The IWAVE Modeling Framework

W. W. Symes and Xin Wang

2010 TRIP Annual Meeting

IWAVE

created by Terentyev, Vdovina and Symes 2007, a software framework for construction of regular grid FD and FE methods for time-dependent PDEs, includes utility software for

- parameter parsing
- i/o methods, sampling operators
- date exchanging with MPI, just like PETSc
- distributed spatial arrays and groupings of arrays

along with a variable density AWE solver in up to 3D

- staggered grid FD scheme of order 2 in time and $2k$ in space
- support either reflecting or absorbing bnd cond
- output traces (seismograms) at specified sample rates and/or movie frames
- *mpi* parallelization via domain decomp and shots (tasks)

IWAVE

now also provides

- a working flow of new solver generation
- two more applications: iso elastic wave equations and Burgers equation

New Solver Generation

FD stencil, the starting point of generating new FD solvers

- memory allocation
- pattern of exchanging data
- ...

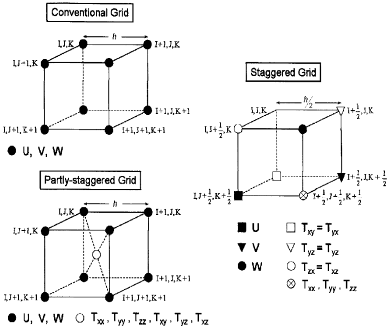


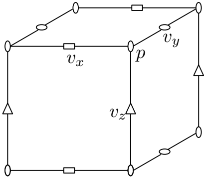
Figure: courtesy of Moczo et al. 2007

but seems there are a lot of choices: orders, grid types

New Solver Generation

before determining stencils, let's take a look at pattern of FDTD wave solvers:

- wave equations have terms of up to 1st order spatial derivative
- FD discretization **along each spatial axis** has up to 2 different types of grids,
  - primal grid: integer grid, index-0 grid $\Rightarrow 0$
  - dual grid: half integer grid, index-0 grid $\Rightarrow 1/2$



e.g., pressure p on primal grids along 3 axes, v_x on dual grid along x-axis and primal grids along y-axis and z-axis

Define FD Stencils

FD stencils determined by

- grid type table for each variable
- variable-dependence relation

e.g., 2D isotropic elastic wave staggered grid FDTD solver

	grid type		dependence relation				
	z-axis	x-axis	σ_{zz}	σ_{xx}	σ_{zx}	u_z	u_x
σ_{zz}	P	P	-	-	-	$\partial/\partial z$	$\partial/\partial x$
σ_{xx}	P	P	-	-	-	$\partial/\partial z$	$\partial/\partial x$
σ_{zx}	D	D	-	-	-	$\partial/\partial x$	$\partial/\partial z$
u_z	D	P	$\partial/\partial z$	-	$\partial/\partial x$	-	-
u_z	P	D	-	$\partial/\partial x$	$\partial/\partial z$	-	-

New Solver Generation

break the process into several parts:

- set grid type and variable-dependence relation
- read grid info $\rightarrow$ basic primal grid assuming single grid determines others
- populate in the parameter data
- assign action list $\rightarrow$ which array updated in which order
- single time stepping function interface, use switch/case to pick right update function for arrays (variables)
- generate FD stencil (IWAVE)
- automating domain decomposition, implicit chop grid into blocks (block decomposition) according to FD stencil (IWAVE)
- modelinit function to setup parameters, assign function pointers (C mechanism to implement inheritance)

IWAVE Elastic Wave Solver

isotropic elastic wave equations in velocity-stress formulation

$$
\begin{aligned}
\frac{\partial \sigma_{zz}}{\partial t} - (\lambda + 2\mu)\frac{\partial u}{\partial z} - \lambda\frac{\partial v}{\partial x} &= 0 \\
\frac{\partial \sigma_{xx}}{\partial t} - \lambda\frac{\partial u}{\partial z} - (\lambda + 2\mu)\frac{\partial v}{\partial x} &= 0 \\
\frac{\partial \sigma_{zx}}{\partial t} - \mu\frac{\partial u}{\partial z} - \mu\frac{\partial v}{\partial x} &= 0 \\
\rho\frac{\partial u}{\partial t} - \frac{\partial \sigma_{zz}}{\partial z} - \frac{\partial \sigma_{zx}}{\partial x} &= 0 \\
\rho\frac{\partial v}{\partial t} - \frac{\partial \sigma_{zx}}{\partial z} - \frac{\partial \sigma_{xx}}{\partial x} &= 0
\end{aligned}
$$

IWAVE Elastic Wave Solver

two layer model

	top	bottom
ρ	2100 $\mathrm{kg/m^3}$	2300 $\mathrm{kg/m^3}$
v_p	2.3 m/ms	3.0 m/ms
v_s	0.93897 m/ms	1.2247 m/ms
λ	7406 MPa	13800 MPa
μ	1851.5 MPa	3450 MPa

- Ricker wavelet with central frequency 15 Hz, free surface bnd cond for all boundaries
- source at depth 40 m and offset 3300 m, receivers at depth 20 m and offset from 100 m to 6100 m with interval 20 m
- 2-4 staggered-grid FD on a grid of size 20 m, wave propagates 3 sec

IWAVE Elastic Wave Solver

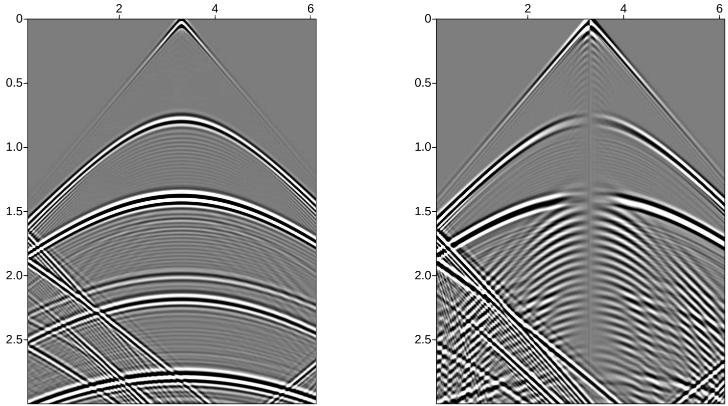


Figure: σ_{zz} (left) and σ_{zx} (right) dome model

- Ricker wavelet with central frequency 15 Hz, free surface bnd cond for all boundaries
- source at depth 40 m and offset 3300 m, receivers at depth 20 m and offset from 100 m to 6100 m with interval 20 m
- 2-4 staggered-grid FD on a grid of size 20 m, wave propagates 3 sec

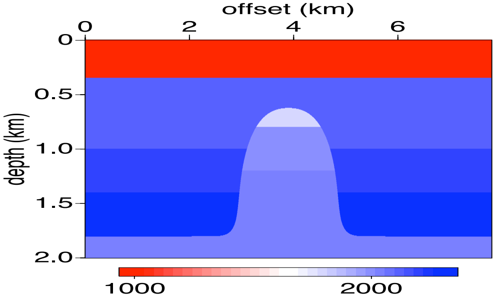


Figure: ρ

IWAVE Elastic Wave Solver

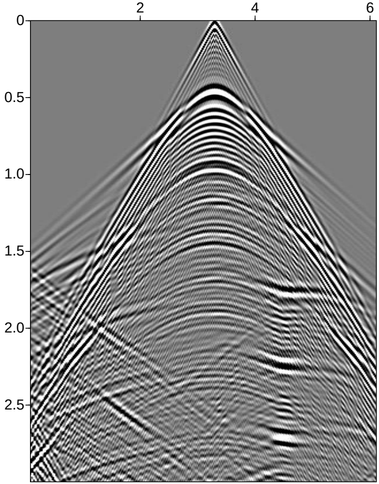


Figure: σ_{zz}

IWAVE Burgers Equation Solver

Burgers equation:

$$u_t + (u^2)_x + (u^2)_y = 0$$

finite volume discretization: take the volume integral over the total volume of the cell, v_i

$$\int_{v_i} \frac{\partial u}{\partial t} \, \mathrm{d}v + \int_{v_i} (u^2)_x + (u^2)_y \, \mathrm{d}v = 0$$

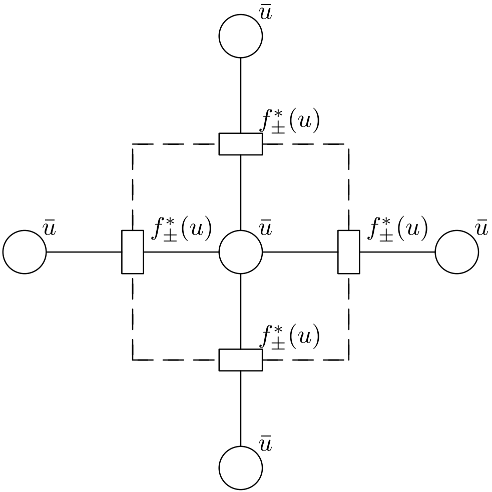
let $\bar{u}_i = \int_{v_i} \frac{\partial u}{\partial t} \, \mathrm{d}v / v_i$ and integration by parts

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{1}{v_i} \oint_{S_i} f^*(u) \cdot n \, \mathrm{d}S = 0$$

numerical flux $f^*(u) = (u^2, u^2)|_{S_i}$ defined on the cell boundary S_i

IWAVE Burgers Equation Solver

stencil of regular grid FV scheme $\Leftrightarrow$ 2nd order staggered-grid stencil



IWAVE Burgers Equation Solver

- initial condition:

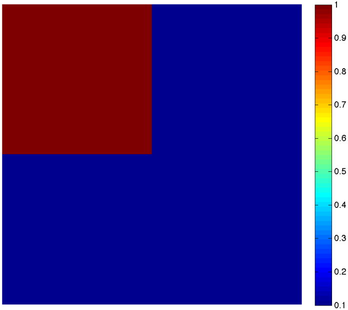


Figure: $u(x, y, 0)$

- periodic boundary condition
- 201×201 grid points with grid size 5 m

IWAVE Burgers Equation Solver

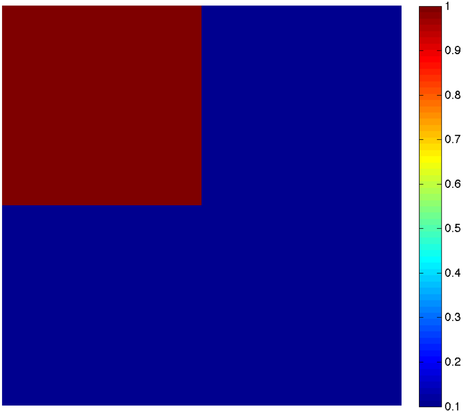
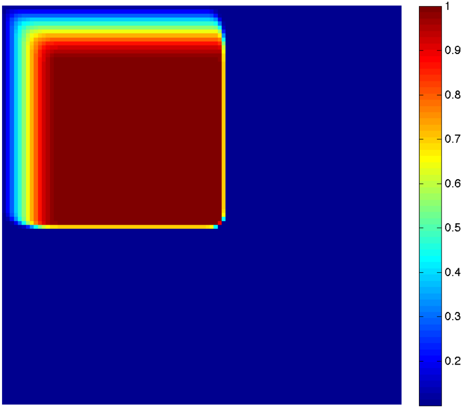


Figure: $t = 0.0s$

previous slide next slide

IWAVE Burgers Equation Solver



previous slide next slide

Figure: $t = 0.1s$

IWAVE Burgers Equation Solver

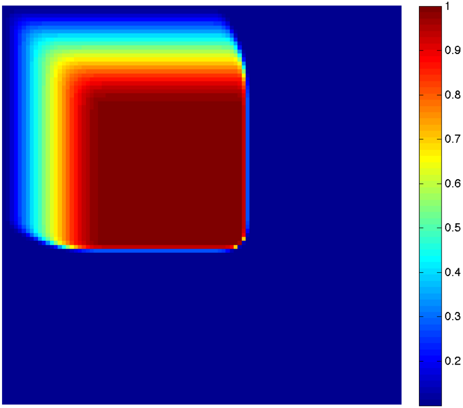


Figure: $t = 0.2s$

◂ previous slide ▸ next slide

IWAVE Burgers Equation Solver

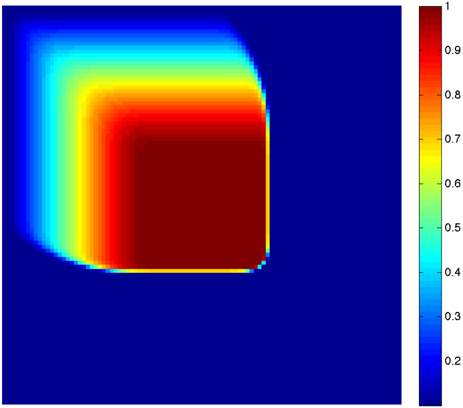


Figure: $t = 0.3s$

◂ previous slide ▸ next slide

IWAVE Burgers Equation Solver

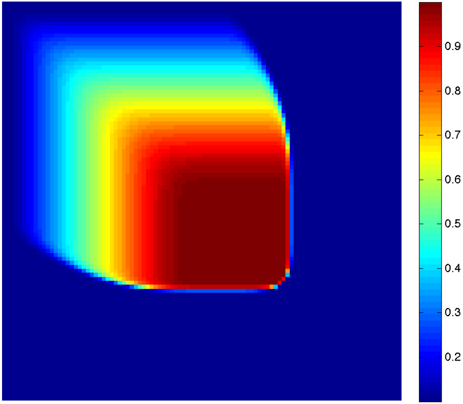


Figure: $t = 0.4s$

IWAVE Burgers Equation Solver

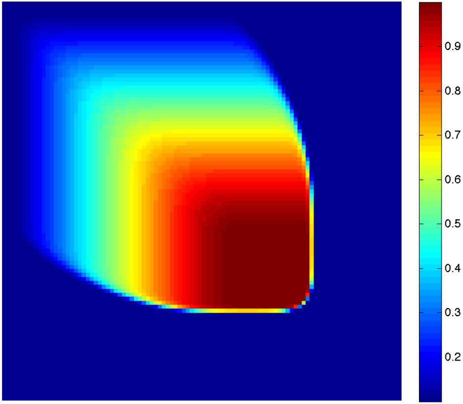


Figure: $t = 0.5s$

◂ previous slide ▸ next slide

IWAVE Burgers Equation Solver

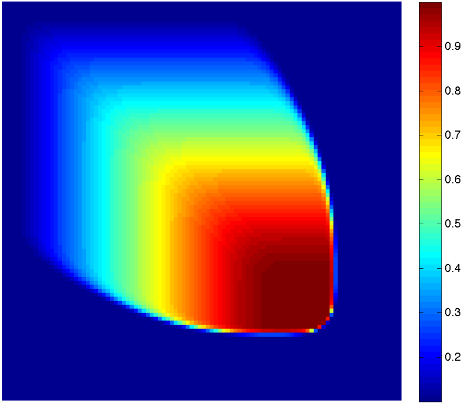


Figure: $t = 0.6s$

IWAVE Burgers Equation Solver

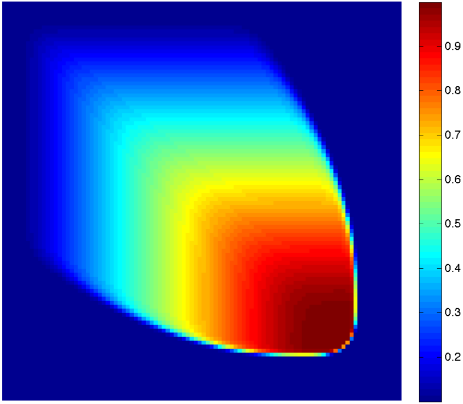


Figure: $t = 0.7s$

previous slide next slide

IWAVE Burgers Equation Solver

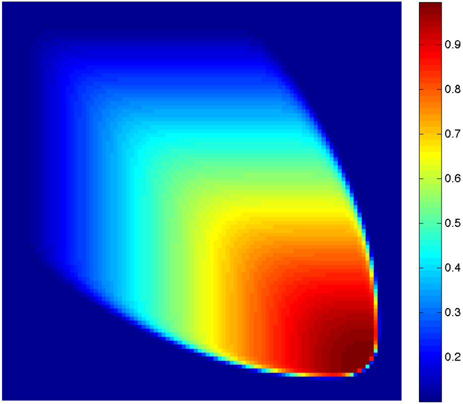


Figure: $t = 0.8s$

previous slide next slide

IWAVE Burgers Equation Solver

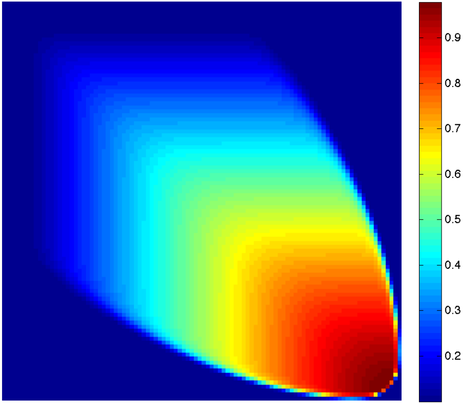


Figure: $t = 0.9s$

◂ previous slide ▸ next slide

IWAVE Burgers Equation Solver

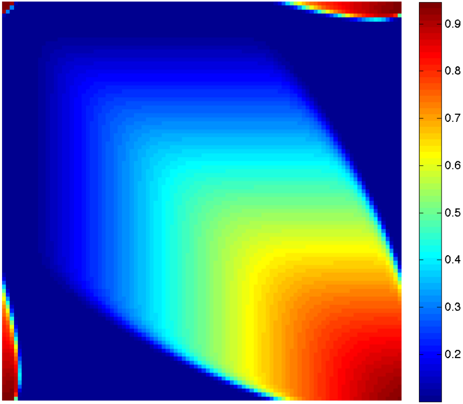


Figure: $t = 1.0s$

previous slide next slide

IWAVE Burgers Equation Solver

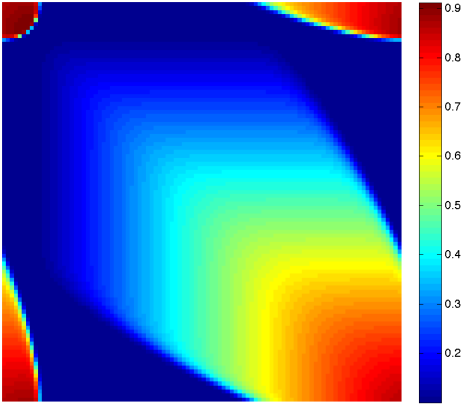
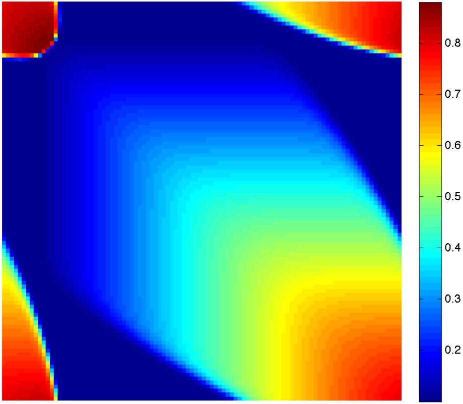


Figure: $t = 1.1s$

◂ previous slide ▸ next slide

IWAVE Burgers Equation Solver



previous slide   next slide

Figure: $t = 1.2s$

IWAVE Burgers Equation Solver

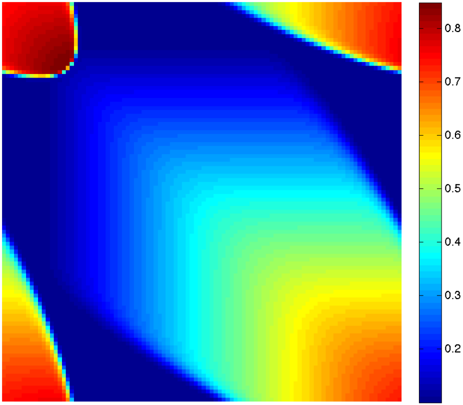


Figure: $t = 1.3s$

IWAVE Burgers Equation Solver

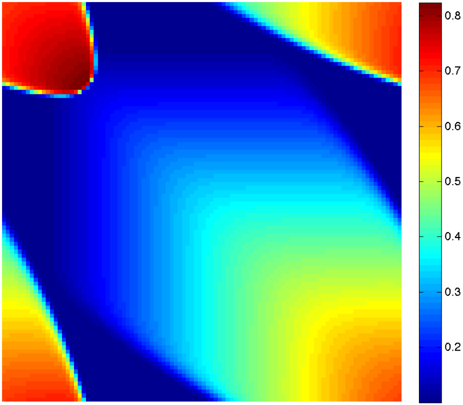


Figure: $t = 1.4s$

IWAVE Burgers Equation Solver

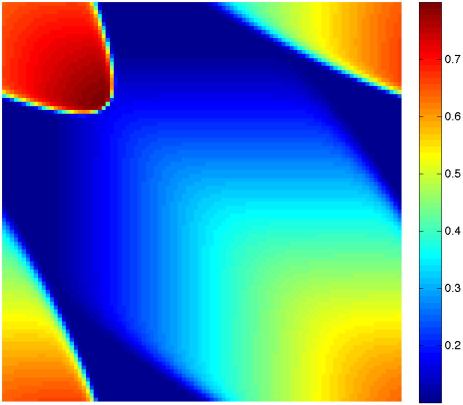


Figure: $t = 1.5s$

previous slide next slide

IWAVE Burgers Equation Solver

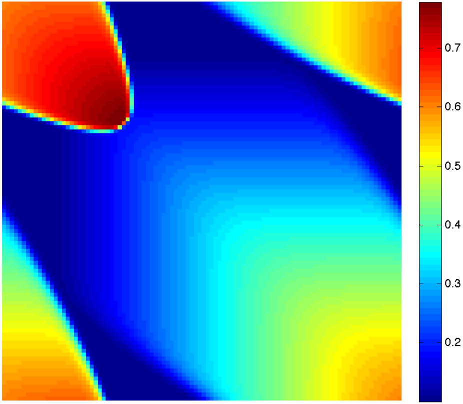


Figure: $t = 1.6s$

previous slide next slide

IWAVE Burgers Equation Solver

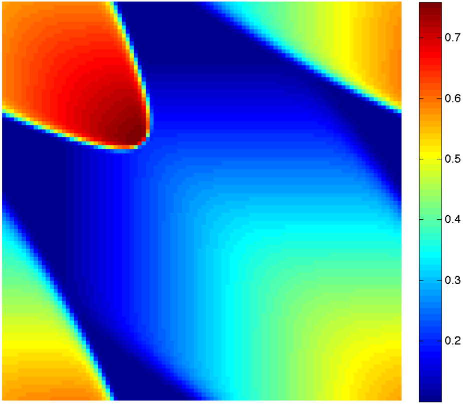


Figure: $t = 1.7s$

IWAVE Burgers Equation Solver

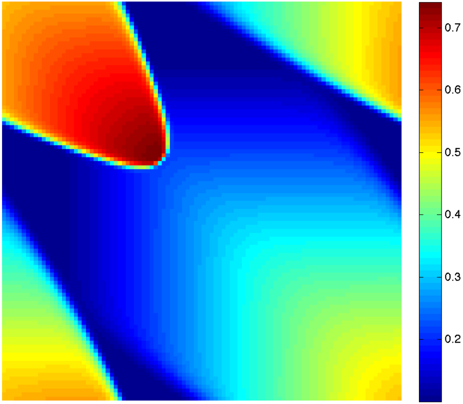


Figure: $t = 1.8s$

◂ previous slide ▸ next slide

IWAVE Burgers Equation Solver

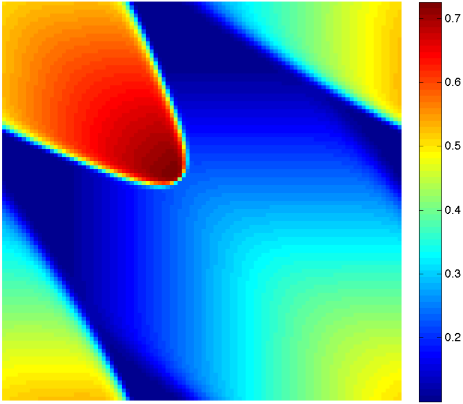


Figure: $t = 1.9s$

previous slide next slide

IWAVE Burgers Equation Solver

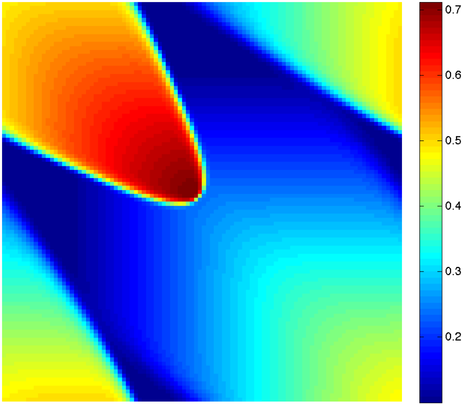


Figure: $t = 2.0s$

IWAVE Burgers Equation Solver

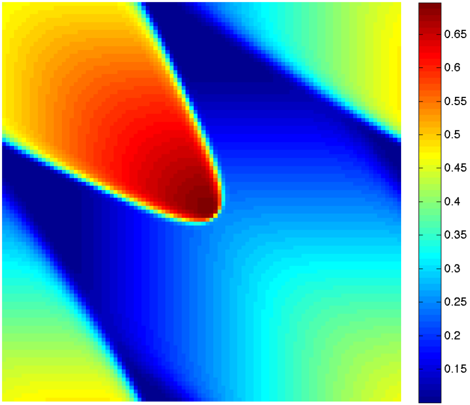


Figure: $t = 2.1s$

◂ previous slide ▸ next slide

IWAVE Burgers Equation Solver

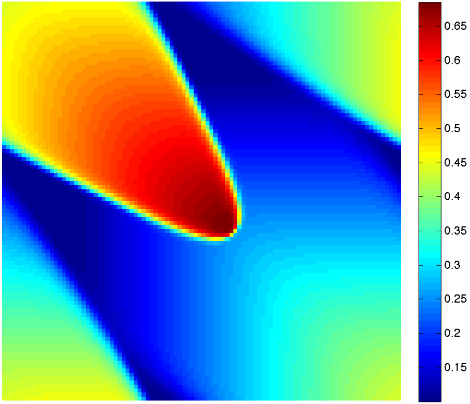


Figure: $t = 2.2s$

IWAVE Burgers Equation Solver

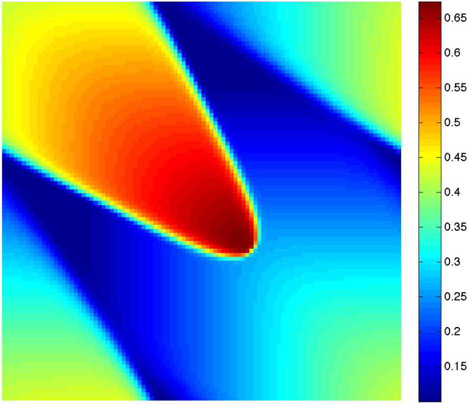


Figure: $t = 2.3s$

◂ previous slide ▸ next slide

IWAVE Burgers Equation Solver

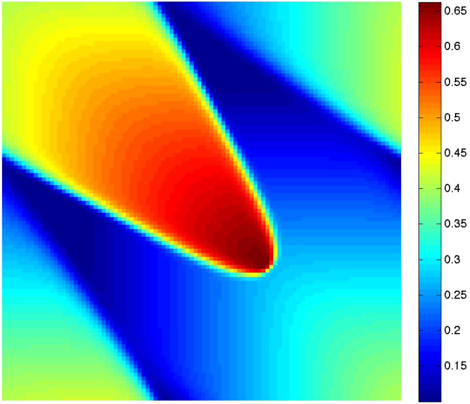


Figure: $t = 2.4s$

◂ previous slide ▸ next slide

IWAVE Burgers Equation Solver

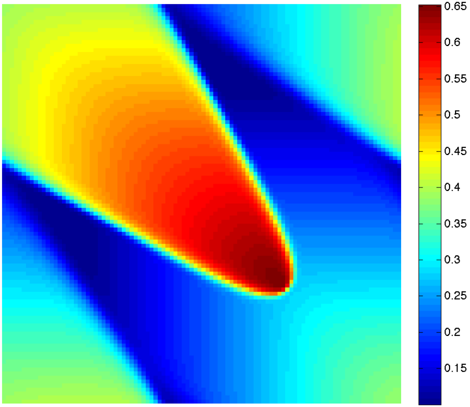
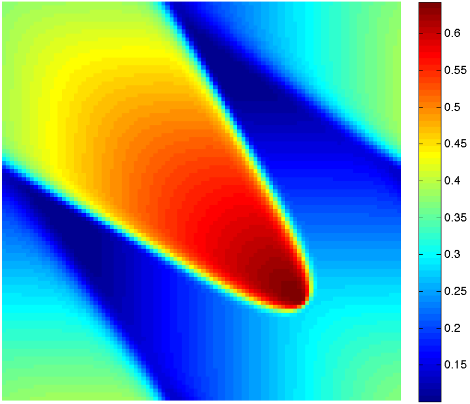


Figure: $t = 2.5s$

previous slide
next slide

IWAVE Burgers Equation Solver



◂ previous slide ▸ next slide

Figure: $t = 2.6s$

IWAVE Burgers Equation Solver

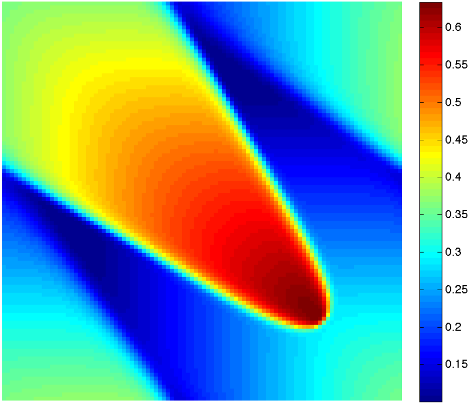
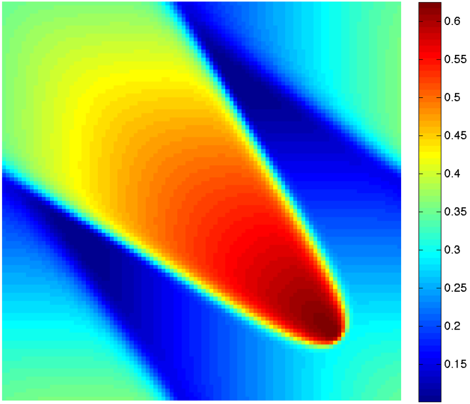


Figure: $t = 2.7s$

IWAVE Burgers Equation Solver



previous slide next slide

Figure: $t = 2.8s$

IWAVE Burgers Equation Solver

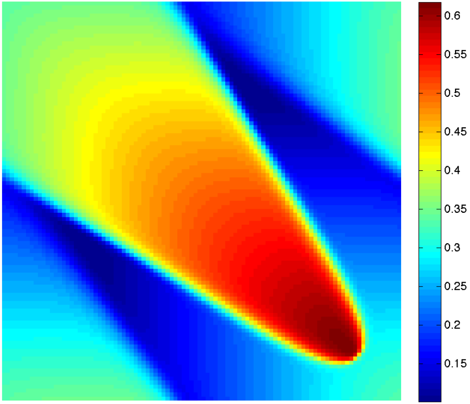


Figure: $t = 2.9s$

previous slide next slide

IWAVE Burgers Equation Solver

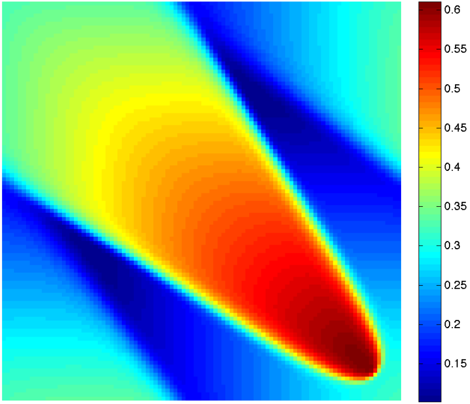


Figure: $t = 3.0s$

previous slide next slide

Future Plan

- absorbing boundary condition for elastic wave solver
- implement DG scheme on regular grid under IWAVE framework
- more applications

Thank You

Q&A