GPU-accelerated Fluid Dynamics

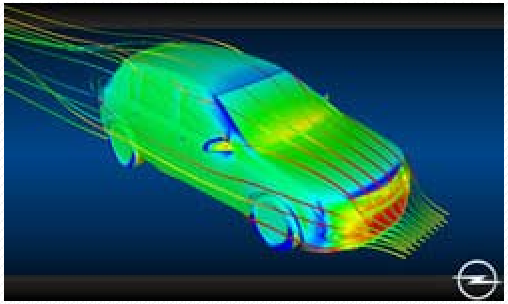
by
Dirk Hekhuis

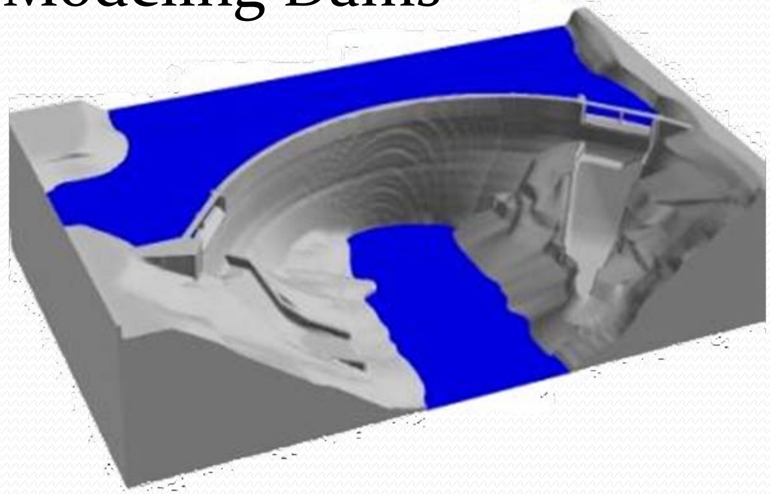
Advisors
Dr. Greg Wolffe
Dr. Christian Trefftz

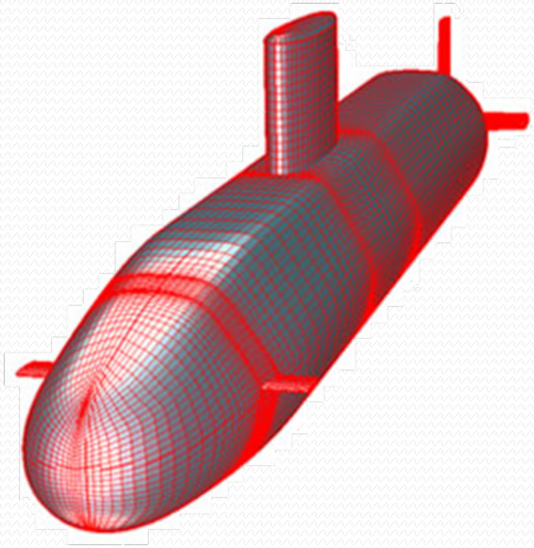
Computational Fluid Dynamics

Applications

- Computational Fluid Dynamics have many applications
  - Automotive Aerodynamics
  - Designing HVAC Systems
  - Water Flow Around Submarines
  - Modeling Dams





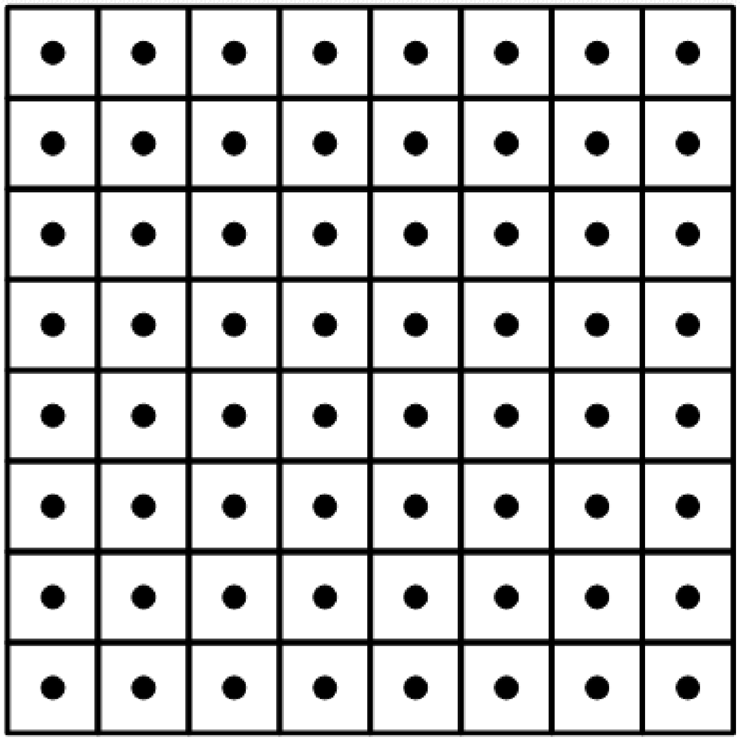


The Physics of Fluids

Navier-Stokes equations for incompressible flow

- Equation for velocity in a compact vector notation
  - $\frac{\partial u}{\partial t} = -(u \cdot \nabla)u + v\nabla^2 \mathrm{u} + \mathrm{f}$
- Equation for density moving through the velocity field
  - $\frac{\partial \rho}{\partial t} = -(u \cdot \nabla)\rho + k\nabla^2 \rho + S$

Fluid Representation



Density and velocity are defined at the center of each cell

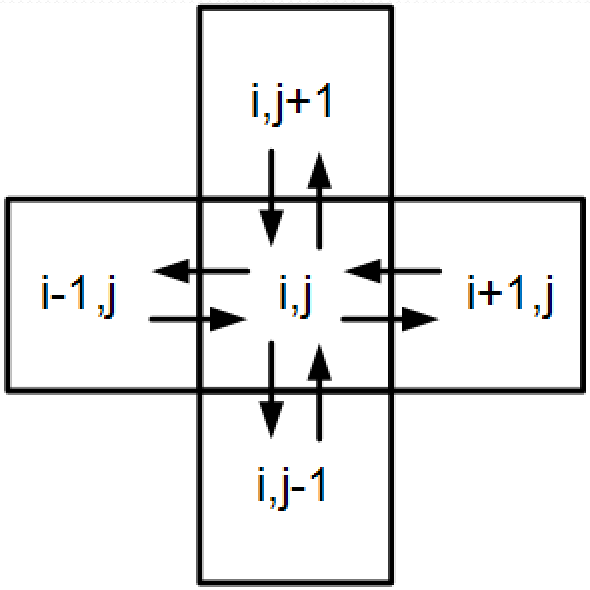
Implementing Navier-Stokes

- External Forces
- Diffusion
- Advection
- Projection

External Forces

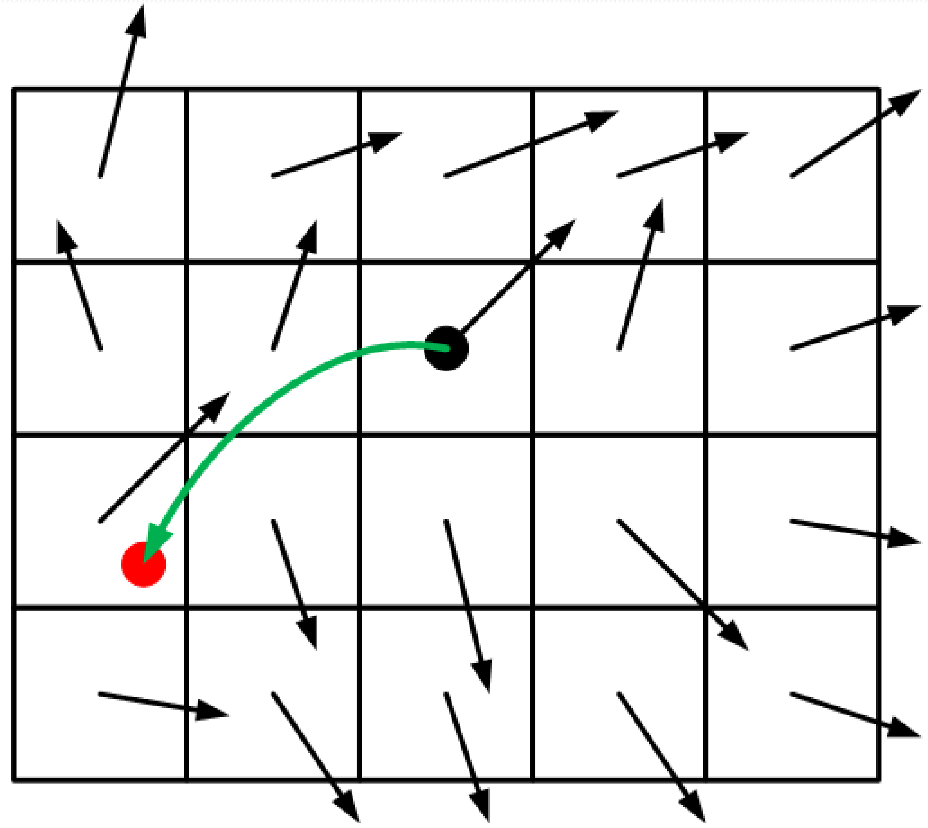
- External forces applied to the fluid can be either ***local forces*** or ***body forces***
- Local forces are applied to a specific region of the fluid – for example the force of a fan blowing air
- Body forces are forces that apply evenly to the entire fluid, like gravity

Diffusion



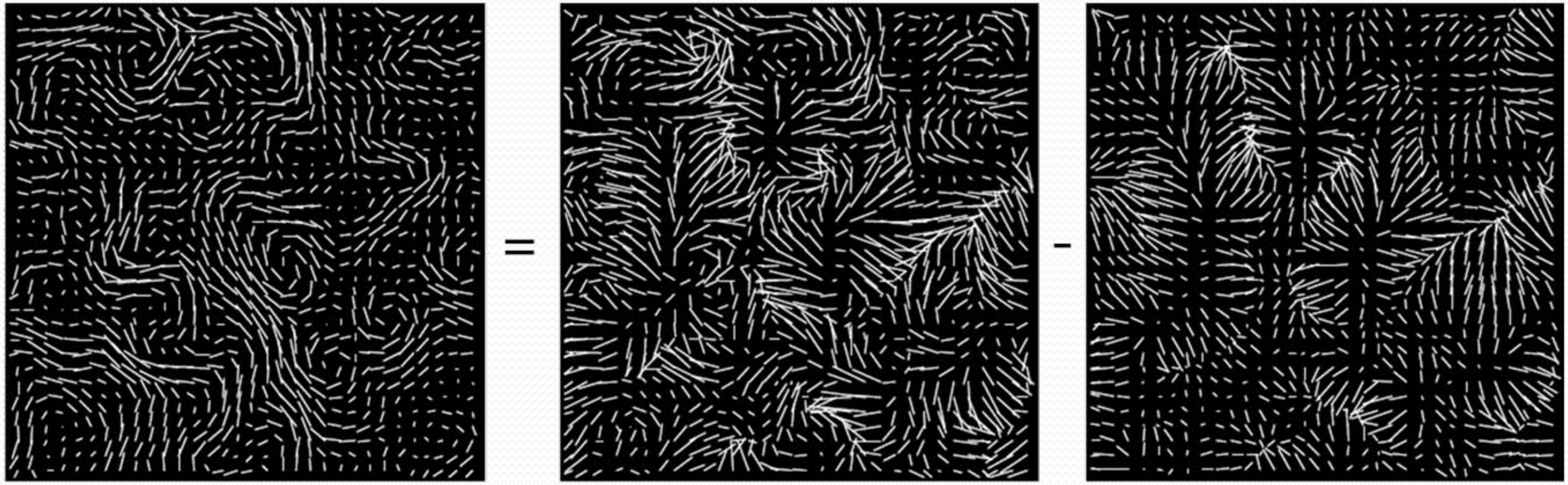
Through diffusion each cell exchanges density with its direct neighbors

Advection



Trace the particle backwards in time

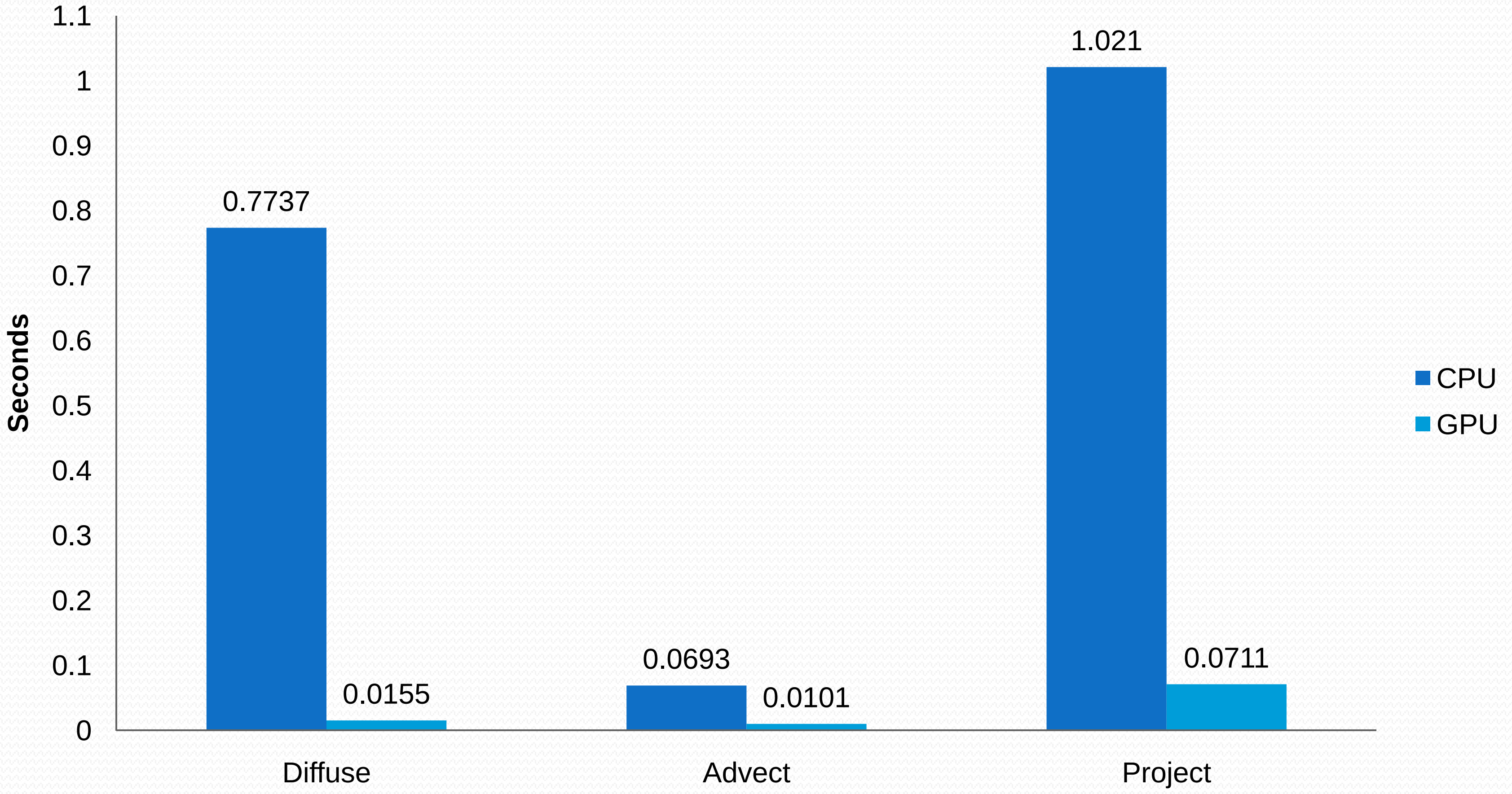
Projection



Mass conserving = our field - gradient

GPU Programming with CUDA

Why use CUDA?



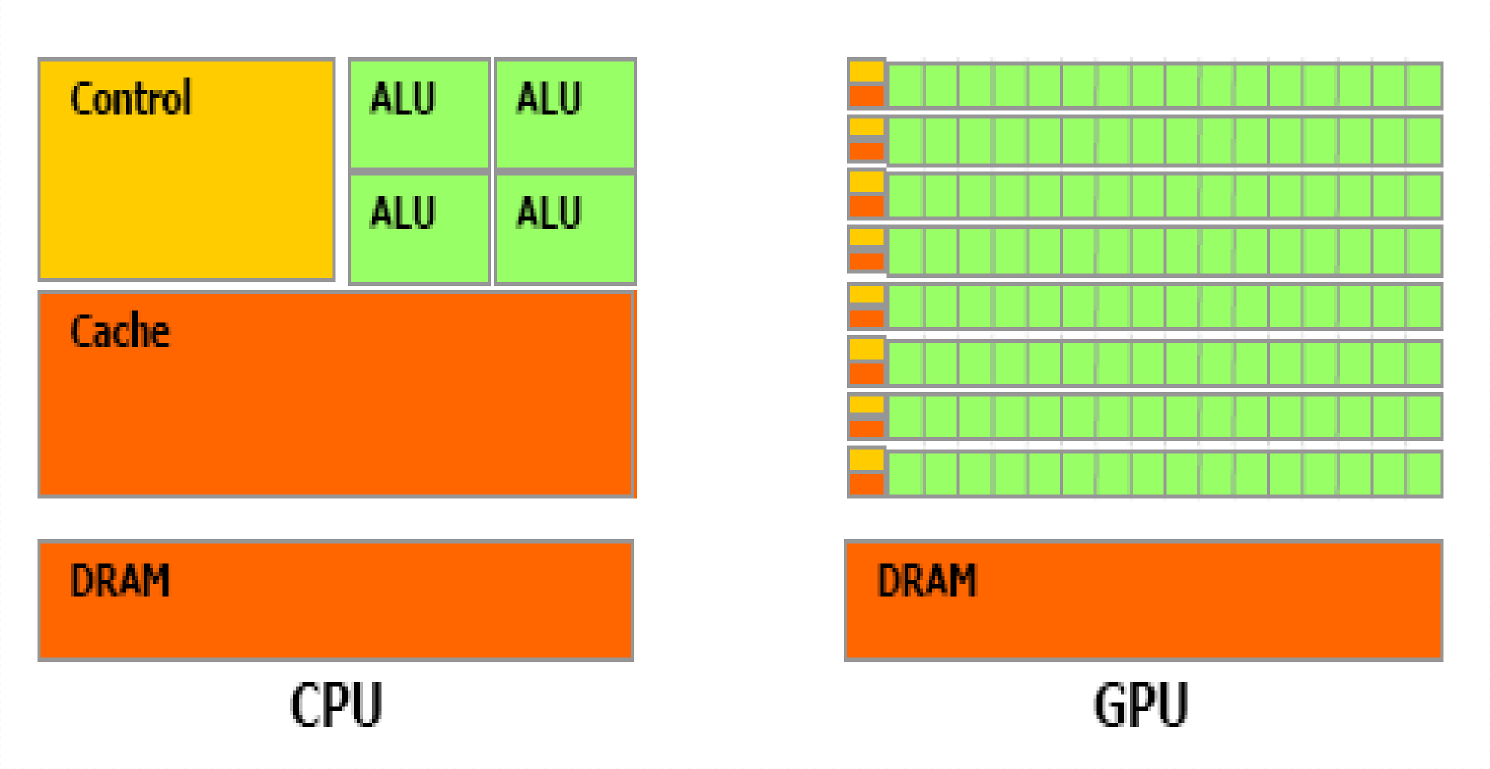
Seconds
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.7737
0.0155
0.0693
0.0101
1.021
0.0711
Diffuse
Advect
Project
CPU
GPU

CPU vs GPU

- CPU
  - Fast caches
  - Branching adaptability
  - High performance
- GPU
  - Multiple ALUs
  - Fast onboard memory
  - High throughput on parallel tasks
    - Executes program on each fragment/vertex

- CPUs are great for ***task*** parallelism
- GPUs are great for ***data*** parallelism

CPU vs GPU - Hardware



- More transistors devoted to data processing

What is CUDA?

- Compute Unified Device Architecture
- NVIDIA’s software architecture for developing and running data-parallel programs
- Programmed in an extension to the C language

Programming CUDA

- Kernel Functions
  - A kernel function is code that runs on the GPU
  - The code is downloaded and executed simultaneously on all stream processors on the GPU
- SIMD Model
  - SIMD stands for Single Instruction, Multiple Data
  - SIMD exploits data level parallelism by performing the same operation on multiple pieces of data at the same time
  - Example: Performing addition on 128 numbers at once

Fluid Dynamics on the GPU

- To implement the Navier-Stokes equations on a GPU we need to write kernel functions for:
  - External Forces
  - Diffusion
  - Advection
  - Projection

Demonstration

Acknowledgements

- “Real-Time Fluid Dynamics for Games” by Jos Stam
- “Fast Fluid Dynamics Simulation on the GPU” by Mark J. Harris
- NVIDIA
  - developer.nvidia.com/CUDA
- “CUDA: Introduction” by Christian Trefftz / Greg Wolffe Grand Valley State University

D.E.N.
DISTRIBUTED EXECUTION NETWORK

http://cis.gvsu.edu/den