2D Spectral Element Scheme for Viscous Burgers' Equation

P. Aaron Lott

University of Maryland

Department of Applied Math &

Scientific Computation

AMSC 664

May 14, 2004

Advisor: Dr. Anil Deane

Motivation/Scientific Context

- To model the dynamics of the Earth's Mantle we treat it as a highly viscous, incompressible Boussinesq fluid.
- It is important to study the affects of flows over long time periods to better constrain the parameter space of the model.
- There is seismic and geochemical evidence of chemical/structural phase transition at the depth of 410 and 670 km. There are viscosity changes of several orders of magnitude. To handle these sharp interfaces one needs a refinement method to efficiently study the flow in this region.

Mantle Convection

- The Governing Equations are:

  $\nabla \cdot u = 0$ Incompressibility

  $\frac{1}{\rho}\nabla p = \nu\nabla^2 u - g\alpha\Delta T$ (Momentum Equation with $\frac{\partial u}{\partial t} = 0$)

  $\frac{\partial T}{\partial t} + u \cdot \nabla T = \kappa\nabla^2 T + \frac{J}{\rho C_p}$ (Thermal energy equation)
- u - velocity, T - temp, p is pressure, ν -viscosity, κ - thermal diffusivity, α - thermal expansion coefficient, ρ - density, and C_p - heat capacity at constant pressure, J - rate of internal pressure per unit volume, g - gravity.
- Note, these are the Incompressible Steady Stokes Equations with a source term fed by the unsteady advection diffusion equation at each time step.

Advection Diffusion Equation

- We are solving the Advection Diffusion equations in 1D and 2D, with advection velocity $\vec{c}$ and viscosity ν.
- 1D

$$\frac{\partial u}{\partial t} + (\vec{c}\frac{\partial u}{\partial x}) = \nu \frac{\partial^2 u}{\partial x^2} \qquad 1)$$

- 2D

$$\frac{\partial u}{\partial t} = \nu \Delta u - \vec{c} \cdot \nabla u \qquad \text{in } \Omega \quad t \geq 0 \qquad 2)$$

- Note $\vec{c} = u$ yields the viscous Burgers' Equations.

Spectral Element Discretization

- We use a Spectral Element Method Discretization to solve 1 and 2, expanding the solution as a linear combination of basis functions $\phi_i(x)$.

$$u_n^k = \sum_{i=0}^{n} u_i^k(t)\phi_i(x) \qquad 3)$$

Spatial Discretization- Basis Functions

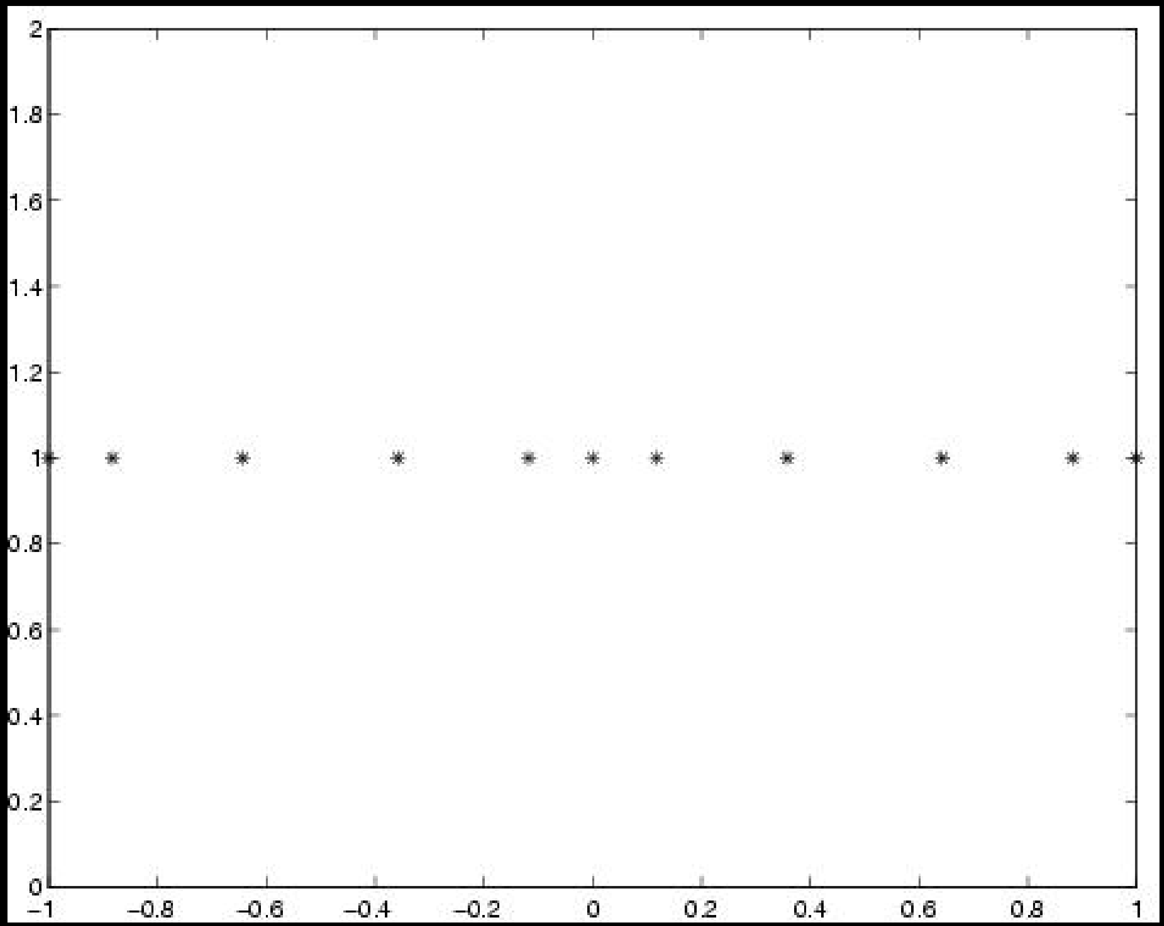


Figure 1: A. GLL Spatial Discretization with 2 elements and 5th degree Polynomials B. GLL Polynomials of degree 1 through 10

Spatial Discretization

- The discretized equation can be written as a non-linear differential equation

$$M\dot{u}(t) = -C(u)u(t) - \nu K u(t) \qquad 4)$$

- M- Mass matrix, K- Stiffness matrix and $C(u)$ is the nonlinear discrete operator. Each of which are block diagonal matrices.
- So to solve for $u(\vec{x},t)$ we will integrate these equations in time.

Spatial Discretization

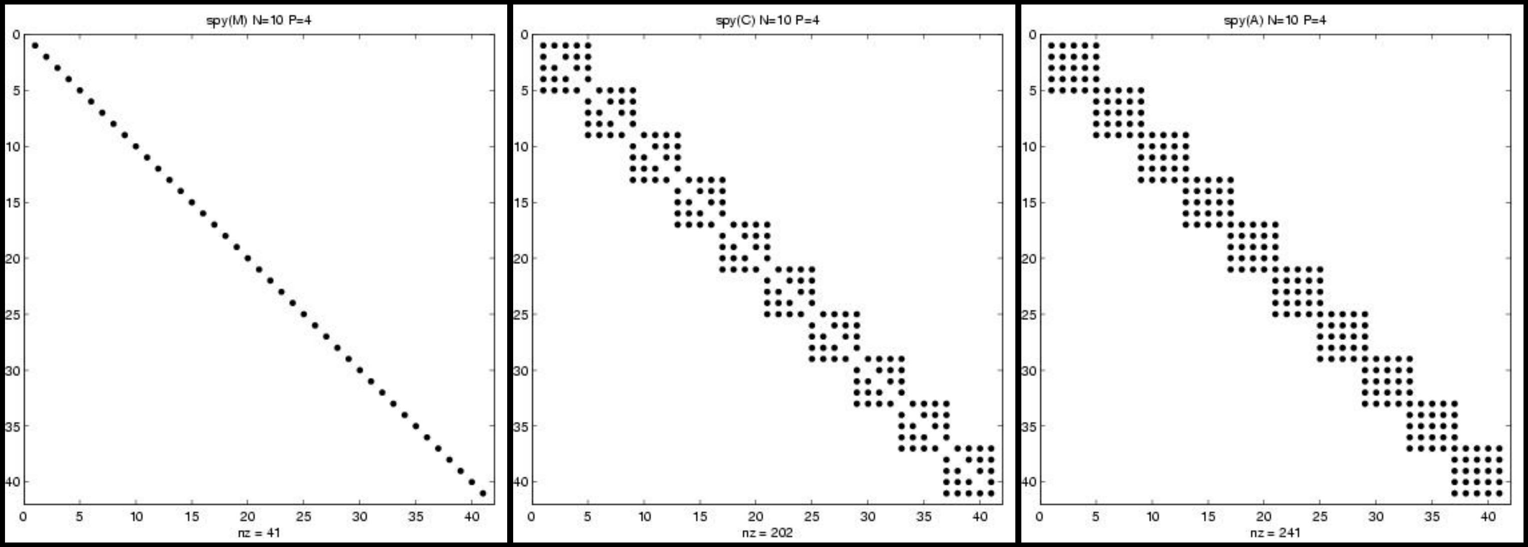


Figure 2: Global System Matrices. From the left 1 Mass,1, Convection, 1 Stiffness(Diffusion)

Time Discretization

- In order to obtain a stable solution in time, one considers the eigenvalues of the operators acting on u, and makes certain that the time marching scheme is stable in this region.
- In our system, C and K act on u
-

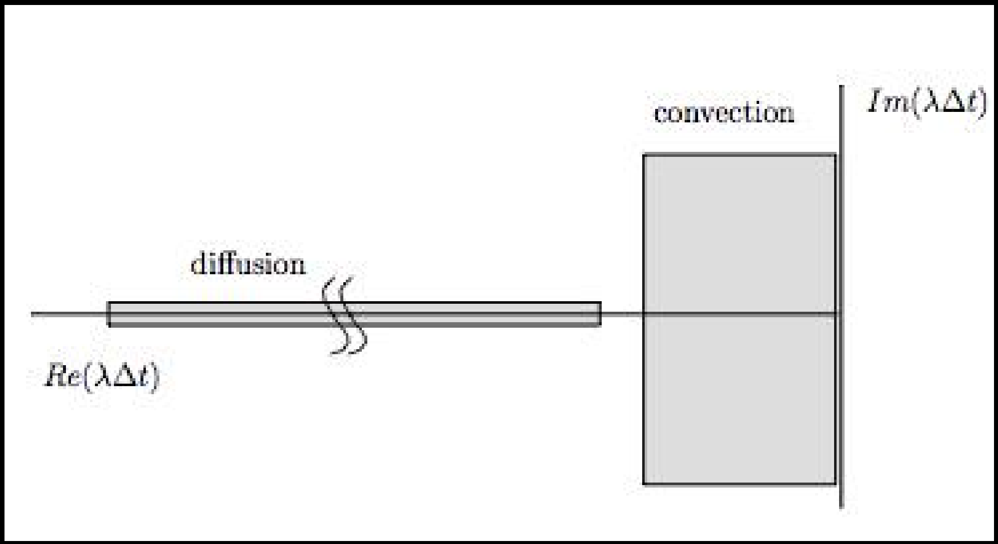


Figure 3: Eigenvalues of the Diffusion and Convection operators

Time Discretization

- For spectral methods the eigenvalues, λ, of the diffusion matrix are real and negative, and the maximum eigenvalue is $O(N^4)$ where N is the maximum polynomial degree. For Spectral Elements, empirical tests show $\lambda \approx O(n_e N^3)$ where n_e is the number of elements.
- The eigenvalues, λ, of the convection operator have an imaginary part and and a negative real part, and the largest eigenvalue is $O(N^2)$.

  Thus, we want a time discretization which is stable on the negative real axis and the imaginary axis.

Time Discretization

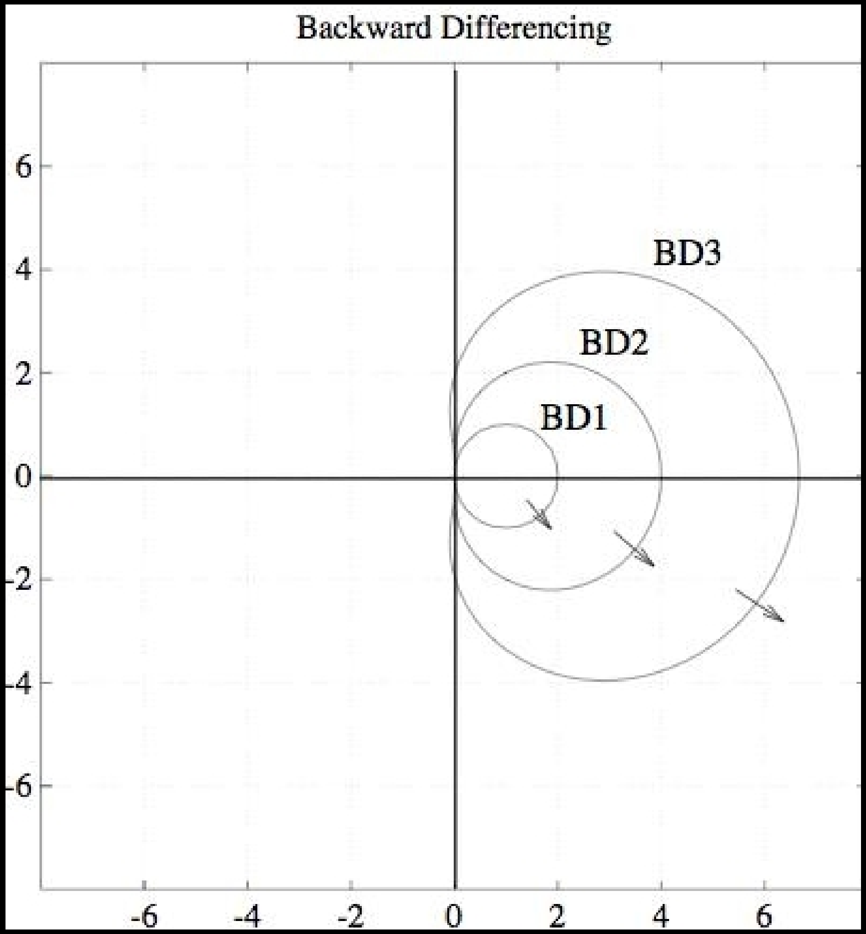


Figure 4: Stability region arrows denote stability outside the corresponding curve for the Backward Difference time marching scheme.

Time Discretization

- In order to achieve 3rd order accuracy, and a stable solution in time we use the BDF3 scheme, and extrapolate the Convection term at each iteration.

$$\left(\frac{11}{6\Delta t}M+\nu K\right)v_i^{n+1}=\frac{M}{\Delta t}\left(3v_i^n-\frac{3}{2}v_i^{n-1}+\frac{1}{3}v_i^{n-2}\right)-Cv_i^{n+1} \qquad 5)$$

where we use 3rd order extrapolation to obtain

$$Cv_i^{n+1}=3Cv_i^n-3Cv_i^{n-1}+Cv_i^{n-2}+O(\Delta t^3) \qquad 6)$$

- For SEM a harsh condition is placed on Δt in order to satisfy the CFL criteria. For basis functions of degree $N-1$,

$$\Delta t \leq \frac{6.5}{\nu}\frac{\pi^2}{N^4} \qquad 7)$$

Computation Localization

- We originally formed the global system matrices, and then iterated over time. However, as we moved into 2D these system matrices have size $(P+1)^2 N_x N_y$, which, even for coarse meshes are quite large.
- Static Condensation

  Earlier we gave an illustration of the coupling between elements. By re-ordering the mapping between local and global indices we could effectively decouple the interiors and only solve the coupled system on elemental boundaries. This could all be done with local element matrices of size $(P+1)^2$.

Computation Localization

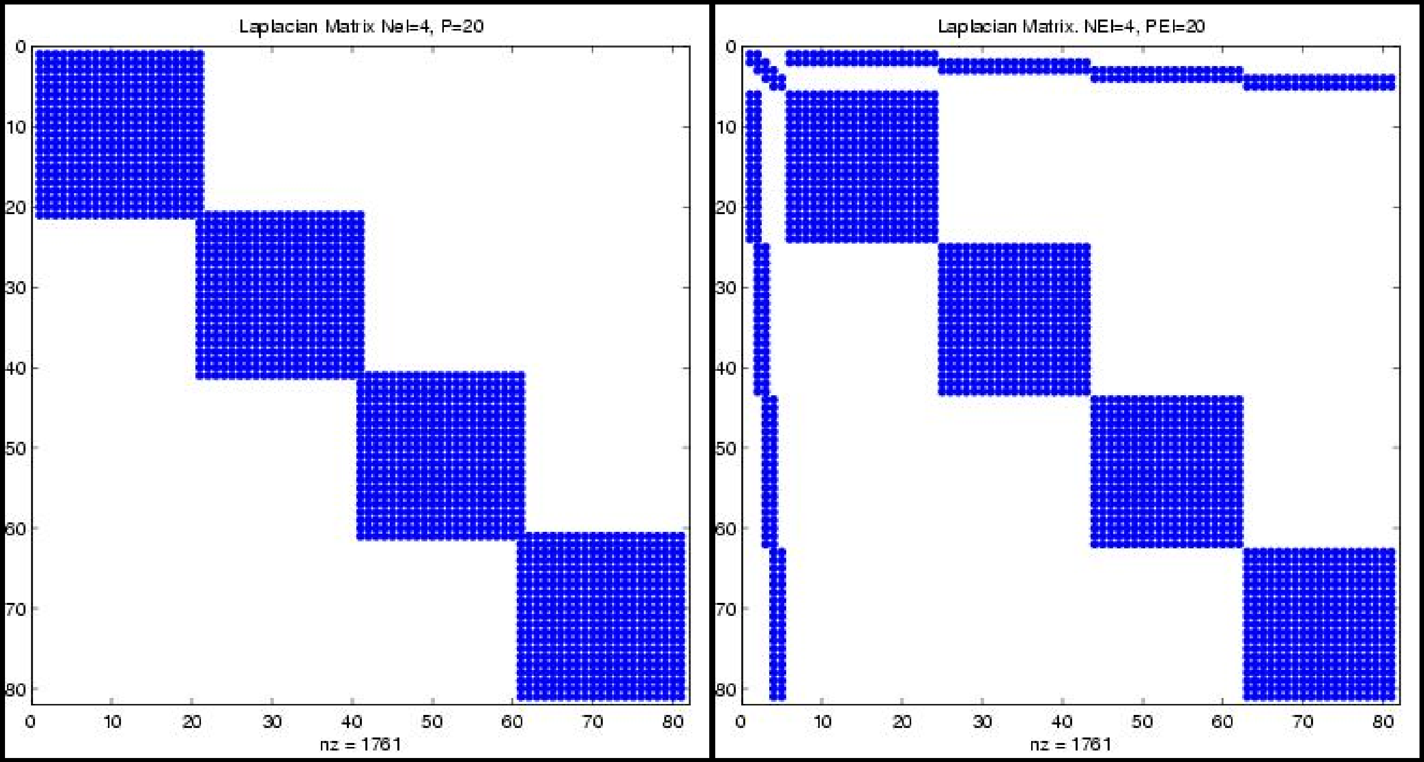


Figure 5: Coupled and Uncoupled Diffusion operator. 1 Big coupled 1 Boundary 1Interior

Computation Localization

- Advantages:

  Reduces communication once the boundary information has been decoupled. No global matrices need to be stored, all calculations are done on element matrices
- Disadvantages: Need to compute Schur compliment.

Computation Localization

- Elementwise operations

  Instead of using static condensation, one can perform operations on local elements and then cleverly add the proper amount to the global solution U without the cost of static condensation.
- A weighting matrix is formed for each element to determine the contribution of the local solution to the global solution. Finally we take care of the element boundary dependencies after each local calculation.

Computation Localization

- Advantages: For large P this should be very efficient since most entries are interior nodes, and each processor can use its cache more efficiently. No global matrices need to be stored, all calculations are done on element matrices
- Disadvantages: For small polynomial degree, more communication compared to calculations

Computation Localization

- In either case, these problems scale well to higher dimensions because the global Matrix operators can be written as tensor products.
- We construct the local operators for all possible polynomial degree (run time parameter), and store them in an easily accessible efficient data structure. These include, M,A,C, Derivative, Iterpolants from $P_n \longrightarrow P_{n-1}$ and vice versa. These structures are accessible through a global data module.

P-type Refinement

- With the local matrices stored for all values of P, and the ability to perform local operations and build the global solution, it is now trivial to compute the derivative of the local solution and perform error analysis with it.
- For example, if the slope of our solution at a local element is greater than some user defined value, then we increase the polynomial degree of that element by one.
- We perform this on each element and then construct a new local to global mapping with respect to the new elements.

P-type Refinement

Figure 6: 3 time steps in solution to burgers' equation starting with N=32 P=4 refining with abs(du/dx) is greater than 8

Validation

- 1D Burgers's Equation

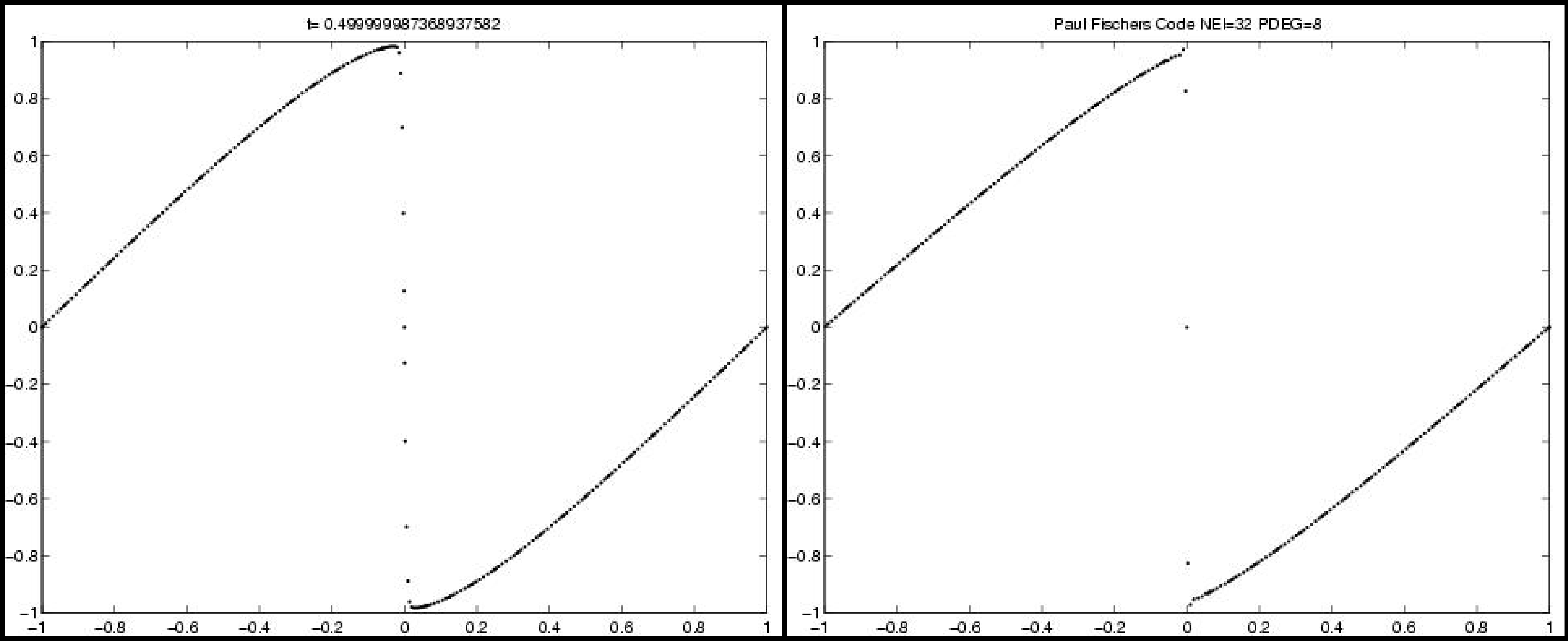


Figure 7: Comparison of Results between Our code (left) and Published Code (right)

Validation

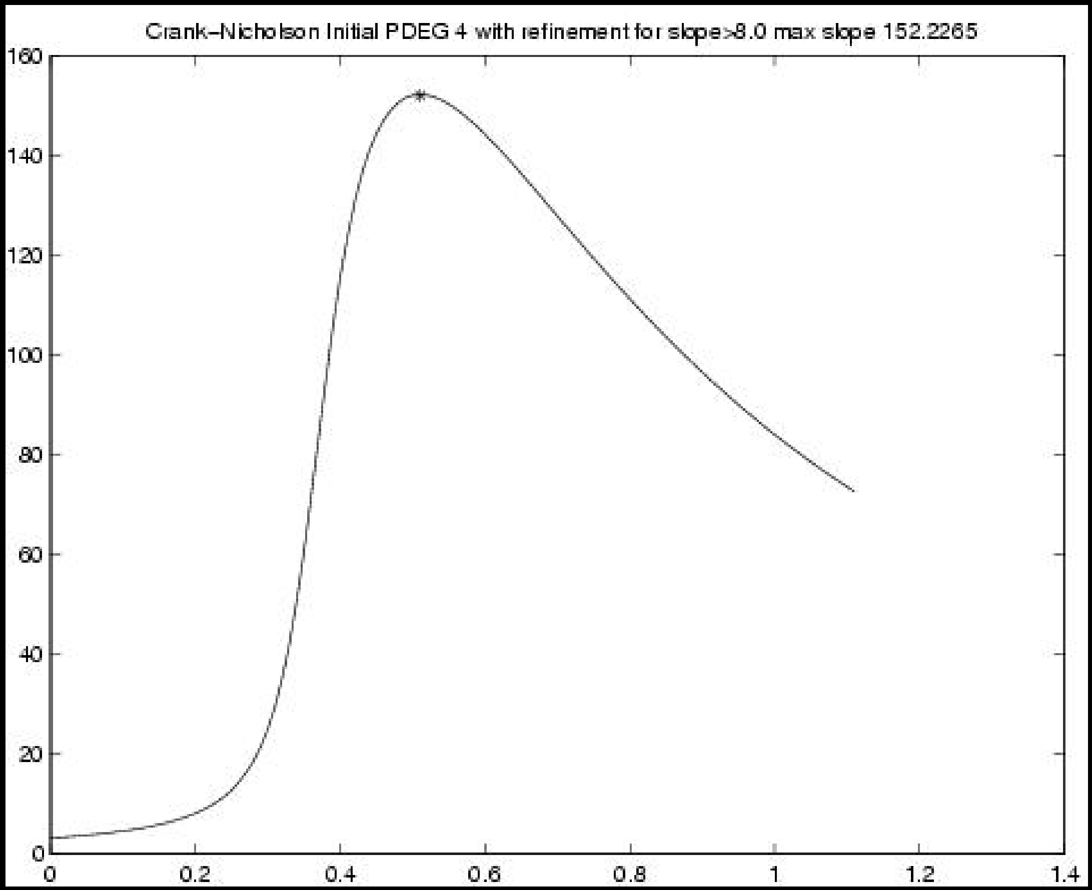


Figure 8: Comparison of results between our code and actual maximum amplitude of slope. We have a value of 152.2265 at t=.5100 analytical value is 152.0051 at t=.5105. The compares very well with other published numerical methods.

2D Results

- 2D Burgers's Equation

  Start with initial conditions

$$u(x,y,0) = .01^{4(x^2+y^2)} \quad \text{on}[-1,1]^2 \qquad 8)$$

  We use periodic boundary conditions, $P = 8$, $N_x = 4$, $N_y = 4$ $\nu = .01$

Future Directions

- Add 2D adaptivity
- Implement Stokes Equation
- Parallelization
- Preconditioned Conjugate Gradient to solve local systems

Conclusions/Summary

- Implemented and verified 1D and 2D viscous Burgers' Equation
- Implemented P adaptivity in 1D, and framework for adaptivity in 2D
- Structured code to begin MPI Parallelization

References

- [1] High-Order Methods for Incompressible Fluid Flows. M.O. Deville, P.F. Fischer, E.H. Mund. Cambridge University Press 2002.

  [2] An overlapping Schwarz method for spectral element solution of the incompressible Navier-Stokes equations. P.F. Fischer. Journal of Computational Physics 1997.

  [3] Spectral/hp Element Methods for CFD. G.E. Karniadakis, S.J. Sherwin. Oxford University Press 1999.

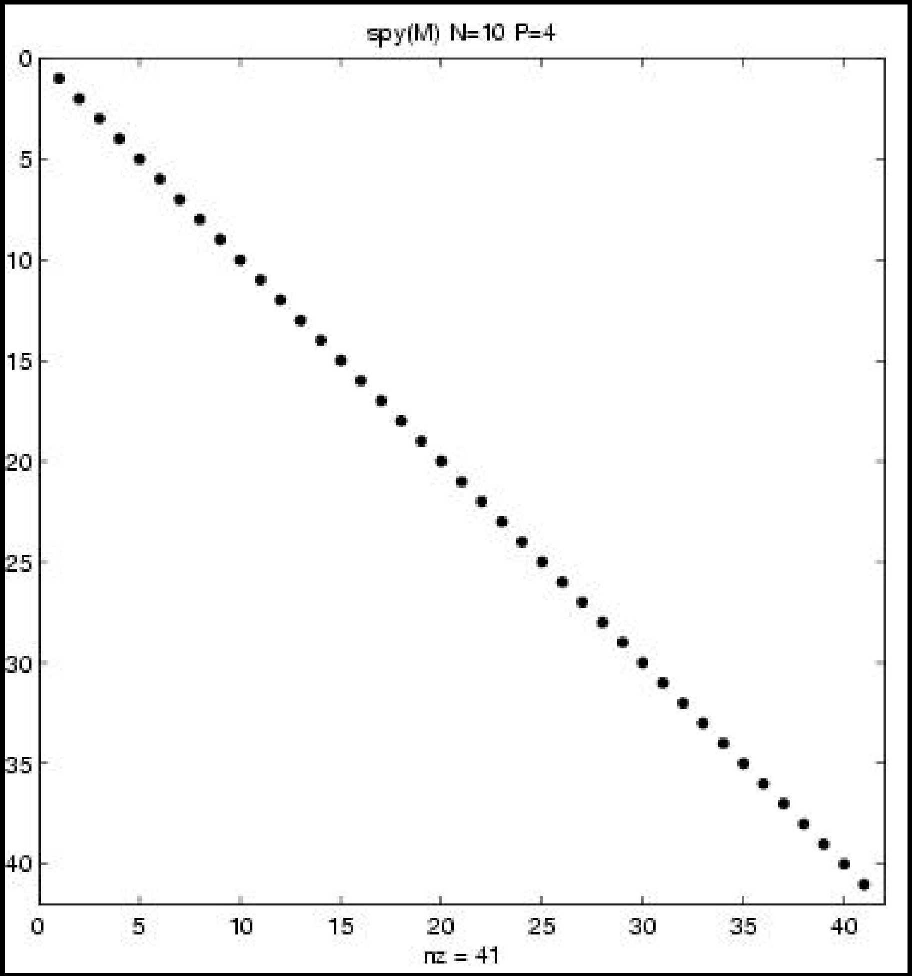
  [4] An Adaptive Finite Element Scheme for Transient Problems in CFD. Rainald Löhner. Computational Methods in Applied Mechanics and Engineering. 61 (1987) 323-338

  [5] Spectral element methods: theory and applications. F.N. van de Vosee, P.D. Minev.

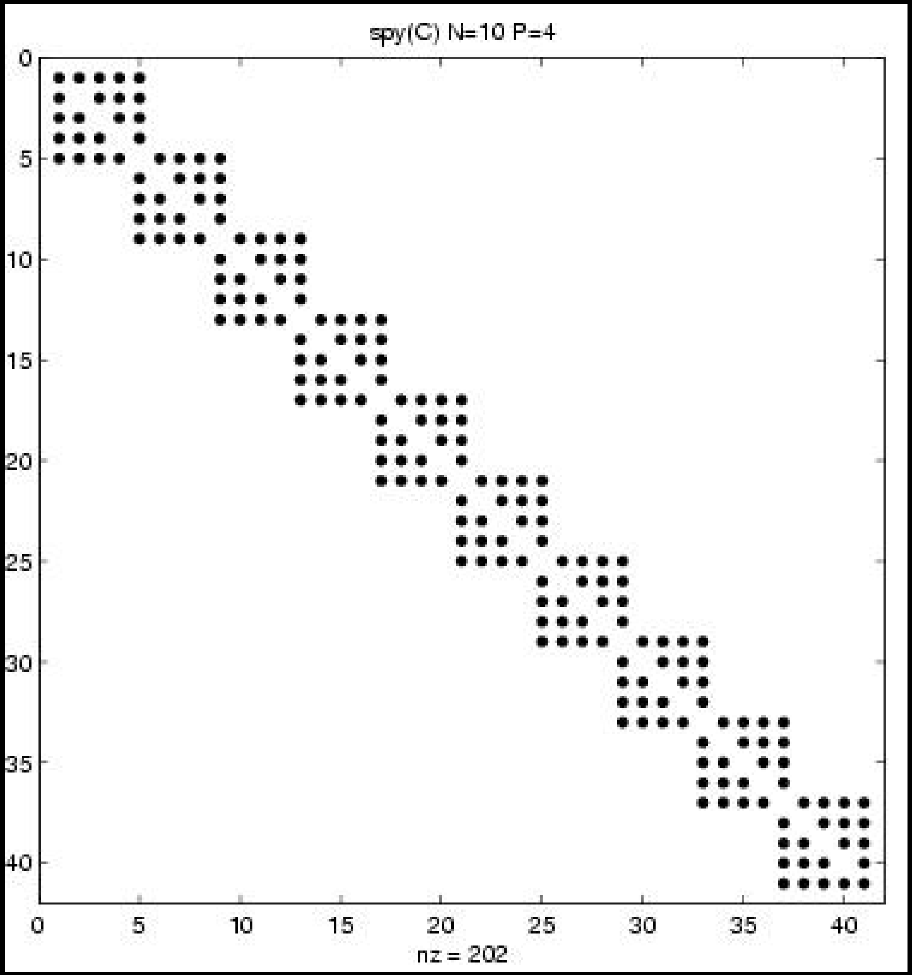
  http://www.mate.tue.nl/people/vosse/docs/vosse96b.pdf

[6] Project Website. http://www.lcv.umd.edu/ palott/research/graduate/66

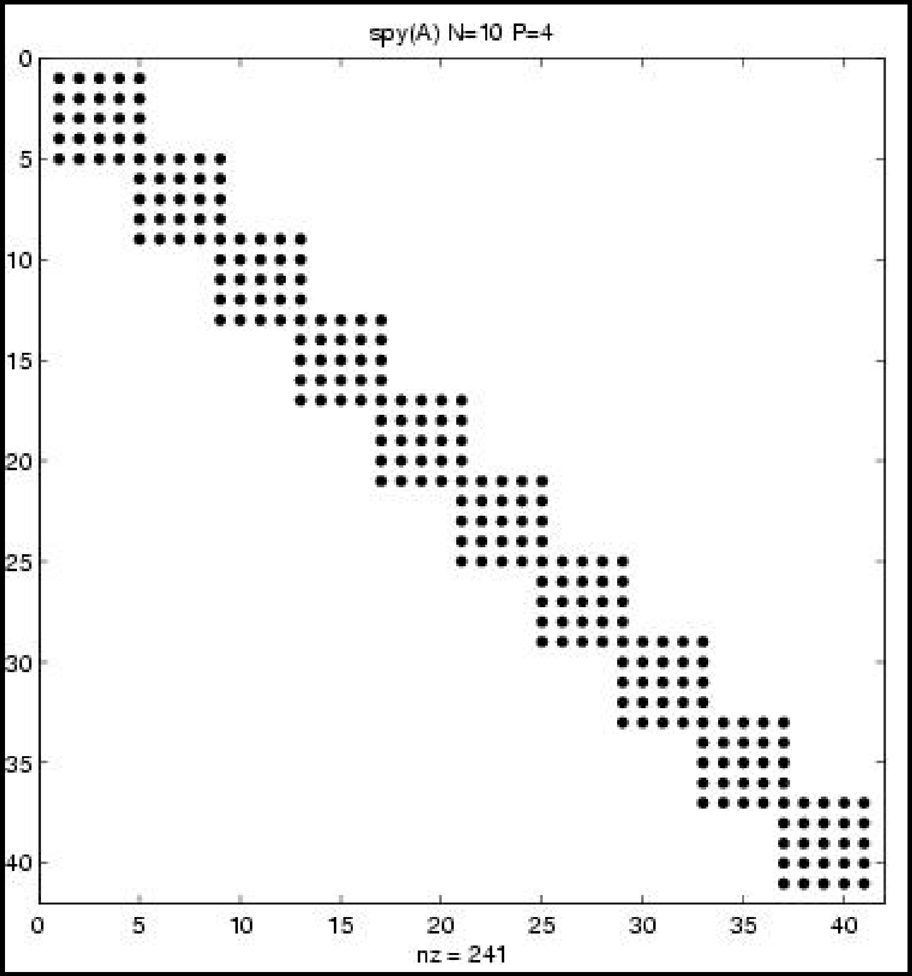
Figures



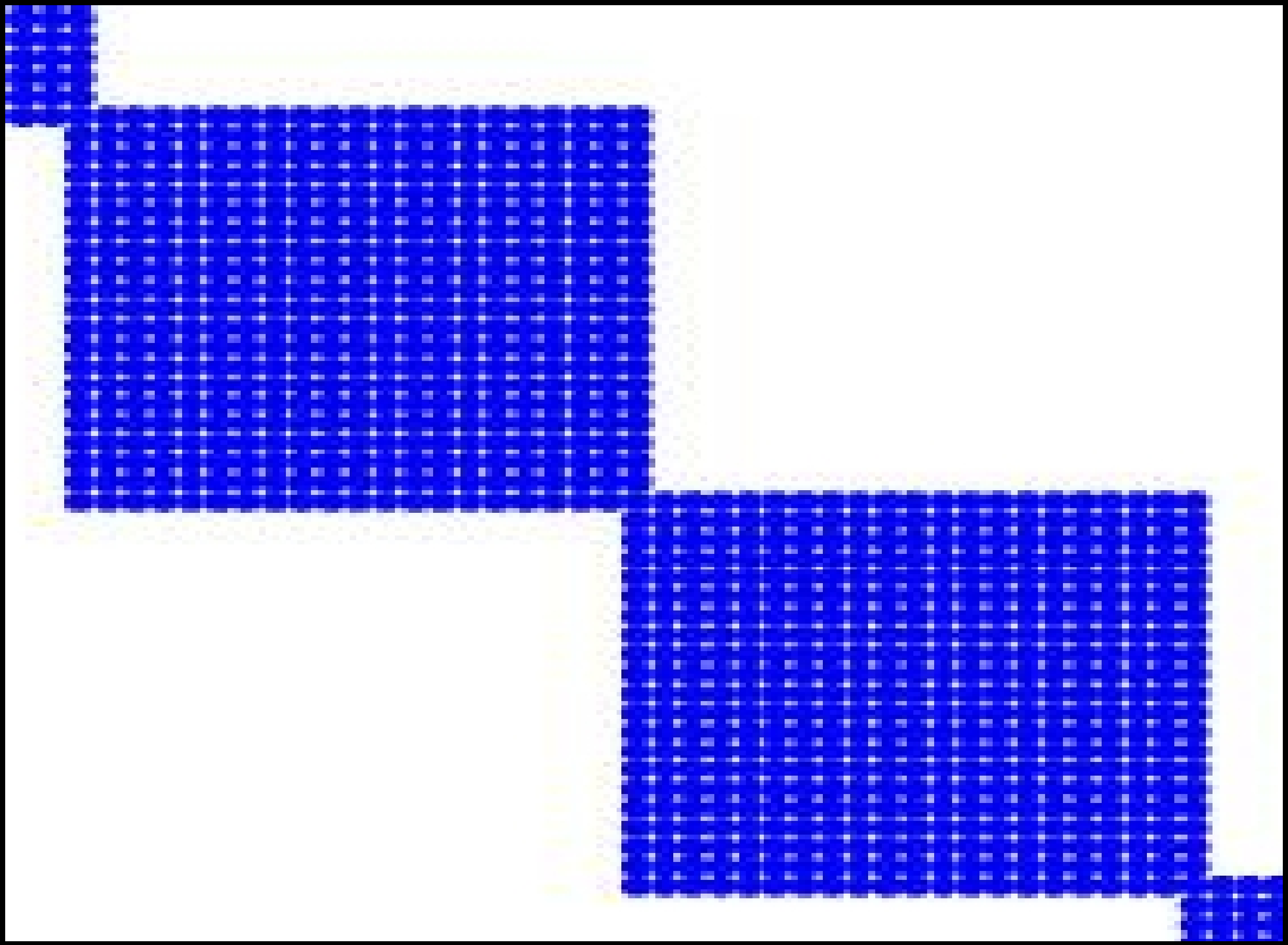
Figures



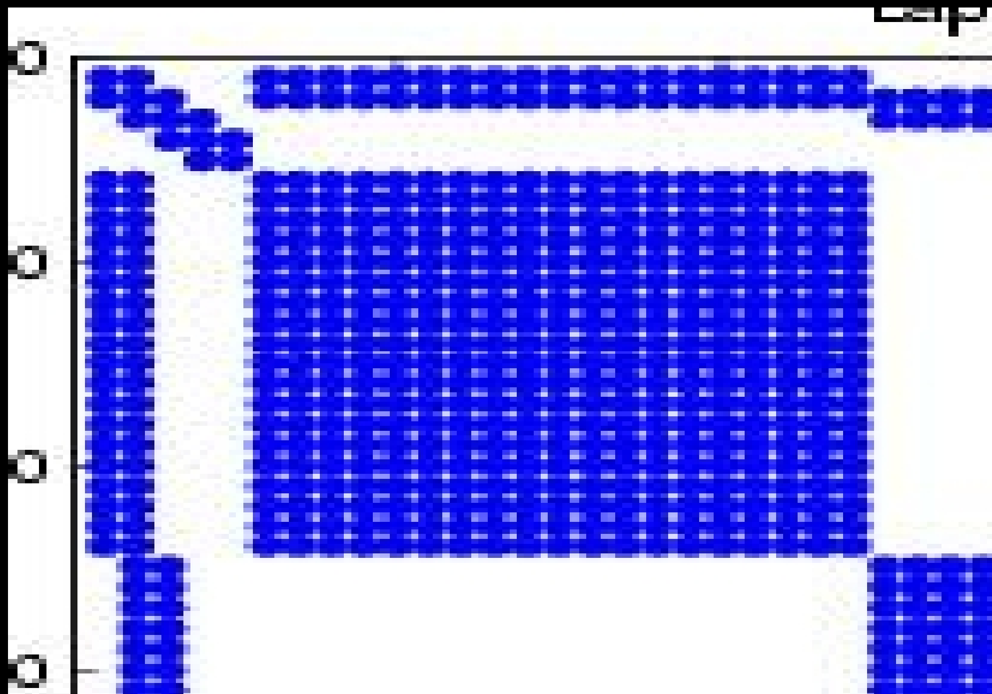
Figures



Figures



Figures

- 

Figures