Comparison of Methods for Capturing Discontinuities

With an Introduction to GPU Programming

Sean Davis and Will Matern

Acknowledgements

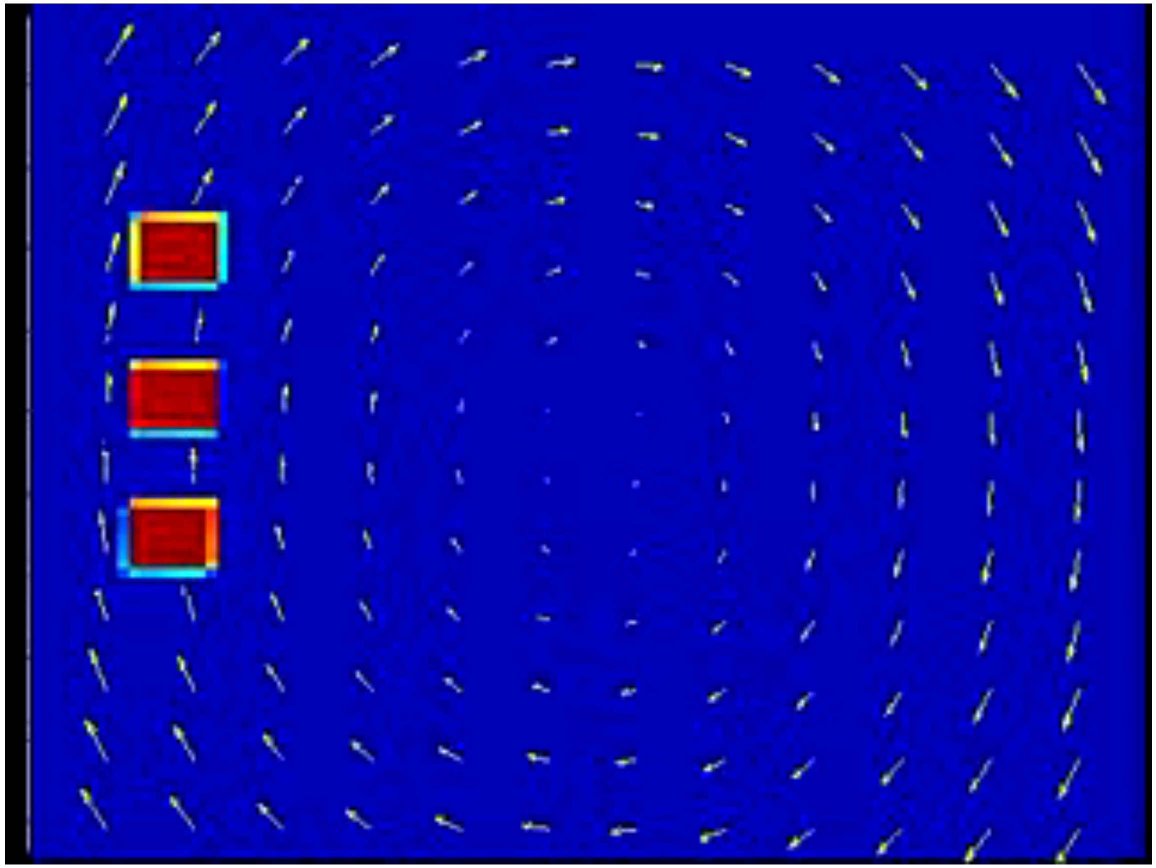
- Mentor: Dr. Bob Robey
- Workshop Coordinator: Dr. Scott Runnels
- Collaborator: David Nicholaeff

Thank you!

Outline

- Motivation
- Problems with capturing discontinuities
- Methods implemented and tested
  - First order Upwind
  - Lax-Wendroff
  - Limiters
  - MUSCL
  - SUPG
- Early MPDATA results
- GPU Programming
  - Introduction
  - Results solving the shallow water equations
- Conclusions

Motivation



A simple model problem showing the advection of radioactive water dumped into the ocean and carried by a rotational current across the Pacific.

1D Advection

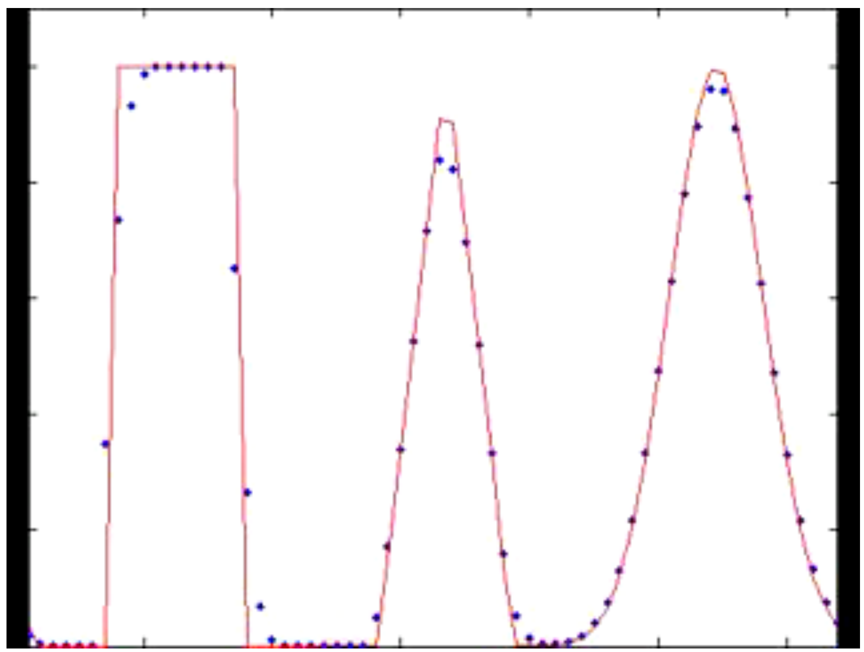
- We are using the 1D advection equation (in our 2D codes) to test various methods:

$$\frac{\partial \psi}{\partial t} + u\frac{\partial \psi}{\partial x} = 0$$

- We are looking at the advection of 3 different types of waves
  - Square Wave
  - Triangular Wave
  - Exponential Smooth Wave

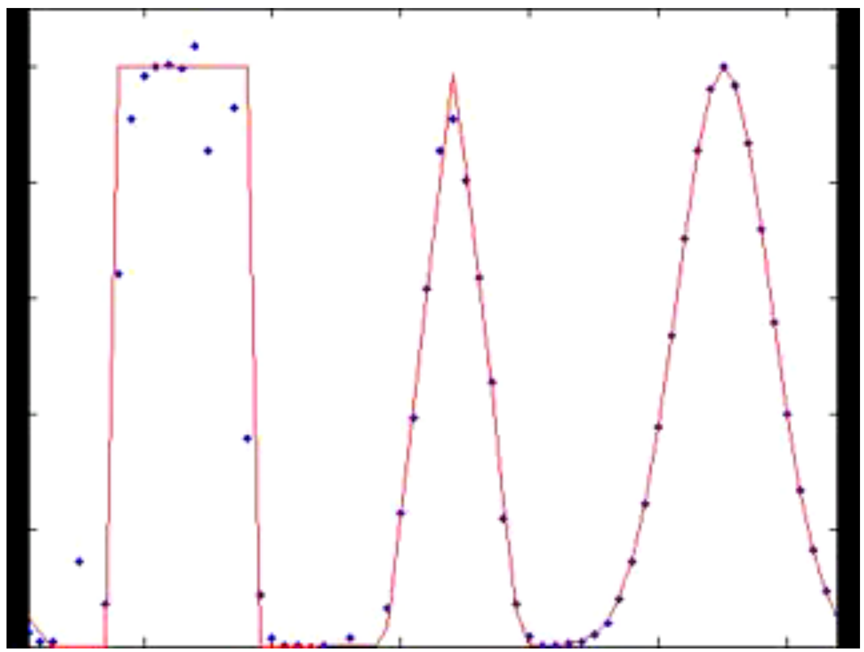
First Order Upwind Scheme

- Conservative 1st order scheme
- Overly dissipative



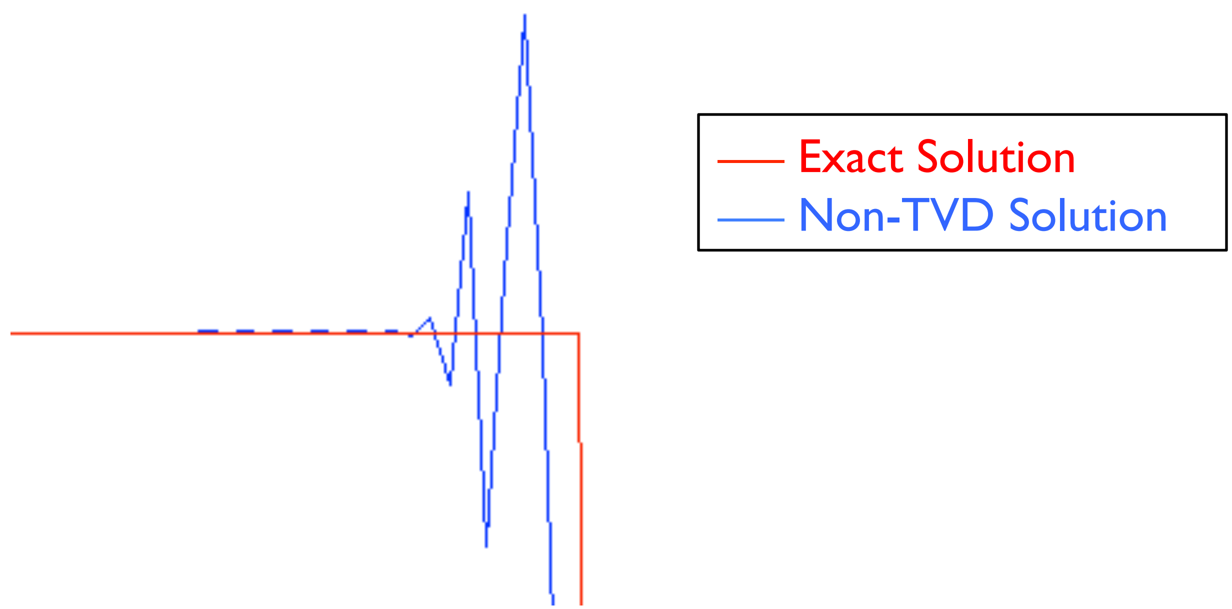
Lax-Wendroff

- Conservative 2nd order scheme
- Approximate solution has oscillations around sharp gradients



Limiters

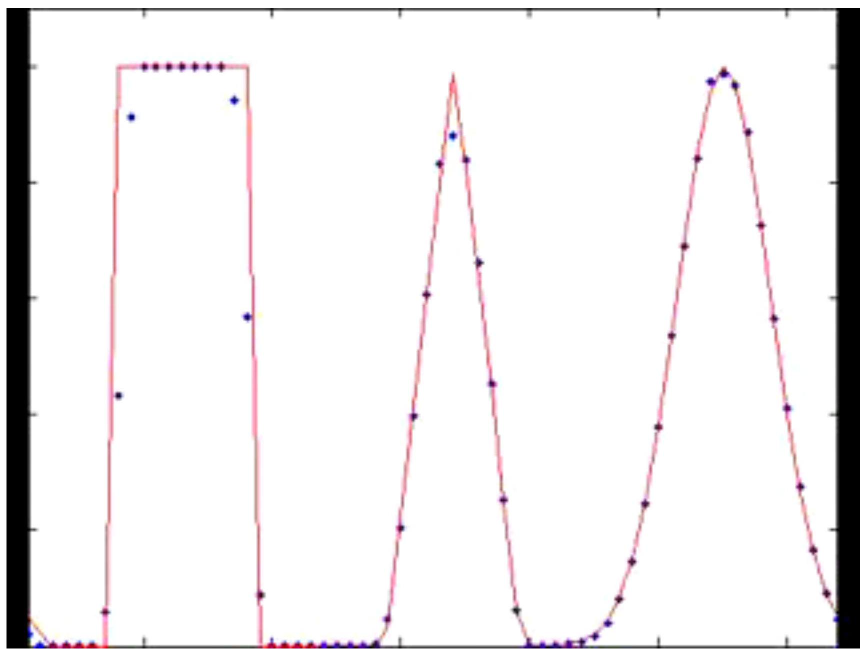
- Godunov's Theorem – Any linear scheme above first order accurate (space) cannot be Total Variation Diminishing (TVD).



- Solution: Limiters (a nonlinear scheme)

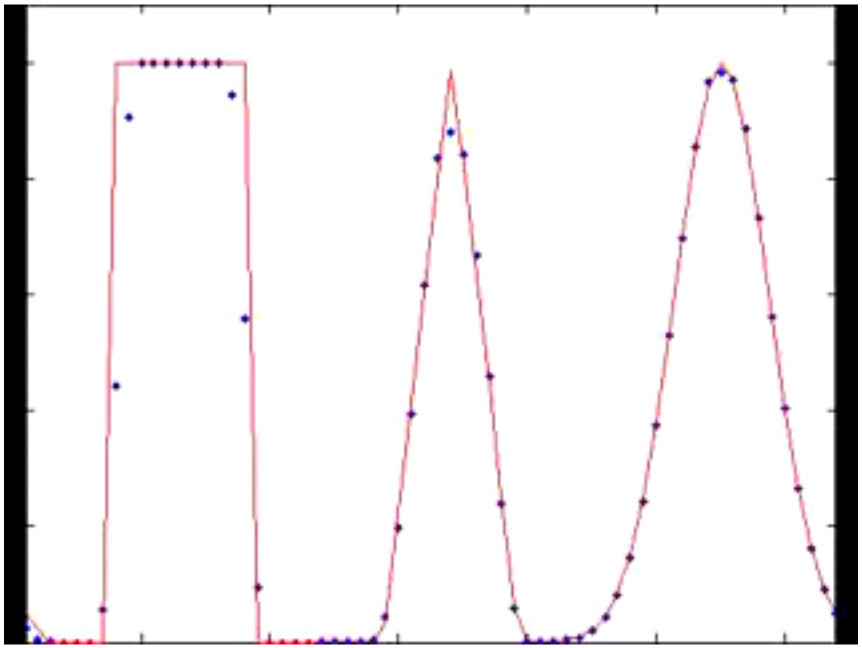
Lax-Wendroff w/ Superbee Limiter

- 1st order near discontinuities, 2nd order elsewhere
- For linear equations is mathematically guaranteed to be TVD



Monotone Upstream Scheme for Conservation Laws (MUSCL)

- Used a modified Osher-Chakravarthy scheme
- Upwind biased
- Second order TVD approximation
- Numerical oscillations avoided using a minmod function



Streamline Upwind Petrov-Galerkin (SUPG)

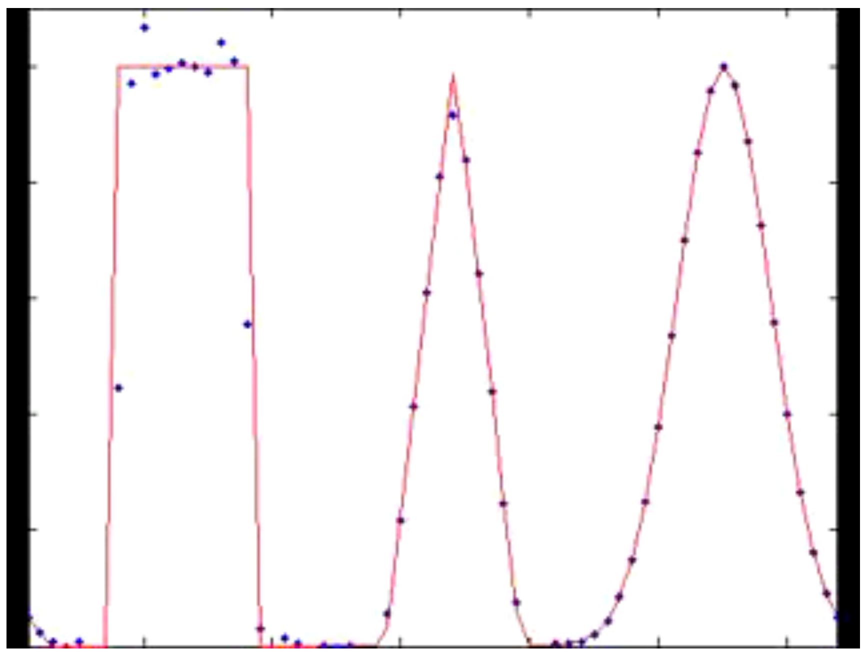
- Trial functions – a set of functions assumed to approximate the underlying solution. The trial functions have degrees of freedom that must be solved for.
- Test functions – a set of functions used to solve for the degrees of freedom of the trial functions. Uses the weak formulation of the governing equation.
- Petrov-Galerkin – Test functions ≠ Trial functions
- SUPG – Choose test functions to be:

$$N + \frac{\alpha h}{2}\left(\frac{u}{\sqrt{u^2+v^2}}\frac{\partial N}{\partial x} + \frac{v}{\sqrt{u^2+v^2}}\frac{\partial N}{\partial y}\right)$$

where N is a trial function

SUPG

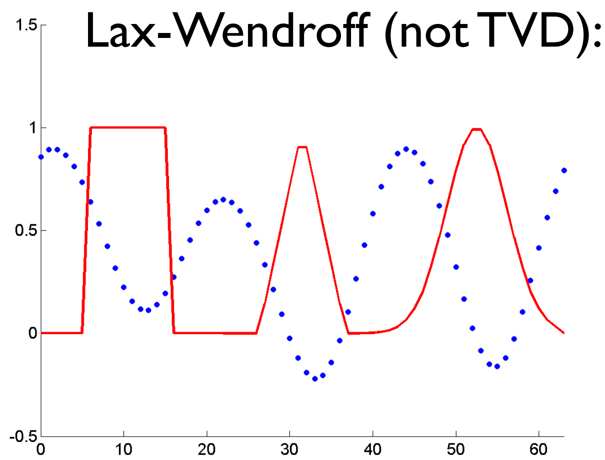
- Inspired by finite difference methods (upwinding and artificial diffusion)
- Applies to linear advection/diffusion equations of any dimension
- Is neither TVD nor conservative

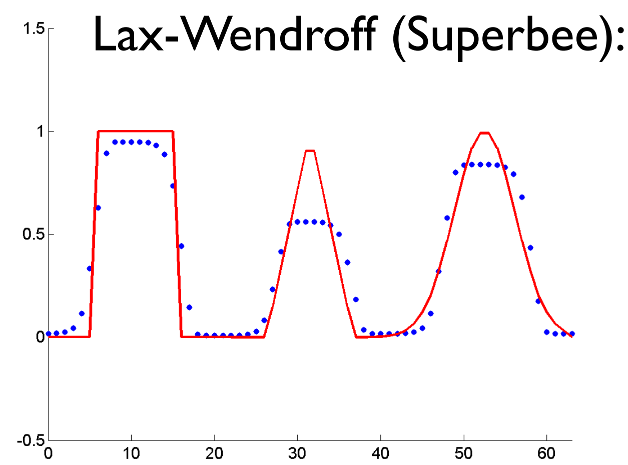


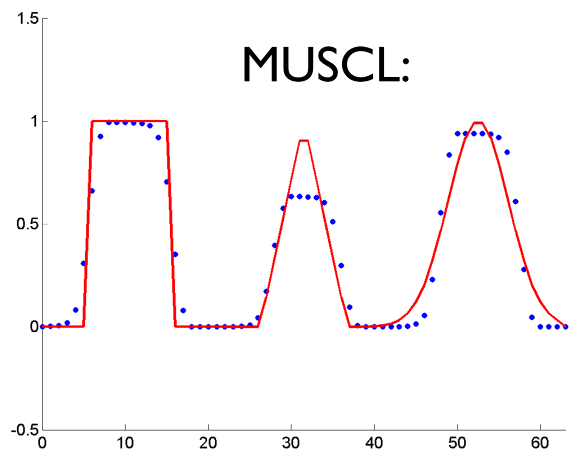
Wave Advection Test Conclusions

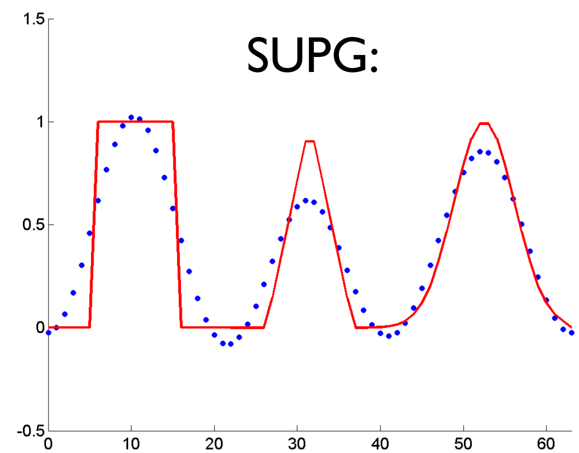
- Lax-Wendroff with slope limiters
  - Effective for sharp discontinuities
  - Overly dissipative for thin waves (cuts off the top of the triangular/gaussian waves)
  - more dissipative than other methods
  - Superbee limiter artificially sharpens smooth waves
- MUSCL with ACM
  - most effective for problems with discontinuities
  - Unnaturally sharpens the solution
- SUPG
  - Balances well between diffusion and discontinuity sharpness
  - Doesn't introduce unnatural sharpening
  - Requires matrix operations
  - Non-conservative

Results for 1D Advection in a 2D Code after 10 Periods









Multidimensional Positive Definite Advection Transport Algorithm (MPDATA)

- A donor-cell approximation is defined in terms of the local Courant number
- Adds a diffusive convective flux
- Subtract out added dissipation using an antidiffusive velocity
- As an example, the 1D advection equation,

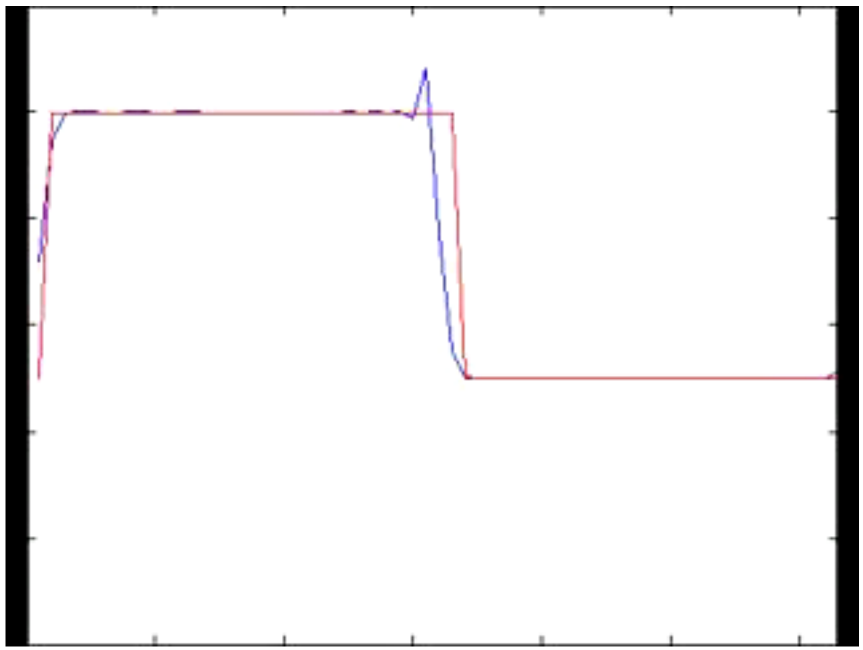
$$\frac{\partial \Psi}{\partial t} = -\frac{\partial}{\partial x}(u\Psi)$$

- becomes,

$$\frac{\partial \Psi}{\partial t} = -\frac{\partial}{\partial x}(u\Psi) - \frac{\partial}{\partial x}\left(-K\frac{\partial \Psi}{\partial x}\right)$$

MPDATA

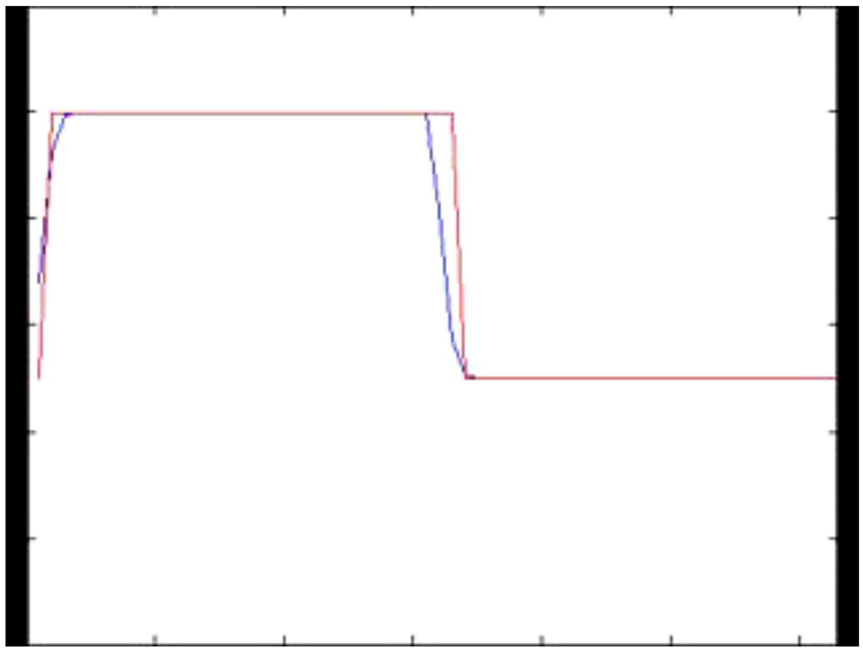
- Antidiffusive correction to the donor cell method causes oscillations



PRELIMINARY

MPDATA

- A Flux Corrected Transport (FCT) algorithm can correct the oscillations
- Performing additional local corrective iterations increases the order of accuracy of the solution

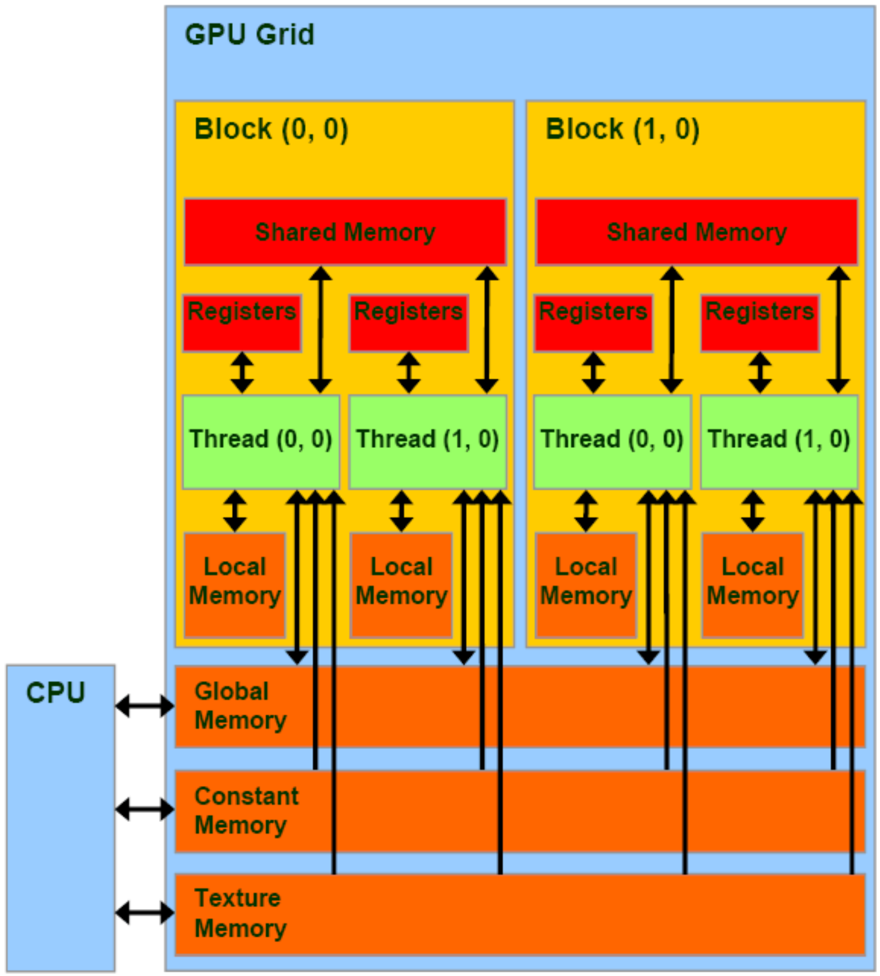


PRELIMINARY

GPUs at a glance

- Much higher FLOPS (floating points operations per second) per dollar than CPUs.
- Different architecture from a CPU.
  - Many low throughput processing units
  - Cores share global memory
  - Warps/Wavefronts – groups of cores run same code.
- Must use a language compiled for GPU
  - OpenCL – Open programming language for multiple CPUs (on same node) and GPU.
  - CUDA – Nvidia only.
- This Project: OpenCL

GPU architecture



http://www.realworldtech.com/gt200/3/

Shallow Water Equations

- Simplification of the Euler Equations
  - Uniform Fluid Density
  - Wavelengths are much longer than fluid depth

$$\frac{\partial h}{\partial t} + \frac{\partial (uh)}{\partial x} + \frac{\partial (vh)}{\partial y} = 0$$

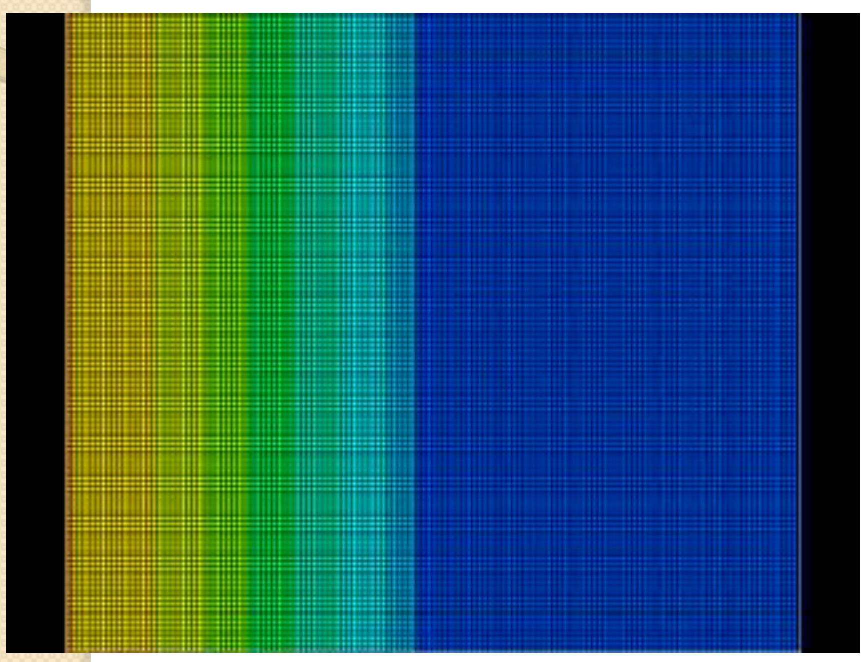
$$\frac{\partial (uh)}{\partial t} + \frac{\partial (u^2 h + 1/2 g h^2)}{\partial x} + \frac{\partial (uvh)}{\partial y} = 0$$

$$\frac{\partial (vh)}{\partial t} + \frac{\partial (uvh)}{\partial x} + \frac{\partial (v^2 h + 1/2 g h^2)}{\partial y} = 0$$

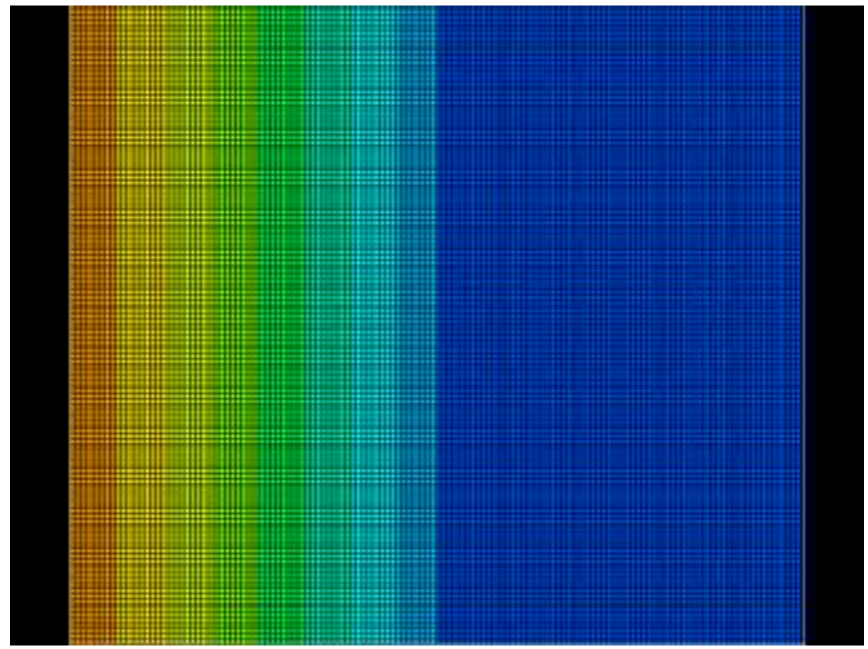
  - Inert Tracer Equation

$$\frac{\partial \psi}{\partial t} + u\frac{\partial \psi}{\partial x} + v\frac{\partial \psi}{\partial y} = 0$$

Shallow Water Simulations on the GPU



Fluid Height



Inert Tracer Concentration

Future Directions

- FEM approach for full Euler equations
- Implement more of these schemes in the GPU code
- Develop full Euler Equation solvers with some of these schemes
- Alternative numerical methods (WENO, Discontinuous Galerkin, etc.)

Question, Comments, Suggestions?