SAND2007-5442C

Improving Magnetic Energy Force Discretization in ALEGRA

7 August 2007

2007 Student Symposium

Sandia National Laboratories

Alan Schiemenz

Brown University

Technical Advisor: Allen Robinson

SNL Org. 1431

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Project Overview

- **ALEGRA magnetohydrodynamics (MHD) modeling**
- **Lagrangian (moving mesh) motion**
- **Improve nodal force computation**
- **Intrepid: templated compatible discretizations technology**
- **Sacado: automatic differentiation library**
- **New force option: combine Intrepid & Sacado in ALEGRA**

Sandia National Laboratories

What is ALEGRA?

- **ALEGRA is a multi-material multi-physics shock hydrocode.**
  - **Lagrangian finite element**
  - **Eulerian multimaterial**
  - **ALE combines techniques to minimize numerical dissipation. Remesh/remap only when mesh becomes badly deformed**
- **ALEGRA is hydro/solid dynamics with other coupled physics (MHD, HEDP)**
- **ALEGRA is built on NEVADA framework capabilities.**

Sandia National Laboratories

The Intrepid Project

- **Templated compatible discretizations technology**
- **Basic idea: match numerical discretization of PDE with exact physics**
- **Do this to ensure exact physical properties, e.g. curl(grad(.)) = 0, div(curl(.)) = 0, etc.**
- **Theoretical foundation in differential geometry and algebraic topology**
- **Differential Forms and the De Rham complex**
- **Pavel Bochev, Denis Ridzal, and others**

Sandia National Laboratories

The Sacado Project

- **C++ automatic differentiation library**
- **Analytic derivatives without hand-coding**
- **Systematic application of the chain rule through your computation, differentiating each statement line-by-line**
- **Eric Phipps and David Gay**

Sandia National Laboratories

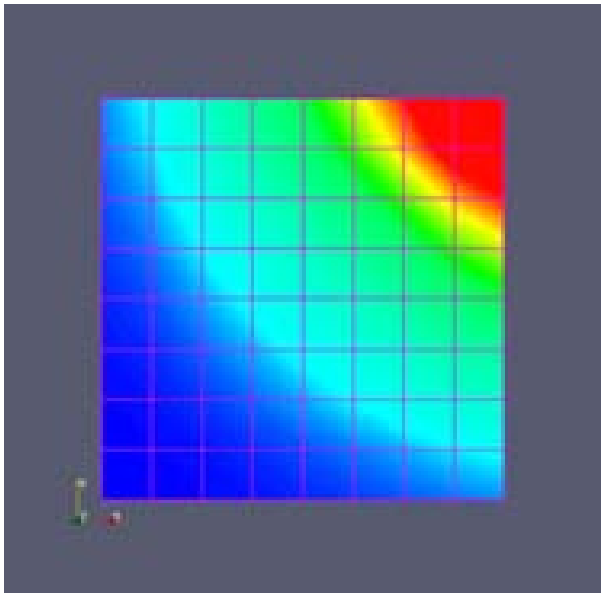
ALE sequence for ~~magnetohydrodynamics (MHD)~~

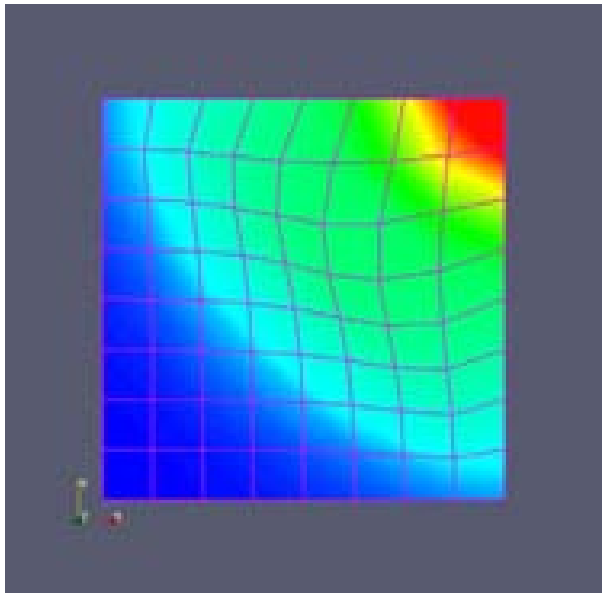
- **2D and 3D multiple material ALE based on unstructured mesh hex and quad finite elements**
- **Lagrangian Steps (Operator split)**
  - **Compute forces and accelerations**
  - **Move nodes (magnetic fluxes or magnetic potential circulations are invariant)**
  - **Implicit magnetic diffusion (eddy currents) and Joule heating, energy transfer through boundary. Vacuum is approximated with a very small conductivity**
- **Remesh – choose a close, new mesh**
- **Remap – compute new values at element centers, nodes, faces and edges. Constrained transport for face centered fluxes**

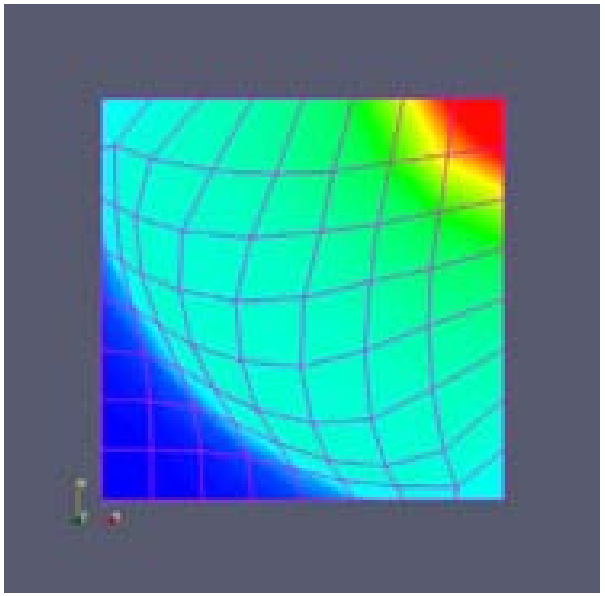
Sandia National Laboratories

Lagrangian description of motion

- **Mesh moves with material**
- **Opposite of Eulerian description, where mesh is fixed**
- **Here: plots of magnitude of B-field**
- **Elements deformed according to nodal (J x B) force**







Sandia National Laboratories

Force Computation

$$\mathrm{J} \times \mathrm{B} = \nabla \cdot \mathrm{T} - \mathrm{B}(\nabla \cdot \mathrm{B})$$

$$\mathrm{T}_{ij} = \frac{1}{\mu}\left(B_i B_j - \frac{1}{2}\delta_{ij} B_k B_k\right)$$

- **TENSOR option: Approximate JxB force by evaluating T at element centers and then utilizing the finite element divergence operator to compute nodal forces.**
- **Issue: Numerical errors lead to nonphysical energy gain/loss**
- **PJXPBPV option: directly compute J x B (but this is even less accurate)**
- **Goal: prevent this by more accurately computing force**

Sandia National Laboratories

Another Idea: MATRIX Force Option

$$\{\mathbf{x}_k\}, \quad 1 \leq k \leq N \text{ nodes on a simplex}$$

$$\frac{d}{dt}\{\text{Kinetic E.} + \text{Magnetic [Potential] E.}\} = 0$$

$$\frac{d}{dt}\left\{\sum_k \left\{\frac{1}{2} m_k {v_k}^2\right\} + E_{\text{mag}}(\mathbf{x}_1, ..., \mathbf{x}_N)\right\} = 0$$

$$\sum_k \left\{m_k v_k a_k + \frac{\partial E_{\text{mag}}}{\partial x_k} v_k\right\} = 0$$

$$m_k a_k + \frac{\partial E_{\text{mag}}}{\partial x_k} = 0 \ \forall k$$

$$\mathbf{F}_k = -\frac{\partial E_{\text{mag}}}{\partial \mathbf{x}_k} = \frac{\partial}{\partial \mathbf{x}_k}\left\{-\frac{1}{2\mu}\int \mathbf{B}\cdot\mathbf{B}dx\right\}$$

Sandia National Laboratories

MATRIX Force (cont.)

- **Force equation:**

$$\mathrm{F}_k = -\frac{\partial E_{\mathrm{mag}}}{\partial \mathrm{x}_k} = \frac{\partial}{\partial \mathrm{x}_k}\left\{-\frac{1}{2\mu}\int \mathrm{B}\cdot\mathrm{B}dx\right\}$$

- **Intrepid gives mass matrix M for magnetic energy computation:**

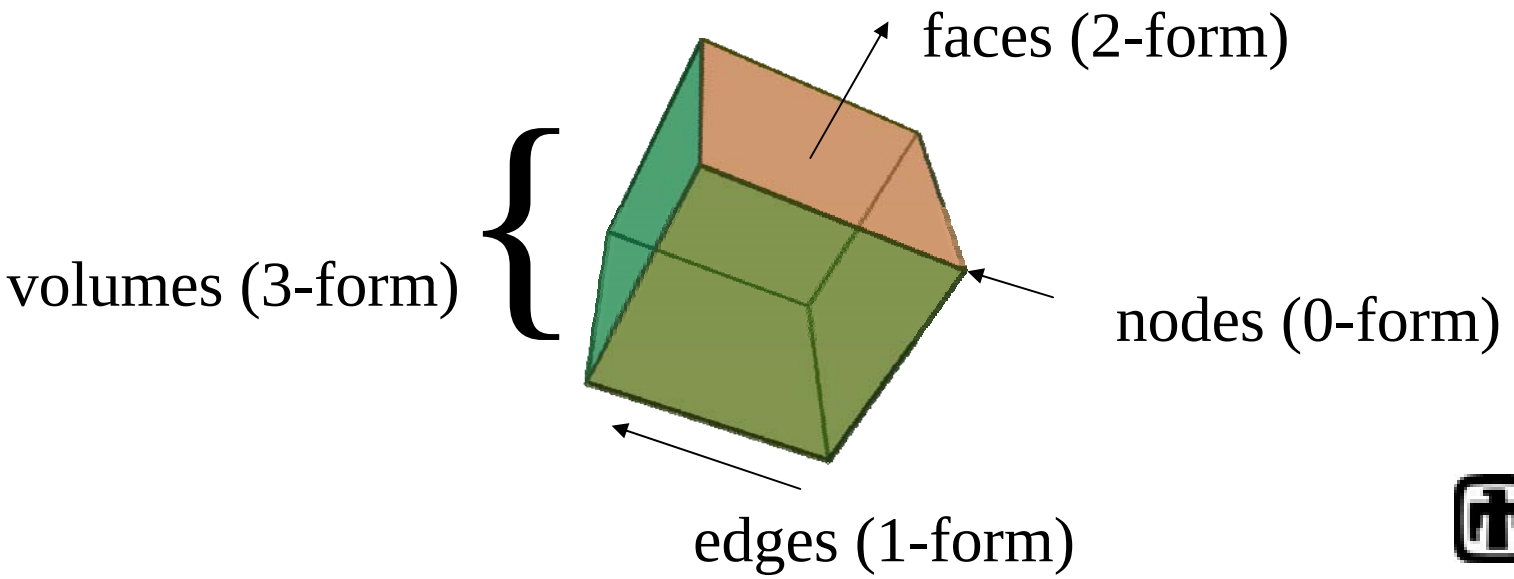
$$E_{\mathrm{mag}} = -\frac{1}{2\mu}\int \mathrm{B}\cdot\mathrm{B}dx = -\frac{1}{2\mu}B_f^T M B_f$$

- **... where degrees of freedom are magnetic fluxes through the element faces (the 2-form) and which are constant for ideal MHD!**
- **Sacado allows us to compute derivatives w.r.t. coordinates**
- **Discretization exactly matches the physics!**

Sandia National Laboratories

How Intrepid Helps Us

- **Handle different topologies (0,1,2,3 – forms)**
- **This allows for different choices for the Deg. of Freedom**
- **Can handle nodal values, edge circulations, flux through faces, or element volumes as DoF**
- **Written as templated library, allowing us to pass in FAD types from Sacado**



Sandia National Laboratories

How Sacado Helps Us

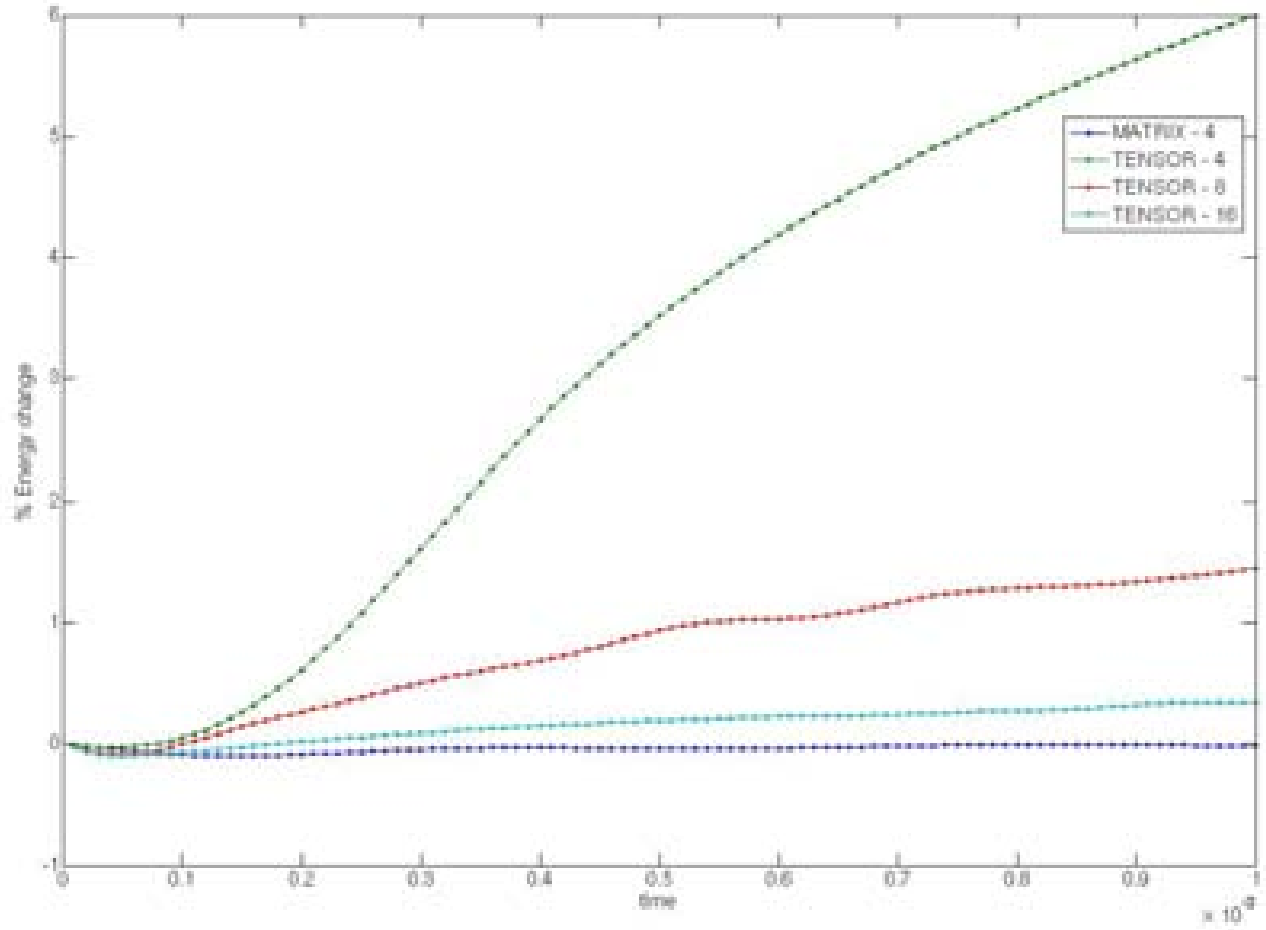
- **All differentiable computations are composition of simple operations [sin(), log(), +, *, /, etc…]**
- **We know the derivatives of these simple operations**
- **Use chain/product rules from calculus**
- **Take "derivative of the code"**
- **Differentiate anything w.r.t. anything without trouble!**

$$\mathrm{F}_k = -\frac{\partial E_{\mathrm{mag}}}{\partial \mathrm{x}_k} = \frac{\partial}{\partial \mathrm{x}_k}\left\{-\frac{1}{2\mu}\int \mathrm{B}\cdot\mathrm{B}dx\right\}$$

Sandia National Laboratories

2-D Results

potential $A(x, y) = x^2 y^2 \mathbf{k}$



coarse mesh (T)

medium mesh (T)

fine mesh (T)

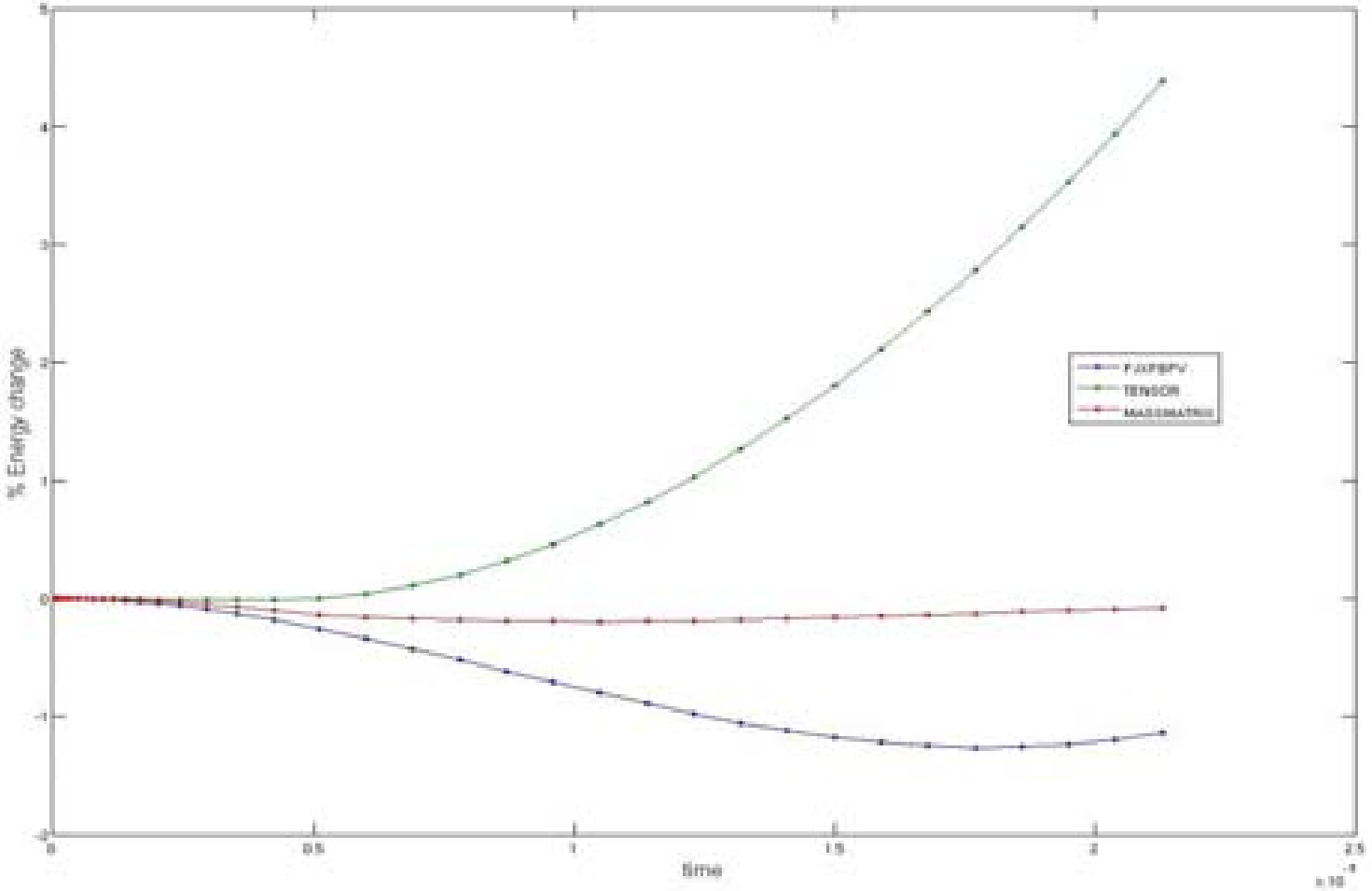
coarse mesh (M)

Sandia National Laboratories

3-D Results

$$A(x, y, z) = \sin((xyz)^6)\mathbf{i} + \cos((xyz)^6)\mathbf{j} + \tan((xyz)^6)\mathbf{k}$$

Mesh: 16 x 16 x 16 Cube



TENSOR

MATRIX

PJXPBPV

Sandia National Laboratories

The Catch: FLOPS

- **Promising results, but high expense**
- **2-D: MATRIX = ~100 x TENSOR**
- **3-D: MATRIX = ~600 x TENSOR**
- **Cost is almost entirely within matrix creation, taking of derivatives (8 in 2-D, 24 in 3-D)**
- **These must be computed at each element, at each time step**
- **Ongoing work: improve efficiency of Intrepid, Sacado**
- **Utilize symmetry of matrices/computations**
- **Allow for less accurate cubature**
- **Cost may not be as noticeable for non-ideal MHD or Eulerian modeling**

Sandia National Laboratories

Summary

- **The idea of using derivatives of the magnetic energy functional in association with a C++ automatic differentiation library works well and gives much improved energy conservation.**
- **There are clearly significant performance issues for ideal MHD modeling**
- **Cost may not be such a large issue for resistive MHD for full Eulerian ALE models because of the cost of matrix solves and remapping costs.**

Sandia National Laboratories

Acknowledgments

- **Allen Robinson, Randy Summers**
- **Pavel Bochev, Denis Ridzal**
- **Eric Phipps**
- **David Day**
- **CSRI**

Sandia National Laboratories