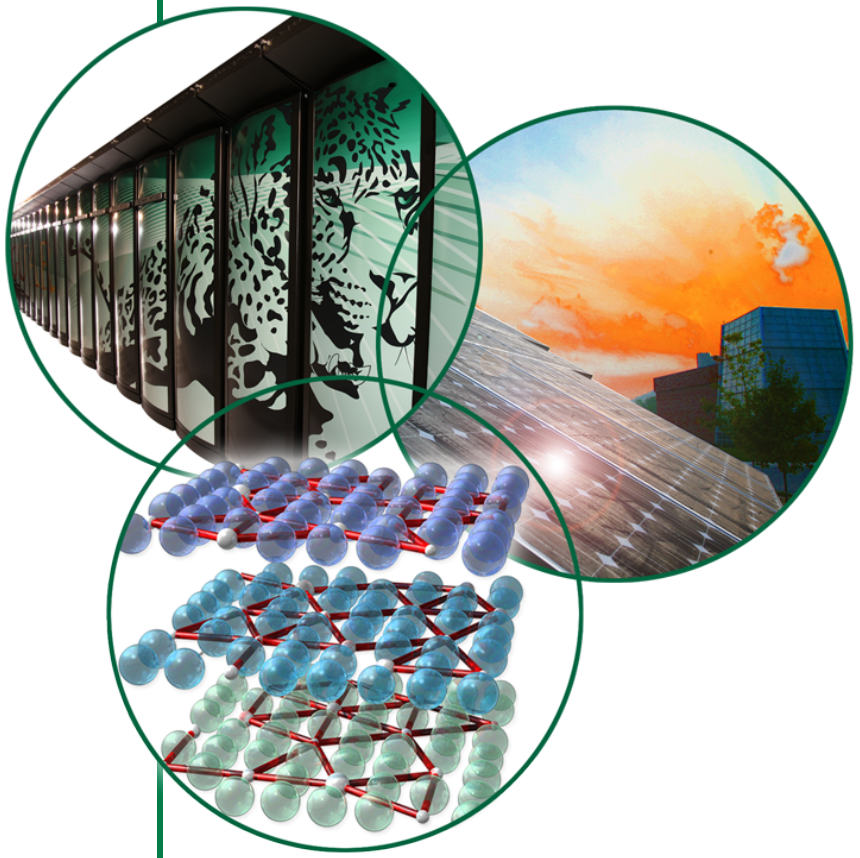
Denovo: A radiation transport code for nuclear applications

Thomas M. Evans

SciDAC SciADS 2010, Snowbird UT



U.S. DEPARTMENT OF ENERGY

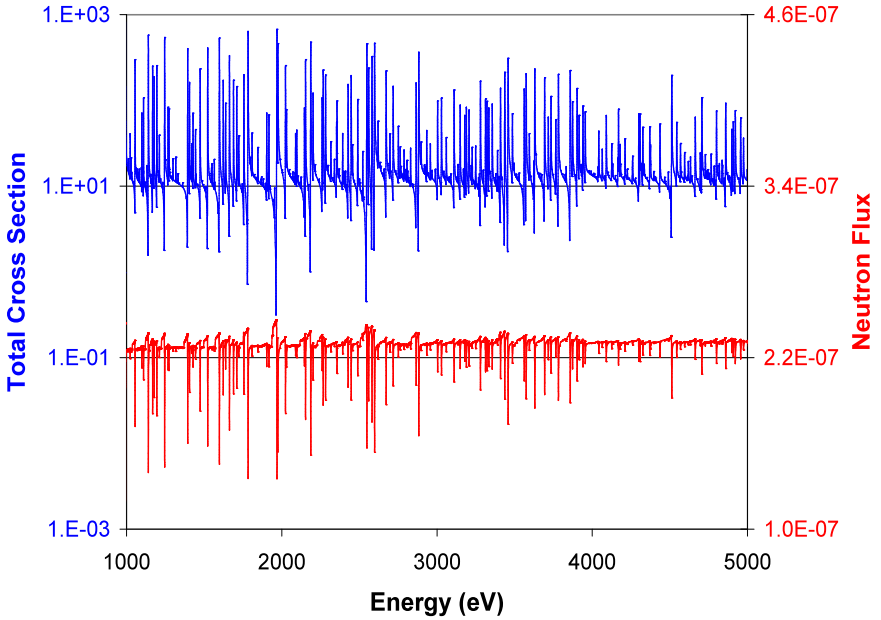
OAK RIDGE NATIONAL LABORATORY

MANAGED BY UT-BATTELLE FOR THE DEPARTMENT OF ENERGY

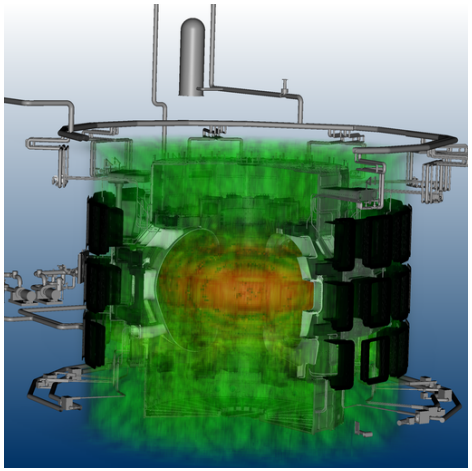
Science Drivers for Neutronics

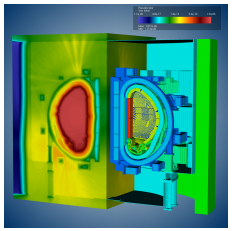
- **Spatial resolution**
  - **To resolve the geometry**
    - **$10^{9\text{-}12}$ unknowns**
    - **mm^3 cells in a m^3 vessel**
  - **Depletion makes it harder**
- **Energy resolution**
  - **To resolve resonances**
    - **$10^{4\text{-}6}$ unknowns**
    - **Done in 0D or 1D today**
- **Angular resolution**
  - **To resolve streaming**
    - **$10^{2\text{-}4}$ unknowns**
  - **Space-energy resolution make it harder**

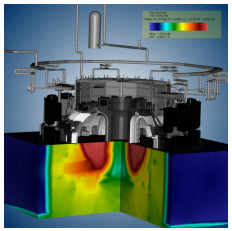
- **Crud and distortion**
- **Control rod insertion**
- ***Ab initio* design**



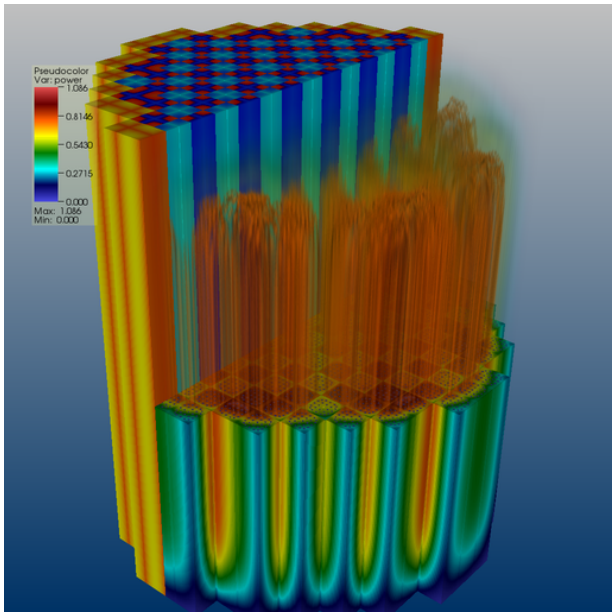
Application Areas



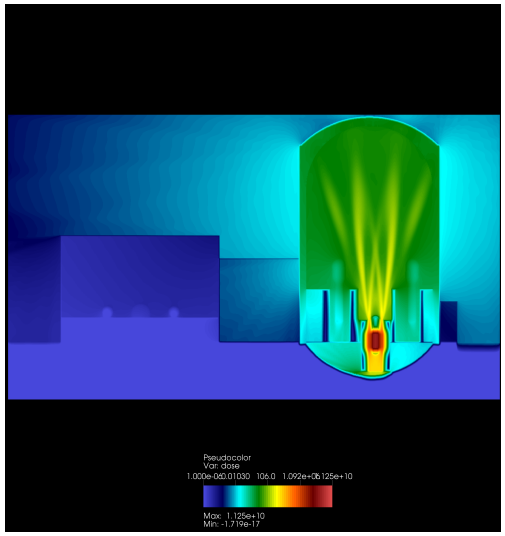




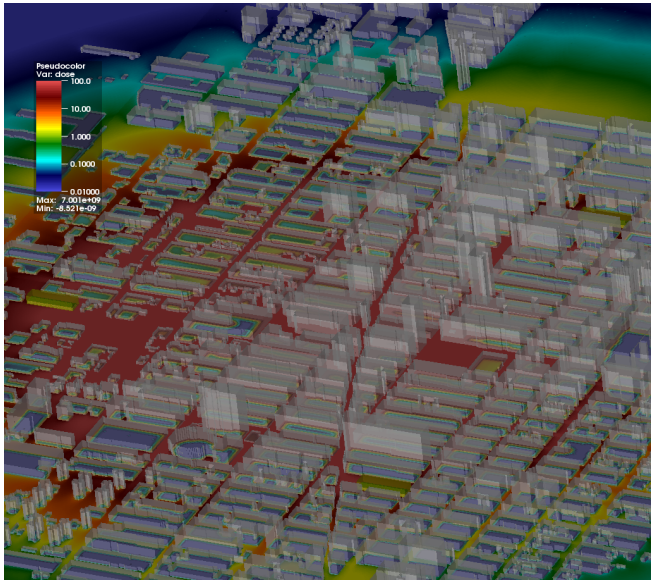
ITER component performance/shielding



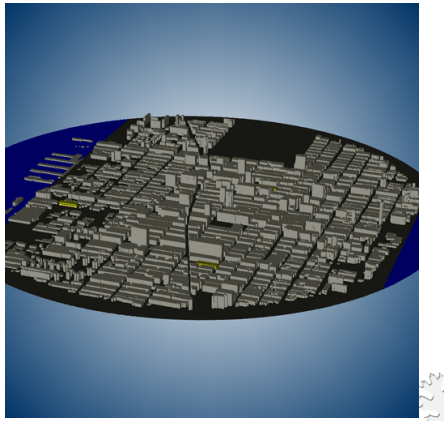
Reactor Analysis



Facility shielding/dosimetry

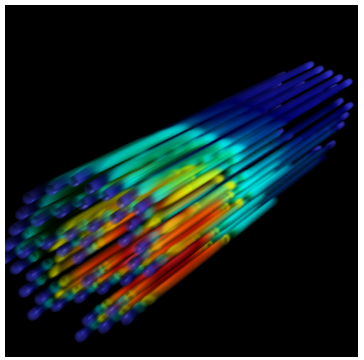


National defense/
Urban modeling



Denovo Capabilities

- State of the art transport methods
  - 3D, non-uniform, regular grid SN
  - Multigroup energy, anisotropic Pn scattering
  - Forward/Adjoint
  - Fixed-source/*k*-eigenvalue
  - 6 spatial discretization algorithms
    - Linear and Trilinear discontinuous FE, step-characteristics, theta-weighted diamond, weighted diamond + flux-fixup
  - Parallel first-collision
    - Analytic ray-tracing (DR)
    - Monte Carlo (DR and DD)
  - Multiple quadratures
    - Level-symmetric
    - Generalized Legendre Product
    - Galerkin



- Modern, Innovative, High-Performance Solvers
  - Within-group solvers
    - Krylov (GMRES, BiCGStab) and source iteration
    - DSA preconditioning (SuperLU/ML-preconditioned CG/PCG)
  - Multigroup solvers
    - Transport Two-Grid upscatter acceleration of Gauss-Seidel
    - Krylov (GMRES, BiCGtab)
  - Eigenvalue solvers
    - Power iteration (with rebalance)
      - CMFD in testing phase
    - Krylov (Arnoldi)
    - Shifted-inverse iteration in development

Power distribution in a BWR assembly

Denovo Capabilities

- **Parallel Algorithms**
  - **Koch-Baker-Alcouffe (KBA) wavefront decomposition**
  - **Domain-replicated (DR) and domain-decomposed first-collision solvers**
  - **Multilevel energy decomposition in development**
  - **Parallel I/O built on SILO/HDF5**

> 5M CPU hours on Jaguar with 2 bugs

2010 INCITE Award
Uncertainty Quantification for Three Dimensional Reactor Assembly Simulations, 8 MCPU-HOURS
2010 ASCR Joule Code
2009-2011 2 ORNL LDRDs

- **Advanced visualization, run-time, and development environment**
  - **3 front-ends (HPC, SCALE, Python-bindings)**
  - **Direct connection to SCALE geometry and data**
  - **Direct connection to MCNP input through ADVANTG**
  - **HDF5 output directly interfaced with VisIt**
  - **Built-in unit-testing and regression harness with DBC**
  - **Emacs-based code-development environment**
  - **Support for multiple external vendors**
    - **GSL, BLAS/LAPACK, TRILINOS (required)**
    - **BRLCAD, SUPERLU/METIS, SILO/HDF5 (optional)**
    - **MPI (toggle for parallel/serial builds)**
    - **SPRNG (required for MC module)**
    - **PAPI (optional instrumentation)**

Discrete Ordinates Methods

- **We solve the first-order form of the transport equation:**
  - **Eigenvalue form for multiplying media (fission):**

$$\hat{\mathbf{\Omega}} \cdot \nabla \psi(\mathbf{r}, \hat{\Omega}, E) + \sigma(\mathbf{r}, E)\psi(\mathbf{r}, \hat{\Omega}, E) = \int dE' \int_{4\pi} d\hat{\Omega}' \; \sigma_{\mathrm{s}}(\mathbf{r}, \hat{\mathbf{\Omega}}' \cdot \mathbf{\Omega}, E' \rightarrow E)\psi(\mathbf{r}, \hat{\Omega}', E') + \frac{1}{k}\frac{\chi(E)}{4\pi} \int dE' \int_{4\pi} d\hat{\Omega}' \; \nu\sigma_{\mathrm{f}}(\mathbf{r}, E')\psi(\mathbf{r}, \hat{\Omega}', E')$$

  - **Fixed source form:**

$$\hat{\mathbf{\Omega}} \cdot \nabla \psi(\mathbf{r}, \hat{\Omega}, E) + \sigma(\mathbf{r}, E)\psi(\mathbf{r}, \hat{\Omega}, E) = \int dE' \int_{4\pi} d\hat{\Omega}' \; \sigma_{\mathrm{s}}(\mathbf{r}, \hat{\mathbf{\Omega}}' \cdot \mathbf{\Omega}, E' \rightarrow E)\psi(\mathbf{r}, \hat{\Omega}', E') + \frac{1}{4\pi} q_{\mathrm{e}}(\mathbf{r}, \hat{\Omega}, E)$$

Discrete Ordinates Methods

- The S_N method is a collocation method in angle.
  - Energy is discretized in groups.
  - Scattering is expanded in Spherical Harmonics.
  - Multiple spatial discretizations are used (DGFEM, Characteristics, Cell-Balance).

$$\mathbf{L}\psi = \mathbf{MS}\phi + Q$$

$$\phi = \mathbf{D}\psi$$

- Dimensionality of operators:

$$t = N_g \times N_c \times N_u \times N_m$$

$$n = N_g \times N_c \times N_u \times N_a$$

$$(n \times n)(n \times 1) = (n \times t)(t \times t)(t \times 1) + (n \times 1)$$

Degrees of Freedom

- **Total number of unknowns in solve:**

$$\text{unknowns} = N_g \times N_c \times N_u \times N_a \times N_m$$

- **An ideal (conservative) estimate.**

$$N_g = 238$$
$$N_c = 1 \times 10^9$$
$$N_u = 4$$
$$N_m = 16$$
$$N_a = 288$$

$$\boxed{\text{unknowns} \geq 4 \times 10^{15}}$$

Solver Taxonomy

The innermost part of each solver are transport sweeps

$$y = \mathbf{T}z = \mathbf{D}\underbrace{\mathbf{L}^{-1}z}$$

$$\mathbf{L}\psi = z$$

"It's turtles all the way down…"

Eigenvalue Solvers

Power iteration
Arnoldi
Shifted-inverse

Multigroup Solvers

Gauss-Seidel
Residual Krylov
Gauss-Seidel + Krylov

Within-group Solvers

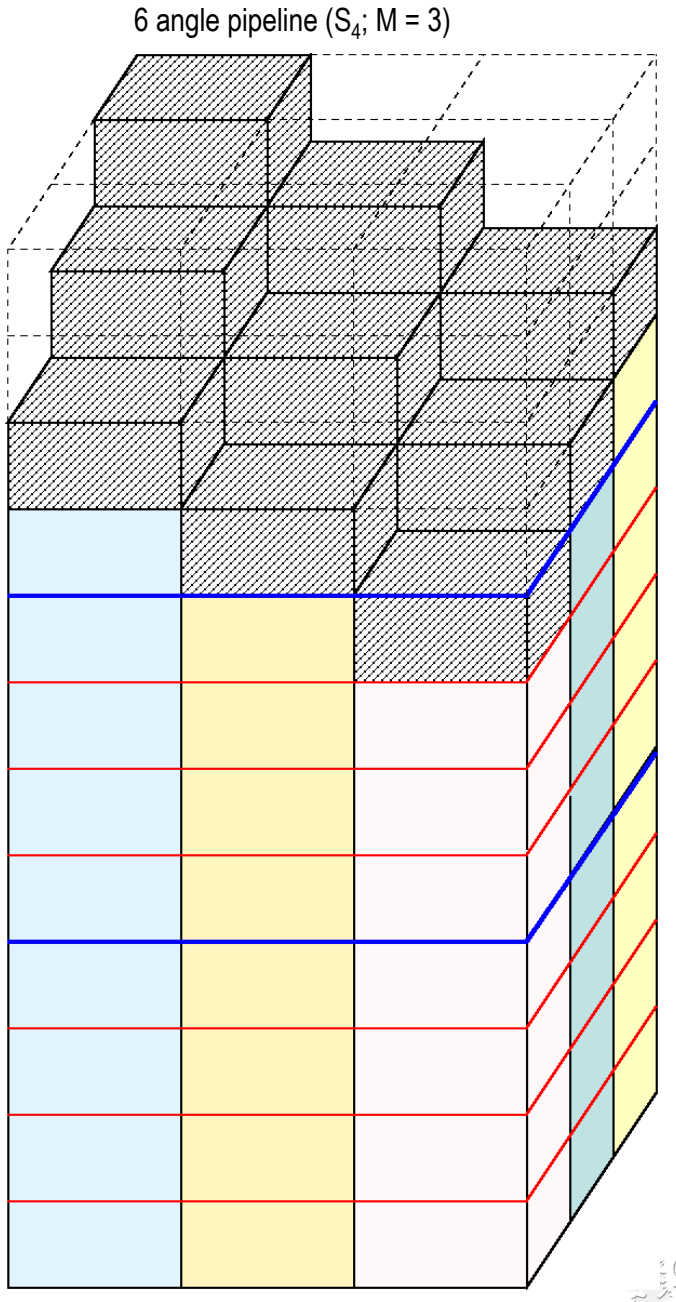
Krylov
Residual Krylov
Source iteration

Parallel Performance

Angular Pipelining

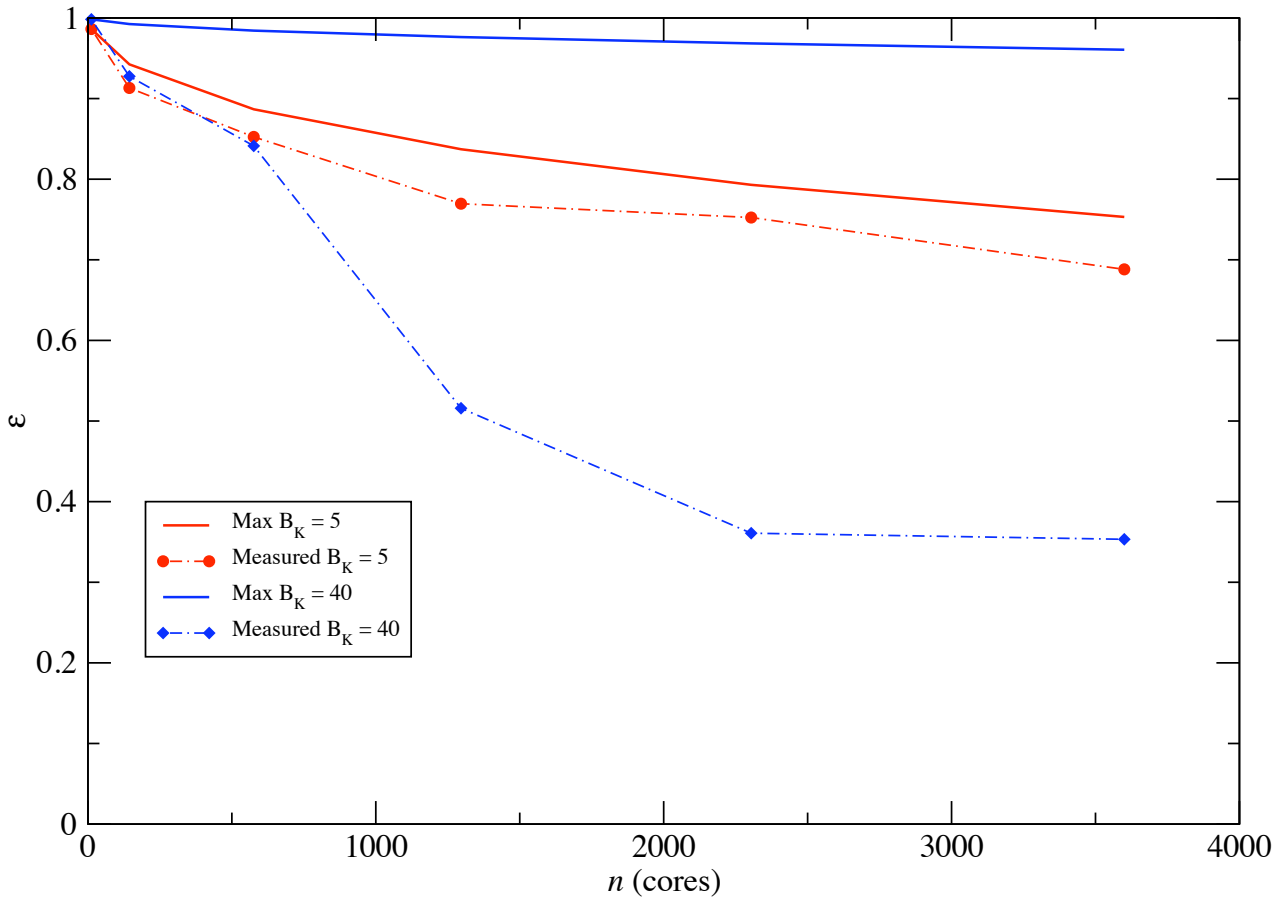
- Angles in ± z directions are pipelined
- Results in 2×M pipelined angles per octant
- Quadrants are ordered to reduce latency

$$\epsilon_{\max} = \frac{2MB_K}{2MB_K + P_I + P_J - 2}$$



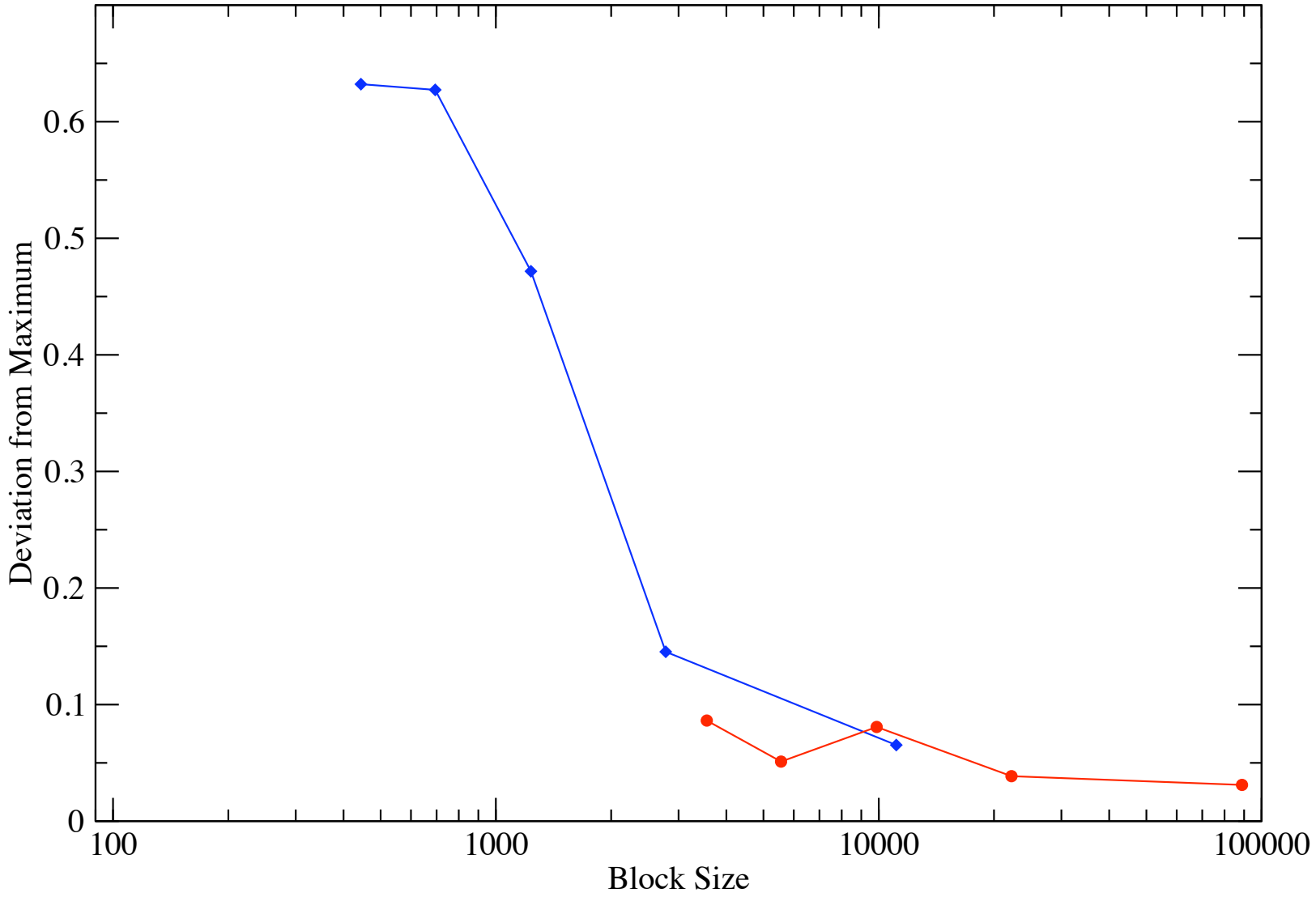
KBA Reality

KBA does not achieve close to the predicted maximum



- **Communication latency dominates as the block size becomes small**
- **Using a larger block size helps achieve the predicted efficency but,**
  - **Maximum achievable efficiency is lower**
  - **Places a fundamental limit on the number of cores that can be used for any given problem**

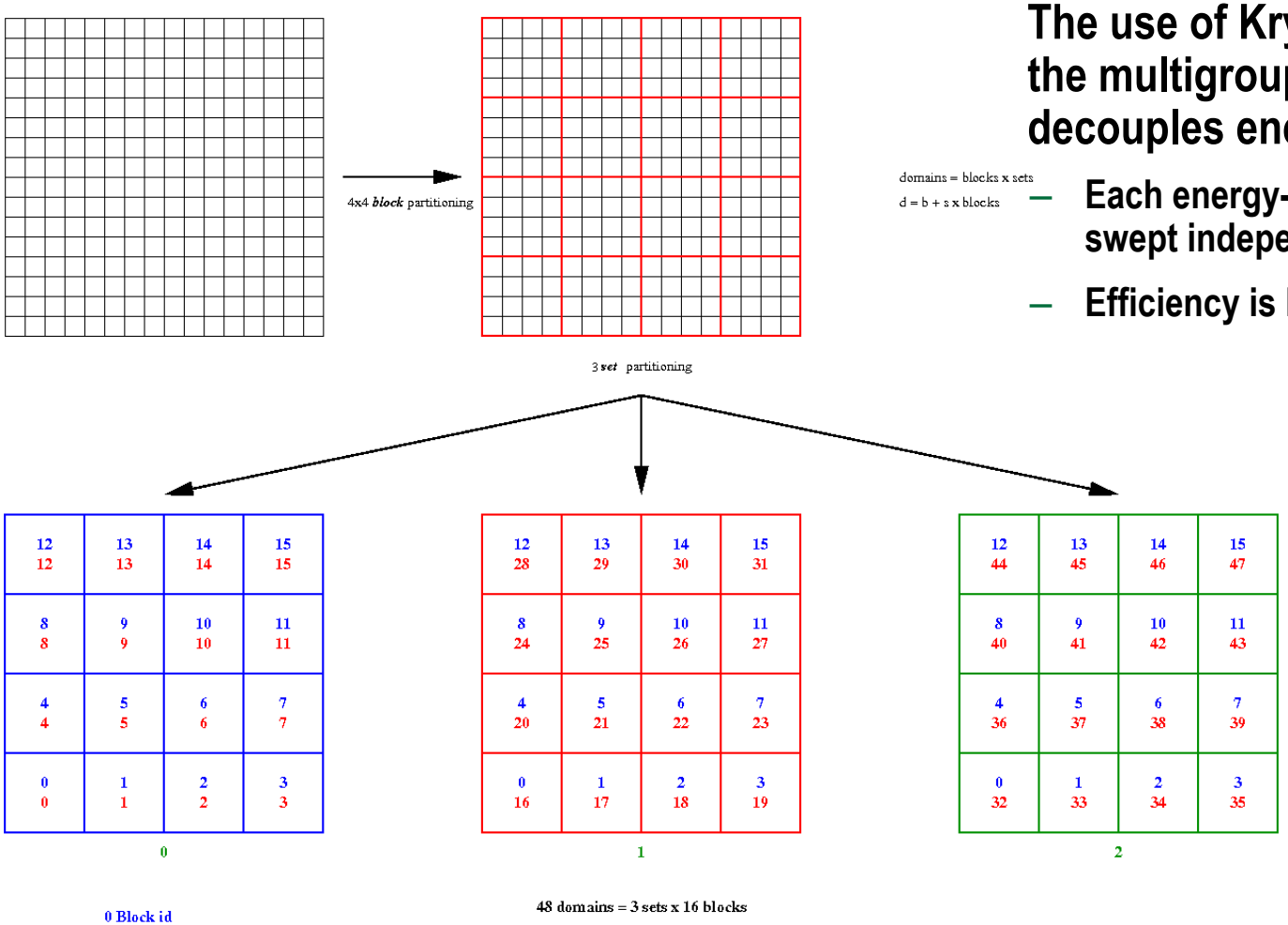
Efficiency vs Block Size



Overcoming Wavefront Challenge

- This behavior is systemic in any wavefront-type problem
  - Hyberbolic aspect of transport operator
- We need to exploit parallelism beyond space-angle
  - Energy
  - Time
- Amortize the inefficiency in KBA while still retaining direct inversion of the transport operator

Multilevel Energy Decomposition



The use of Krylov methods to solve the multigroup equations effectively decouples energy

- Each energy-group S_N equation can be swept independently
- Efficiency is better than Gauss-Seidel

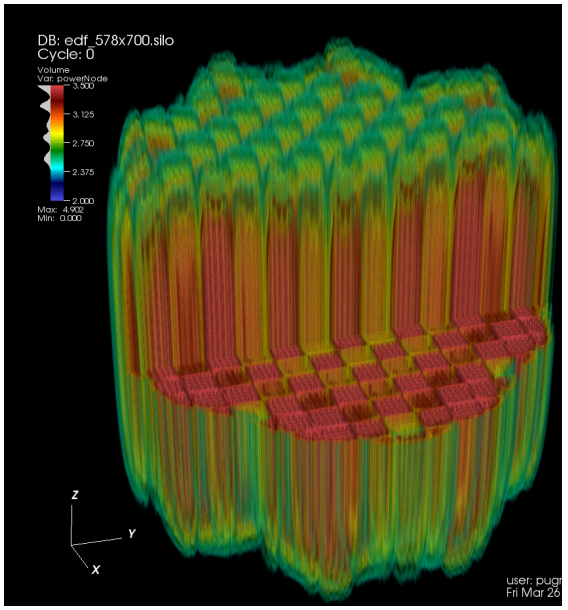
Multilevel Summary

- Energy decomposed into sets.
- Each set contains blocks constituting the entire spatial mesh.
- The total number of domains is

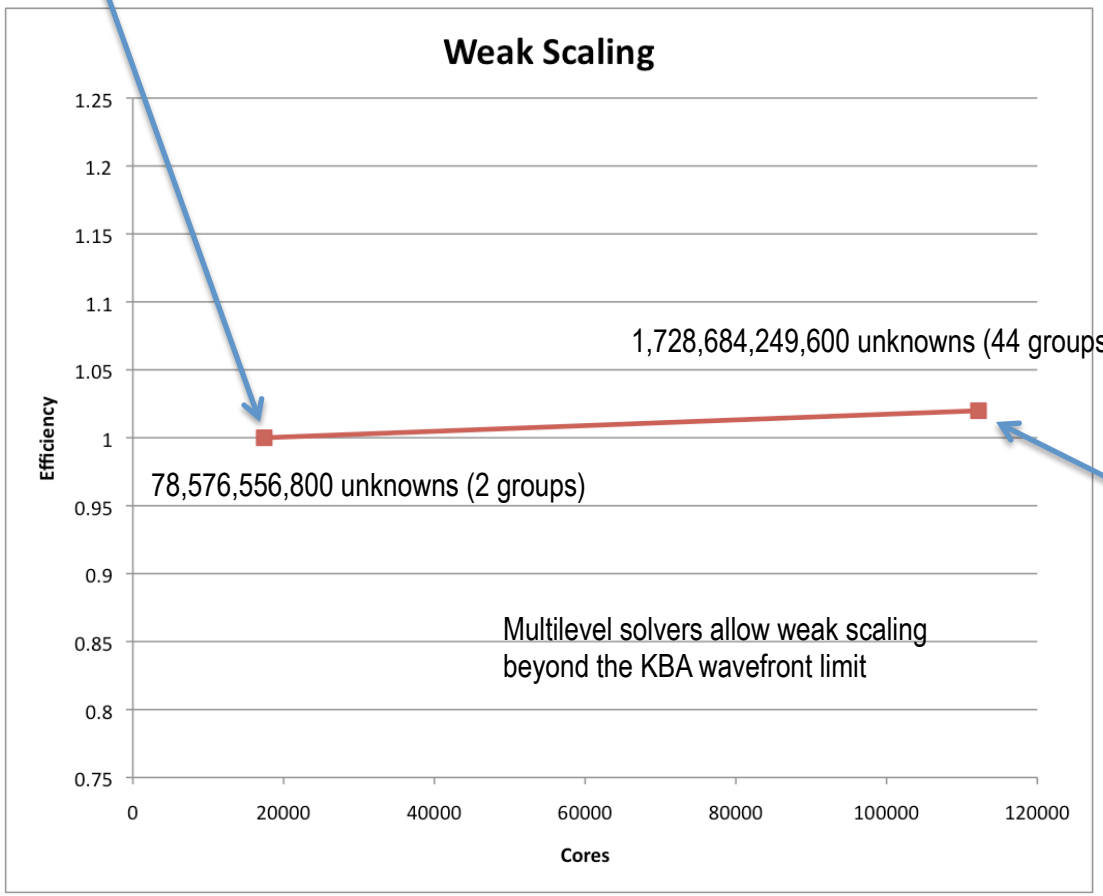
$$\text{domains} = \text{sets} \times \text{blocks}$$

- KBA is performed for each group in a set across all of the blocks.
  - Not required to scale beyond *O(1000)* cores.
- Scaling in energy across sets should be linear.
- Allows scaling to O(100K) cores and enhanced parallelism on accelerators.

Parallel Scaling



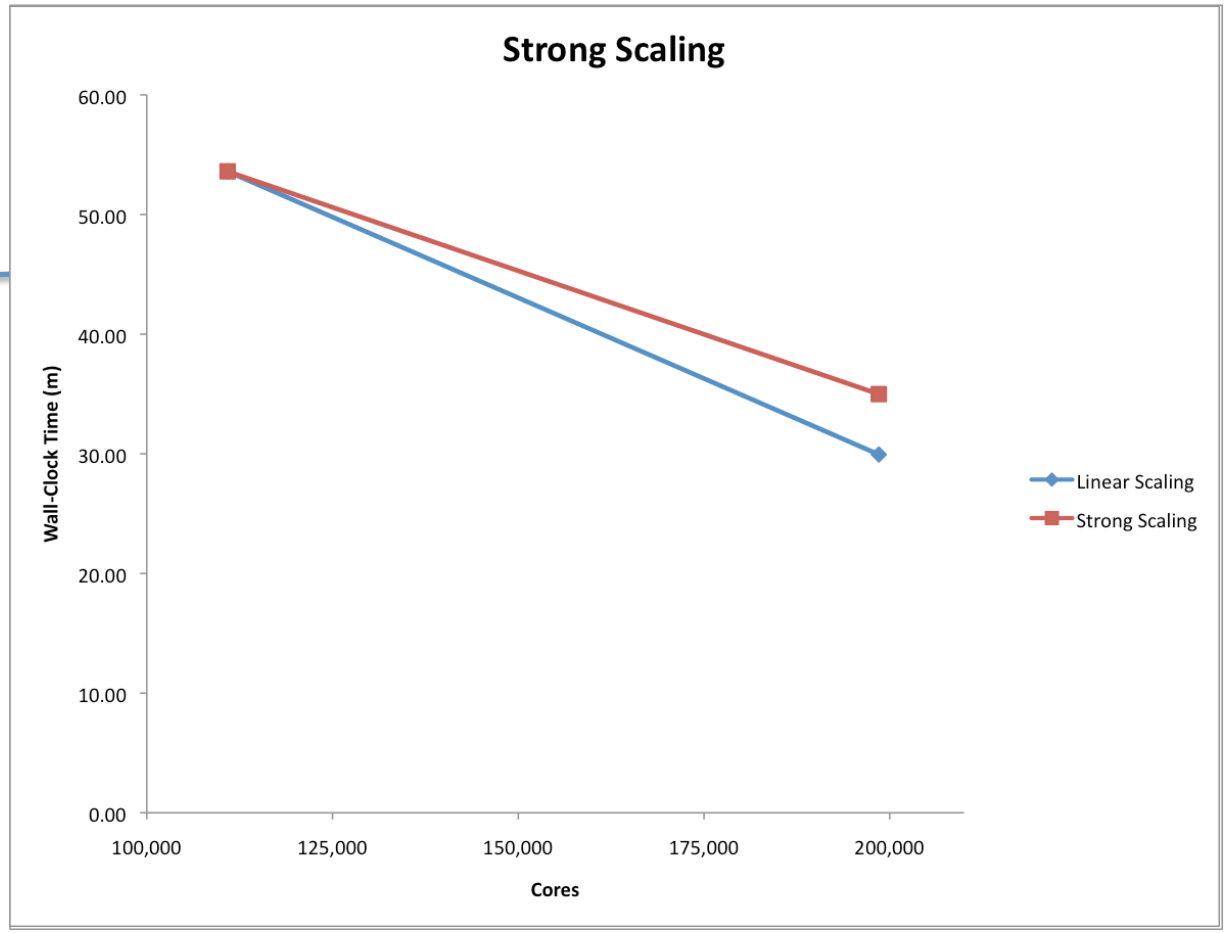
17,424 cores is effectively the maximum that can be used by KBA alone



MG Krylov solver partitioned across 11 sets

Strong Scaling

Optimized communication gave performance boost to 100K core job, number of sets = 11



- Communication improvements were significant at 100K core level (using 11 sets).
- They do not appear to scale to 200K core. Why?
  - The problem isn't big enough to demonstrate strong scaling.
  - We are not using the optimal block decomposition.
  - Communication collision on torus across full machine.
  - Multiset communication latency across entire machine dominates.

What do we need to do?

- Optimize scaling out to 200K cores
- Investigate multithreading to reduce latency in space-angle sweep
- GPU kernels for sweep
  - Already in testing phase
- New strategies for pure-downscatter (lower triangular) systems?
- New preconditioners
  - Multigrid in energy
  - Upper/lower diagonal energy matrix
- Time-dependence
- Multiphysics coupling