Implicit Finite Element Solver

Boundary Condition & Geometry Group

W. Lowrie, S. Lukin, G. Marklin, E. Meier, U. Shumlak

PSI Center Annual Meeting

University of Washington

August 13, 2007

Goals

- Develop a 3D, parallel, fully implicit, finite element solver for the MHD equations.
  - Using PETSc (Portable, Extensible Toolkit for Scientific Computation) for parallel matrix data structure and solver
  - And/Or Using SuperLU for matrix solver
  - Flux source based for easy equation/physics specification
  - Integrate an easy to use structured mesh generator into the code (CAD $\longrightarrow$ Structured Mesh)

Current Development

- **Solving Pseudo-1D Finite Element Equations**

$$\frac{\partial \vec{Q}}{\partial t} + \frac{\partial \vec{F}}{\partial x} = \vec{S}$$

where

$$\vec{Q} = \begin{bmatrix} \rho A \\ \rho u A \\ e A \end{bmatrix}, \vec{F} = \begin{bmatrix} \rho u A \\ (\rho u^2 + p) A \\ u(e+p) A \end{bmatrix}, \text{ and } \vec{S} = \begin{bmatrix} 0 \\ p\frac{dA}{dx} \\ 0 \end{bmatrix}$$

and $e = \frac{p}{\gamma - 1} + \frac{1}{2}\rho u^2$ for an ideal gas.

- **Taylor-Galerkin discretization**

Taylor Temporal Expansion:

$$\Delta Q = \Delta t \frac{\partial Q^n}{\partial t} + \frac{\Delta t^2}{2}\frac{\partial^2 Q^n}{\partial t^2} + \dots \quad (1)$$

Galerkin Spatial Expansion:

$$\int_\Omega N_i \{\Delta Q\} \partial\Omega = \Delta t \left[\int_\Omega N_i S^n \partial\Omega - \int_\Omega N_i \frac{\partial F^n}{\partial x} \partial\Omega \right] + \frac{\Delta t^2}{2} \left[\int_\Omega N_i Z(S - \frac{\partial F}{\partial x}) \partial\Omega - \int_\Omega N_i \frac{\partial}{\partial x} \left(A(S - \frac{\partial F}{\partial x}) \right) \right] \quad (2)$$

Current Development

- **Linear elements**

  Linear basis functions used for initial simplicity.

- **Implicit terms using the Θ-Method**

$$\left[\mathbf{M} - \theta\Delta t(\mathbf{M}Z^n + \mathbf{K}A^n - \mathbf{B}A^n)\right]\{\Delta q\} = \Delta t\left[\mathbf{M}\{s^n\} + \mathbf{K}\{f^n\} - \mathbf{B}\{f^n\}\right]$$

  where $\mathbf{M} = \int_\Omega N_i N_j \partial\Omega$, $\mathbf{K} = \int_\Omega \frac{\partial N_i}{\partial x} N_j \partial\Omega$, $\mathbf{B} = [N_i N_j]_\Omega$
  and $A^n = \frac{\partial F}{\partial Q}$, $Z^n = \frac{\partial S}{\partial Q}$,
  which are the flux Jacobians for the flux vector and source vector respectively.

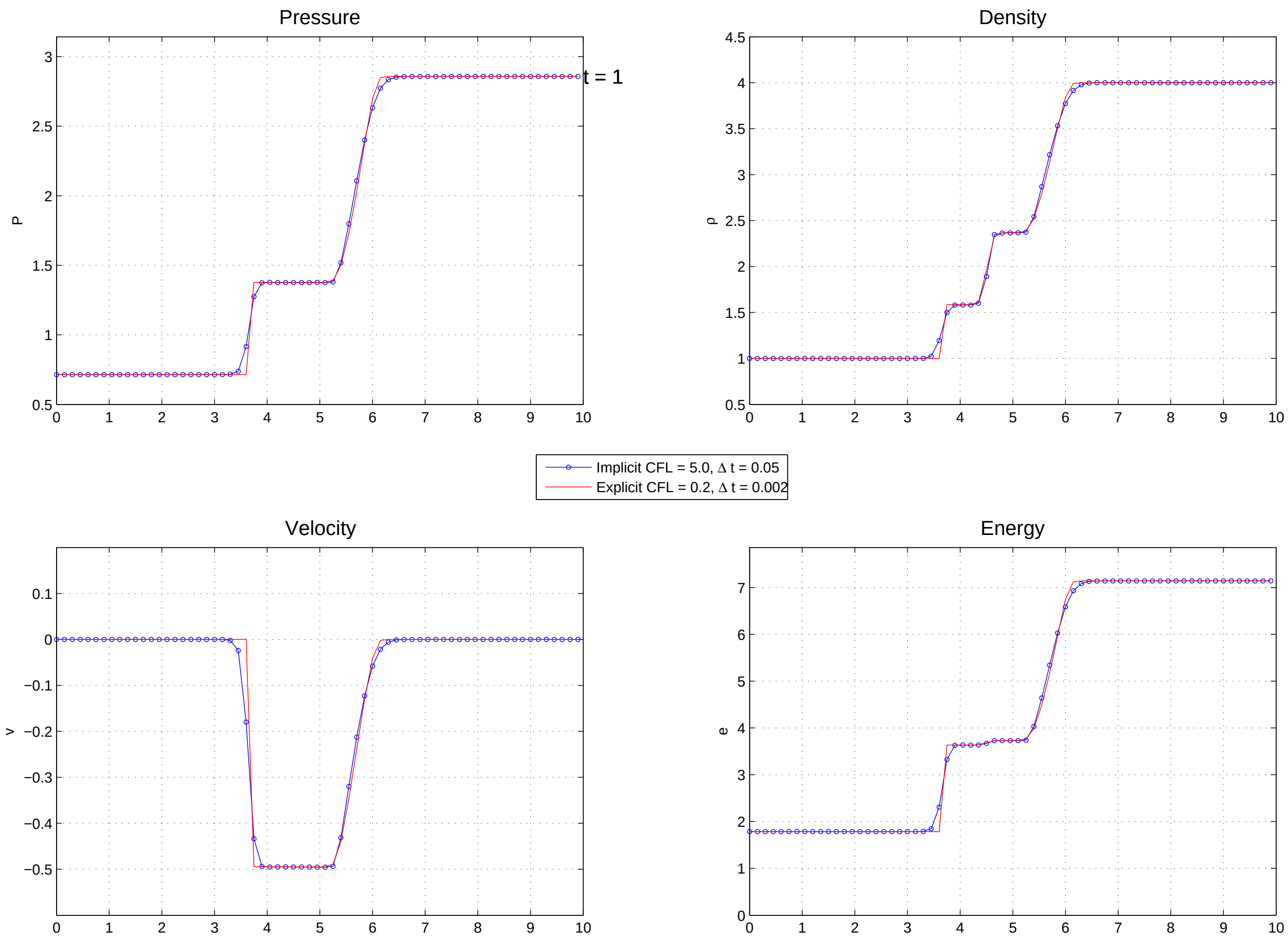
- **Artificial dissipation for both explicit and implicit terms**

  $\epsilon\nabla^2 Q$ term is added to both implicit and explicit terms. Equation is modified:

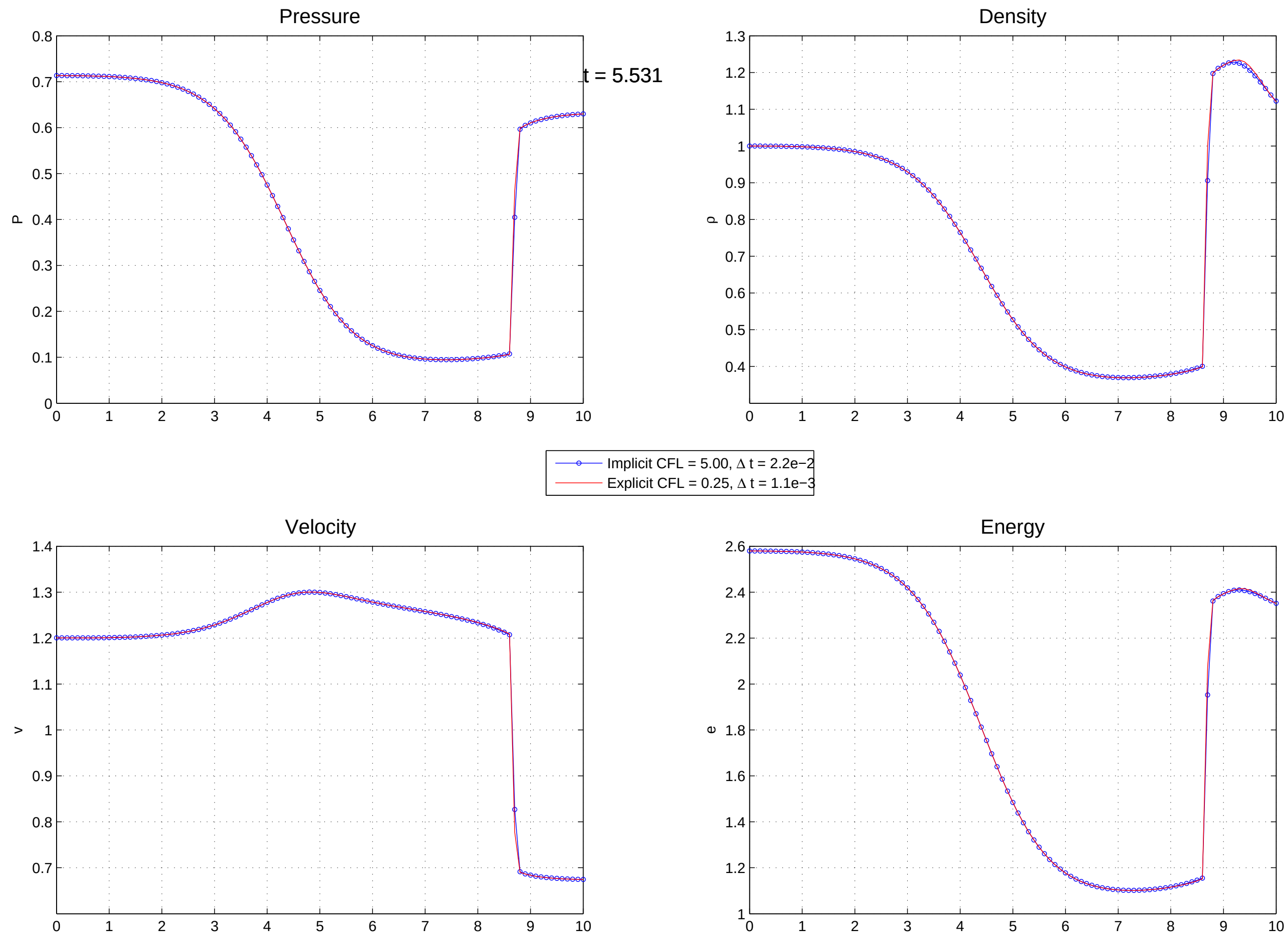
$$\left[\mathbf{M} - \theta\Delta t(\mathbf{M}Z^n + \mathbf{K}A^n - \mathbf{B}A^n) + \epsilon_i\mathbf{V}\right]\{\Delta q\} =$$
$$\Delta t\left[\mathbf{M}\{s^n\} + \mathbf{K}\{f^n\} - \mathbf{B}\{f^n\} - \epsilon_e\mathbf{V}\{q^n\}\right]$$

  where $\mathbf{V} = \int_\Omega \frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x}\partial\Omega$

Shock Tube Result



Pseudo-1D Nozzle Result



Where to Next...

- PETSc implementation for parallel matrix data structures and solvers
- Also, implement SuperLU as parallel solver
- High order basis functions
- Implement more equations sets (i.e ideal MHD)
- Increase dimensionality (2D and then 3D)
- Integrate the CUBIT mesh generator (structured, hexahedral, CAD input)