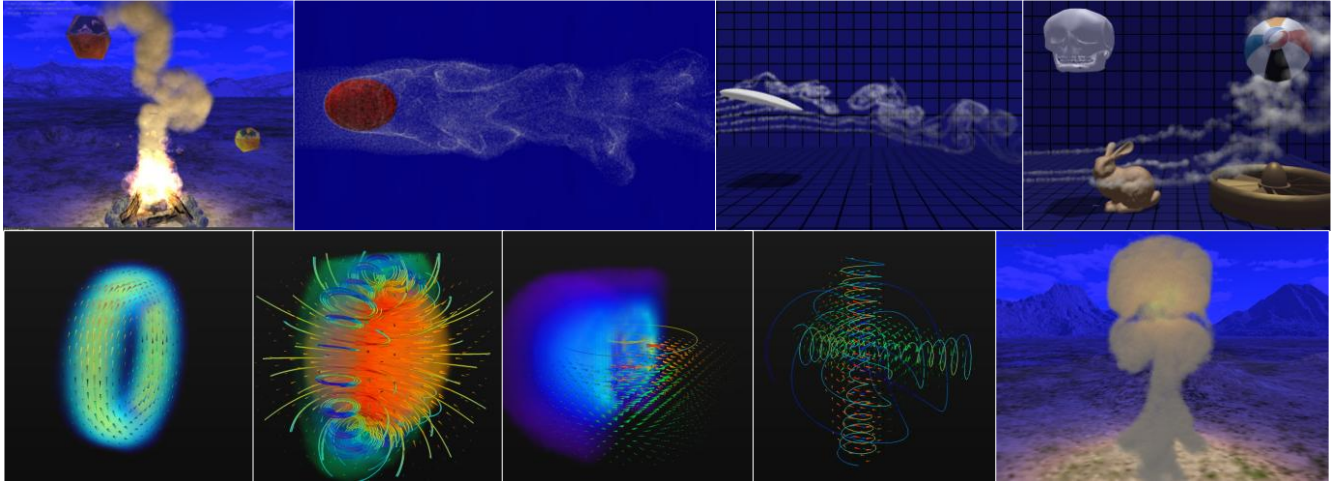


# Fluid-Body Simulations using Vortex Particle Operations

Michael J. Gourlay

University of Central Florida / Electronic Arts

mgourlay@fiea.ucf.edu



## 1 Introduction

Modeling continuous media such as fluids remains an elusive goal for interactive simulations. Fluids are particularly challenging because of the complexity imparted by the non-linear equations of motion, and the difficulty in creating stable simulations that retain spatial detail.

Content creators for video games and other interactive simulations are familiar with using particle systems to model fluid-like systems such as smoke, but those systems traditionally lack the complexity and response to passing bodies that real fluids have. We aim to enhance particle systems so that effects authors can continue to use familiar systems, and provide a better approximation of fluid-like motion which has adequate performance for use in interactive, real-time simulations.

Fluids move chaotically and chaos has appeal in visual effects. Particles that move under the influence of vortex systems with 3 or more vortices behave chaotically. Vortex simulations introduce appealing fluid-like chaotic motion into particle systems with minimal cost. We present an algorithm that exploits vortex particle methods to introduce fluid-like chaotic motion into traditional particle systems, yielding a fast, steerable and interactive fluid simulation.

### 1.1 Parallelized vorton methods for fluid motion

Our technique, the code for which we provide publically, treats fluid simulation as operations incorporated into a canonical library of particle operations typically available in existing commercial particle systems. We therefore discretize vortices as particles (not filaments or sheets).

Using the map-reduce paradigm and Intel Threading Building Blocks, we implemented scalably parallelized versions of several solvers, including integral (direct summation, treecode and a novel fast monopole) and differential (Jacobi, red-black Gauss-Seidel with successive over-relaxation and multi-grid) methods.

For fluid-body interactions we introduce a novel amortized localized approach to obtain a global solution.

## 2 Methods

Reusing and enhancing an existing particle system provides an economy of effort and familiarity with existing tools – important features in commercial game development. The simulation includes several phases, each implemented as particle operations:

1. Compute velocity from vorticity.
2. Compute stretching and tilting, viscous diffusion etc.
3. Apply other, canonical, particle operations.
4. Advect particles (vorton and tracer).
5. Compute fluid-body interactions.

While several of these steps appear in other fluid simulations, our approach has two key differences: Each of these steps occurs as a particle operation, reusing an existing particle system simulation, so tools already in-place can be reused, retaining the workflow familiar to effects authors. Also, step 3 allows the fluid simulation operations to cooperate with existing particle operations, such as wind and forces, permitting effects authors to steer the flow beyond the restraints of accurate physical simulation. Furthermore, as material quantities, vortex particle play multiple roles, including sources of velocity, carriers of momentum and markers for visualization. So although other techniques exist that combine fluid simulations with particle systems, this treatment provides additional economy due to using particles as the impetus for fluid motion, in addition to their traditional roles in visual effects.

## 3 Results

Results include practical interactive visual effects and canonical flow scenarios that demonstrate the viability of the algorithm. The user can control the motion of rigid bodies and move them through fluid and the fluid pushes and spins objects, all in real time. Validation studies indicate the relative accuracy of the six different solvers.

Test cases simulate and render at least 60 frames per second even when using only a single core, with nearly linear scaling up to over a dozen cores. Comparisons with other vortex methods reveal our technique runs faster, with lower time complexity and better scaling with the number of particles.