Automating Scientific Computing with Discontinuous Petrov-Galerkin Finite Elements

Truman Ellis

Advisors: Leszek Demkowicz, Robert Moser,

Collaborators: Nate Roberts (Argonne), Jesse Chan (Rice)

THE UNIVERSITY OF TEXAS AT AUSTIN

INSTITUTE FOR COMPUTATIONAL ENGINEERING & SCIENCES

Table of Contents

THE UNIVERSITY OF TEXAS AT AUSTIN

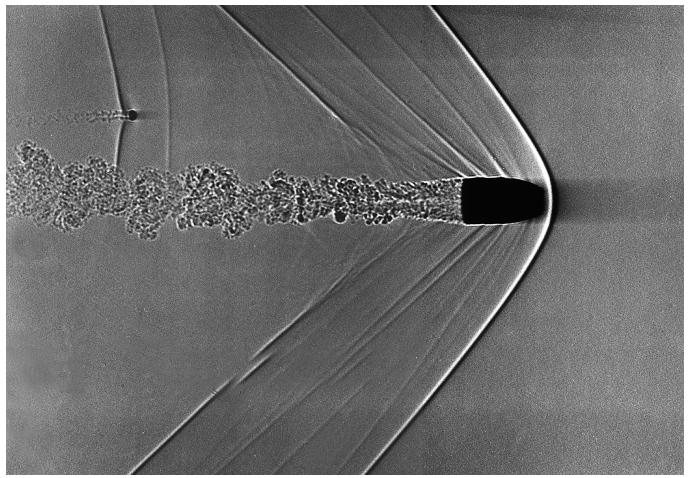
Navier-Stokes Equations

Numerical Challenges

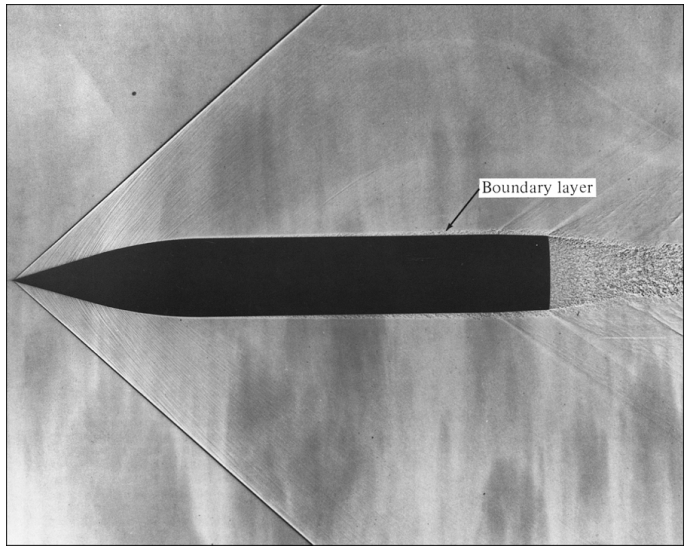
THE UNIVERSITY OF TEXAS AT AUSTIN

Robust simulation of unsteady fluid dynamics remains a challenging issue.

- Resolving solution features (sharp, localized viscous-scale phenomena)
  - Shocks
  - Boundary layers - resolution needed for drag/load
  - Turbulence (non-localized)
- Stability of numerical schemes
  - Nonlinearity
  - Nature of PDE changes for different flow regimes
  - Coarse/adaptive grids
  - Higher order



Shock

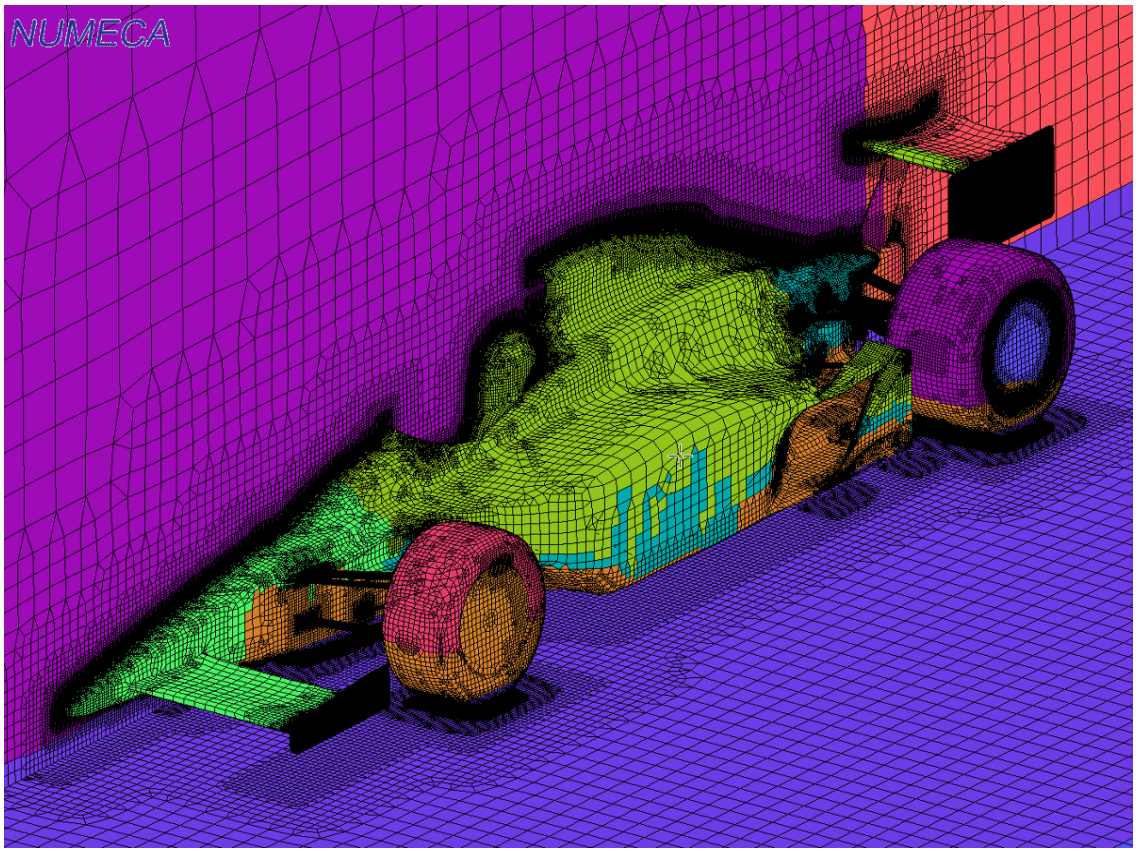


Boundary layer

Motivation

Initial Mesh Design is Expensive and Time-Consuming

- Surface mesh must accurately represent geometry
- Volume mesh needs sufficient resolution for asymptotic regime
- Engineers often forced to work by trial and error
- Bad in the context of HPC



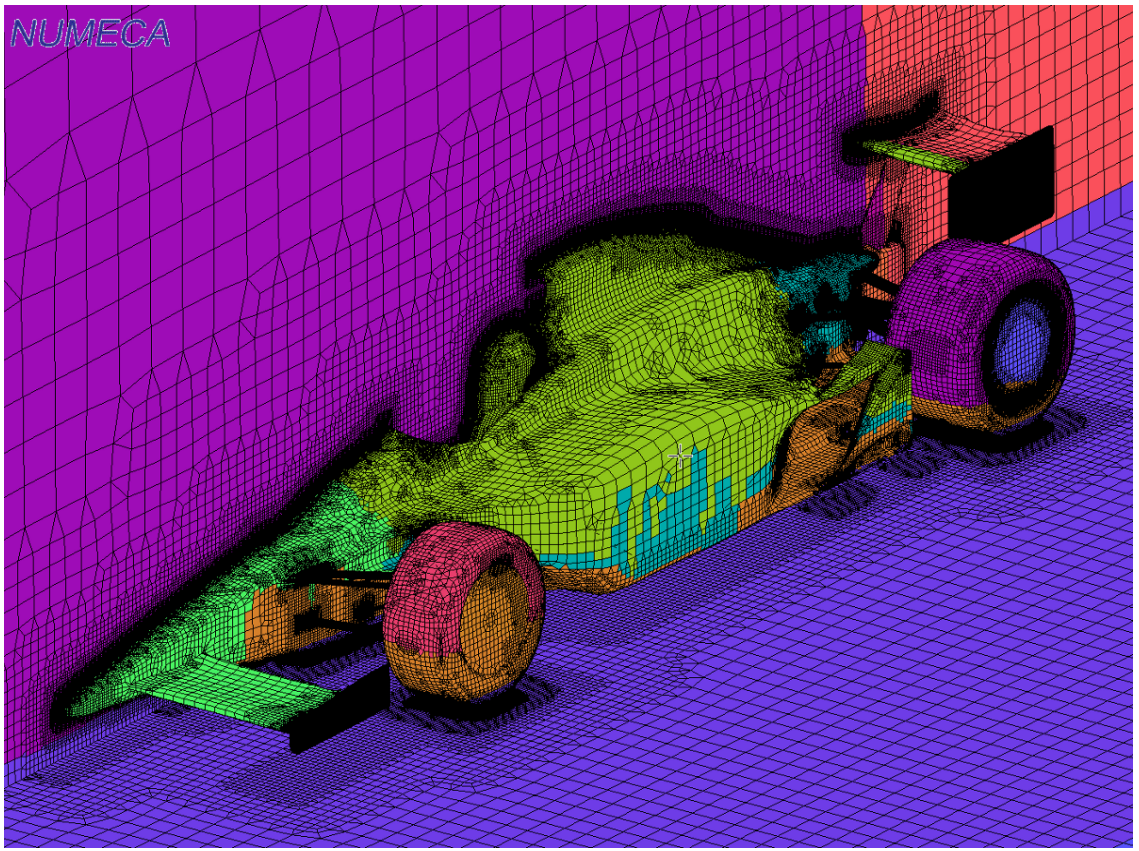
Formula 1 Mesh by Numeca

Motivation

Initial Mesh Design is Expensive and Time-Consuming

THE UNIVERSITY OF TEXAS AT AUSTIN

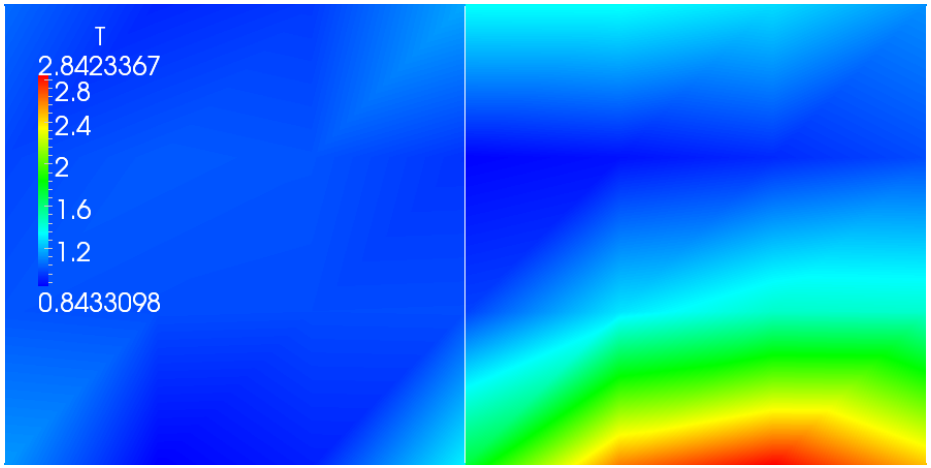
- Surface mesh must accurately represent geometry
- Volume mesh needs sufficient resolution for asymptotic regime
- Engineers often forced to work by trial and error
- Bad in the context of HPC
- We desire an automated computational technology



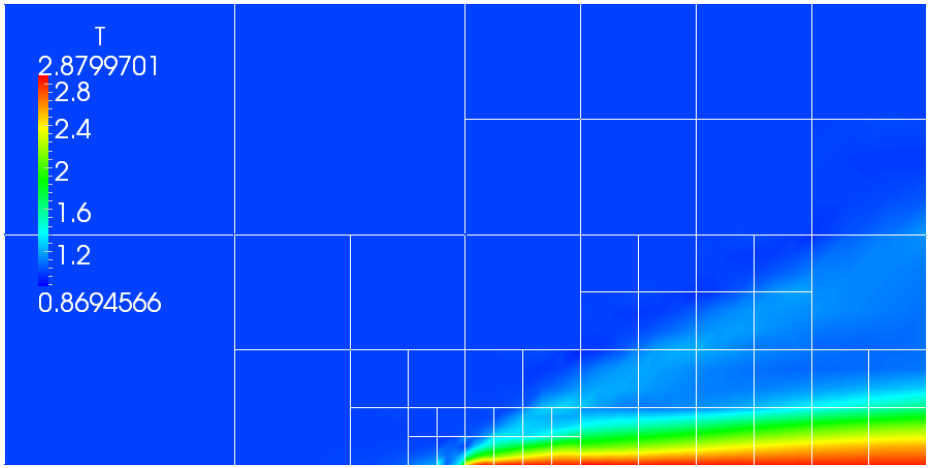
Formula 1 Mesh by Numeca

DPG on Coarse Meshes

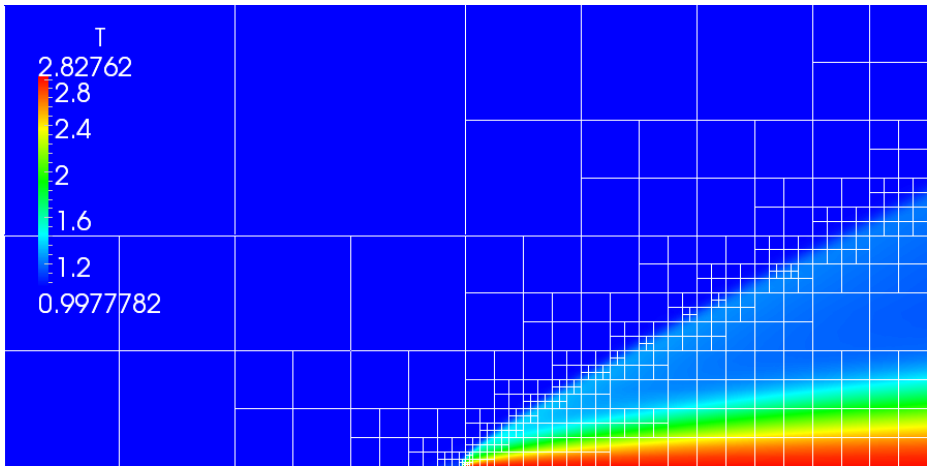
Adaptive Solve of the Carter Plate Problem[1] $Re = 1000$



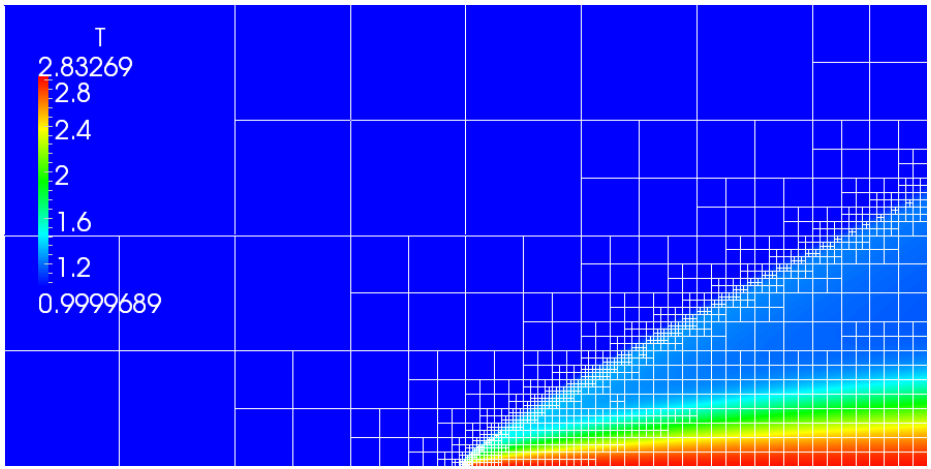
Temperature on Initial Mesh



Temperature after 4 Refinements



Temperature after 8 Refinements



Temperature after 11 Refinements

[1] J.L. Chan. ``A DPG Method for Convection-Diffusion Problems''. PhD thesis. University of Texas at Austin, 2013.

Lessons from Other Methods

THE UNIVERSITY OF TEXAS AT AUSTIN

Streamline Upwind Petrov-Galerkin: Adaptively changing the test space can produce a method with better stability.

Discontinuous Galerkin: Discontinuous basis functions are a legitimate option for finite element methods.

Hybridized DG: Mesh interface unknowns can facilitate static condensation -- reducing the number of DOFs in the global solve.

Least-Squares FEM: The finite element method is most powerful in a minimum residual context (i.e. as a Ritz method).

Space-Time FEM: Highly adaptive methods should have adaptive time integration. Superior framework for problems with moving boundaries. Requires a method that is both temporally and spatially stable.

Table of Contents

THE UNIVERSITY OF TEXAS AT AUSTIN

Overview of DPG

THE UNIVERSITY OF TEXAS AT AUSTIN

DPG is a Minimum Residual Method

Find $u \in U$ such that

$$b(u, v) = l(v) \quad \forall v \in V$$

with operator $B : U \to V'$ defined by $b(u, v) = \langle Bu, v \rangle_{V' \times V}$.
This gives the operator equation

$$Bu = l \quad \in V' .$$

We wish to minimize the residual $Bu - l \in V'$:

$$u_h = \arg\min_{w_h \in U_h} \frac{1}{2} \|Bw_h - l\|_{V'}^2 .$$

Dual norms are not computationally tractable. Inverse Riesz map moves the residual to a more accessible space:

$$u_h = \arg\min_{w_h \in U_h} \frac{1}{2} \left\| R_V^{-1} (Bw_h - l) \right\|_V^2 .$$

Overview of DPG

Petrov-Galerkin with Optimal Test Functions

THE UNIVERSITY OF TEXAS AT AUSTIN

Taking the Gâteaux derivative to be zero in all directions $\delta u \in U_h$ gives,

$$\left(R_V^{-1}(Bu_h - l), R_V^{-1}B\delta u\right)_V = 0, \quad \forall \delta u \in U,$$

which by definition of the Riesz map is equivalent to

$$\left\langle Bu_h - l, R_V^{-1}B\delta u_h \right\rangle = 0 \quad \forall \delta u_h \in U_h \,,$$

with optimal test functions $v_{\delta u_h} := R_V^{-1}B\delta u_h$ for each trial function δu_h.

Resulting Petrov-Galerkin System

This gives a simple bilinear form

$$b(u_h, v_{\delta u_h}) = l(v_{\delta u_h}),$$

with $v_{\delta u_h} \in V$ that solves the auxiliary problem

$$(v_{\delta u_h}, \delta v)_V = \langle R_V v_{\delta u_h}, \delta v \rangle = \langle B\delta u_h, \delta v \rangle = b(\delta u_h, \delta v) \quad \forall \delta v \in V.$$

Overview of DPG

Mixed Formulation

Identifying the error representation function:

$$\psi := R_V^{-1}(Bu_h - l)$$

allows us to develop an alternative interpretation of DPG.

DPG as a Mixed Problem

Find $\psi \in V$, $u_h \in U_h$ such that

$$\begin{aligned} (\psi, \delta v)_V - b(u_h, \delta v) &= -l(\delta v) \qquad \forall \delta v \in V \\ b(\delta u_h, \psi) &= 0 \qquad \forall \delta u_h \in U_h \end{aligned}$$

In this unconventional saddle-saint problem, the approximate solution u_h comes from a finite-dimensional trial space and plays the role of the Lagrange multiplier for the error representation function

Overview of DPG

THE UNIVERSITY OF TEXAS AT AUSTIN

DPG is the Most Stable Petrov-Galerkin Method

Babuška's theorem guarantees that *discrete stability and approximability imply convergence*. If bilinear form $b(u, v)$, with $M := \|b\|$ satisfies the discrete inf-sup condition with constant γ_h,

$$\sup_{v_h \in V_h} \frac{|b(u, v)|}{\|v_h\|_V} \geq \gamma_h \|u_h\|_U \ ,$$

then the Galerkin error satisfies the bound

$$\|u_h - u\|_U \leq \frac{M}{\gamma_h} \inf_{w_h \in U_h} \|w_h - u\|_U \ .$$

Optimal test function realize the supremum guaranteeing that $\gamma_h \geq \gamma$.

Energy Norm

If we use the energy norm, $\|u\|_E := \|Bu\|_{V'}$ in the error estimate, then $M = \gamma = 1$. Babuška's theorem implies that the minimum residual method is the most stable Petrov-Galerkin method (assuming exact optimal test functions).

Overview of DPG

Other Features

THE UNIVERSITY OF TEXAS AT AUSTIN

Discontinuous Petrov-Galerkin

- Continuous test space produces global solve for optimal test functions
- Discontinuous test space results in an embarrassingly parallel solve

Hermitian Positive Definite Stiffness Matrix

Property of all minimum residual methods

$$b(u_h, v_{\delta u_h}) = (v_{u_h}, v_{\delta u_h})_V = \overline{(v_{\delta u_h}, v_{u_h})_V} = \overline{b(\delta u_h, v_{u_h})}$$

Error Representation Function

Energy norm of Galerkin error (residual) can be computed without exact solution

$$\|u_h - u\|_E = \|B(u_h - u)\|_{V'} = \|Bu_h - l\|_{V'} = \left\|R_V^{-1}(Bu_h - l)\right\|_V$$

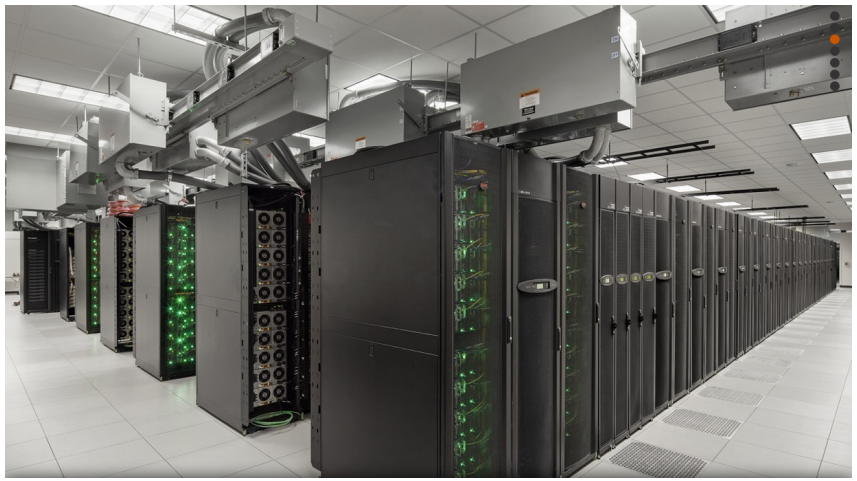
Overview of DPG

High Performance Computing

THE UNIVERSITY OF TEXAS AT AUSTIN

Eliminates human intervention

- Stability
- Robustness
- Adaptivity
- Automaticity
- Compute intensive
- Embarrassingly parallel local solves
- Factorization recyclable
- Low communication
- SPD stiffness matrix
- Multiphysics



Stampede Supercomputer at TACC

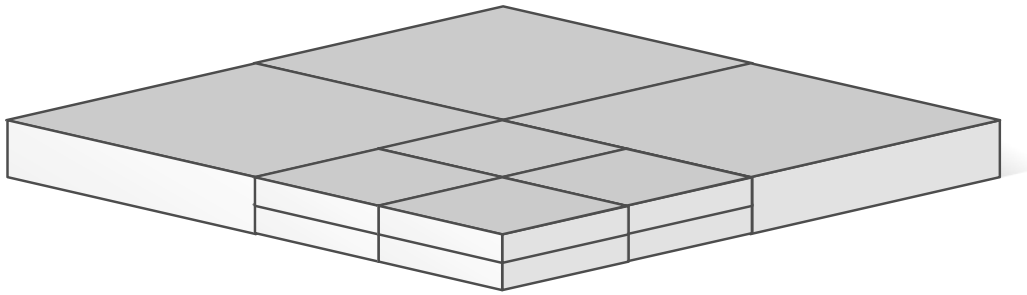
Mira Supercomputer at Argonne

Space-Time DPG

Extending DPG to Transient Problems

- Time stepping techniques are not ideally suited to highly adaptive grids
- Space-time FEM proposed as a solution
  - ✓ Unified treatment of space and time
  - ✓ Local space-time adaptivity (local time stepping)
  - ✓ Parallel-in-time integration
  - ✗ Spatially stable FEM methods may not be stable in space-time
  - ✗ Need to support higher dimensional problems
- DPG provides necessary stability and adaptivity

space-time slab



spatial mesh

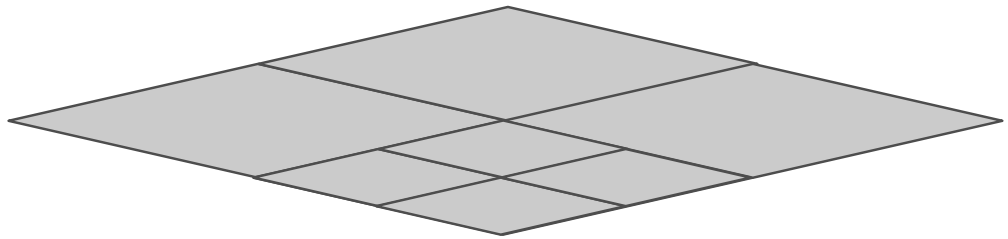


Table of Contents

THE UNIVERSITY OF TEXAS AT AUSTIN

Space-Time DPG for Convection-Diffusion

THE UNIVERSITY OF TEXAS AT AUSTIN

Space-Time Divergence Form

Equation is parabolic in space-time.

$$\frac{\partial u}{\partial t} + \boldsymbol{\beta} \cdot \nabla u - \epsilon \Delta u = f$$

This is just a composition of a constitutive law and conservation of mass.

$$\boldsymbol{\sigma} - \epsilon \nabla u = 0$$

$$\frac{\partial u}{\partial t} + \nabla \cdot (\boldsymbol{\beta} u - \boldsymbol{\sigma}) = f$$

We can rewrite this in terms of a space-time divergence.

$$\frac{1}{\epsilon}\boldsymbol{\sigma} - \nabla u = 0$$

$$\nabla_{xt} \cdot \begin{pmatrix} \boldsymbol{\beta} u - \boldsymbol{\sigma} \\ u \end{pmatrix} = f$$

Space-Time DPG for Convection-Diffusion

Ultra-Weak Formulation with Discontinuous Test Functions

Multiply by test function and integrate by parts over space-time element K.

$$\left(\frac{1}{\epsilon}\boldsymbol{\sigma}, \boldsymbol{\tau}\right)_K + (u, \nabla \cdot \boldsymbol{\tau})_K - \langle \hat{u}, \boldsymbol{\tau} \cdot \boldsymbol{n}_x \rangle_{\partial K} = 0$$

$$-\left(\begin{pmatrix} \boldsymbol{\beta} u - \boldsymbol{\sigma} \\ u \end{pmatrix}, \nabla_{xt} v\right)_K + \langle \hat{t}, v \rangle_{\partial K} = f$$

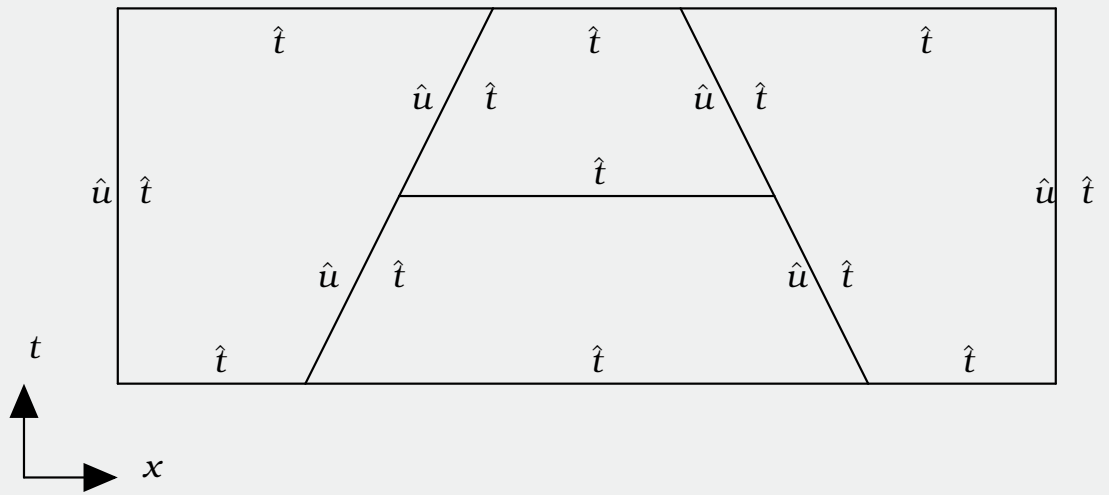
where

$$\hat{u} := \mathrm{tr}(u)$$

$$\hat{t} := \mathrm{tr}(\boldsymbol{\beta} u - \boldsymbol{\sigma}) \cdot \boldsymbol{n}_x + \mathrm{tr}(u) \cdot n_t$$

- Trace $\hat{u}$ defined on spatial boundaries
- Flux $\hat{t}$ defined on all boundaries

Support of Trace Variables



Space-Time Convection-Diffusion

Robust Norms

Bilinear form with group variables:

$$b\left(\left(u,\hat{u}\right),v\right)=\left(u,A_h^*v\right)_{L^2(\Omega_h)}+\langle\widehat{u},[\![v]\!]\rangle_{\Gamma_h}$$

For conforming v^* satisfying $A^*v^*=u$

$$\|u\|^2_{L^2(\Omega_h)}=b(u,v^*)=\frac{b(u,v^*)}{\|v^*\|_V}\|v^*\|_V$$

$$\leq \sup_{v^*\neq 0}\frac{|b(u,v^*)|}{\|v^*\|}\|v^*\|=\|u\|_E\|v^*\|_V$$

Necessary robustness condition:

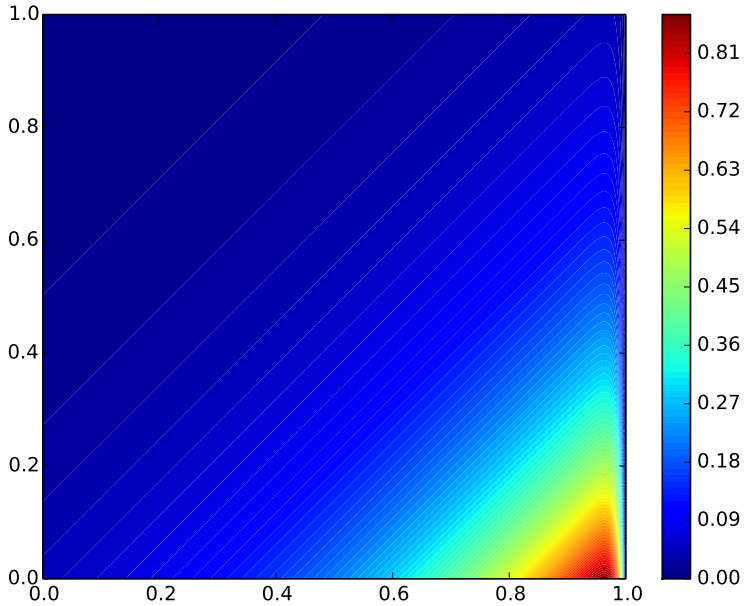
$$\|v^*\|_V\lesssim\|u\|_{L^2(\Omega_h)}$$
$$\Rightarrow\|u\|_{L^2(\Omega_h)}\lesssim\|u\|_E$$

Analytical Solution

$$u=e^{-lt}(e^{\lambda_1(x-1)}-e^{\lambda_2(x-1)})$$

$$\lambda_{1,2}=\frac{-1\pm\sqrt{1-4l\epsilon}}{-2\epsilon}$$

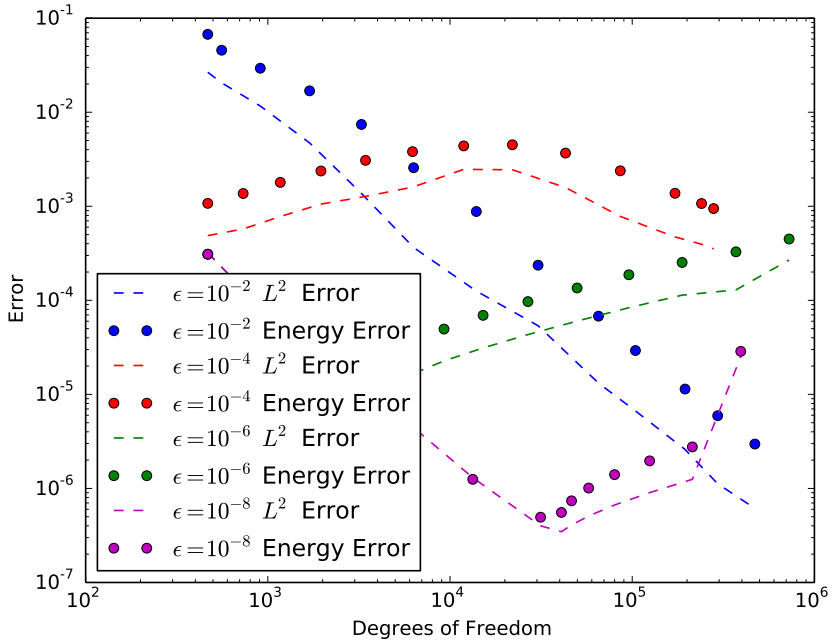
where $l=3,\quad \epsilon=10^{-2}$

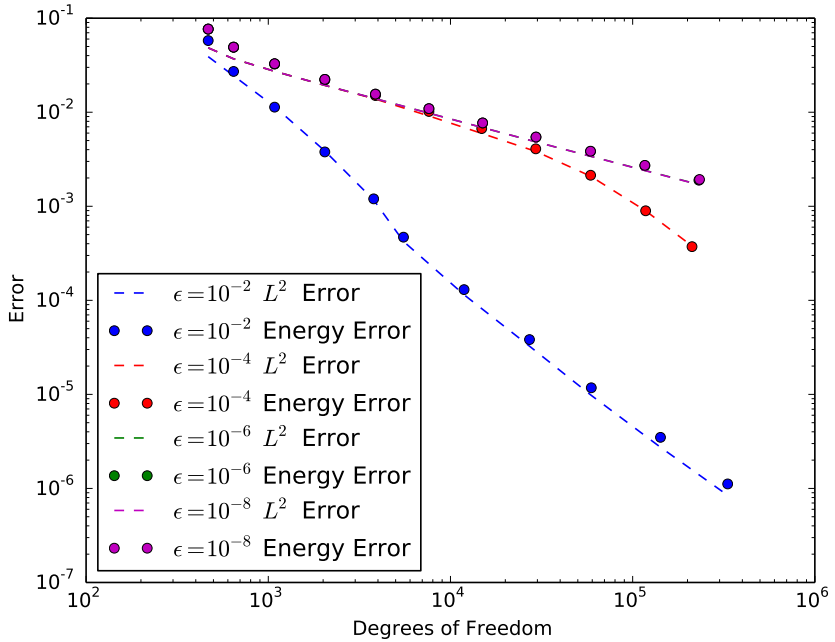


Space-Time Convection-Diffusion

Robust Norms

A norm should be: bounded by $\|u\|_{L^2(\Omega_h)}$, have good conditioning, not produce boundary layers in the optimal test function.





$$\|(v,\tau)\|^2 = \left\|\nabla\cdot\tau - \tilde{\beta}\cdot\nabla_{xt} v\right\|^2 + \left\|\frac{1}{\epsilon}\tau + \nabla v\right\|^2 + \|v\|^2 + \|\tau\|^2$$

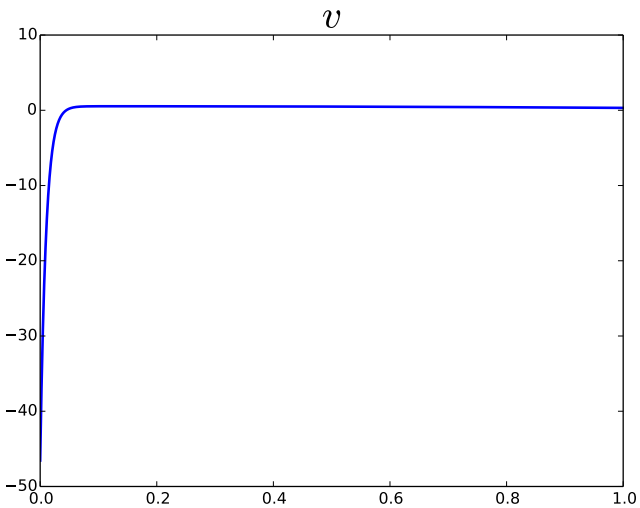
$$\|(v,\tau)\|^2 = \left\|\nabla\cdot\tau - \tilde{\beta}\cdot\nabla_{xt} v\right\|^2 + \min\left(\frac{1}{h^2},\frac{1}{\epsilon}\right)\|\tau\|^2 + \epsilon\|\nabla v\|^2 + \|\beta\cdot\nabla v\|^2 + \|v\|^2$$

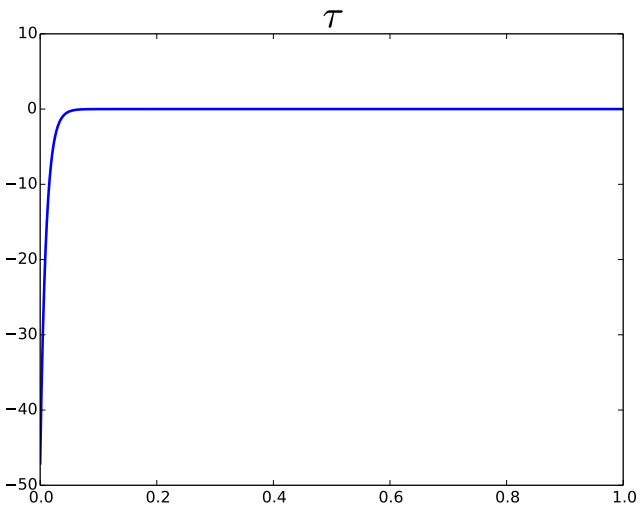
Space-Time Convection-Diffusion

THE UNIVERSITY OF TEXAS AT AUSTIN

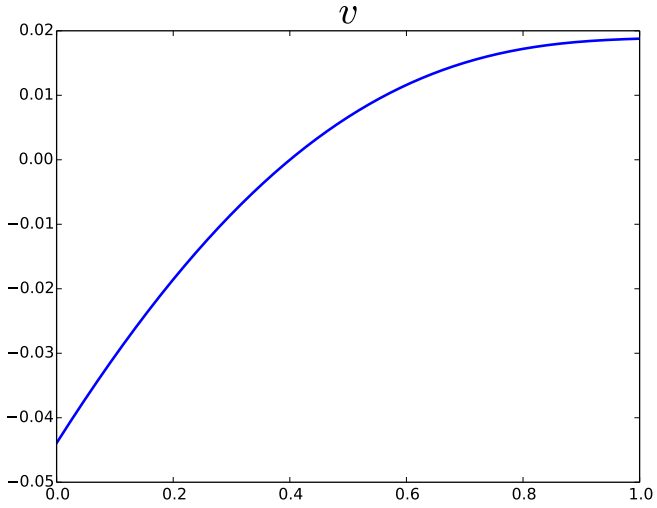
Ideal Optimal Shape Functions

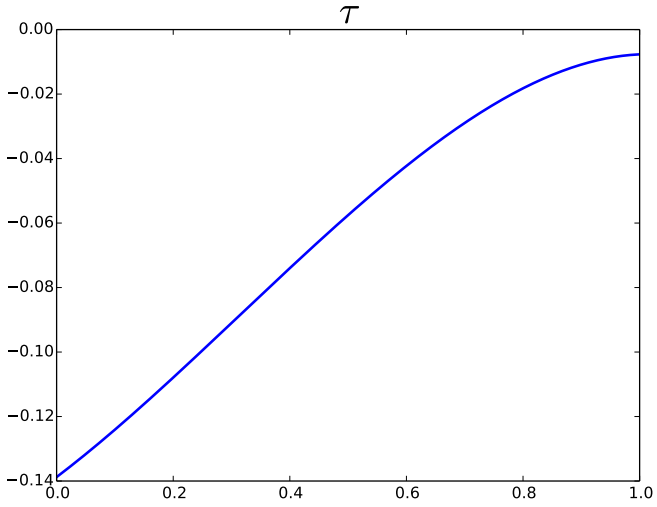
Graph Norm





Robust Norm

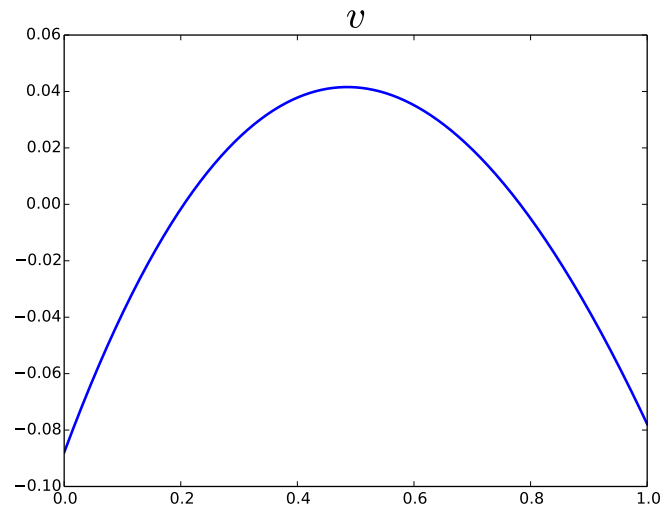


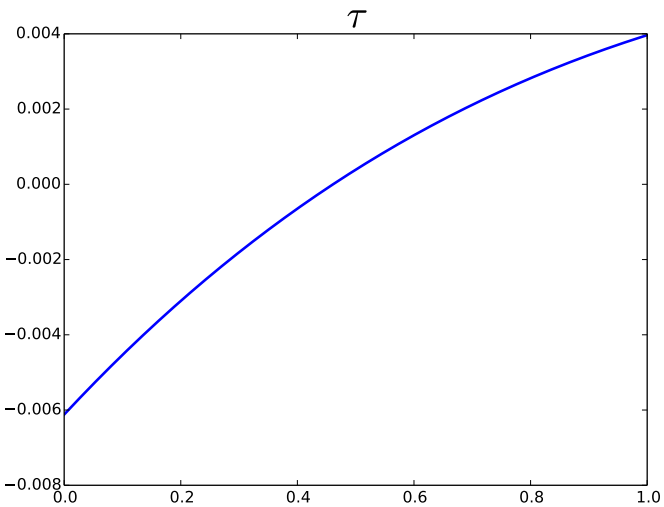


Space-Time Convection-Diffusion

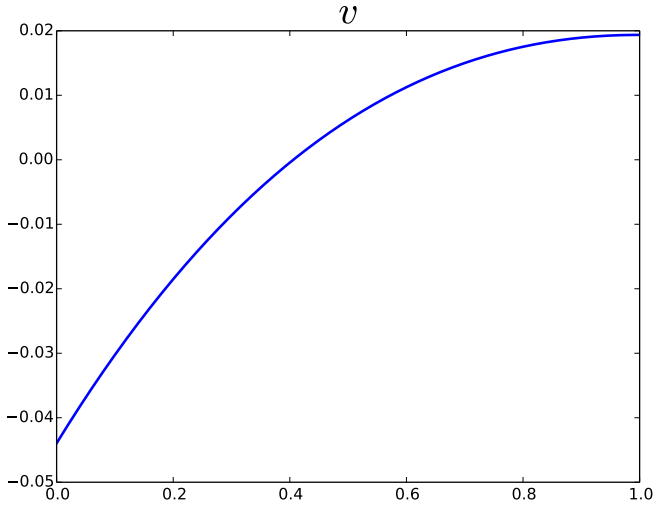
Approximated ($p = 3$) Optimal Shape Functions

Graph Norm





Robust Norm



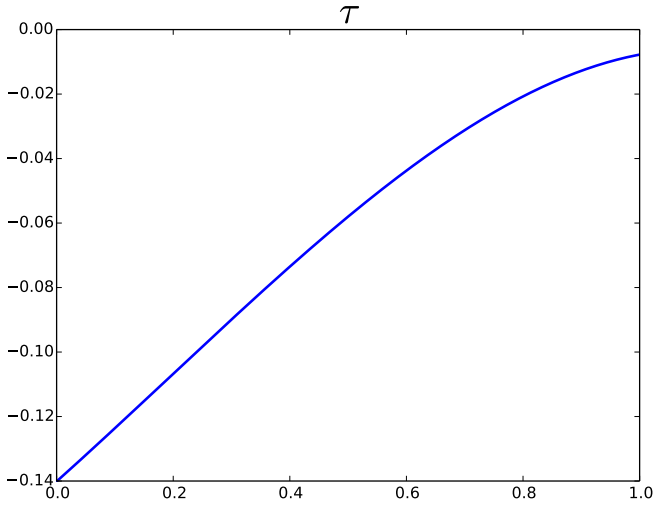


Table of Contents

THE UNIVERSITY OF TEXAS AT AUSTIN

Camellia: DPG for the Masses

THE UNIVERSITY OF TEXAS AT AUSTIN

Overview

Design Goal

Make DPG research and experimentation as simple as possible, while maintaining computational efficiency and scalability.

Built on Trilinos (Teuchos, Intrepid, Shards, Epetra, etc).

Mature support for:

- Rapid specification of DPG variational forms, inner products, etc.
- Distributed computation of stiffness matrix
- 1D - 3D geometries
- Curvilinear elements
- h- and p-refinements (anisotropic in h)

Experimental support for:

- Space-time computations
- Iterative solvers (tested up to 32,768 cores)

Convection-Diffusion in Three Slides

Building the Bilinear Form

```
VarFactory vf;
//fields:
VarPtr u = vf.fieldVar("u", L2);
VarPtr sigma = vf.fieldVar("sigma", VECTOR_L2);

// traces:
VarPtr u_hat = vf.traceVar("u_hat");
VarPtr t_n = vf.fluxVar("t_n");

// test:
VarPtr v = vf.testVar("v", HGRAD);
VarPtr tau = vf.testVar("tau", HDIV);

double eps = .01;
FunctionPtr beta_x = Function::constant(1);
FunctionPtr beta_y = Function::constant(2);
FunctionPtr beta = Function::vectorize(beta_x, beta_y);

BFPtr bf = Teuchos::rcp( new BF(vf) );

bf->addTerm((1/eps) * sigma, tau);
bf->addTerm(u, tau->div());
bf->addTerm(-u_hat, tau->dot_normal());

bf->addTerm(sigma - beta * u, v->grad());
bf->addTerm(t_n, v);

RHSPtr rhs = RHS::rhs();
```

Find $u \in L^2(\Omega_h)$, $\boldsymbol{\sigma} \in \boldsymbol{L}^2(\Omega_h)$, $\hat{u} \in H^{\frac{1}{2}}(\Gamma_h)$, $\hat{t}_n \in H^{-\frac{1}{2}}(\Gamma_h)$ such that

$$\frac{1}{\epsilon}(\boldsymbol{\sigma}, \boldsymbol{\tau}) + (u, \nabla \cdot \boldsymbol{\tau}) - \langle \hat{u}, \boldsymbol{\tau} \cdot \boldsymbol{n} \rangle - (\boldsymbol{\beta} u - \boldsymbol{\sigma}, \nabla v) + \langle \hat{t}_n, v \rangle = (f, v)$$

for all $v \in H^1(K)$, $\boldsymbol{\tau} \in \boldsymbol{H}(\text{div}, K)$,

where $\epsilon = 10^{-2}$, $\boldsymbol{\beta} = (1, 2)^T$ and

$$f = 0.$$

Convection-Diffusion in Three Slides

Boundary Conditions and Mesh

```
int k = 2;
int delta_k = 2;
MeshPtr mesh = MeshFactory::quadMesh(bf, k+1, delta_k);

BCPtr bc = BC::bc();

SpatialFilterPtr y_equals_one = SpatialFilter::matchingY(1.0);
SpatialFilterPtr y_equals_zero = SpatialFilter::matchingY(0);
SpatialFilterPtr x_equals_one = SpatialFilter::matchingX(1.0);
SpatialFilterPtr x_equals_zero = SpatialFilter::matchingX(0.0);

FunctionPtr zero = Function::zero();
FunctionPtr x = Function::xn(1);
FunctionPtr y = Function::yn(1);
bc->addDirichlet(t_n, y_equals_zero, -2 * (1-x));
bc->addDirichlet(t_n, x_equals_zero, -1 * (1-y));
bc->addDirichlet(u_hat, y_equals_one, zero);
bc->addDirichlet(u_hat, x_equals_one, zero);
```

Create a square mesh $[0,1] \times [0,1]$ with boundary conditions

- $\hat{t}_n = 2x - 2$ on $y = 0$
- $\hat{t}_n = x - 1$ on $x = 0$
- $\hat{u} = 0$ on $y = 1$
- $\hat{u} = 0$ on $x = 1$

Note

- Can subclass SpatialFilter to match any geometry
- Adding new mesh readers is straightforward

Convection-Diffusion in Three Slides

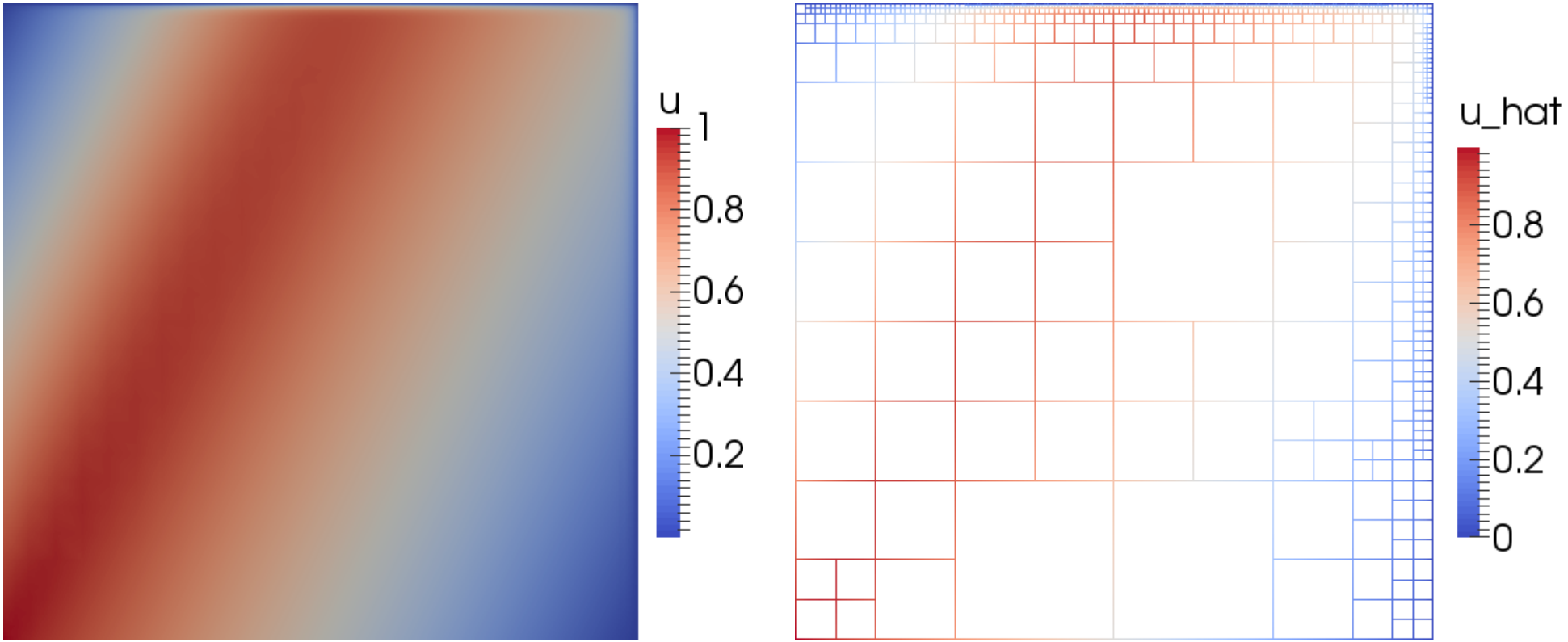
Test Norm, Solving, and Adaptivity

THE UNIVERSITY OF TEXAS AT AUSTIN

```
IPPtr ip = bf->graphNorm();

SolutionPtr soln = Solution::solution(mesh, bc, rhs, ip);

double threshold = 0.20;
RefinementStrategy refStrategy(soln, threshold);

int numRefs = 10;

ostringstream refName;
refName << "ConvectionDiffusion";
HDF5Exporter exporter(mesh,refName.str());

for (int refIndex=0; refIndex < numRefs; refIndex++) {
  soln->solve();

  double energyError = soln->energyErrorTotal();
  cout << "After " << refIndex << " refinements, energy error is " << energyError << endl;

  exporter.exportSolution(soln, vf, refIndex);

  if (refIndex != numRefs)
    refStrategy.refine();
}
```

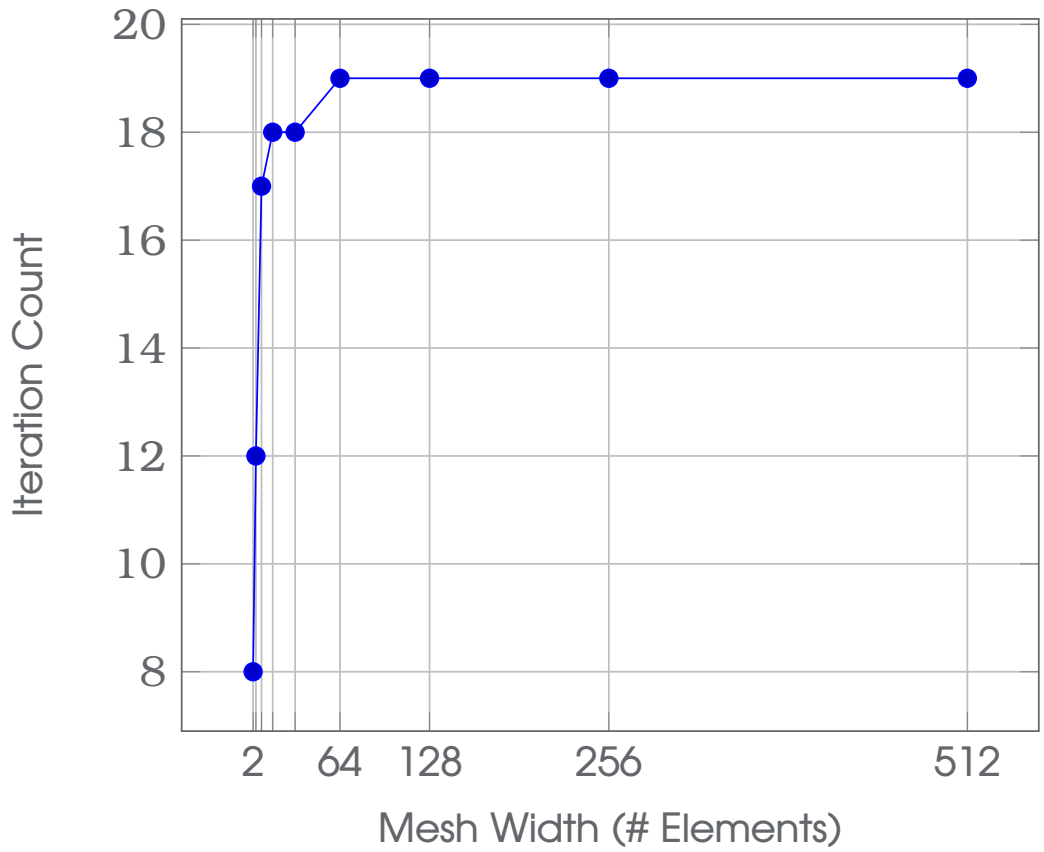
Convection-Diffusion in Three Slides

Computed Solution



Towards a Robust Iterative Solver

Poisson 1D, p-multigrid Preconditioners, $k = 16$

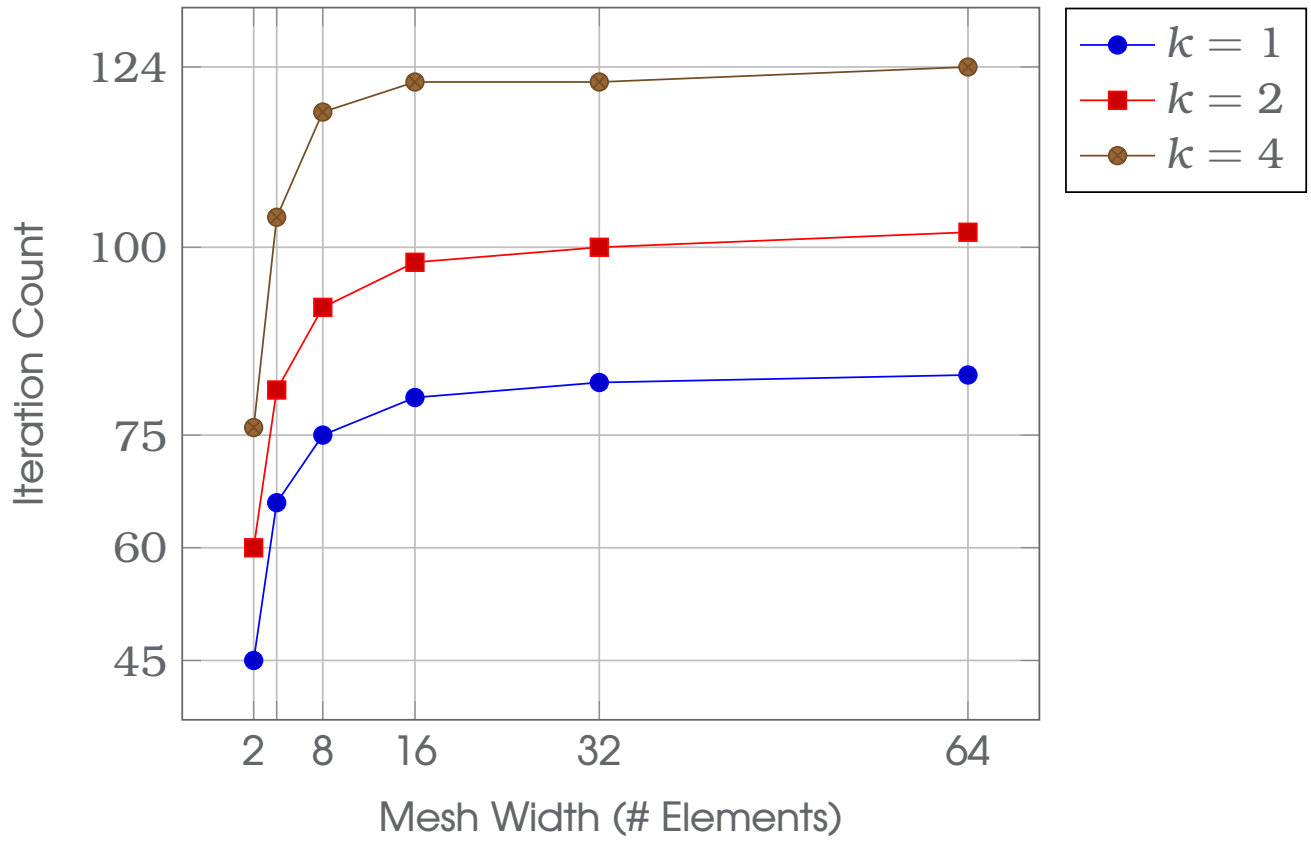


Poisson 1D: number of CG iterations to reduce error by a factor of 10^{10} using p-multigrid preconditioners with Schwarz smoother with 0 overlap for $k = 16$. The results for $k = 1, 2, 4$, and 8 are essentially identical.

Towards a Robust Iterative Solver

THE UNIVERSITY OF TEXAS AT AUSTIN

Stokes 2D, p-multigrid Preconditioners, Schwarz algebraic overlap 0

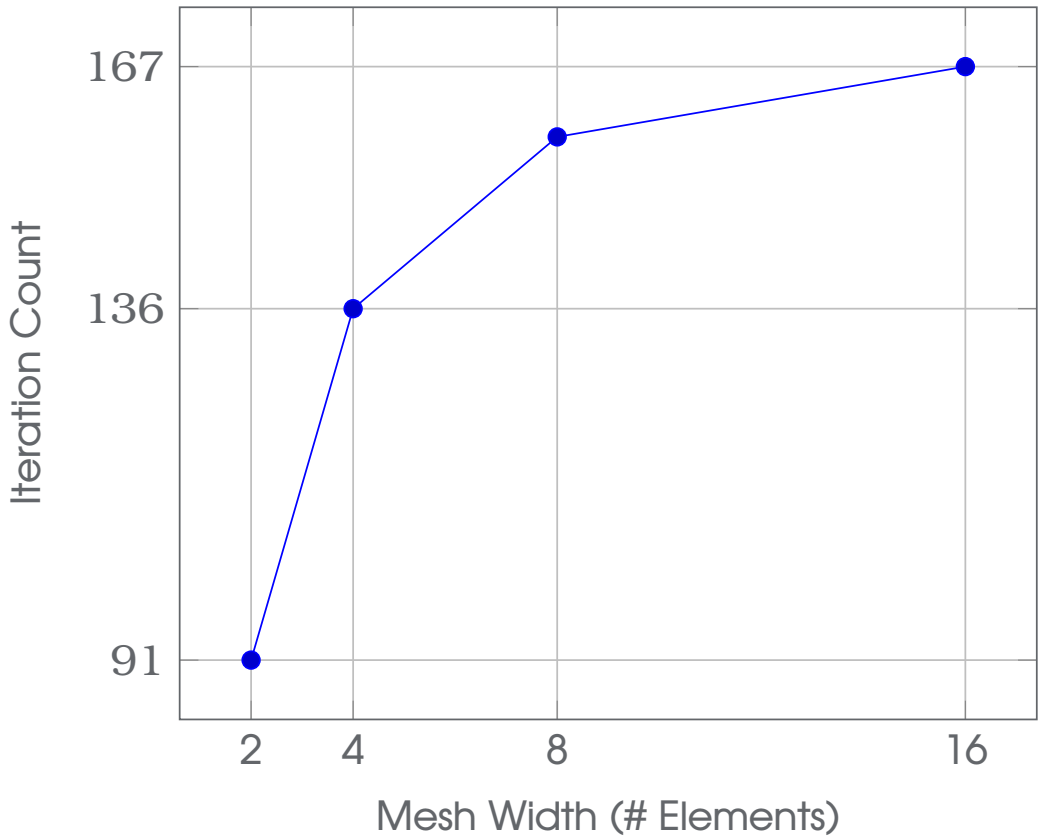


Stokes 2D: number of CG iterations to reduce error by a factor of 10^{10} using p-multigrid, algebraic Schwarz smoother with 0 overlap.

Towards a Robust Iterative Solver

THE UNIVERSITY OF TEXAS AT AUSTIN

Stokes 3D, p-multigrid Preconditioners, $k = 2$, Schwarz with Incomplete Cholesky



Stokes 3D: number of CG iterations to reduce error by a factor of 10^{10} using statically condensed system matrix with p-multigrid, algebraic Schwarz smoother, using Incomplete Cholesky factorization (fill ratio 5) for the Schwarz blocks.

Table of Contents

THE UNIVERSITY OF TEXAS AT AUSTIN

Space-Time Navier-Stokes

First Order System

Assuming Stokes hypothesis, ideal gas law, and constant viscosity:

$$\frac{1}{\mu}\mathbb{D} - \left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T\right) + \frac{2}{3}\nabla \cdot \boldsymbol{u}\mathbb{I} = 0$$

$$\frac{Pr}{C_p \mu}\boldsymbol{q} + \nabla T = 0$$

$$\nabla_{xt} \cdot \begin{pmatrix} \rho \boldsymbol{u} \\ \rho \end{pmatrix} = f_c$$

$$\nabla_{xt} \cdot \begin{pmatrix} \rho \boldsymbol{u} \otimes \boldsymbol{u} + \rho RT\mathbb{I} - \mathbb{D} \\ \rho \boldsymbol{u} \end{pmatrix} = \boldsymbol{f}_m$$

$$\nabla_{xt} \cdot \begin{pmatrix} \rho \boldsymbol{u}\left(C_v T + \frac{1}{2}\boldsymbol{u} \cdot \boldsymbol{u}\right) + \rho RT\boldsymbol{u} + \boldsymbol{q} - \boldsymbol{u} \cdot \mathbb{D} \\ \rho\left(C_v T + \frac{1}{2}\boldsymbol{u} \cdot \boldsymbol{u}\right) \end{pmatrix} = f_e\,,$$

Space-Time Navier-Stokes

Compact Notation

Conserved quantities

$$C_c := \rho$$

$$\boldsymbol{C}_m := \rho \boldsymbol{u}$$

$$C_e := \rho(C_v T + \frac{1}{2}\boldsymbol{u} \cdot \boldsymbol{u})$$

Euler fluxes

$$\boldsymbol{F}_c := \rho \boldsymbol{u}$$

$$\mathbb{F}_m := \rho \boldsymbol{u} \otimes \boldsymbol{u} + \rho R T \mathbb{I}$$

$$\boldsymbol{F}_e := \rho \boldsymbol{u} \left(C_v T + \frac{1}{2}\boldsymbol{u} \cdot \boldsymbol{u} \right) + \rho R T \boldsymbol{u}$$

Viscous fluxes

$$\boldsymbol{K}_c := \boldsymbol{0}$$

$$\mathbb{K}_m := \mathbb{D}$$

$$\boldsymbol{K}_e := -\boldsymbol{q} + \boldsymbol{u} \cdot \mathbb{D}$$

Viscous variables

$$\mathbb{M}_{\mathbb{D}} := \frac{1}{\mu}\mathbb{D}$$

$$\boldsymbol{M}_{\boldsymbol{q}} := \frac{Pr}{C_p \mu}\boldsymbol{q}$$

Viscous relations

$$\boldsymbol{G}_{\mathbb{D}} := 2\boldsymbol{u}$$

$$G_{\boldsymbol{q}} := -T$$

Space-Time Navier-Stokes

THE UNIVERSITY OF TEXAS AT AUSTIN

Define Group Variables

Group terms

$$C := \{C_c\,,\ \boldsymbol{C}_m\,,\ C_e\}$$
$$F := \{\boldsymbol{F}_c\,,\ \mathbb{F}_m\,,\ \boldsymbol{F}_e\}$$
$$K := \{\boldsymbol{K}_c\,,\ \mathbb{K}_m\,,\ \boldsymbol{K}_e\}$$
$$M := \{\mathbb{M}_{\mathbb{D}}\,,\ \boldsymbol{M}_{\boldsymbol{q}}\}$$
$$G := \{\boldsymbol{G}_{\mathbb{D}}\,,\ G_{\boldsymbol{q}}\}$$
$$f := \{f_c\,,\boldsymbol{f}_m\,,f_e\}$$

Group variables

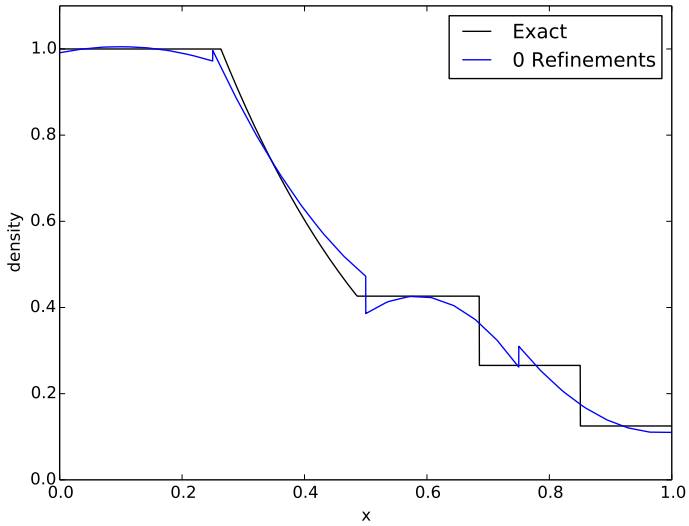
$$W := \{\rho\,,\ \boldsymbol{u}\,,\ T\}$$
$$\hat{W} := \left\{2\hat{\boldsymbol{u}}\,,\ -\hat{T}\right\}$$
$$\Sigma := \{\mathbb{D}, \boldsymbol{q}\}$$
$$\hat{t} := \{\hat{t}_e\,,\ \hat{\boldsymbol{t}}_m,\ ,\ \hat{t}_e\}$$
$$\Psi := \{\mathbb{S}\,,\ \boldsymbol{\tau}\}$$
$$V := \{v_c\,,\ \boldsymbol{v}_m,\ ,\ v_e\}\ .$$

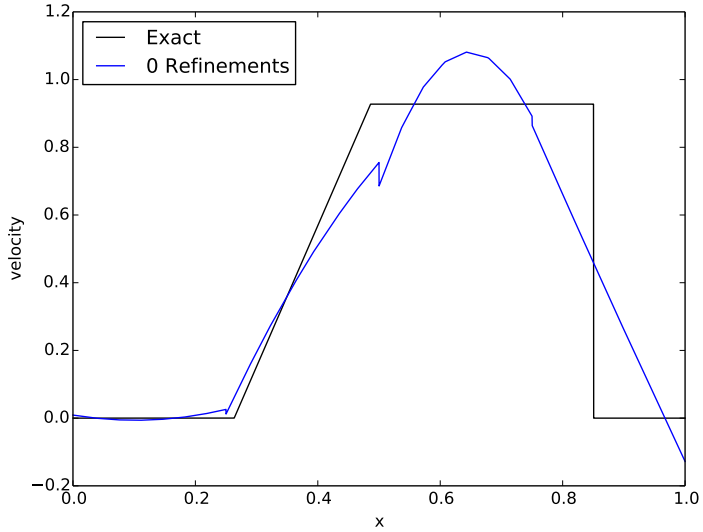
Navier-Stokes variational formulation is

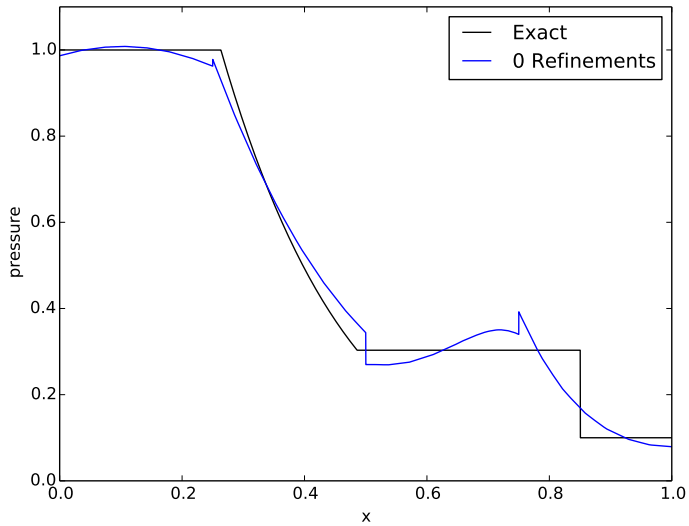
$$(M, \Psi) + (G, \nabla \cdot \Psi) - \left\langle \hat{W}, \Psi \cdot \boldsymbol{n}_x \right\rangle = 0$$
$$-\left(\begin{pmatrix} F - K \\ C \end{pmatrix}, \nabla_{xt} V\right) + \langle \hat{t}, V \rangle = (f, V)\ .$$

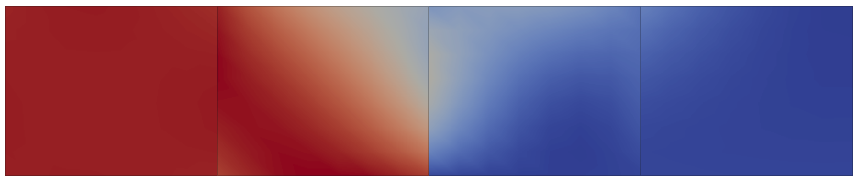
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



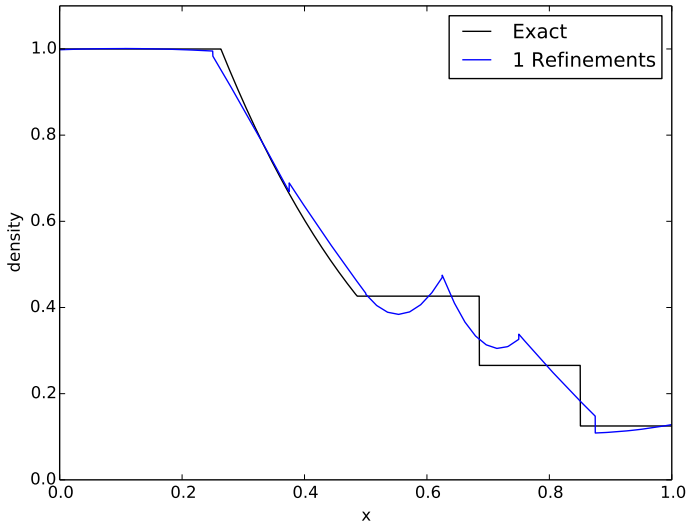


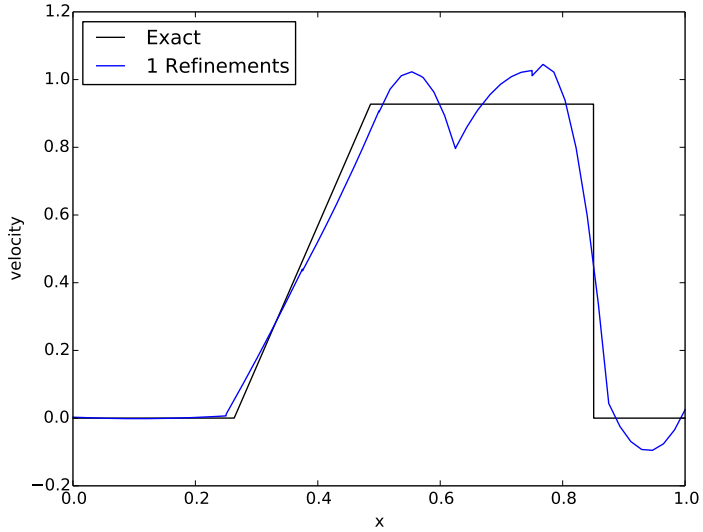


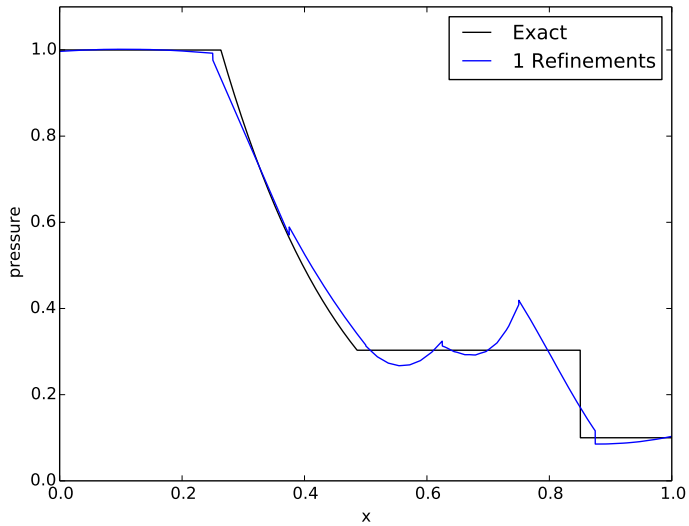


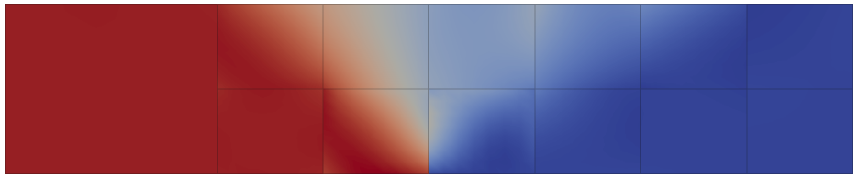
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



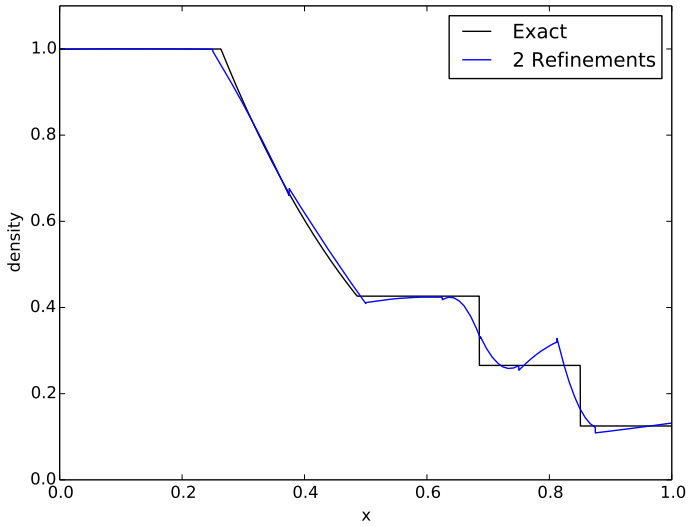


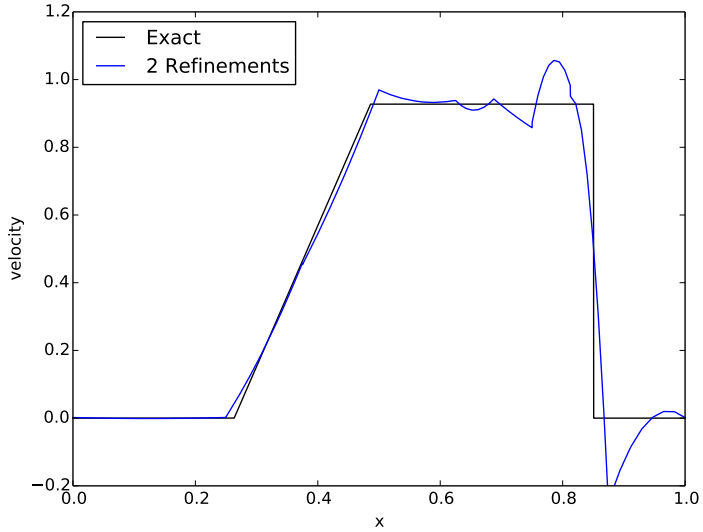


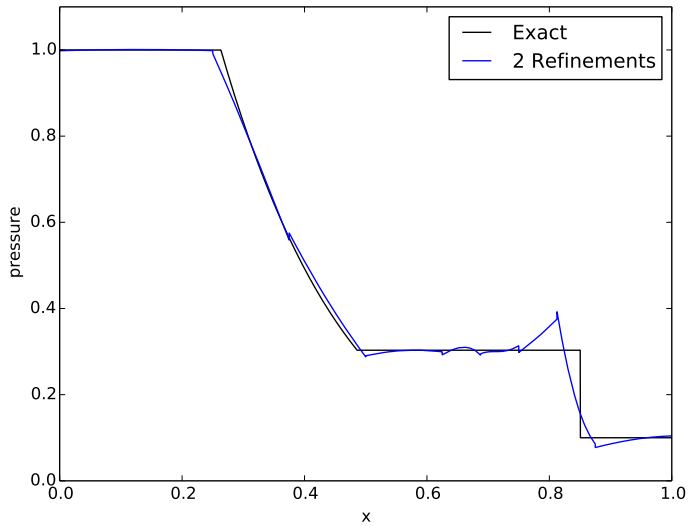


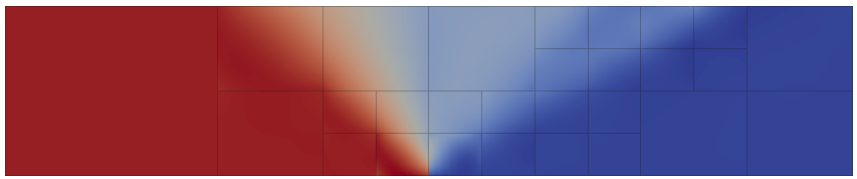
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



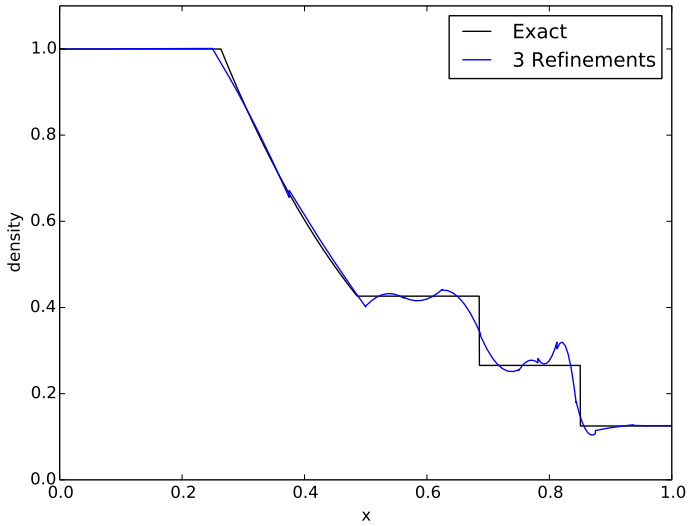


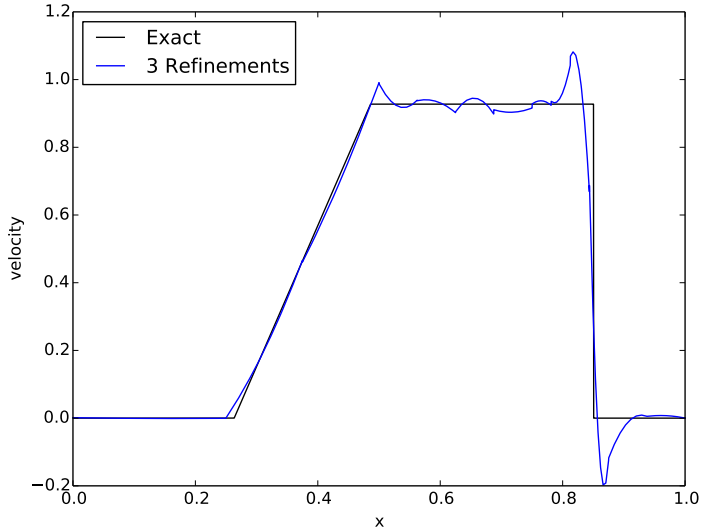


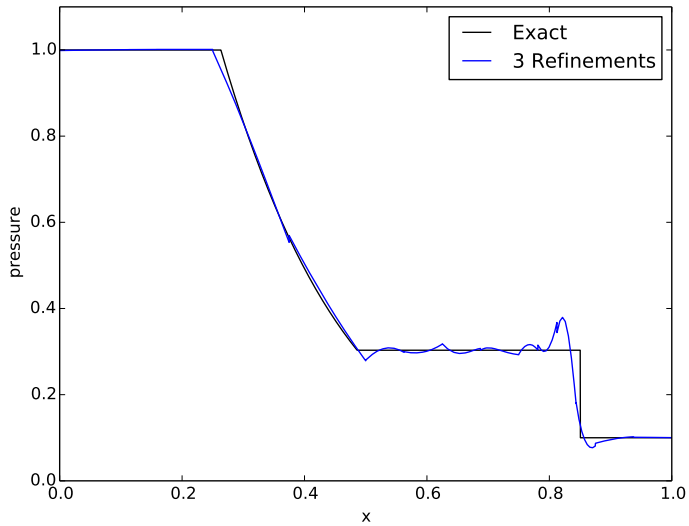


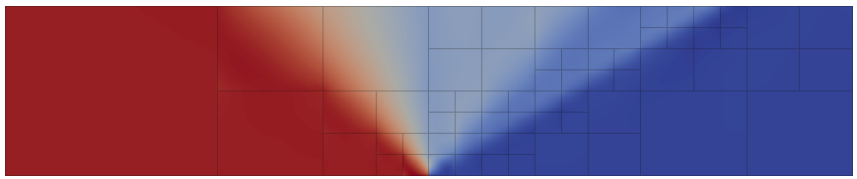
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



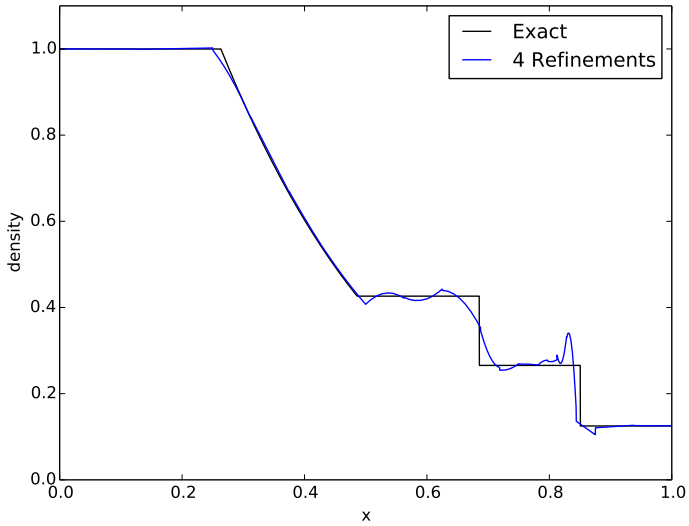


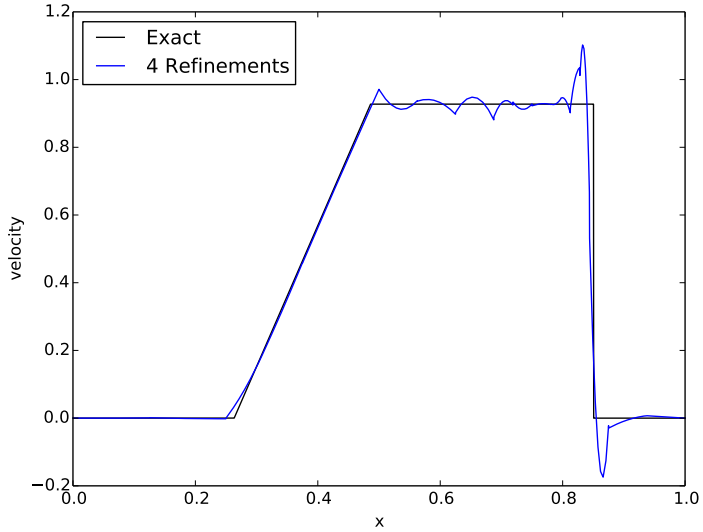


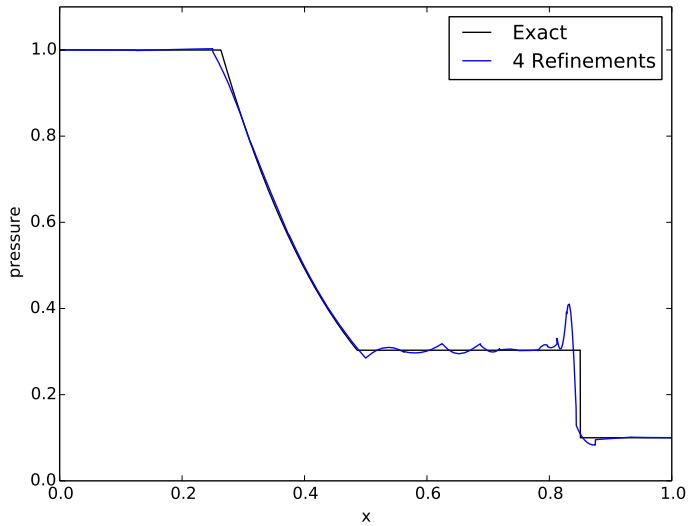


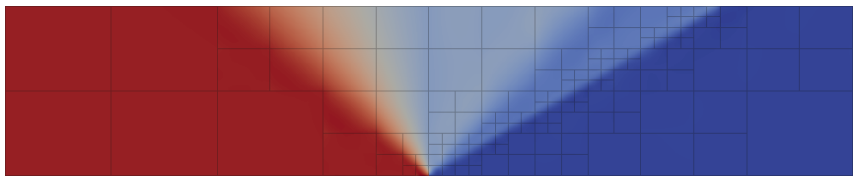
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



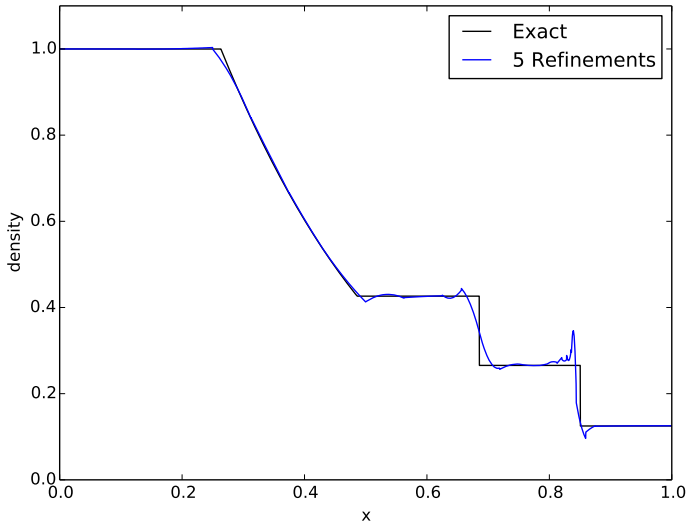


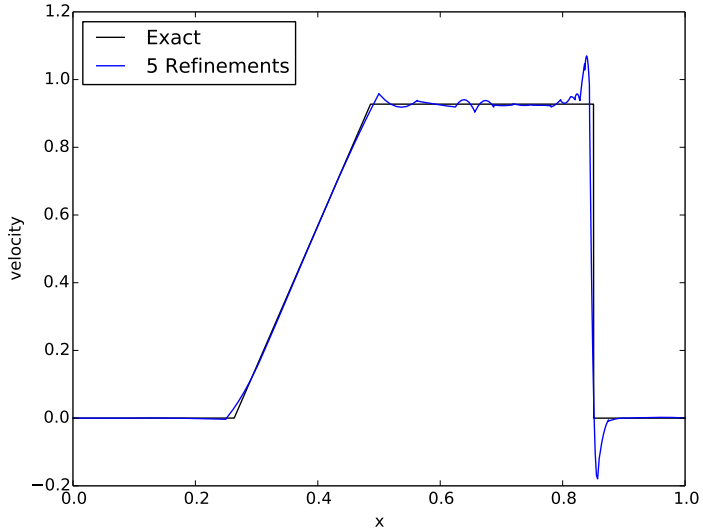


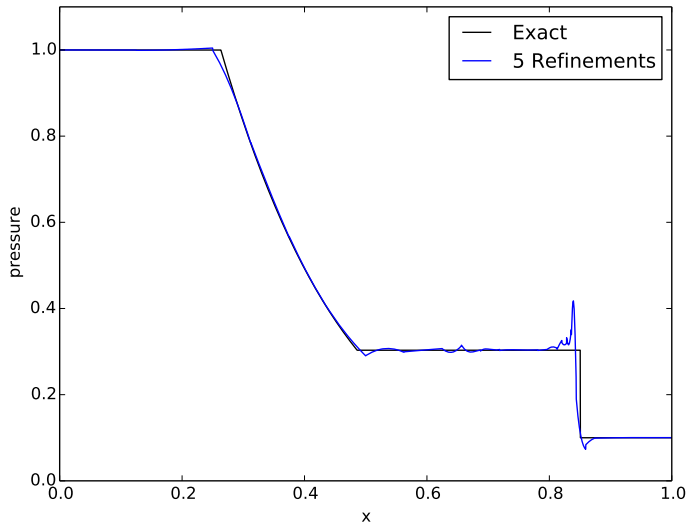


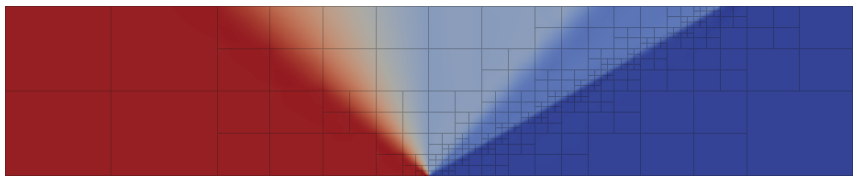
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



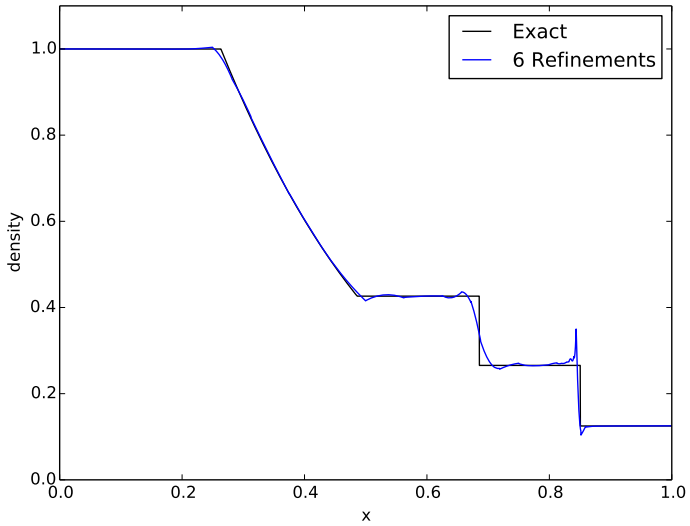


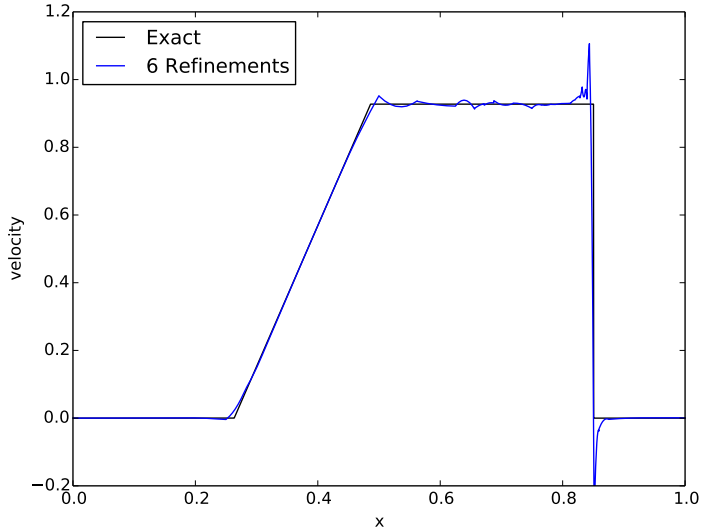


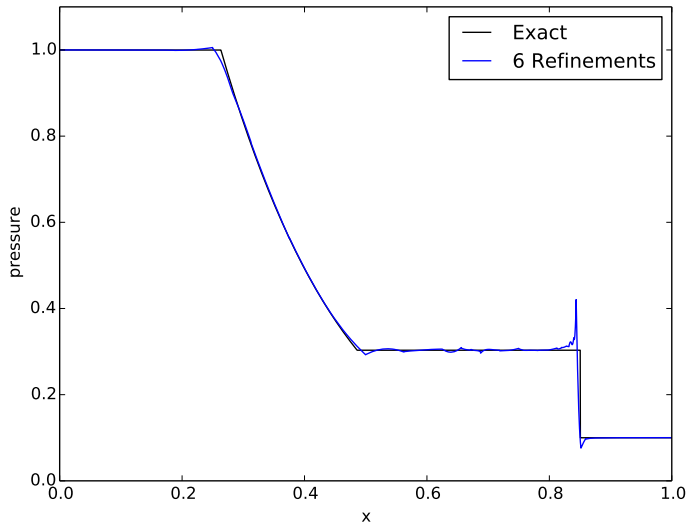


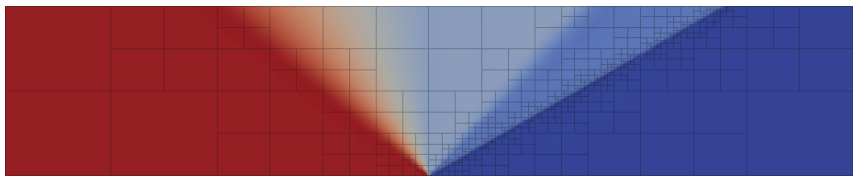
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



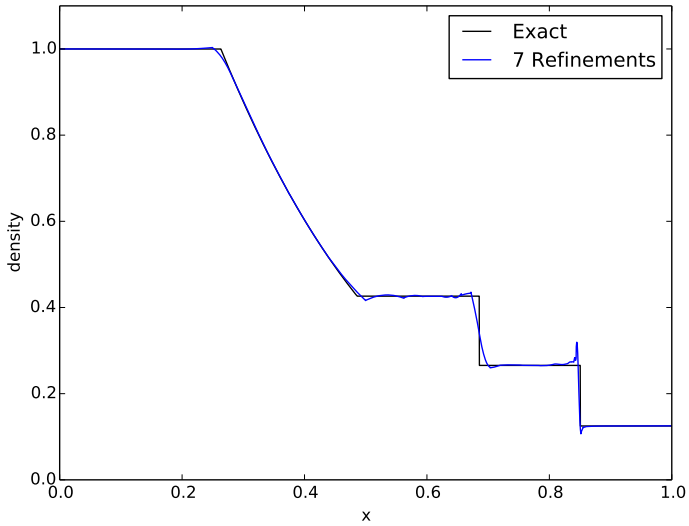


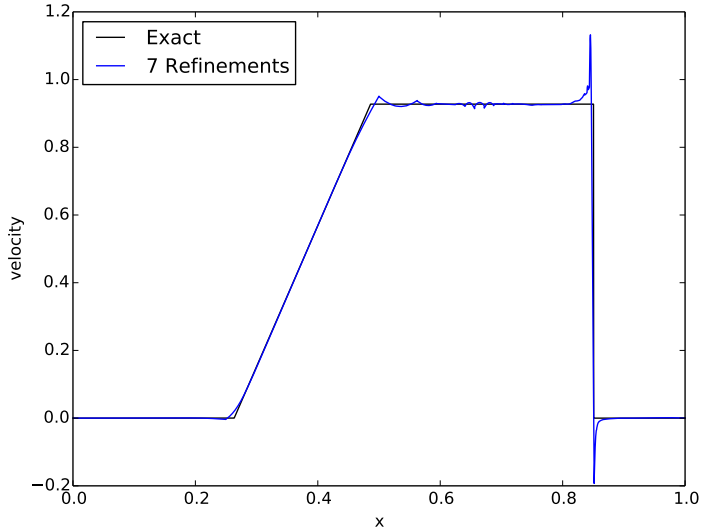


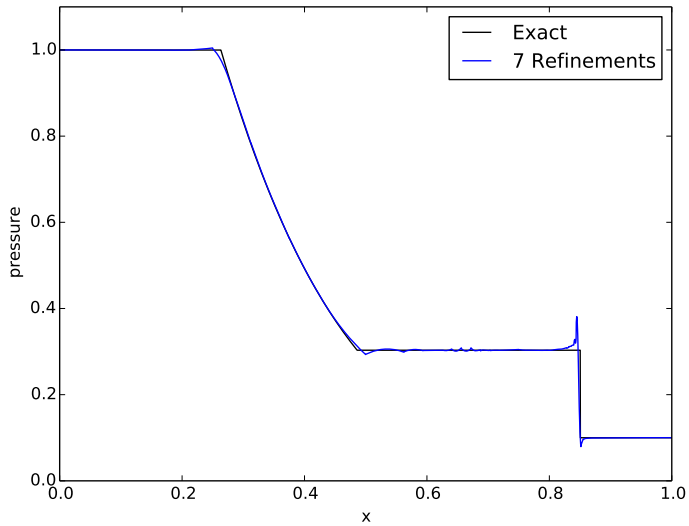


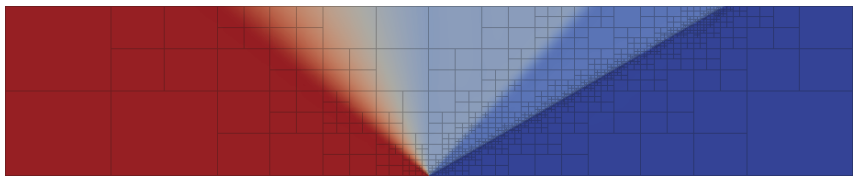
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



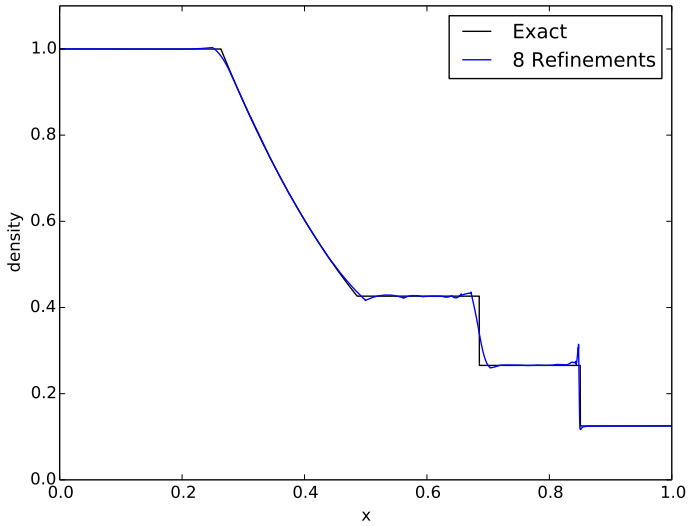


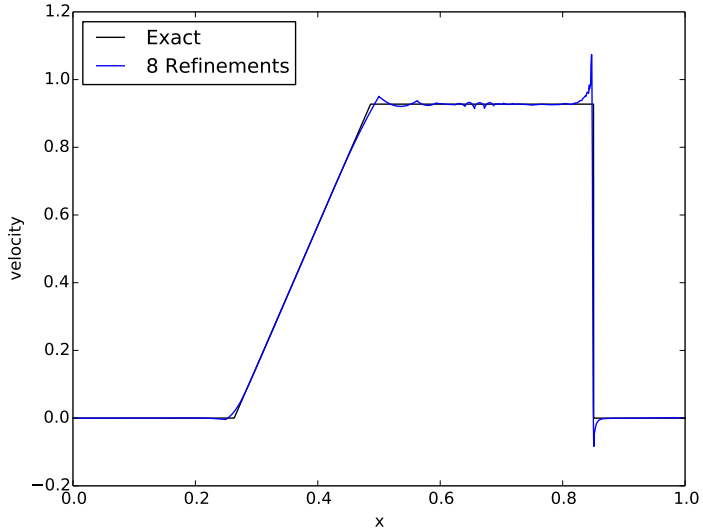


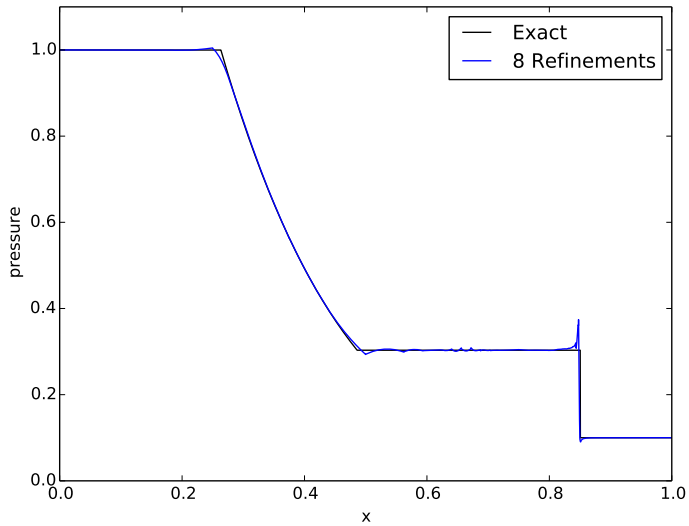


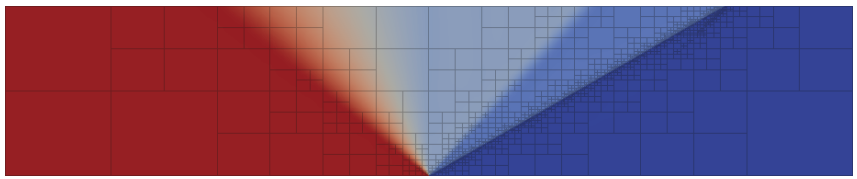
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



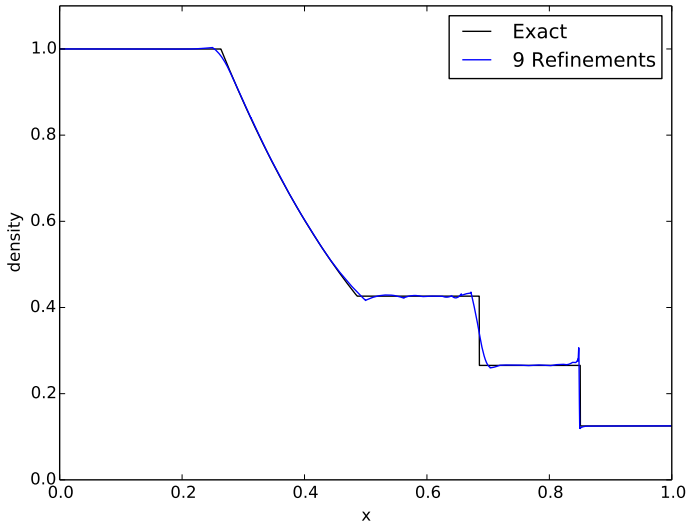


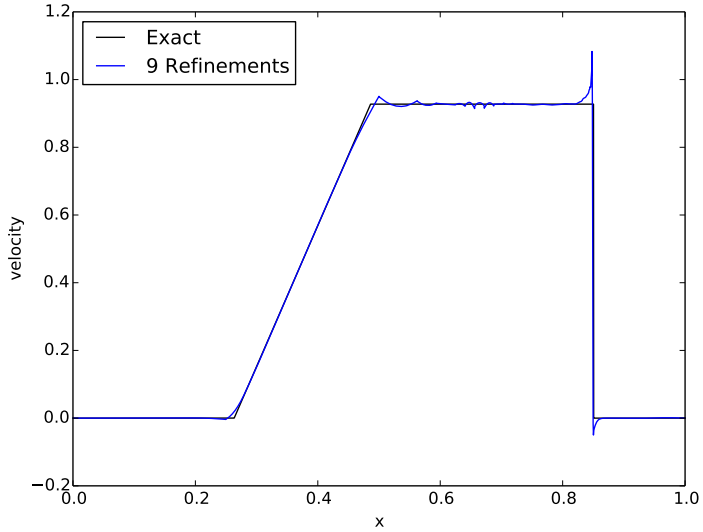


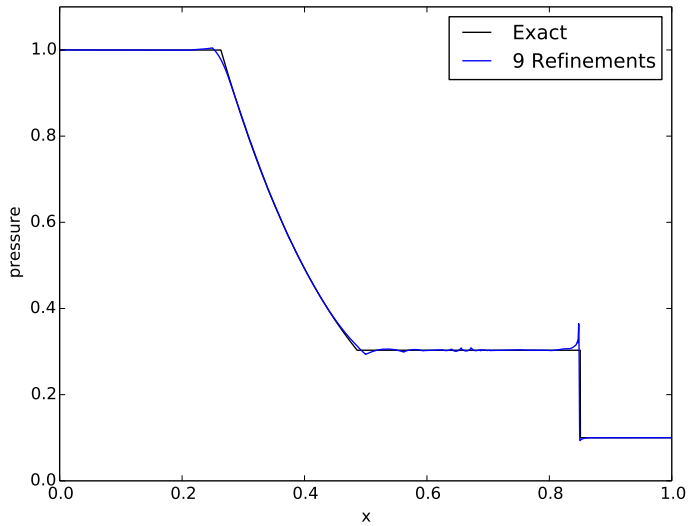


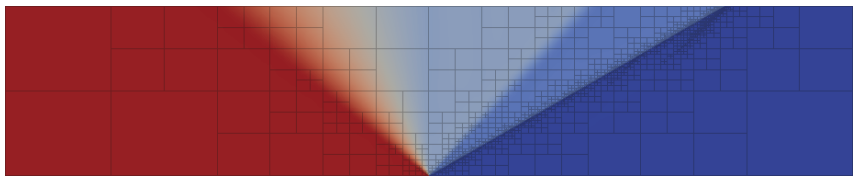
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



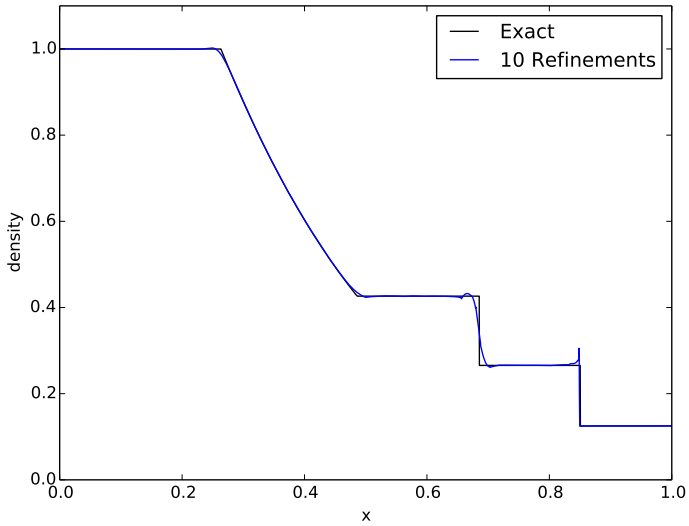


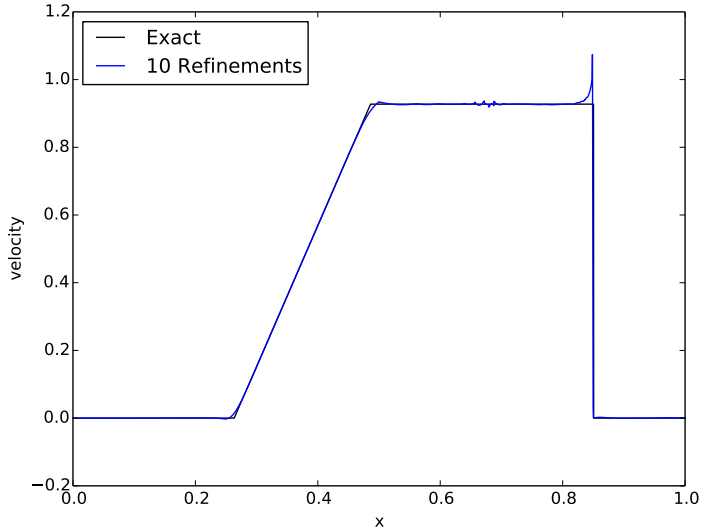


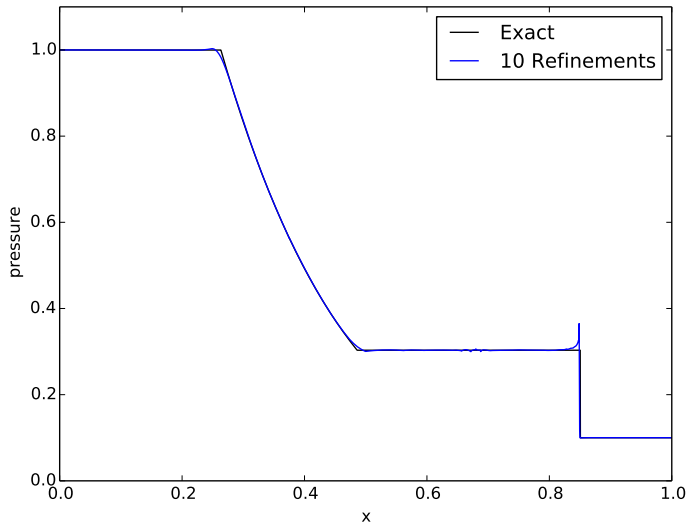


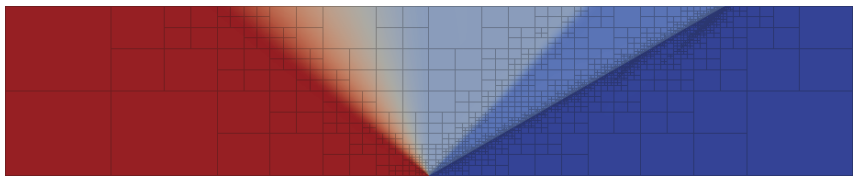
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



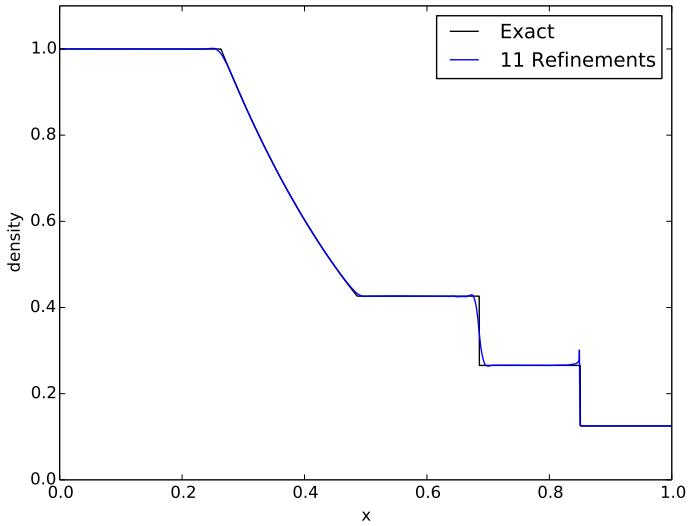


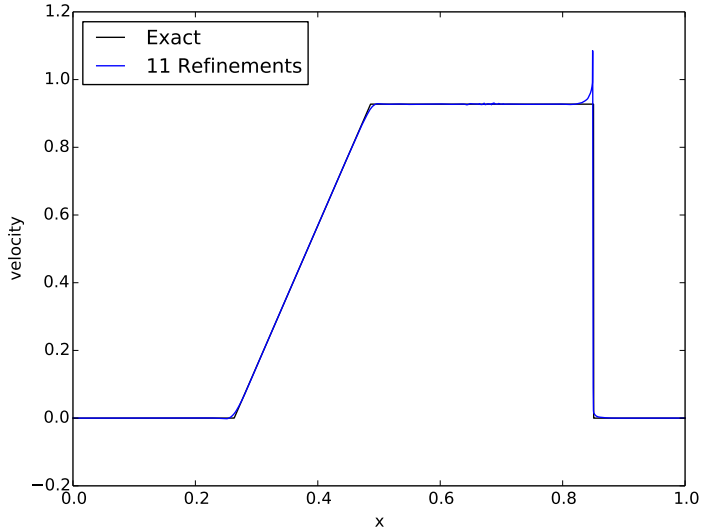


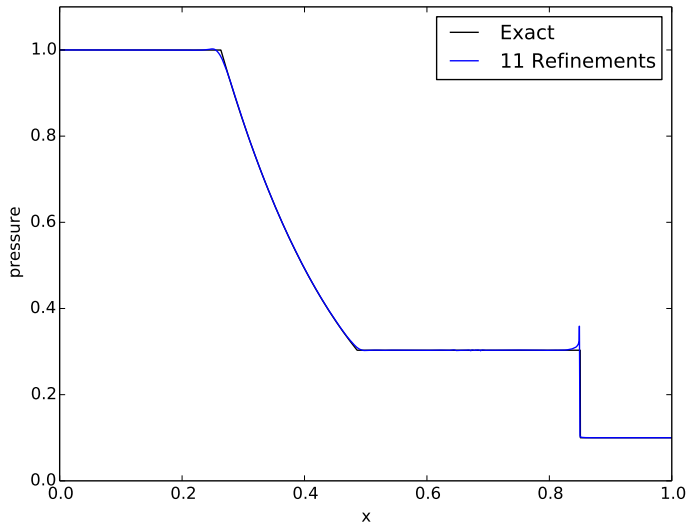


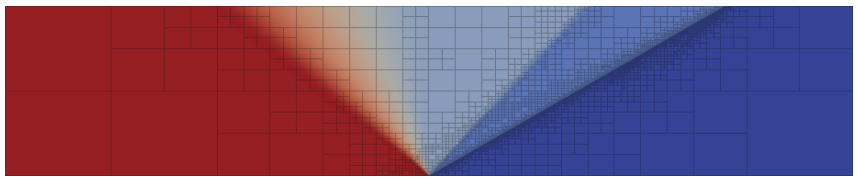
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



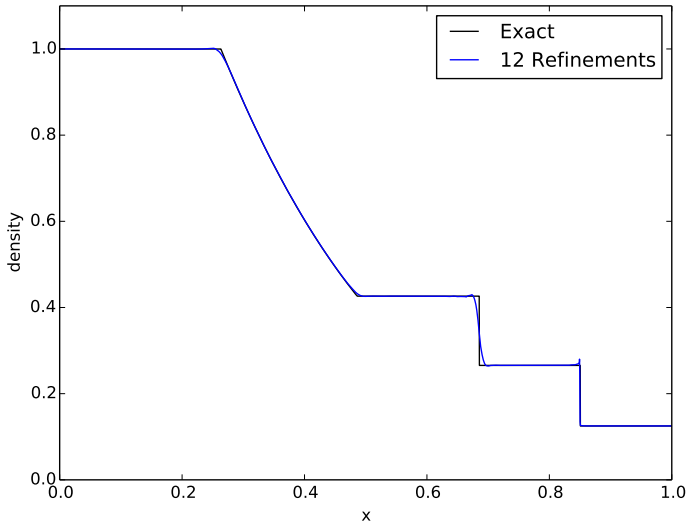


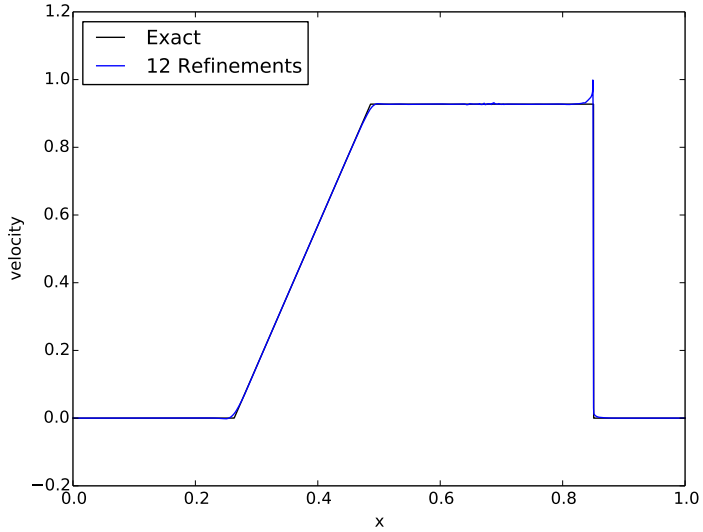


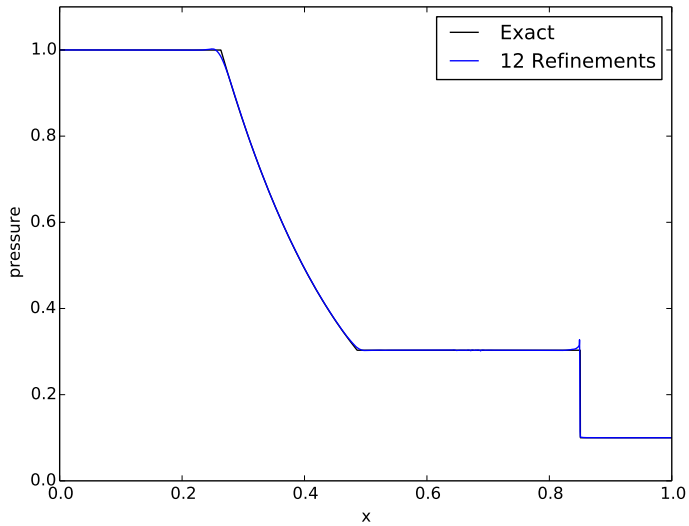


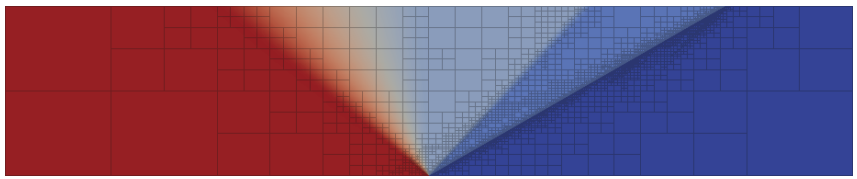
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



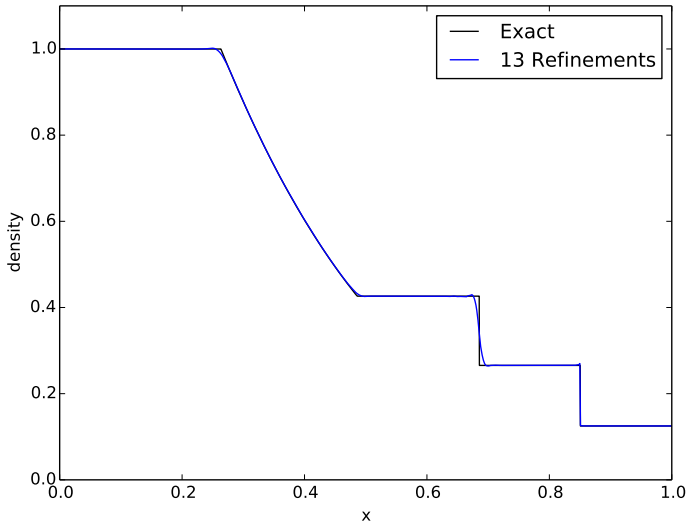


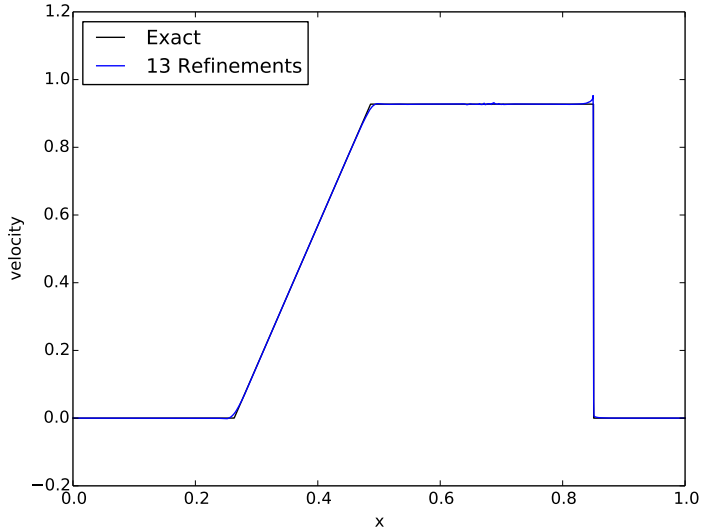


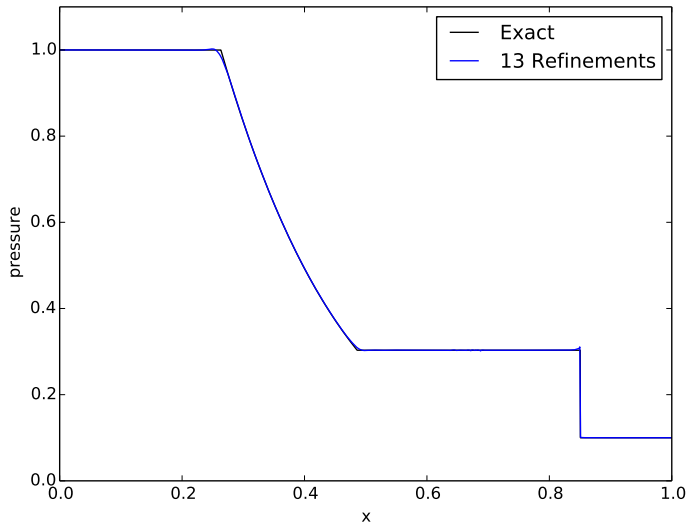


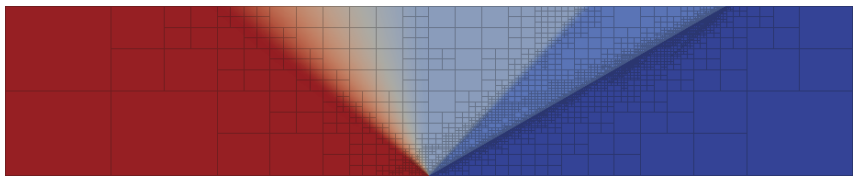
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$



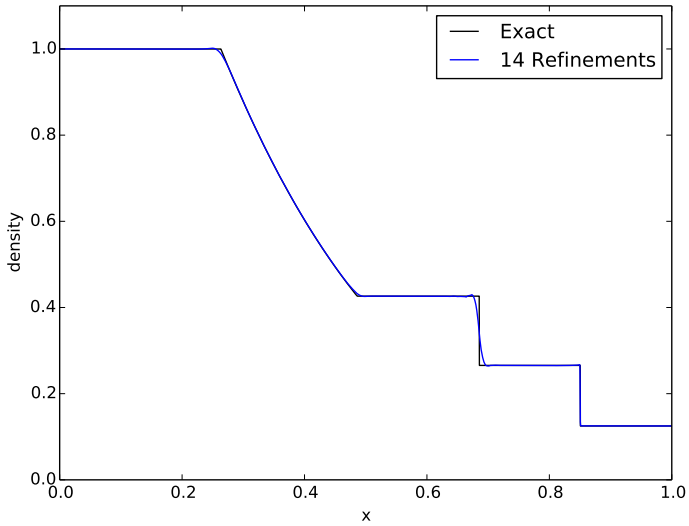


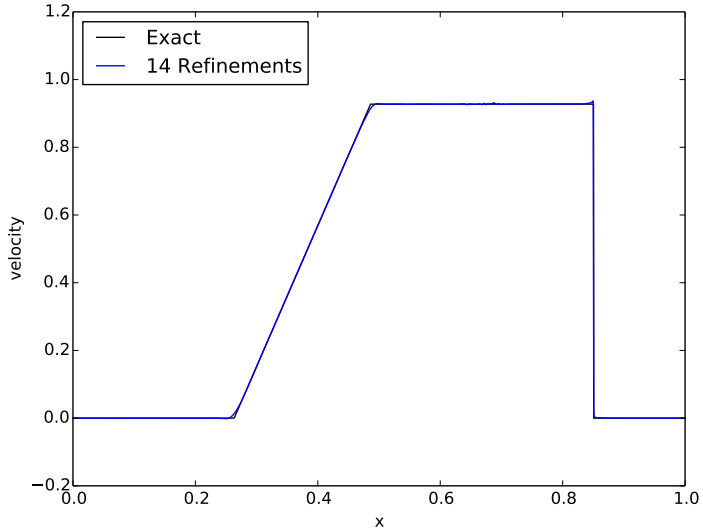


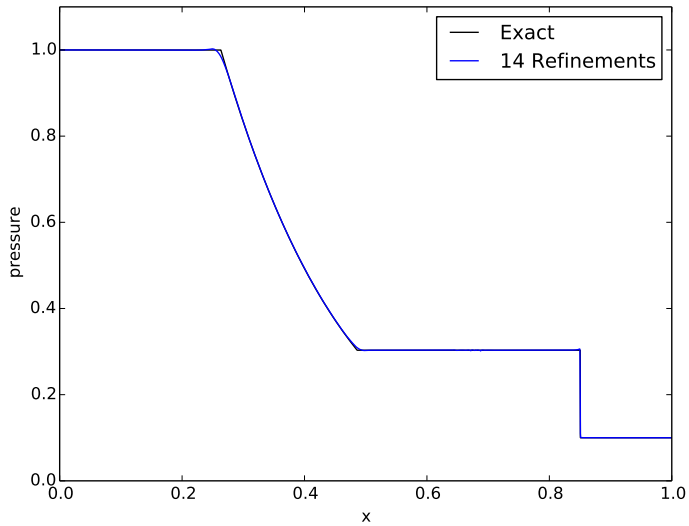


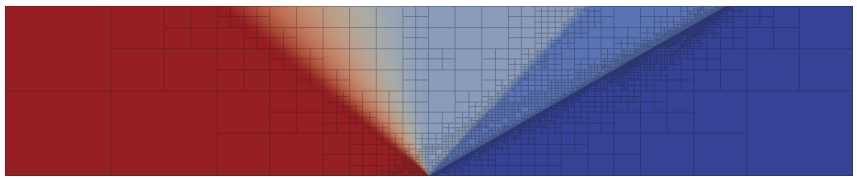
Compressible Navier-Stokes

Sod Shock Tube with $\mu = 10^{-5}$





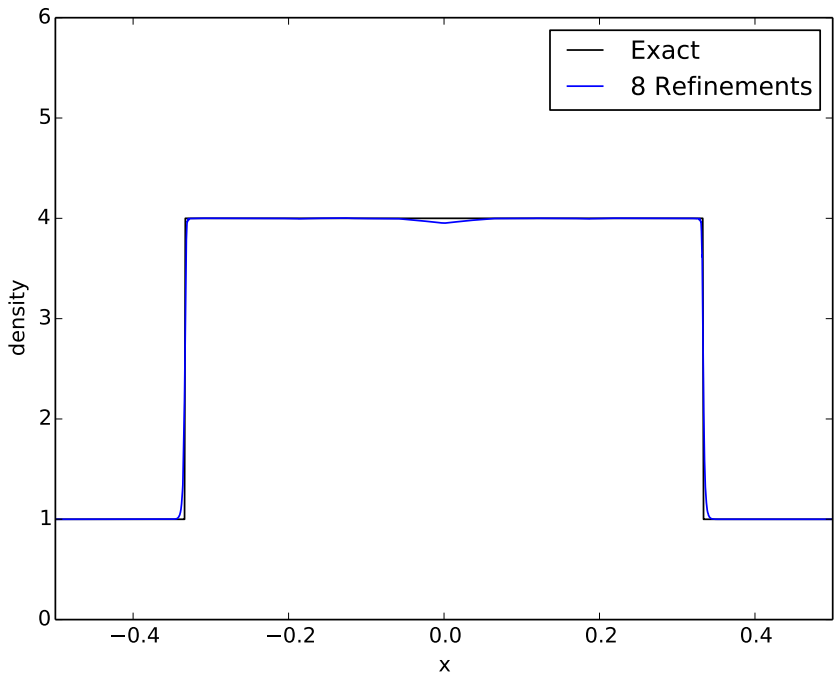


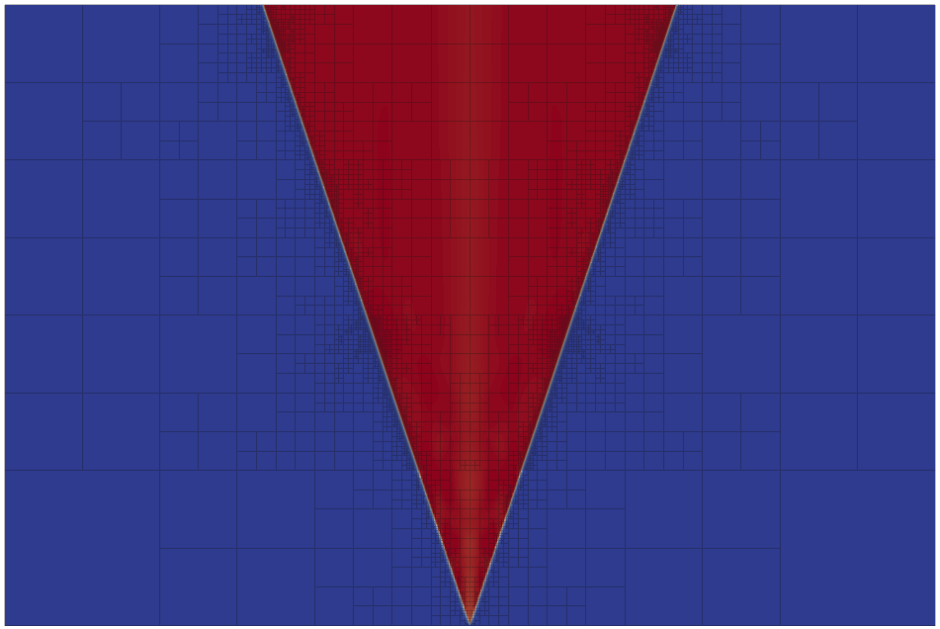


Compressible Navier-Stokes

Noh Implosion with $\mu = 10^{-3}$

Infinitely strong shock propagation.





Sequence of 4 time slabs

Table of Contents

THE UNIVERSITY OF TEXAS AT AUSTIN

Related Research

THE UNIVERSITY OF TEXAS AT AUSTIN

Past and Present Topics in DPG Research

- Multiphysics
  - Heat conduction (Poisson and Heat equation)
  - Wave problems (Helmholtz and Maxwell)
  - Linear elasticity and plate problems
  - Convection-Diffusion, Stokes, incompressible Navier-Stokes, compressible Navier-Stokes, Euler
- Natively nonlinear DPG
- DPG for non-Hilbert L^p spaces
- Local conservation
- Iterative solvers
- Entropy scaling for physically meaningful test norms
- General polyhedral elements

Thank You!

THE UNIVERSITY OF TEXAS AT AUSTIN

Recommended References

- J.L. Chan. ``A DPG Method for Convection-Diffusion Problems''. PhD thesis. University of Texas at Austin, 2013.
- L.F. Demkowicz and J. Gopalakrishnan. ``Recent Developments in Discontinuous Galerkin Finite Element Methods for Partial Differential Equations (eds. X. Feng, O. Karakashian, Y. Xing)''. In: vol. 157. IMA Volumes in Mathematics and its Applications, 2014. Chap. An Overview of the DPG Method, pp. 149–180.
- T.E. Ellis, L.F. Demkowicz, and J.L. Chan. ``Locally Conservative Discontinuous Petrov-Galerkin Finite Elements For Fluid Problems''. In: *Comp. Math. Appl.* 68.11 (2014), pp. 1530 –1549.
- N.V. Roberts. ``Camellia: A Software Framework for Discontinuous Petrov-Galerkin Methods''. In: *Comp. Math. Appl.* 68.11 (2014). Minimum Residual and Least Squares Finite Element Methods, pp. 1581 –1604.
- L.F. Demkowicz and N. Heuer. ``Robust DPG Method for Convection-Dominated Diffusion Problems''. In: *SIAM J. Numer. Anal.* 51.5 (2013), pp. 1514–2537.
- J. Chan et al. ``A robust DPG method for convection-dominated diffusion problems II: Adjoint boundary conditions and mesh-dependent test norms''. In: *Comp. Math. Appl.* 67.4 (2014). High-order Finite Element Approximation for Partial Differential Equations, pp. 771 –795.
- N. Roberts, T. Bui-Thanh, and L. Demkowicz. ``The DPG method for the Stokes problem''. In: *Comp. Math. Appl.* 67.4 (2014). High-order Finite Element Approximation for Partial Differential Equations, pp. 966 –995.