LA-UR- 11-01410

*Approved for public release; distribution is unlimited.*

| | |
|---|---|
| *Title:* | Neutron Transport in the Capsaicin Project |
| *Author(s):* | Massimiliano Rosa<br>Computational Physics (CCS-2) |
| *Intended for:* | CASL Transport PIs Meeting<br>March 7, 2011<br>Oak Ridge National Laboratory<br>Oak Ridge, TN |

Los Alamos
NATIONAL LABORATORY
EST.1943



Form 836 (7/06)

**Neutron Transport in the Capsaicin Project**

*Massimiliano Rosa*
*Computational Physics (CCS-2)*
*Los Alamos National Laboratory*

CASL Transport PIs Meeting
Oak Ridge National Laboratory
Oak Ridge, TN, March 7th, 2011

## ABSTRACT

We review the neutron transport capabilities available in the Capsaicin code package developed and maintained by the Capsaicin Team in the Computational Physics Group (CCS-2) at Los Alamos National Laboratory. We discuss how the above capabilities are provided to the VERA-TR neutronics code framework being developed as part of the Consortium for the Advanced Simulation of Light water reactors (CASL) effort within the Department of Energy's Nuclear Energy Modeling and Simulation Hub.

# Neutron Transport in the Capsaicin Project

Massimiliano Rosa (maxrosa@lanl.gov)

Capsaicin Project Team

Kent Budge, Jae Chang, Erin Fichtl, Max Rosa, Jim Warsa

Computational Physics (CCS-2)

CASL Transport PIs Meeting

Oak Ridge National Laboratory

March 7th, 2011

Los Alamos NATIONAL LABORATORY EST.1943

LA-UR-11-????

NNSA

# Outline

1. Capsaicin within CASL
2. Review of Capsaicin neutron transport capabilities
3. Status update

# 1. Capsaicin within CASL

- Provide parallel multigroup neutron transport capability on unstructured meshes to VERA-TR
- Provide interface specification and libraries for VERA-TR to access this capability
- Interface specification
  - Mesh description and parallel decomposition
  - Material properties and related data
  - Iteration and solution method controls

# 2. Review of Capsaicin capabilities

A. Data

B. Problems

   1. Fixed-source and adjoint

   2. Eigenvalue

C. Algorithms

   1. Solvers

   2. Sweepers

   3. Discretizations

# A. Nuclear data

- LANL nuclear data interface (NDI) cross-section libraries
- Analytic cross-section data
- ANISN cross-section data
- User-defined cross-section data (via interface)

# B. Steady-state fixed-source problem

- Operator notation

$$\mathbf{L}\psi = \mathbf{MS}\phi + q, \quad \phi = \mathbf{D}\psi$$

- $M$ discrete ordinates, $G$ energy groups, $N$ moments
- $\phi$ vector of length $NG$, $\psi$ and $q$ vectors of length $MG$
- **L** is ($MG \times MG$), **M** is ($MG \times NG$), **S** is ($NG \times NG$) and **D** is ($NG \times MG$)
- Linear system formulation of fixed-source problem

$$\mathbf{A}\phi = b$$

$$\mathbf{A} = \left(\mathbf{I} - \mathbf{DL}^{-1}\mathbf{MS}\right) \qquad b = \mathbf{DL}^{-1}q$$

# B. Eigenvalue problem

- Operator notation: $\mathbf{L}\psi = \mathbf{M}\left(\mathbf{S} + \frac{1}{k}\mathbf{F}\right)\phi$
- **F** is (*NG*×*NG*)
- Maximum eigenvalue *k* is "critical" eigenvalue of interest
- Generalized eigenproblem: $\mathbf{A}\phi = \frac{1}{k}\mathbf{B}\phi$
- Standard eigenproblem: $\mathbf{A}^{-1}\mathbf{B}\phi = k\phi$

$$\mathbf{A} = \left(\mathbf{I} - \mathbf{D}\mathbf{L}^{-1}\mathbf{M}\mathbf{S}\right) \qquad \mathbf{B} = \mathbf{D}\mathbf{L}^{-1}\mathbf{M}\mathbf{F}$$

# C. Solvers

- Linear solvers for fixed-source problem are parallel non symmetric Krylov solvers
  - Source (Richardson) iteration
  - GMRES(m)
  - BiCGStab
- Preconditioners for linear fixed-source problem
  - Diffusion Synthetic Acceleration (DSA)
  - Transport Synthetic Acceleration (TSA)
  - Linear Multi-frequency Grey (LMFG)
  - Full DSA (FDSA)

# C. Solvers

- Eigenvalue problem as non-linear fixed point iteration

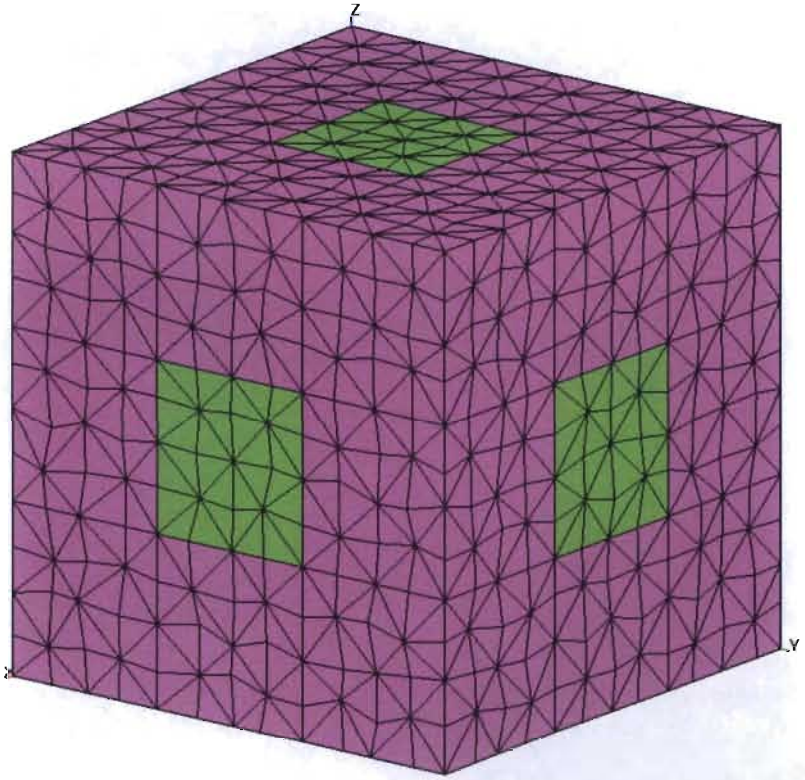
$$u_{z+1} = u_z - F\left(u_z\right)$$

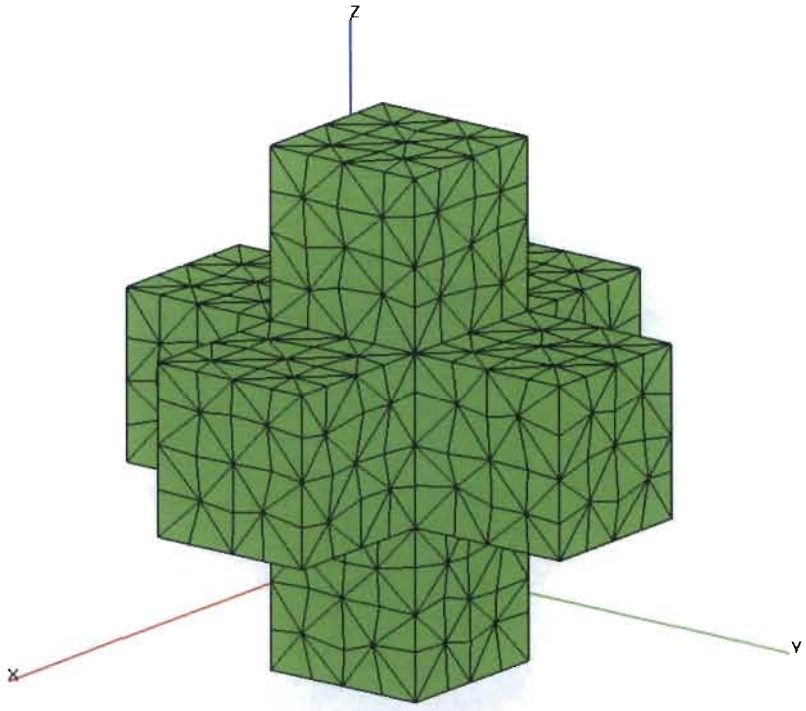
$$u = \begin{bmatrix} \phi \\ k \end{bmatrix} \qquad F\left(u\right) = \begin{bmatrix} \left(\mathbf{I}-\mathbf{P}\right)\phi \\ k - k\dfrac{E^T\mathbf{F}\mathbf{P}\phi}{E^T\mathbf{F}\phi} \end{bmatrix} \qquad \mathbf{P} = \mathbf{A}^{-1}\mathbf{B}$$

- Non-linear solvers for eigenvalue problem
  - Fixed-point (Picard) iteration
  - Jacobian-Free Newton-Krylov (JFNK) method
  - Carlson-Miller method

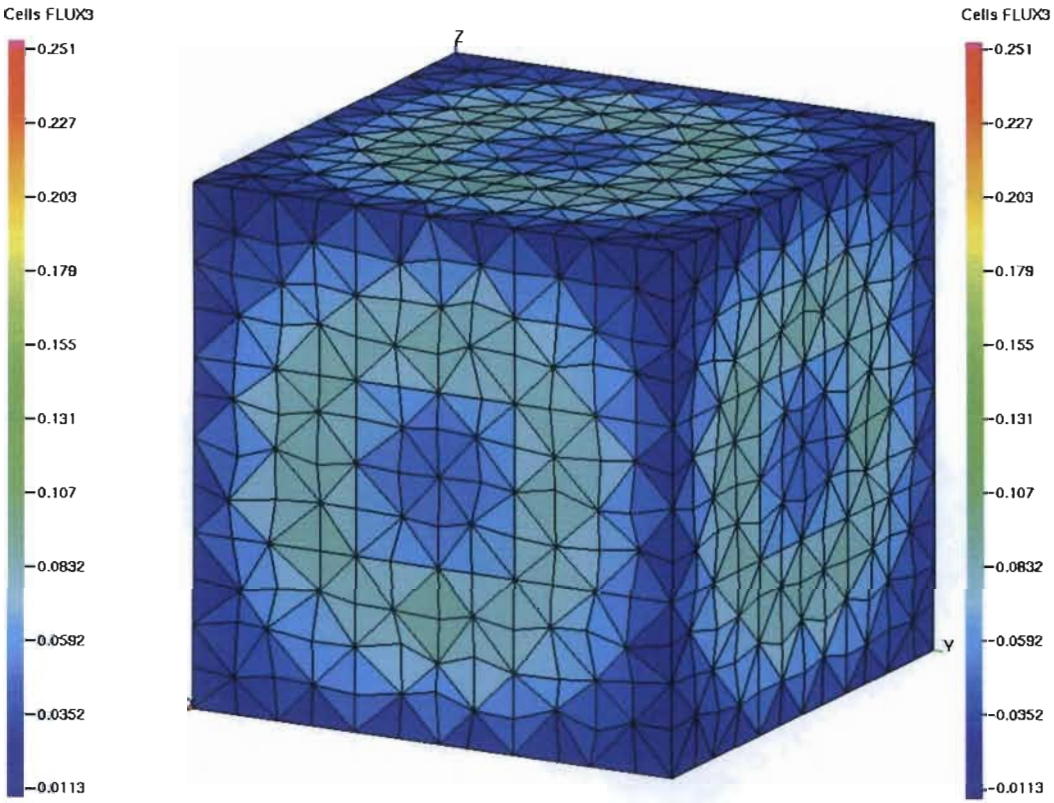
# C. Solvers

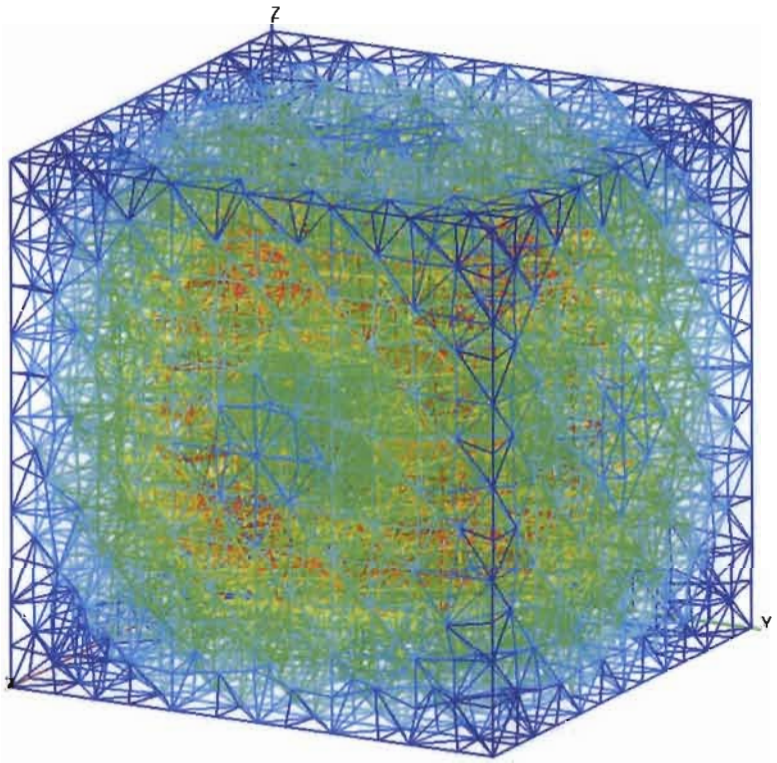
- 3D JFNK eigenvalue computation
- HEU in central region surrounded by water



# C. Solvers


Cells FLUX3
0.251
0.227
0.203
0.179
0.155
0.131
0.107
0.0832
0.0592
0.0352
0.0113
Z
Y
Cells FLUX3
0.251
0.227
0.203
0.179
0.155
0.131
0.107
0.0832
0.0592
0.0352
0.0113


Z
Y

# C. Sweepers

- Full Parallel Sweeps (FPS): apply $\mathbf{L}^{-1}$ to a vector at every Krylov iteration

$$\psi^{(\ell+1)} = \mathbf{L}^{-1}\mathbf{MSD}\psi^{(\ell)}$$

- A "sweep schedule" is used to order the cell-wise sweep through a parallel-decomposed mesh for every angle
- Various scheduling options for **unstructured** 2D/3D meshes
  - RANDOM: choose priorities randomly
  - B_LEVEL/BFDS/DFDS/DFHDS: strategies based on b-levels
  - LDS: determine priority based on minimum number of graph edges to processor boundary

# C. Sweepers

- Max edges per communication: how much data is accumulated before communicating it to downwind processor
- Max visits per receive check: how often to probe for incoming upwind information
- Group set size: number of energy groups in a single task
- Level/group sequencing: finish a level/group before moving on to next, or do all simultaneously?

# C. Sweepers

- Scheduling options for unstructured meshes can be non optimal for structured meshes
- Koch - Baker - Alcouffe (KBA) scheduling algorithm implemented for orthogonal 2D and 3D meshes

# C. Sweepers

- Inexact Parallel block-Jacobi (IPBJ)
  - Split **L** into local and interprocessor boundary: $\mathbf{L} = \mathbf{L}_o + \mathbf{L}_b$
  - $\psi^{(\ell+1)} = \mathbf{L}_o^{-1}\left(\mathbf{MSD} - \mathbf{L}_b\right)\psi^{(\ell)}$
  - Implemented for unstructured 2D meshes
- Cell-wise block-Gauss-Seidel (bGS)
  - Split **L** into cell-interior and cell-boundary: $\mathbf{L} = \mathbf{L}_c + \mathbf{L}_b$
  - $\psi^{(\ell+1)} = -\left(\mathbf{L}_c - \mathbf{MSD}\right)^{-1}\mathbf{L}_b\psi^{(\ell)}$
  - Implemented on Roadrunner hybrid computer architecture for unstructured 2D meshes
- Communication delays associated with FPS are eliminated

# C. Discretizations

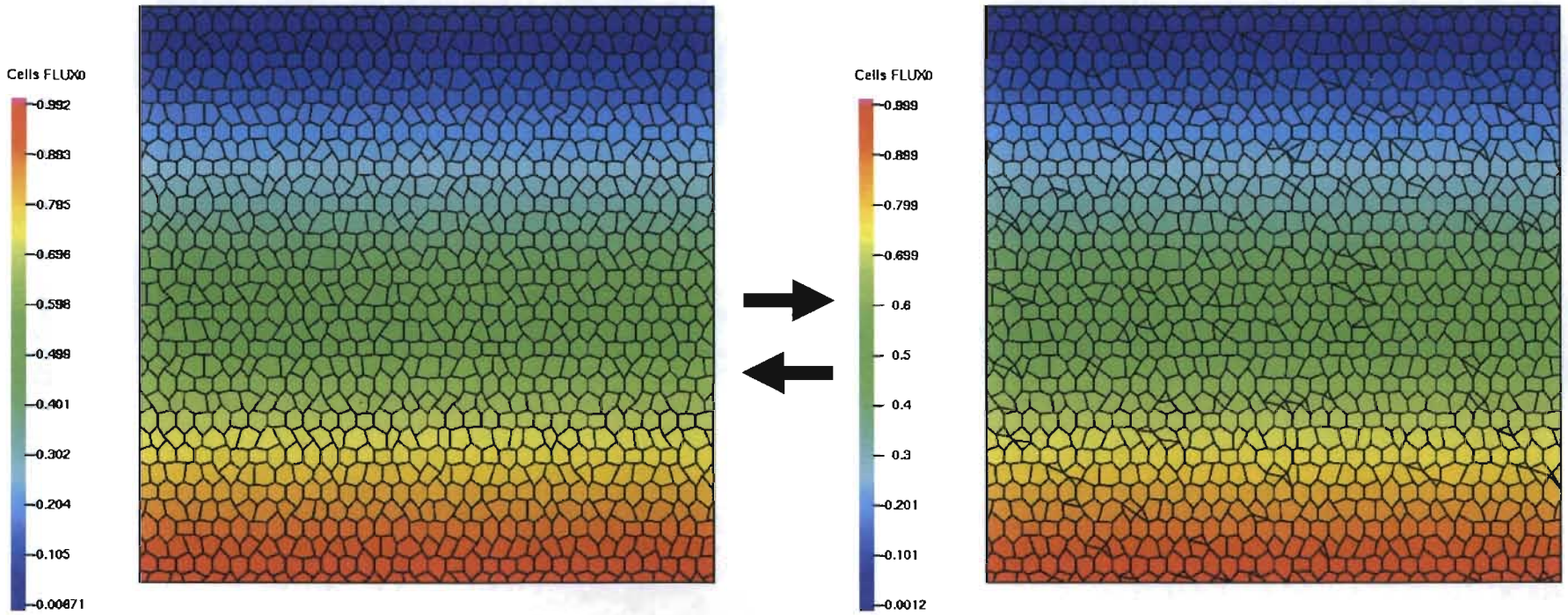
- Energy dependence
  - Multigroup approximation
  - Parallel energy decomposition
- Angular dependence
  - Discrete ordinates ($S_N$)
  - Spherical harmonics ($SP_N$)
  - Diffusion
  - Quasi-Diffusion

# C. Discretizations

- DFEM for unstructured meshes in 1D (slab, spherical, cylindrical), 2D (XY, RZ) and 3D (XYZ)
  - Linear Bars, Triangles, Quadrilaterals, Polygons (CFEM-based DFEM)
  - Quadratic Bars, Triangles
  - Linear Tetrahedra
- DFEM available both lumped and un-lumped
- Structured meshes in 3D
  - Diamond difference (DD)
  - Lumped linear discontinuous (LD)

# C. CFEM-based DFEM

- Capability to detect non-convex mesh, compute on convex mesh and project onto non-convex mesh




non-convex mesh

convex mesh

# 3. Update on L4 milestones

1. SN.01.05 - Capsaicin source code deposited into the CASL repository
2. SN.01.06 - Capsaicin successfully running its neutronics acceptance tests on leadership architecture at ORNL
3. SN.01.07 - VERA-TR successfully running Capsaicin's neutronics acceptance tests on unstructured meshes
4. SN.01.08 - VERA-TR running a 2D lattice physics benchmark problem with the same data on both structured and unstructured meshes