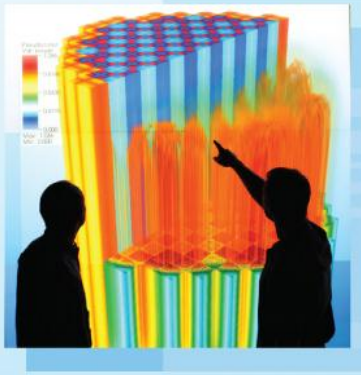
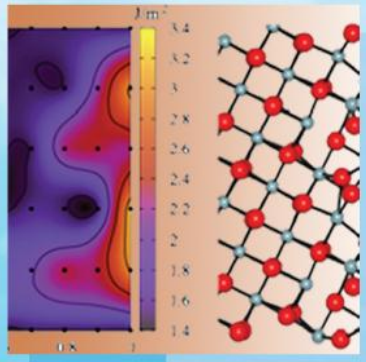
CASL-U-2012-0041-000

Power uprates
and plant life extension



Engineering design
and analysis

Science-enabling
high performance
computing



Fundamental science

Plant operational data

Consortium for Advanced Simulation of LWRs

L3:RTM.PRT.P4.02

Bill Martin
University of Michigan

Completed: 3/13/12

U.S. DEPARTMENT OF ENERGY | Nuclear Energy

Consortium for Advanced Simulation of LWRs

Consortium for Advanced Simulation of Light Water Reactors
Bldg. 5700, Room F317, MS 6003
P.O. Box 2008
Oak Ridge, TN 37831-6003
Email: casl-info@ornl.gov

Date: 3/13/2012

To: Bill Martin

c: John Wagner

From: Thomas Evans, RTM/ORNL

Level 3 Milestone Deliverable – L3:RTM.PRT.P4.02

Milestone Due Date: 3/31/2012

Milestone Completion Date:3/12/2012

Description of Milestone:
Demonstrate 3D full-core pin-homogenized transport with WB-1.

Completion Proof of the Milestone:
A series of WB-like 3D ¼ core models were run using macro-mixed materials for fuel pins, guide tubes, and other reactor features. See the paper, "Full Core Reactor Analysis" in the attachment section of this memo for full details.

Tasks to Complete the Milestone:
Multilevel parallelism was required to accommodate the large size of these problems. This involved implementing parallelism over energy using a multistep approach combined with the Koch-Baker-Alcouffe (KBA) algorithm for parallelism of space-angle. Additionally, macro-mixing was required in order to conserve fissionable fuel volumes to numerical precision. Full details on the calculational sequences are given in the "Full Core Reactor Analysis" attachment to this memo.

Person(s) Responsible for Completing the Milestone:
Thomas Evans, ORNL
Greg Davidson, ORNL
Joshua Jarrell, ORNL
Andrew Godfrey, ORNL
Steven Hamilton, ORNL

Regards,

Thomas Evans

PHYSOR 2012 Advances in Reactor Physics Linking Research, Industry, and Education
Knoxville, Tennessee, USA, April 15-20, 2012, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2012)

Full Core Reactor Analysis: Running Denovo on Jaguar

Joshua J. Jarrell, Andrew T. Godfrey, Thomas M. Evans, and Gregory G. Davidson
Oak Ridge National Laboratory
PO Box 2008, Oak Ridge, TN 37831
jarrelljj@ornl.gov; godfreyat@ornl.gov; evanstm@ornl.gov; davidsongg@ornl.gov

ABSTRACT

Fully-consistent, full-core, 3D, deterministic neutron transport simulations using the orthogonal mesh code Denovo were run on the massively parallel computing architecture Jaguar XT5. Using energy and spatial parallelization schemes, Denovo was able to efficiently scale to more than $160k$ processors. Cell-homogenized cross sections were used with step-characteristics, linear-discontinuous finite element, and trilinear-discontinuous finite element spatial methods. It was determined that using the finite element methods gave considerably more accurate eigenvalue solutions for large-aspect ratio meshes than using step-characteristics.

Key Words: Deterministic Transport, Reactor Analysis, Denovo

1. INTRODUCTION

Fully-consistent, three-dimensional, deterministic transport analysis of nuclear reactors is computationally expensive, in both time and memory. To solve these problems in a reasonable time period, deterministic codes have been developed to take advantage of massively parallel architectures. At Oak Ridge National Laboratory (ORNL), the three-dimensional, neutral-particle, orthogonal grid transport code Denovo [1] has been developed to utilize massively parallel machines, such as Oak Ridge Leadership Computing Facility's (OLCF) Jaguar XT5 *, a Cray XT5 supercomputer. Jaguar consists of $224,256$ AMD Opteron cores, has 300 terabytes of memory, and has a peak theoretical speed of 2.33 petaflops.

The goal of this work was twofold:

- Ensuring Denovo could be efficiently utilized to solve realistic reactor core models, and
- Evaluating the most appropriate spatial methods for different meshes.

The main concerns for modeling in Denovo were reactor input design, efficient scaling to hundreds of thousands of processors, memory limitations for large models, effects of orthogonal grid on accuracy, and visualization of output data.

2. DENOVO METHODOLOGY

Denovo has two general classes of k-eigenvalue solvers, power iteration and Arnoldi. These solvers can use a Krylov subspace or a Gauss-Seidel multigroup solver and can use a Krylov

*see http://www.nccs.gov.

subspace or a source iteration within-group solver. Denovo uses the Koch-Baker-Alcouffe (KBA) algorithm [2] for spatial parallelism and distributed energy groups for energy parallelism [3]. It supports forward and adjoint calculations and has multiple acceleration schemes. Whereas the KBA algorithm has been available for a number of years, the energy-set parallelism in Denovo is a recent development that we will briefly discuss.

We begin with a brief review of the fundamental solver strategies in Denovo; Ref. [1] can be consulted for full details. The multigroup S_N equations for k-effective eigenvalue problems can be written in operator form as

$$\mathbf{L}\psi = \mathbf{MS}\phi + \frac{1}{k}\mathbf{M}\chi\mathbf{f}^T\phi \,. \tag{1}$$

The state of these equations is defined in angular flux moments, ϕ, that are related to the discrete angular flux through

$$\phi = \mathbf{D}\psi \,, \tag{2}$$

where $\mathbf{D}$ is the discrete-to-moment operator that integrates discrete angles into angular flux moments using quadrature rules. $\mathbf{L}$ is the first-order linear differential transport operator, $\mathbf{M}$ is the moment-to-discrete operator that projects angular flux moments into discrete angle space, and $\mathbf{S}$ is the group-to-group scattering matrix. In the eigenvalue form of the equation, $\mathbf{f}^T$ is the rectangular matrix of fission cross sections, χ is the rectangular matrix of fission spectrums, and k is the largest eigenvalue.

Operating by $\mathbf{T} = \mathbf{DL}^{-1}$ and rearranging terms, the eigenvalue problem becomes

$$(\mathbf{I} - \mathbf{TMS})\phi = \frac{1}{k}\mathbf{TMF}\phi \,, \tag{3}$$

where $\mathbf{F} = \chi\mathbf{f}^T$ is the fission matrix. The operator $\mathbf{L}^{-1}$ can be formed into a lower-triangular system if one sweeps the space-angle grid in the direction of neutron travel. The resulting transport *sweep* is the operation that is parallelized using the KBA algorithm. This matrix is never formed in practice; only the action of the operator on a vector, $y = \mathbf{L}^{-1}v$, is required.

For eigenvalue problems, we have implemented an Arnoldi solver using the Trilinos Anasazi package [4] that can (1) take full advantage of the energy parallelism and (2) be more efficient than power iteration. Arnoldi iteration requires the eigenproblem to be written in standard form:

$$\mathbf{A}x = \lambda x \,. \tag{4}$$

Arnoldi iteration can be implemented with either an energy-dependent or an energy-independent eigenvector as follows:

$$\mathbf{A}\phi = k\phi \,, \quad \mathbf{A} = (\mathbf{I} - \mathbf{TMS})^{-1}\mathbf{TMF} \,, \qquad \text{(energy dependent)} \,, \tag{5}$$

$$\mathbf{A}\Gamma = k\Gamma \,, \quad \mathbf{A} = \mathbf{f}^T(\mathbf{I} - \mathbf{TMS})^{-1}\mathbf{TM}\chi \,, \quad \text{(energy independent)}, \tag{6}$$

where $\Gamma = \mathbf{f}^T\phi$. In either case, the implementation of the Arnoldi iteration requires a matrix-vector multiplication at each iterate of the form

$$y^k = \mathbf{A}v^k \,. \tag{7}$$

For the energy-dependent case, we have

$$z^k = \mathbf{TMF}v^k \,, \quad \text{(matrix-vector multiply and sweep)}, \tag{8}$$

$$(\mathbf{I} - \mathbf{TMS})y^k = z^k \,, \quad \text{(fixed-source solve)}. \tag{9}$$

Similarly, for the energy-independent eigenvector, the steps are

$$z^k = \mathbf{TM}\chi v^k \,, \quad \text{(matrix-vector multiply and sweep)}, \tag{10}$$

$$(\mathbf{I} - \mathbf{TMS})y^k = z^k \,, \quad \text{(fixed-source solve)}, \tag{11}$$

$$y^k \leftarrow \mathbf{f}^T y^k \,, \quad \text{(dot product)}. \tag{12}$$

Both methods require a fixed-source solve each iteration, ie. Eqs. 9 and 11. The inner multigroup fixed-source problem is solved using the generalized minimal residual (GMRES) method. Including energy in the GMRES vectors enables the following benefits:

- the energy variable is decoupled, allowing groups to be treated independently;
- Krylov subspace iteration is more efficient and robust than Gauss-Seidel iteration;
- preconditioning a Krylov iteration is generally more robust and stable than Gauss-Seidel acceleration.

Furthermore, including energy in the Krylov vector does not invalidate any of the existing sweep mechanics that are already implemented in Denovo.

For multigroup fixed-source problems in the form of Eq. (11), application of a Krylov method requires a full energy-space-angle sweep at each iteration to calculate the action of the operator on the kth Krylov basis vector, v^k,

$$y^k = (\mathbf{I} - \mathbf{TMS})v^k \,. \tag{13}$$

We note that this vector is dimensioned $v^k \equiv \{v^k_{g,c,n,l,m}\}$ where g is the energy group, c is the cell index, n is the spatial unknown index in the cell, and (l, m) are the spherical harmonic moment indices. The energy-independent approach allows only energy-domain parallelization over the fixed-source solve, and the eigenvalue solve is parallel only over space-angle. However, this decomposition is sometimes more efficient because the eigenvector is smaller, especially when work is dominated by the inner multigroup fixed-source solve.

Denovo employs a multilevel parallel scheme that is described in Ref. [3]. The S_N multilevel energy-space decomposition is illustrated in Fig. 1. In this decomposition, space is partitioned into *blocks* and energy is partitioned into *sets*. Each set contains the full mesh (all of the blocks) in order to eliminate space-angle coupling between sets. Every $(\text{block}, \text{set})$ combination is termed a *domain*. The total number of domains is currently the same as the number of MPI processes in a parallel job. Within each set, the space-angle partition is decomposed and solved using the well-known KBA wavefront algorithm. The benefit of using multiple sets is that KBA does not have to scale beyond $O(1,000)$ processes because the multiset algorithm provides parallelism over energy.

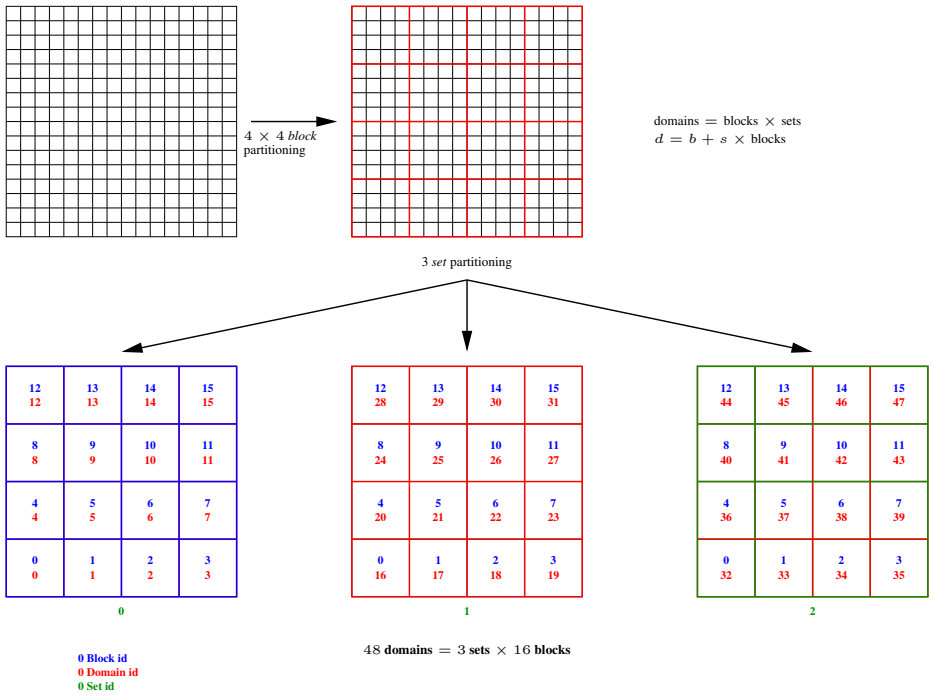


Figure 1. Multilevel energy-space decomposition in Denovo. This example has 3 *sets*, each containing 16 *blocks*, resulting in 48 total *domains*. The block (blue), set (green), and domain (red) IDs are indicated by b, s, and d, respectively.

3. BENCHMARK PROBLEM

3.1. Problem Description

We chose to model a typical Westinghouse $3,411$ MWth four-loop pressurized water reactor (PWR) for these calculations. An initial startup quarter-core loading pattern was developed using three fresh fuel regions and discrete lumped burnable poison (BP) assemblies. The fuel enrichments for the design are 1.5, 2.5, and 3.0 wt% $^{235}U/U$. The BP assemblies contained 2.0% B_4C with either 8, 12, or 16 rodlets per assembly. As in many first-cycle cores, the majority of the highest enriched fuel was positioned on the core periphery; and the lowest enrichment assemblies were placed in the core interior in a checkerboard pattern under control rod clusters. This loading pattern is displayed in Fig. 2.

The fuel dimensions used in the model are consistent with traditional Westinghouse 17×17 fuel, including the small inter-assembly gap. The core height is 12 feet, and the BP height is assumed to be equal to and aligned with the fuel column. Fuel, clad, and moderator densities and temperatures were chosen based on average full-power conditions, and the soluble boron concentration selected for the water moderator was $1,000$ ppm. Finally, a homogeneous mixture of 40% moderator and 60% stainless steel was used as an axial and radial reflector material to simulate the neutron current at the core boundaries. Spacer grids and regulating control rod banks

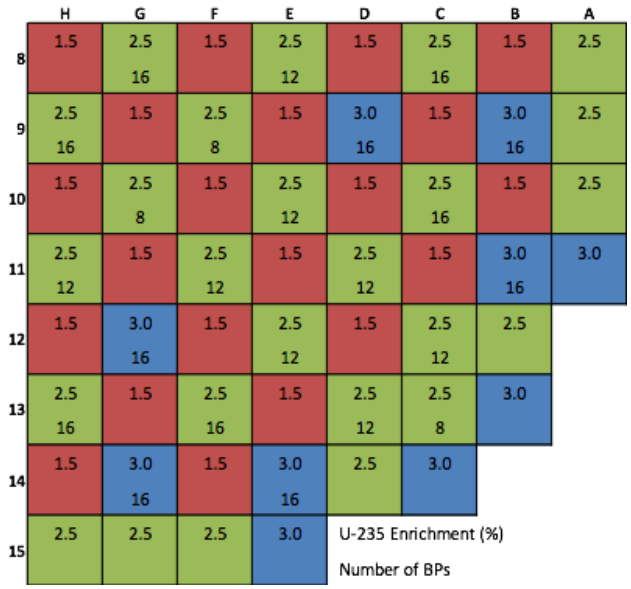


Figure 2. Core loading pattern where first number in each assembly is the ^{235}U enrichment and the second number (if shown) is the number of BPs.

were excluded from the model for simplicity.

Macroscopic cross sections in eight energy groups were created by sequences in the SCALE 6.1 [5] analysis package. The TRITON/NEWT sequence was used to collapse the microscopic cross sections from the built-in 238-group ENDF/VI structure to only eight groups using a 2D lattice problem-dependent flux spectrum. This new library was then provided to the CSASI sequence, which calculated the resonance self-shielded macroscopic cross sections from the one-dimensional continuous-energy pin cell code CENTRM. These cross sections were then provided directly to Denovo for the radiation transport calculation, which used P_1 scattering to perform the solve.

3.2. Denovo Problem Setup

In order to set up the problem inputs and run the problem on Jaguar, we used Denovo's Python front end to create input files that were then used by Denovo's high-performance computing front end to allow submission to the Jaguar queueing system. The variables that were changed for various runs were the spatial discretization method, the quadrature set, and the spatial mesh.

The spatial methods considered were the step-characteristics (SC), linear-discontinuous (LD) finite element, and the trilinear-discontinuous (TLD) finite element method [1] . We chose to use the rectangular quadruple range (QR) quadrature sets [6,7] and varied the number of polar levels in each octant between two and eight, and the number of points per polar level in each octant between four and eight. We describe the mesh using the 2D mesh in each pin cell and the axial mesh size. The 2D mesh varied from a 4×4 pin cell mesh up to a 12×12 pin cell mesh. This results in between 16 and 144 cells per pin cell. The axial mesh size was varied between 0.2 cm

Table I. Eigenvalue differences using a 2.54 cm (1 in.) axial mesh.

Case ID	2D Mesh Discretization	Angles per Polar Level	Polar Levels Per Octant	SC Error (pcm)	LD Error (pcm)
1	4	4	2	-204	-16
2	6	6	4	-113	33
3	6	6	8	-109	
4	8	6	4	-120	
5	10	6	4	-150	

(0.079 in.) in the reference case up to 15.24 cm (6 in.) in the coarsest case. These variations resulted in meshes as small as 9.7 million cells and as large as 1.6 billion cells, as well as quadrature sets as small as 8 angles per octant and as large as 48 angles per octants.

4. RESULTS

The reference case was a deterministic solution that used the SC method with a QR quadrature set having four polar levels per octant and six angles per level in each octant. It had a 6×6 2D mesh and a 0.2 cm (0.079 in.) axial mesh, which resulted in 1.6 billion cells and an aspect ratio of 1.05. We ran the reference case on $163,200$ processors for 70 minutes.

The first problem we analyzed used a fairly coarse axial mesh of 15.24 cm (6 in.). For a pin cell that had a 6×6 2D mesh, this resulted in an aspect ratio of 72.5 in each cell and a total of 21 million cells. Using a QR quadrature set with 6 polar levels and 6 angles per level in each octant (36 angles per octant), the SC eigenvalue had an error of 960 pcm, while the LD and TLD eigenvalue errors were 9 and 26 pcm, respectively. The LD and TLD methods require four and eight times more memory, respectively, than the SC method. The SC method ran for 17 minutes on $9,600$ cores, whereas the LD and TLD methods took 1.4 and 6.6 times longer on the same number of processors.

We then analyzed the effect of using a 2.54 cm (1 in.) axial mesh. For the 6×6 2D mesh, this resulted in an aspect ratio of 12.1 and a total of 130 million cells. On this problem, we ran five SC cases and two LD cases. The results are shown in Table I. Table I implies that regardless of the 2D spatial mesh or the quadrature refinement, the SC method is significantly less accurate than the LD method using the same quadrature set and spatial cells. Between $3,000$ and $19,200$ processors were used to ensure enough memory was available on all processors. The timing results for the two cases in which SC and LD methods were run with the same spatial and quadrature mesh can be seen in Table II. For case 1, the total amount of cpu-time (number of processors $\times$ runtime) was 6.1 times larger for the LD case. For case 2, the total amount of cpu-time was 4.4 times larger for the LD case.

The last problem analyzed used a 1 cm (0.39 in.) axial mesh which resulted in an aspect ratio of 4.8 and 320 million cells, if a 6×6 2D mesh was used. On this problem, four SC cases and one LD case were run. The results are shown in Table III. For this problem, the SC method yields

Table II. Runtime statistics using a 2.54 cm (1 in.) axial mesh.

Case ID	Number of Processors (SC)	SC Runtime (min)	Number of Processors (LD)	LD Runtime (min)
1	3,000	19.9	5,184	67.5
2	4,800	76.3	19,200	80.2

Table III. Eigenvalue differences using a 1 cm (0.39 in.) axial mesh.

Case ID	2D Mesh Discretization	Angles per Polar Level	Polar Levels Per Octant	SC Error (pcm)	LD Error (pcm)
6	4	4	2	-90	
7	6	6	4	-5	35
8	6	6	6	-1	
9	12	8	4	-4	

accurate results for quadrature sets that have more than two polar levels. We note that the reference case used the SC method, which explains the discrepancy in the LD solution. For Case 7, the SC runtime on $28,800$ processors was 88 minutes and the LD runtime on $96,000$ processors was 64.5 minutes, which resulted in a total cpu-time increase of 2.4.

The scalar flux and fission data were output in HDF5 format and read into the VisIt parallel visualization tool for post-processing and analysis. Figure 3 illustrates the power profile of the reactor using a 6×6 2D mesh, 2.54 cm (1 in.) axial mesh, and the SC spatial method. Figure 4 shows the power profile of the bottom half of a single fuel assembly (which was run with grid spacers and reflecting boundaries) using a 10×10 2D mesh, a 2.54 cm (1 in.) axial mesh, and the SC spatial method.

The core simulations were run with a variety of space-energy decompositions so that the number of blocks ranged from $1,500$ to $50,400$ and the number of energy sets ranged from 1 to 8. The largest problem contained more than 10^{13} total unknowns. Weak scaling results for all of the runs were accumulated using the following definition of parallel efficiency:

$$\epsilon = \frac{\tau_{\text{ref}}}{\tau_{\text{P}}} \left(\frac{D_{\text{P}}}{D_{\text{ref}}} \right) , \quad \tau = t \times N_c \, , \tag{14}$$

where t is the wall-clock time, D is the total number of unknowns, and N_c is the number of processors (cores). The "ref" subscript denotes a reference problem run, and the "P" subscript refers to the target problem. Figure 5 shows the weak scaling results. The variation in Fig. 5 results from the fact that the solver depends on parallelized independent variables differently. For example, increasing the number of spatial dimensions has a different effect on parallel efficiency than does increasing the number of angles. Nonetheless, this figure shows that Denovo makes good use of the resource over a wide range of problem configurations and cores.

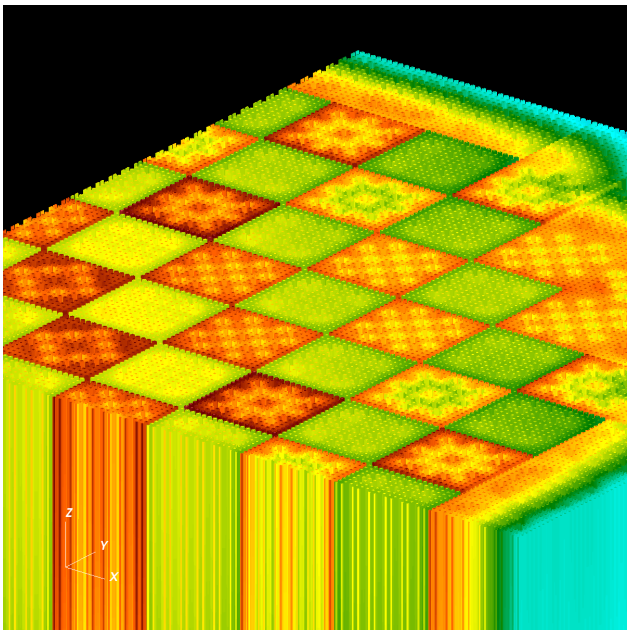


Figure 3. The power profile in the core using a 6×6 2D mesh, a 2.54 cm (1 in.) axial mesh, and the SC spatial method.

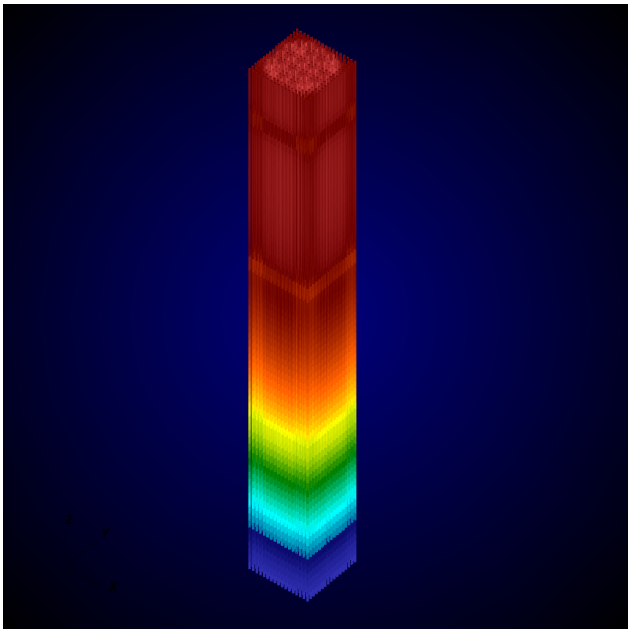
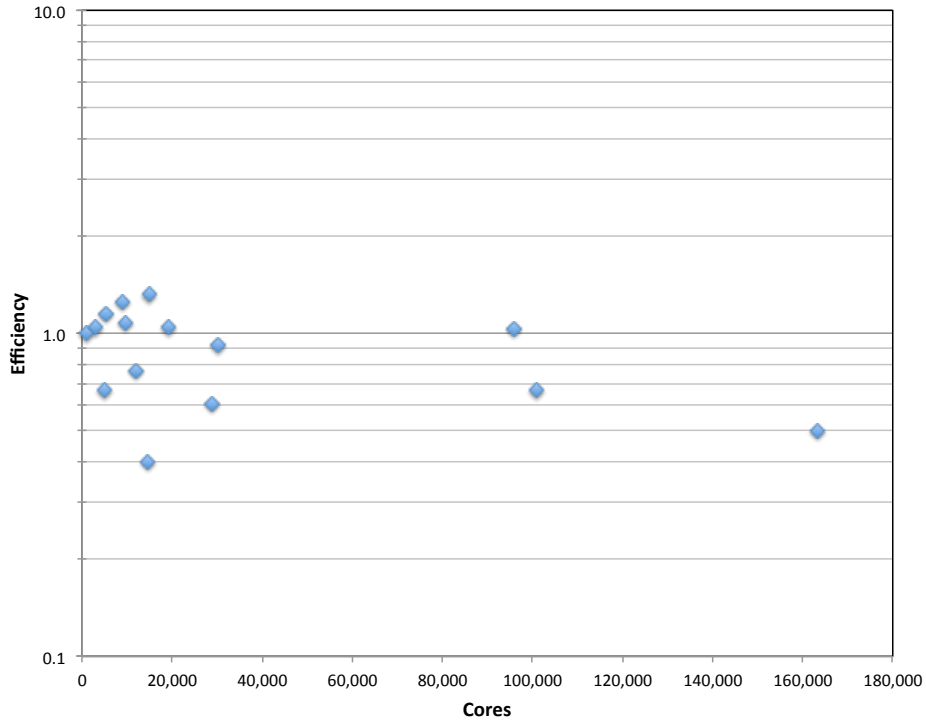
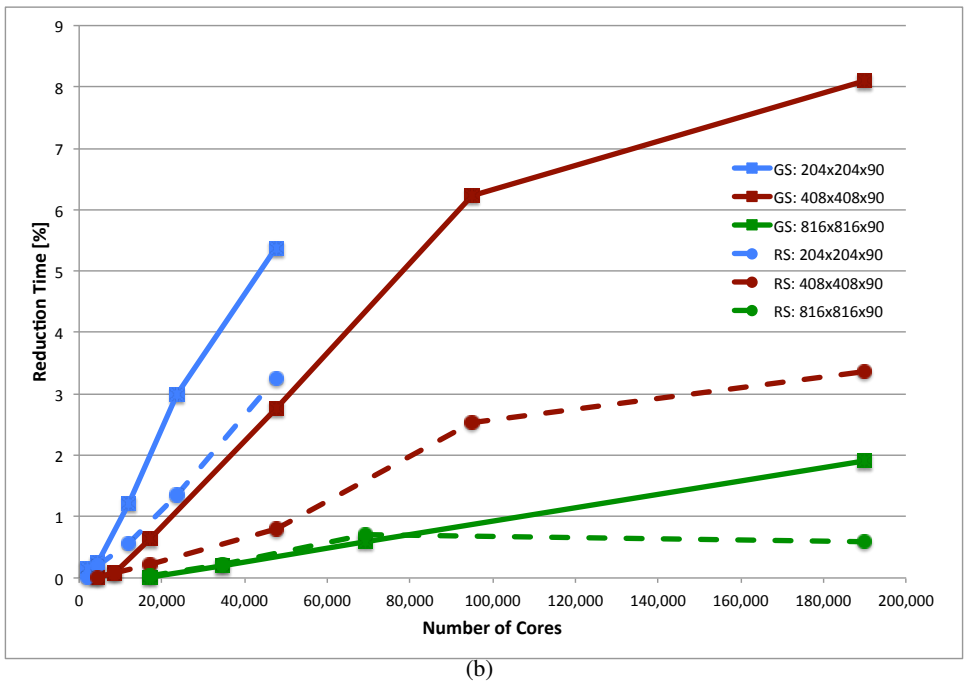


Figure 4. The power profile in a single assembly using a 10×10 2D mesh, a 2.54 cm (1 in.) axial mesh, and the SC spatial method.



(a)



(b)

Figure 5. (a) Weak scaling of Denovo S_N on the quarter-core test problem. (b) Time spent in multiset reductions; RS refers to the reduce-scatter, and GS refers to the global reduction. The numbers give the mesh size (in (x, y, z)).

We have recently optimized a communication bottleneck in the multiset reduction algorithm that has been affecting the scaling. At the end of each multigroup iteration, the right-hand source must be calculated using a block-sparse matrix-vector multiply. This operation occurs across all sets, and at the time the scaling results of Fig. 5a were generated, we were using a global reduction to perform it. This global operation has since been replaced with a reduce-scatter operation that has lowered the time spent in the reduction significantly, as illustrated in Fig. 5b. Using this scheme, we have achieved improvements in the multiset reduction of up to a factor of 3 in large problems. This should improve the scaling performance of the algorithm; we are currently rerunning the scaling suite to quantify the overall gain in performance.

We also used the PAPI hardware counters to register floating point operations and cycles. Using this data and a hypothetical maximum performance rate of 4 FLOP/cycle/core, the Denovo S_N solver achieved a minimum peak performance of 0.98%, a maximum peak performance of 3.53%, and an average of 2.11%. The median value was 1.89%. These results are very reasonable for a sparse, hyperbolic PDE solver.

5. CONCLUSIONS

For all axial meshes, the finite element methods (LD and TLD) yielded an error in the eigenvalue of no more than 40 pcm, while the SC method had an error of 960 pcm for the 15.24 cm (6 in.) axial mesh case. As the aspect ratio is reduced, the SC method converges to the correct solution and requires four times less memory requirements than the LD method.

For eigenvalue calculations, the same level of accuracy can be attained with significantly fewer mesh cells by using LD instead of SC. However, LD can introduce negativities in the flux solutions (even with strictly positive sources), which may be unacceptable for shielding problems, pin power comparisons, and hybrid-Monte Carlo calculations. By restricting sources to be strictly positive and converging the flux tolerance highly enough, SC is guaranteed to maintain positivity regardless of the spatial mesh.

Denovo has the capability to scale to hundreds of thousands of processors (or more) and can therefore be used to analyze three-dimensional reactor cores. In the future, we plan to extend reactor core modeling to OLCF's next-generation high-performance platform, Titan. We plan to perform more extensive testing in the future by comparing flux profiles with Monte Carlo results and analyzing additional timing results. We also will evaluate the effect of quadrature set type and refinement on the eigenvalue.

We note that the orthogonal mesh (which allows this scaling) has inherit accuracy limitations for certain reactor problems: i.e. gadolinium reactor pins, spray-on BP, and other geometrically small features. We plan to investigate different spatial methods that may address these difficult features.

6. ACKNOWLEDGEMENTS

Work for this paper was supported by Oak Ridge National Laboratory, which is managed and operated by UT-Batelle, LLC, for the US Department of Energy under Contract No.

DE-AC05-00OR22725 and by the Consortium for Advanced Simulation of Light Water Reactors (www.casl.gov), an Energy Innovation Hub (http://www.energy.gov/hubs) for Modeling and Simulation of Nuclear Reactors under US Department of Energy Contract No. DE-AC05-00OR22725.

This manuscript has been authored by UT-Battelle, LLC, under Contract No. DE-AC05-00OR22725 with the US Department of Energy. The United States government retains, and the publisher, by accepting the article for publication, acknowledges that the United States government retains, a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States government purposes.

REFERENCES

[1] T. Evans, A. Stafford, R. Slaybaugh, and K. Clarno, "DENOVO: A new three-dimensional parallel discrete ordinates code in SCALE," *Nuclear Technology*, **171**, pp. 171–200 (2010).

[2] K. Koch, R. Baker, and R. Alcouffe, "A Parallel Algorithm for 3D S_n Transport Sweeps," Technical report, LA-CP-92-406, Los Alamos National Laboratory (1992).

[3] G. Davidson, T. Evans, R. Slaybaugh, and C. Baker, "Massively Parallel Solutions to the k-Eigenvalue Problem," *Trans Am Nucl Soc*, **81**, pp. 318–320 (2010).

[4] C. Baker, U. Hetmaniuk, R. Lehoucq, and H. Thornquist, "Anasazi software for the numerical solution of large-scale eigenvalue problems," *ACM Transactions on Mathematical Software (TOMS)*, **36**(3), p. 13 (2009).

[5] Oak Ridge National Laboratory, *SCALE: A Modular Code System for Performing Standardized Computer Analyses for Licensing Evaluations* (2009), Version 6.

[6] I. Abu-Shumays, "Angular Quadratures for Improved Transport Computations," *Transport Theory and Statistical Physics*, **30(2&3)**, pp. 169–204 (2001).

[7] J. Jarrell, M. Adams, and J. Risner, "Application of Quadruple Range Quadratures to Three-Dimensional Model Shielding Problems," *Nuclear Technology*, **168(2)**, p. 424 (2009).