SAND2012-0792 C

SAND2012-0792C

Mantevo project

Analyzing and improving performance via Mini-Applications

Project Goals:

- Predict performance of real applications
- Foster communication between developers of applications, libraries and computer systems
- Guide design choices for both computer systems and application software

Mike Heroux et al.,
Sandia National Laboratories

Mini-applications:

- Represent key performance characteristics of “real” applications
- Small, self-contained, easily ported to new systems
- Freely available as open-source

- **miniFE**
- HPCCG
- miniGhost
- miniMD
- phdMesh
- miniXyce

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy’s National Nuclear Security Administration under contract DE-AC04-94AL85000.

"miniFE" is a Finite-Element mini-application

Implements algorithms from an implicit finite-element application

- Assemble a sparse linear-system from the steady-state conduction equation on a domain of hexahedral elements.

$$Ku = f \qquad K = \sum_e K^e \qquad K^e = \int_{\Omega_e} \frac{\partial \psi}{\partial x_m} k_{mn} \frac{\partial \psi}{\partial x_n} dx$$

$$f = \sum_e Q^e \qquad Q^e = \int_{\Omega_e} \psi Q dx$$

- Solve the linear-system using the Conjugate Gradient algorithm:
  - Per iteration:
    - 2 dot-products
    - 3 axpys
    - 1 matrix-vector product

$r_0 = b - Ax_0$

Loop {

If ***k*** **== 1**

$p_1 = r_0$

else

$\beta_k = r_{k-1}^T r_{k-1} / r_{k-2}^T r_{k-2}$

$p_k = r_{k-1} + \beta_k p_{k-1}$

$\alpha_k = r_{k-1}^T r_{k-1} / p_k^T A p_k$

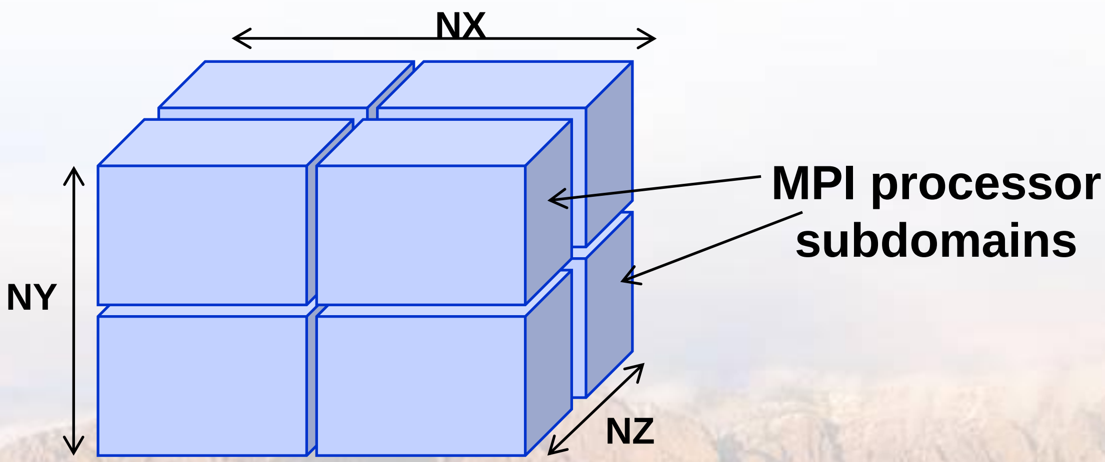
$x_k = x_{k-1} + \alpha_k p_k$

$r_k = r_{k-1} - \alpha_k A p_k$

}

Finite Element mini-app: miniFE

- **miniFE sets up a brick-shaped problem domain of hexahedral elements**
  - Parameters **nx, ny, nz** specify global number of elements in each dimension
  - Global number of equations in linear-system corresponds to finite element vertices (nodes): (nx+1)(ny+1)(nz+1)
- **RCB partitioning splits the global problem domain among MPI processors**
- **Vertices on processor boundaries are shared**



miniFE code origins, overview

- **Evolved from Mike Heroux's HPCCG**
  - Small, portable, self-contained implementation of linear Conjugate Gradient solve.
  - MPI parallel, with many derivative implementations that explored threading, etc.

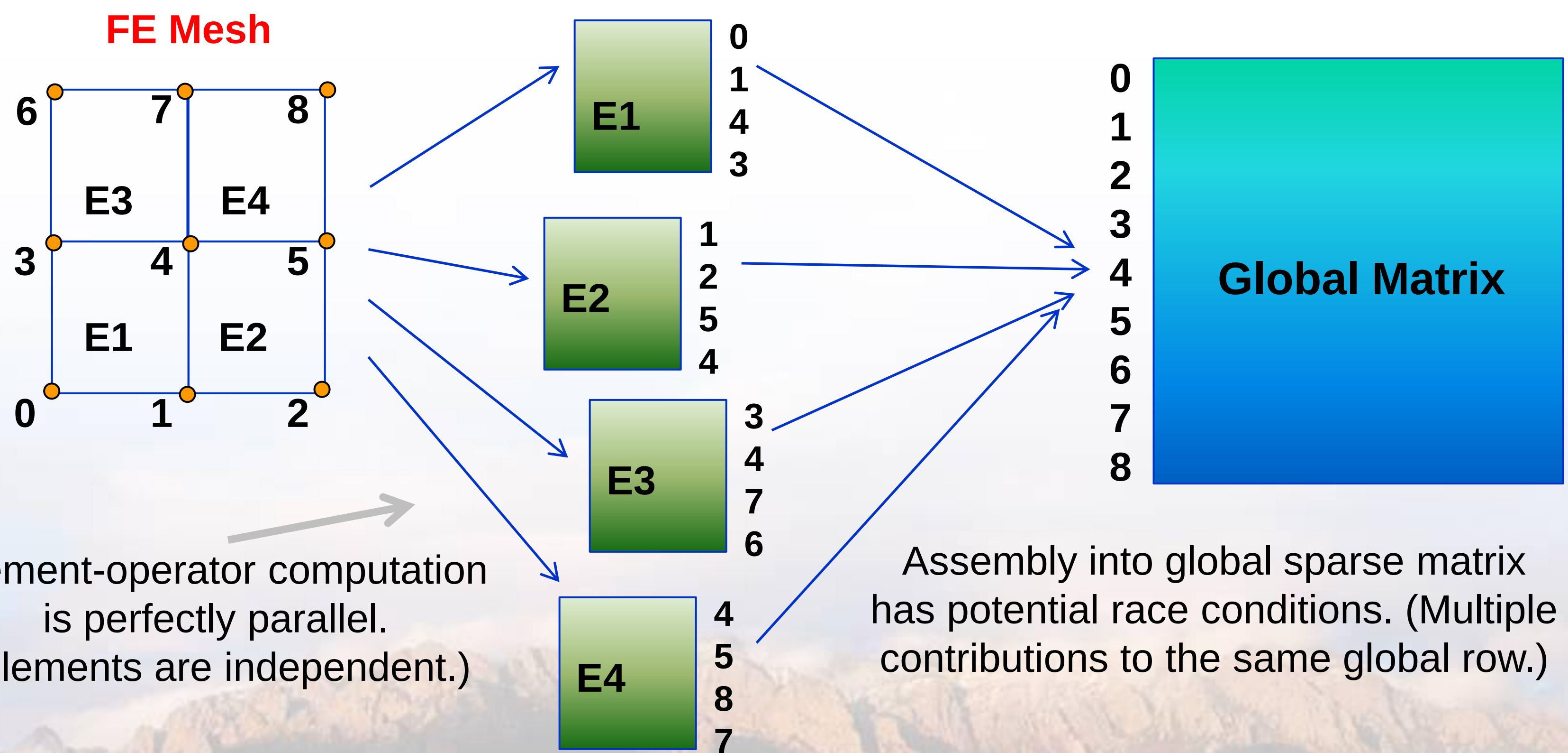
- **Evolution to miniFE added several features:**
  - Finite-Element assembly from conduction equation
  - Optionally store mesh data in Sierra Toolkit Mesh (STK Mesh)
  - Entire code is parameterized (C++ templates) on floating-point and integer types
  - ComputeNode programming model (Baker et. al.) provides abstract interface to fine-grained parallelism (CPU threading, GPU co-processing).

Description of Finite-Element linear-system assembly: for each element in the mesh, 3 operations

1. Get node-ids & coordinates

2. Compute dense element-operators

3. Scatter-assemble into global sparse linear-system



Element-operator computation is perfectly parallel. (Elements are independent.)

Assembly into global sparse matrix has potential race conditions. (Multiple contributions to the same global row.)