A GPU-Based Approach for Real-Time Haptic Rendering of 3D Fluids

Meng Yang*
University of Pennsylvania

Jingwan Lu
Hong Kong University of Science and Technology

Zehua Zhou
University of Pennsylvania

Alla Safonova
University of Pennsylvania

Katherine J. Kuchenbecker
University of Pennsylvania

1 Introduction

Real-time haptic rendering of three-dimensional fluid flow will improve the interactivity and realism of video games and surgical simulators, but it remains a challenging undertaking due to its high computational cost. In this work we propose an innovative GPU-based approach that enables real-time haptic rendering of high-resolution 3D Navier-Stokes fluids. We show that moving the vast majority of the computation to the GPU allows for the simulation of touchable fluids at resolutions and frame rates that are significantly higher than any other recent real-time methods [Baxter and Lin 2004; Mora and Lee 2008; Dobashi et al. 2006].

2 System Overview

Our simulation lets the user move a virtual object inside a 3D fluid flow by maneuvering the handle of a commercial haptic device, as shown in Figure 1. Throughout the interaction, the user can feel the fluid's response as forces from the device handle and can simultaneously see the fluid's overall motion on the screen.

Texture Preparation The CPU manages communication with the haptic device, reading its position at a rate of 1000 Hz and steadily passing this three-element vector to the GPU. When ready, the GPU takes the most recent position measurement and prepares a 3D texture that marks the cells in the fluid grid as *inside*, *on the boundary of*, *surrounding*, or *outside* the virtual object. This approach is based on that of [Crane et al. 2007], modified to include the additional category of *surrounding* to facilitate the haptic force computation discussed below. A 2D sample of our cell categorizaton algorithm is shown in the inset of Figure 1.

Fluid Simulation The GPU then updates the voxelized fluid simulation using a semi-Lagrangian approach to implicitly solving the Navier-Stokes equation, largely following the GPU-based implementation of [Crane et al. 2007]. This process includes density and velocity advection, pressure computation, vorticity confinement, and displacement of fluid by the manipulated object.

Force Computation After updating the simulation, the GPU must estimate the force of the interaction between the virtual object and the fluid. This step is performed by using the prepared 3D texture to locate the *surrounding* cells: the interaction force is the vector sum of the *surrounding* cell pressures acting on all of the exposed faces of the adjacent *boundary* cells. Though it represents only a small portion of the computation required in the simulation, parallel reduction on the GPU enables this force computation to be $O(\log N)$, where N is the total number of cells in the simulation, while it would be $O(N)$ on the CPU.

Force Feedback Each time the GPU completes the three steps described above, the CPU obtains a copy of the newly computed interaction force vector. Because forces updated at rates below about 500 Hz feel rough and unnatural to the user, the CPU uses a discrete-time low-pass filter on the raw force values to send smoothed haptic feedback commands to the haptic interface at a rate of 1000 Hz, as suggested by [Baxter and Lin 2004]. The bandwidth of this low-pass filter can be adjusted based on the current frame rate of the GPU.

*e-mail: mengyang@seas.upenn.edu

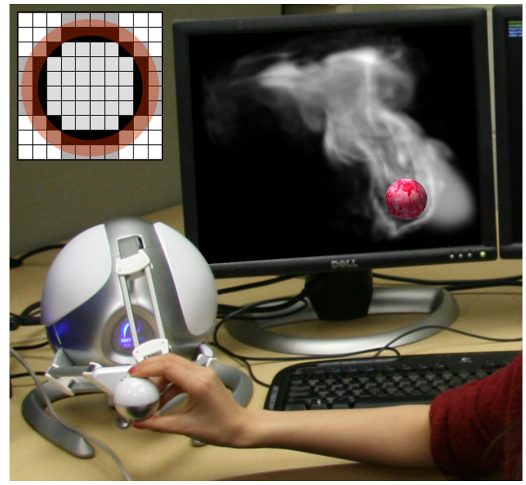

Figure 1: *The user grasps the handle of the Novint Falcon to move the spherical object and feel the forces it experiences in the fluid. The inset shows a sample labeling of cells in the sphere's vicinity, where light gray, black, dark gray, and white represent inside, boundary, surrounding, and outside cells respectively.*

Table 1: *Performance Comparison*

Method	Max. Fluid Size	Optimal FPS
Baxter and Lin's 2D	128×128	$40 \sim 70$
Mora and Lee's 2D$^+$	15×15 $(\times 15)$	30
Our GPU-Based 3D	$100 \times 100 \times 70$	$30 \sim 75$

3 Results and Conclusion

As detailed in Table 1, our GPU-based approach allows real-time haptic rendering of 3D fluids with physical accuracy at higher resolutions and/or higher rates than previous approaches. Our future work will seek to further optimize the GPU-based computations and apply this multimodal simulation method to a wide variety of fluids and virtual objects.

References

BAXTER, W., AND LIN, M. C. 2004. Haptic interaction with fluid media. In *GI '04: Proceedings of Graphics Interface 2004*.

CRANE, K., LLAMAS, I., AND TARIQ, S. 2007. Real-time simulation and rendering of 3D fluids. In *GPU Gems 3*.

DOBASHI, Y., SATO, M., HASEGAWA, S., YAMAMOTO, T., KATO, M., AND NISHITA, T. 2006. A fluid resistance map method for real-time haptic interaction with fluids. In *VRST '06*.

MORA, J., AND LEE, W.-S. 2008. Real-time 3D fluid interaction with a haptic user interface. In *Proceedings of IEEE Symposium on 3D User Interfaces 2008*.