An Area-Preserving Model For GPU Two-Dimensional Fluid Simulation Using Particle Systems

Yu Wang

Prepared for CMSC 601

May 9, 2011

Department of Computer Science and Electrical Engineering
University of Maryland, Baltimore County

Contents

Introduction

Fluid Simulation

Background and Related Work

- Solve the Navier-Stokes Equation

$$\rho\left(\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{dy}+w\frac{\partial u}{\partial z}\right)=$$

$$\rho g_x-\frac{\partial p}{\partial x}+\frac{\partial}{\partial x}\left[2\mu\frac{\partial u}{\partial x}+\lambda\nabla\cdot\mathbf{V}\right]+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial w}{\partial x}+\frac{\partial u}{\partial z}\right)\right]$$

$$\rho\left(\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{dy}+w\frac{\partial v}{\partial z}\right)=$$

$$\rho g_y-\frac{\partial p}{\partial y}+\frac{\partial}{\partial y}\left[2\mu\frac{\partial v}{\partial y}+\lambda\nabla\cdot\mathbf{V}\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right]+\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\right]$$

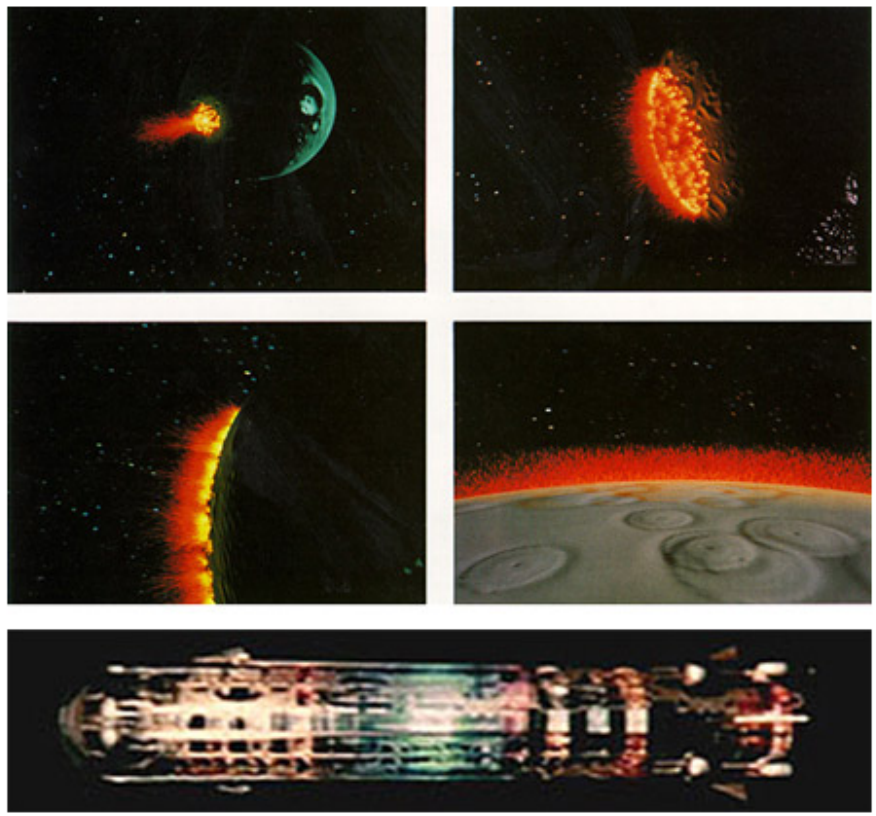
$$\rho\left(\frac{\partial w}{\partial t}+u\frac{\partial w}{\partial x}+v\frac{\partial w}{dy}+w\frac{\partial w}{\partial z}\right)=$$

$$\rho g_z-\frac{\partial p}{\partial z}+\frac{\partial}{\partial z}\left[2\mu\frac{\partial w}{\partial z}+\lambda\nabla\cdot\mathbf{V}\right]+\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial w}{\partial x}+\frac{\partial u}{\partial z}\right)\right]+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right]$$

Background and Related Work

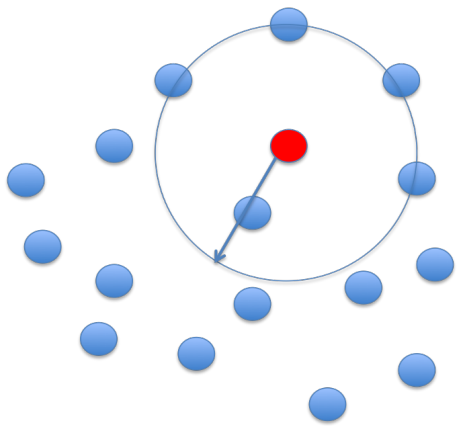
- Particle Systems

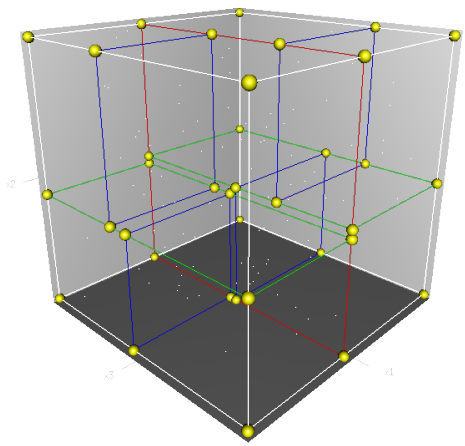
1. generated into space with initial states: velocity, color, etc
2. update every frame; life span



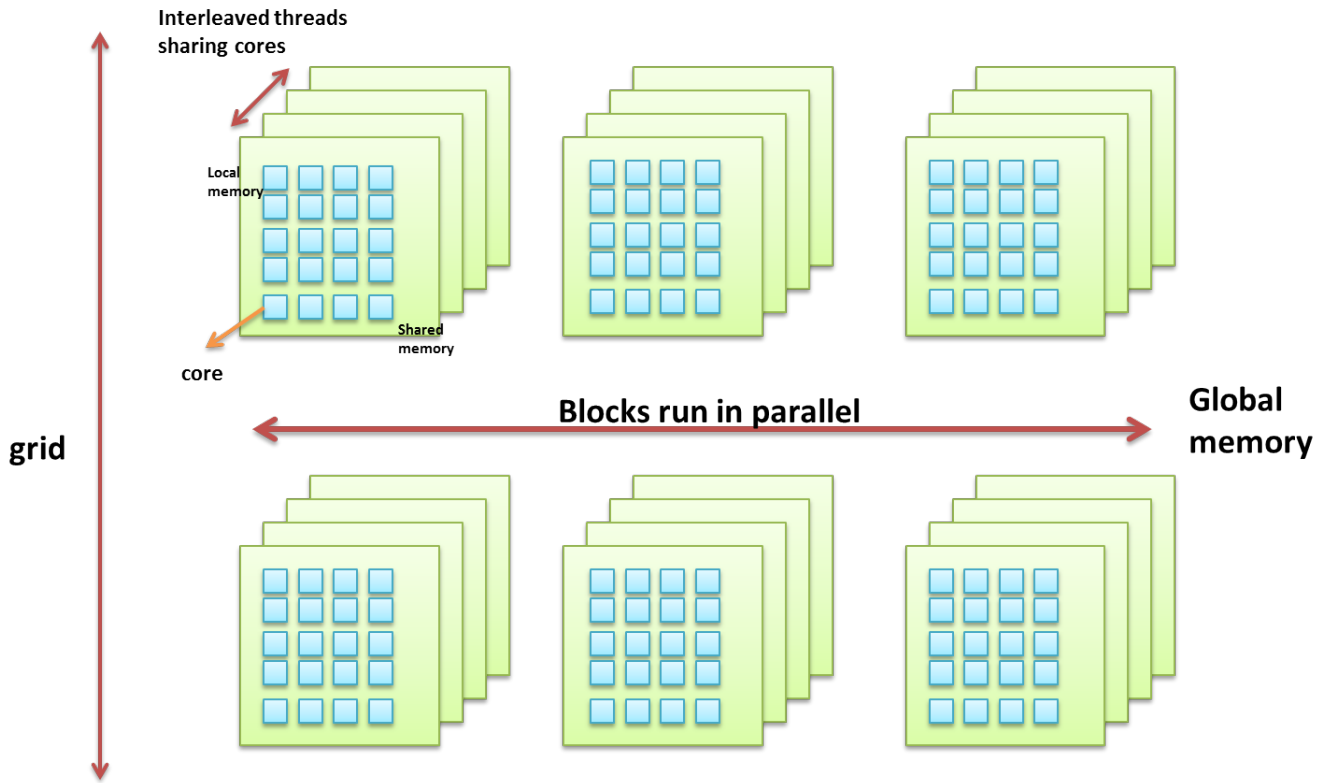
K-Nearest Neighbor Search Algorithm

- Nearest Neighbor Search: Given a set R of reference points in space S and a query point $q \in S$, find the closest point in R to q
- K-Nearest Neighbor Search: locating the closest k neighbors of query point q in S based on their Euclidean distances





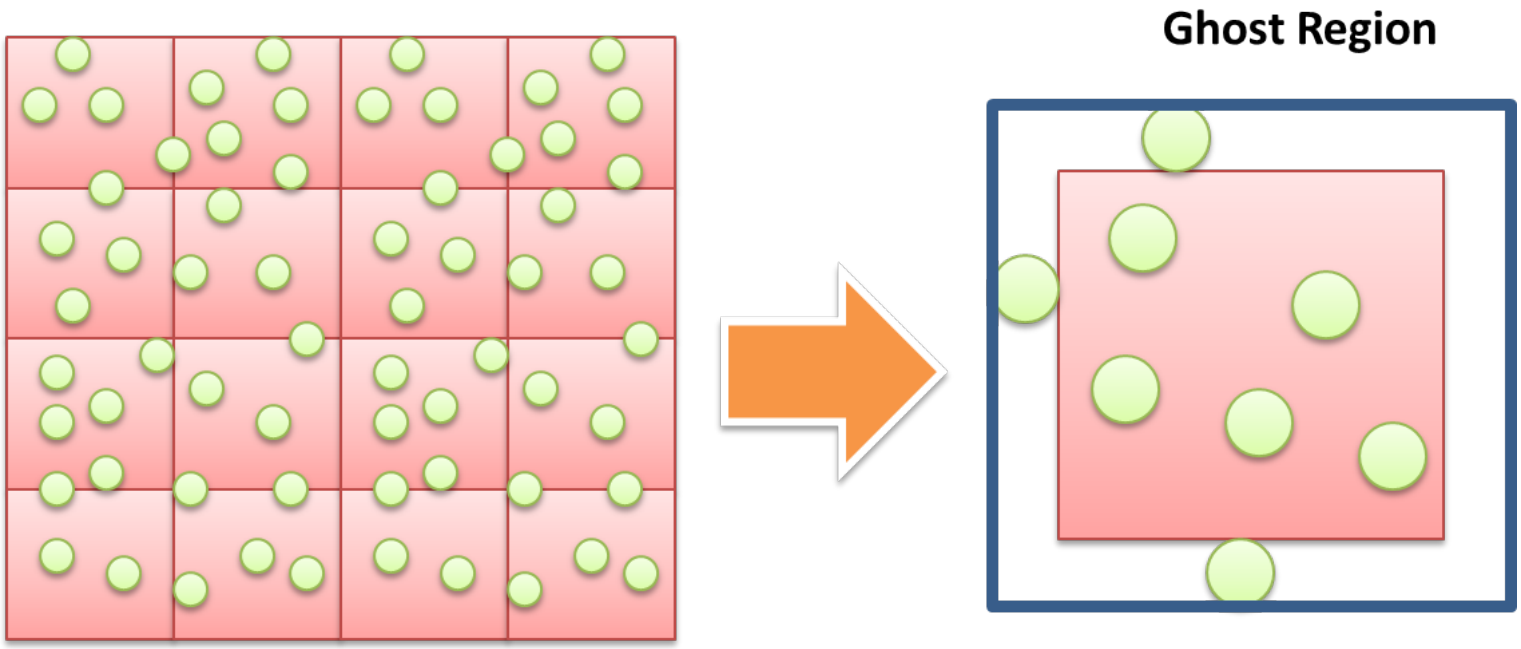
Modern Graphics Processing Units (GPU)



Memory	Size	Speed	Sharing Scope
register	small	Very fast	thread
local	small	Fast	thread
shared	small	Fast	block
global	large	Slow	grid
constant	small	Fast	grid(read only)

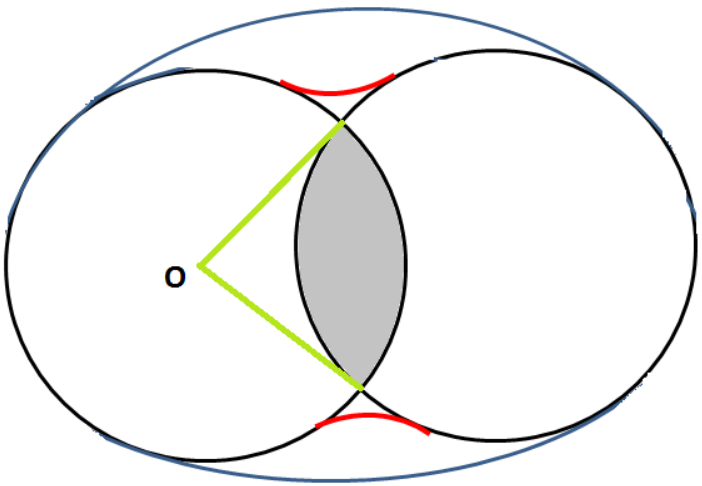
Proposed Model

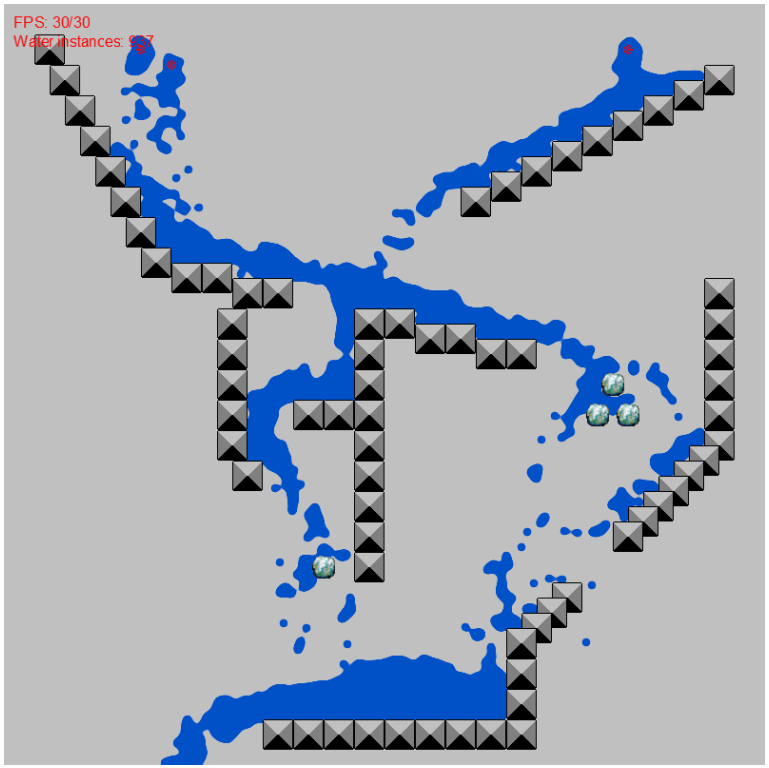
Position Update



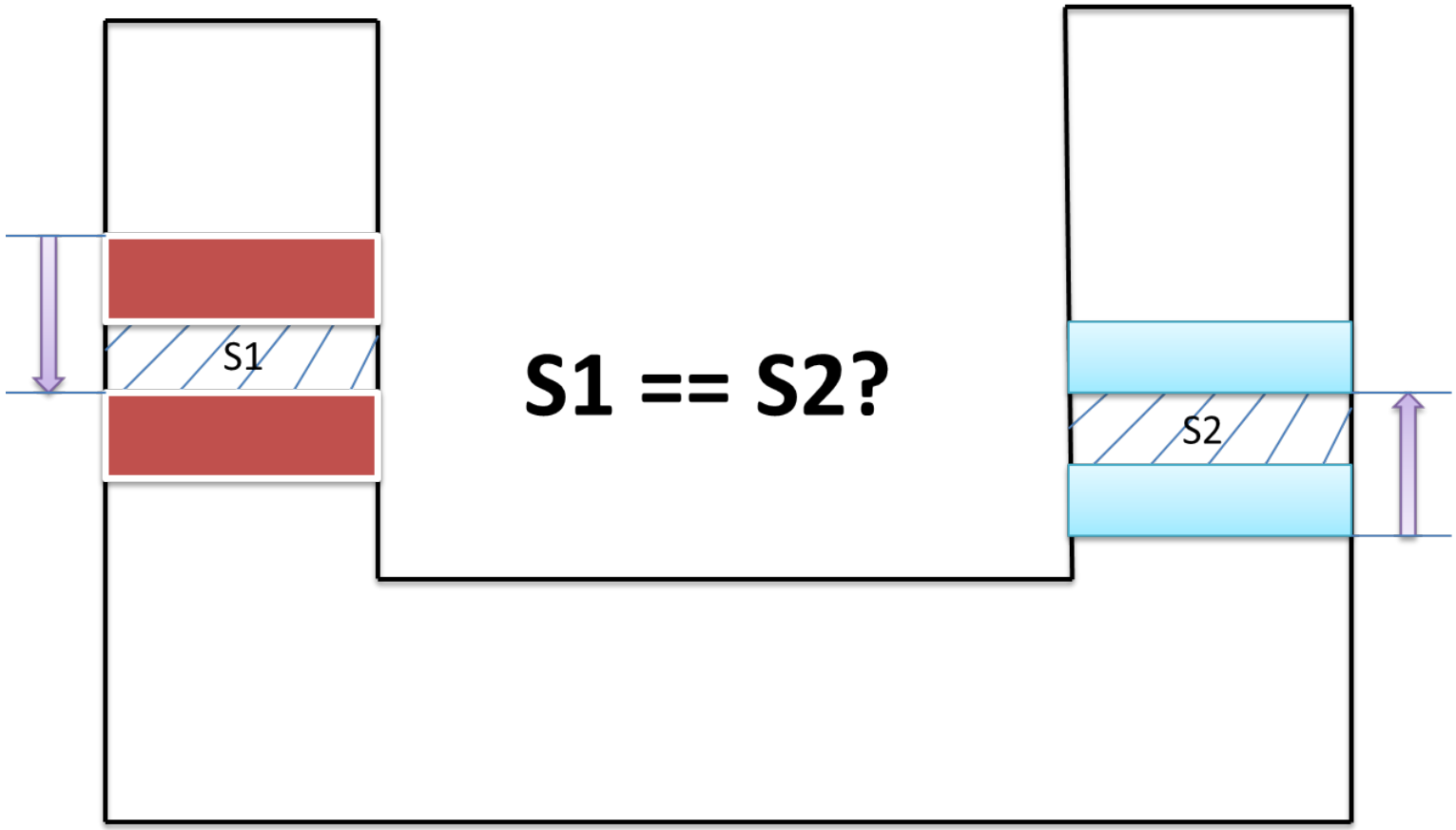
Proposed Model

Area Preserving





Evaluation



Challenge

- Data structure of the particle states
- Shared Memory; Global Memory; Host Memory
- Deform multiple intersecting particles