Joint International Topical Meeting on Mathematics & Computation and Supercomputing in Nuclear Applications (M&C + SNA 2007)
Monterey, California, April 15-19, 2007, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2007)

A METHOD OF CHARACTERISTICS SOLUTION TO THE OECD/NEA 3D NEUTRON TRANSPORT BENCHMARK PROBLEM

J. Bryce Taylor
The Pennsylvania State University
Department of Mechanical and Nuclear Engineering
137 Reber Building, University Park, PA 16802
jbt120@psu.edu

Dave Knott
Studsvik Scandpower, Inc.
1015 Ashes Drive, Suite 205, Wilmington, NC 28405
Dave.Knott@StudsvikScandpower.com

Anthony J. Baratta
Professor Emeritus of Nuclear Engineering
The Pennsylvania State University
ab2@psu.edu

ABSTRACT

A method of characteristics for steady-state transport calculations that can be applied to problems with a full and explicit 3D Cartesian system model has been implemented into the new code, MOCK-3D. Procedures for performing 3D ray-tracing are derived from methods that are employed in most lattice physics codes and simplifications to the discretized problem model minimize the expense associated with the storage of a "tracking file." MOCK-3D has been applied to one of the OECD/NEA 3D Neutron Transport Benchmarks, and comparison of these results to the reference solution is presented.

Key Words: method of characteristics, 3D deterministic neutron transport, lattice physics, ray trace

1. INTRODUCTION

Nuclear reactor kinetics codes that calculate the time-dependent neutron flux in a reactor core are an essential component in the suite of tools available for light water reactor (LWR) design and analysis. Traditionally, such codes have employed the nodal diffusion approximation, which yields a computationally efficient solution to typical LWR problems. The accuracy of diffusion theory is significantly reduced, however, when strong material and/or flux gradients are present, when neutron streaming is significant, or when neutron scattering includes a strongly anisotropic component. Such physical phenomena could occur when MOX fuel assemblies are loaded into a LWR core, when the streaming of neutrons through the core externals to ex-core instrumentation is of interest, due to voiding in the core region during boiling water reactor (BWR) instability transients, and during detailed analysis of control rod motion and/or rod ejection, especially at the control rod blade tips. Many advanced LWR and Generation IV reactor designs also raise physical issues to which diffusion theory may not be applicable. In light of these numerous issues, the need for an improved nuclear reactor kinetics methodology is apparent.

The widespread use of nodal diffusion theory for reactor kinetics calculations has resulted in part from the significant computational expense associated with more accurate solutions to the time-dependent neutron transport equation. In light of advances in computer technology that facilitate larger memory storage and more powerful computations, successful implementation of a higher-order transport formulation now may be feasible. Recent investigations have included the development of time-dependent formulations for discrete ordinates (S_N) methods [1,2], spherical harmonics (P_N) methods [3], variational nodal methods [4], and incorporation of the simplified-P_n approximation into pre-existing diffusion theory codes [5,6]. While success has been achieved to varying degrees with all of these approaches, no single method has yet been proven to be optimal for use in a high-order three-dimensional (3D) reactor kinetics methodology. Many of these methods are adaptable to nodal theory and to code parallelization techniques, yet computational expense remains a significant impediment to practical application.

An alternative technique that has not been considered previously for use in a 3D reactor kinetics methodology is the method of characteristics. This method is currently incorporated into most commercial lattice physics codes for the solution of steady-state, two-dimensional (2D) assembly transport problems. The method of characteristics has been shown to be uniquely applicable to such problems, which involve detailed descriptions of the problem geometry and include severe flux gradients and material heterogeneity. The method of characteristics may be applicable to time-dependent problems that require a similarly complicated and detailed description of the problem space; however, a 3D steady-state method of characteristics must be developed before a time-dependent formulation can be pursued. This steady-state calculation is required to perform the “null transient” calculation that provides values for the angular and scalar neutron flux distributions at the beginning of a time-dependent calculation. This task is not trivial, as recent efforts to extend lattice physics methods to model 3D reactor cores have been impeded by the computational expense associated with performing a 3D method of characteristics calculation.

The feasibility of a 3D steady-state method of characteristics that avoids many of the inefficiencies encountered by lattice physics methods will be demonstrated in this paper. This formulation has been implemented into the code MOCK-3D, the details of which are discussed in Section 2. While this code is intended for use in a 3D reactor kinetics methodology, the methods presented below draw on previous experience with lattice physics applications of the method of characteristics and results may be of interest to researchers in that field. In fact, derivation and solution of the characteristic equation in MOCK-3D is identical to standard techniques that have been employed elsewhere. The primary difference between MOCK-3D and lattice physics codes lies with techniques that have been developed for performing ray-tracing in 3D geometry. In MOCK-3D, the computational expense associated with the generation of a 3D track set is minimized via a set of simplifying constraints that, while reasonable in the context of a reactor kinetics methodology, would be over-limiting to most lattice physics applications. In Section 3 the feasibility of this approach to the method of characteristics is demonstrated via application to one of the OECD/NEA Neutron Transport Benchmark Problems [7]. Results for this problem demonstrate that a reasonably accurate solution can be attained while minimizing memory storage and run-time requirements and without recourse to advanced acceleration and/or parallelization techniques.

2. 3D STEADY-STATE METHOD OF CHARACTERISTICS

The method of characteristics is a class of solutions to the neutron transport equation via a transformation to a Lagrangian coordinate system; i.e, to the frame-of-reference of the neutron in motion. Whereas most transport methods solve a neutron balance problem, the method of characteristics solves a neutron propagation problem. It is assumed that neutrons will travel in straight lines until experiencing collision events, so that the spatial domain can be discretized into a set of straight linear characteristics along which all neutrons are assumed to travel. Given this discretization, the 3D steady-state transport equation is transformed to a 1D equation that can be solved by direct integration along the neutron propagation path. One advantage to this approach is that spatial discretization techniques such as finite differencing, which present a significant limitation to accuracy for many other transport methods, are explicitly avoided. Characteristics methods, however, require procedures to determine the length and orientation of the neutron propagation paths – a ray-tracing algorithm – and potential gains in accuracy are offset by an increase in computational expense.

The 3D steady-state method of characteristics presented in this section draws on experience gained from developments in the field of lattice physics. In Section 2.1, the most relevant literature is reviewed, specifically as it relates to advances towards a 3D lattice physics implementation for the method of characteristics. The MOCK-3D ray-tracing methodology, presented in Section 2.2, utilizes many of these techniques and introduces simplifications to the problem geometry as are appropriate for a 3D reactor kinetics application. In so doing, 3D ray-tracing is rendered feasible at minimal computational expense. Iterative solution of the characteristic equation in MOCK-3D is described in Section 2.3. MOCK-3D utilizes techniques that are documented in detail elsewhere, and References 8 and 9 provide a more detailed discussion of these well-understood aspects of the method of characteristics.

2.1 Recent Advances in the Method of Characteristics

The first method of characteristics formulations for nuclear applications were derived by V. S. Vladimirov in 1959 [10]. In 1972, the development of a 2D steady-state method of characteristics was reported by J. R. Askew [11], as implemented by M. J. Halsall into the lattice physics code CACTUS eight years later [8]. The "CACTUS method" has since become the most common approach to solution of the transport equation for 2D steady-state assembly transport problems and a plethora of commercial lattice physics codes are based on this formulation [12,13,14,15]. The primary role of these codes is to perform assembly-level energy group collapse and spatial homogenization calculations, thereby generating the set of few-group material cross-sections that is required for solution of the multi-group nodal diffusion equations. Assembly-level transport calculations include extremely detailed models of the reactor internals, which for a typical LWR fuel assembly will include the fuel pin, fuel cladding, and gas gap, as well as external assembly structures, control rods, and the coolant channels. The spatial model for such a calculation may include a combination of rectangular, cylindrical, and/or hexagonal features, requiring substantial spatial flexibility by the transport solver. The method of characteristics has become the norm for such calculations due to its unique spatial discretization, which yields accurate results for problems with fine detail and severe spatial heterogeneity.

Traditionally, lattice physics applications of the method of characteristics have been limited to 2D geometry for small sub-sections of a reactor model. A typical calculation will model a single fuel assembly or a small group of adjacent assemblies and treat these as infinitely periodic in the radial direction and infinite in the axial direction. Such a limitation was acceptable for older commercial reactor core designs, as these reactor cores were loaded with identical fuel assemblies of simple design and axial detail could safely be approximated as uniform. Many of the features associated with advanced fuel designs demand a more accurate spatial treatment, however, and current advances in the method of characteristics focus on extension to whole-core radial analysis with axial variations. The computational expense of large-scale 2D and 3D formulations has hindered practical application. Much of this expense results from the requirements of ray-tracing. In large problems an extremely large set of track segments is needed and the computational burdens of ray-tracing become manifest in two ways: (1) a large amount of memory is required to store the track length information and (2) iterative computation for the entire track set can lead to long runtimes.

Most 2D implementations of the method of characteristics utilize a ray-tracing methodology similar to that included in CACTUS. When applied to the model of an entire reactor core, these procedures are not particularly efficient. Additionally, numerical procedures for calculating track lengths can lead to a violation of basic principles such as conservation of area and neutron balance and in certain instances numerical instabilities can arise. Such problems are typically resolved via the introduction of corrections to enforce balance and constraints to avoid instabilities. A significant and fundamental redefinition of the CACTUS ray-tracing methods was developed by S. Kosaka and E. Saji and is known as the direct neutron path linking technique (DNPL) [16]. DNPL requires that each propagation path entering or leaving a cell must connect directly to a path in the adjacent cell. The outgoing angular flux for any track is thus identical to the incoming flux for a track in an adjacent cell, and a naturally occurring set of continuity boundary conditions can be employed at all internal cell interfaces. Direct connection of tracks is also enforced at external system boundaries, with reflection between directions being governed by Snell's Law for optical reflection. By defining the track set in this manner, DNPL allows for regions with similar material properties – identical fuel assemblies, for example – to be modeled via cells with uniform track sets, greatly reducing the memory storage requirements for ray-tracing in large problems.

Incorporation of DNPL into lattice physics applications has enabled the practical pursuit of whole-core 2D assembly transport calculations. Further extension of these techniques to 3D geometry remains problematic due to the added expense associated with explicit modeling of the axial dimension. One means to ameliorate this expense involves the development of hybrid computational strategies that decompose the 3D problem into simpler one-dimensional (1D) and 2D calculations. One class of such methods performs a whole-core 2D method of characteristics calculation to analyze the radial reactor plane and a separate 1D calculation to account for details along the axial direction. By coupling the results of these two calculations, a full 3D angular flux distribution can be generated [17,18]. Alternatively, a series of 2D radial calculations at various axial levels can be coupled to generate a full 3D profile [19]. Hybrid techniques can utilize pre-existing 1D and 2D ray-tracing procedures; however, such an approach will place limitations on the accuracy that can be obtained by the characteristics solution.

Ideally, accurate calculations should incorporate explicit models of the 3D problem space and, to date, two methods for performing 3D ray-tracing within lattice physics codes have been developed. The first of these was suggested by I. R. Suslov and incorporated into MCCG [20]. This approach decomposes 3D space into a set of coupled orthogonal 2D planes – the x-y plane and the axial plane – and then applies CACTUS-like procedures to each of these 2D domains in sequence. The ray-tracing procedures discussed in Section 2.2 will utilize a similar approach. An alternative procedure has been incorporated into the DRAGON lattice physics code [21]. In DRAGON, the problem domain is again decomposed into a coupled set of 2D planes; however, the planes are defined to be perpendicular/parallel to the direction of neutron travel. While differences between these two methods affect the precise nature of the ray-tracing algorithm, effects on the final track set should be negligible. Application of 3D characteristics methods to problems of practical interest has been limited by the computational expense associated with the large track set required for large 3D problems with fine spatial detail. Parallel processing could provide a means to render such calculations feasible [22]; however, to date a successful commercial application for 3D assembly transport remains unachieved.

2.2 Ray-Tracing for Explicit 3D Cartesian Geometry

Development of a ray-tracing method for the method of characteristics in a 3D reactor kinetics methodology can draw on much of the lattice physics experience discussed above, yet there exist fundamental differences that can be exploited to simplify ray-tracing in the present context. While any method of characteristics formulation must utilize a fine-mesh discretization of the problem space, mixed geometry features that are included in most assembly transport models – such as the circle-in-a-square that describes a fuel pin cell – are of much less importance to the kinetics calculation. In fact, if a sufficiently small mesh size is employed, all of the internal cell details can be neglected and each cell treated assumed to be homogeneous. The kinetics method should still be able to model various geometries – Cartesian, cylindrical, hexagonal, etc. To simplify initial development, however, the MOCK-3D ray-tracing algorithm models only 3D Cartesian cells, which are most easily generalized to any problem geometry.

Where a homogeneous fine-mesh spatial discretization is tolerable, the ray-tracing problem can be simplified via additional constraints on the spatial model. In MOCK-3D, two such constraints are imposed. First, all cells are required to have identical spatial dimensions – a constraint which could be relaxed by allowing for multiple regions, each of which contains cells of a certain size and shape. The second constraint enforces periodicity of the characteristics by requiring direct connection of adjacent track segments across cell boundaries. The latter constraint is consistent with DNPL, as described in Section 2.1. A direct result of this application of DNPL is that, where all cells are required to have the same size and shape, every cell in the system model will possess an identical periodic track set. Ray-tracing requirements are reduced to generating the track set for a single “representative” cell, the results of which can be applied uniformly to all cells in the system model.

The angular domain is discretized via the discrete ordinates approximation, which requires that characteristics be projected across the 3D Cartesian cell for each in a set of directions, Ω, as described by a pair of angles – the azimuthal angle, φ, and the polar angle, θ. The azimuthal angle is defined to be the angle between the x-axis and a projection of the 3D characteristic onto the x-y plane. This plane is also called the *azimuthal plane*. The polar angle is defined to be the angle between the z-axis and the 3D characteristic, both of which lie within the *polar plane*. Fig. 1 shows that the azimuthal and polar planes lie perpendicular to one another. The implied separation of the 3D direction space into a pair of 2D planes will be utilized as in Reference 20 to facilitate the 3D ray-tracing process.

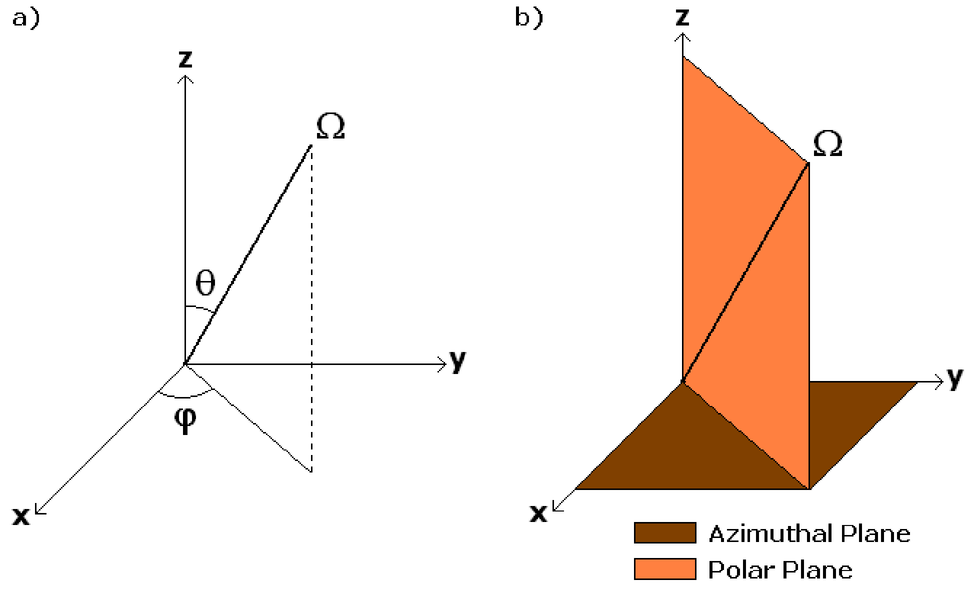

Figure 1: a) Definition of the Characteristic Direction, Ω, b) Definition of the Azimuthal and Polar Planes.

In the discrete ordinates approximation, all directions not explicitly included in the discrete set of angles are accounted for via a set of weighting functions, ω_m, where

$$\sum_m \omega_m = 1\,. \tag{1}$$

The combined set of angles and weights is known as a quadrature set and must be specified prior to execution of the ray-tracing algorithm. In principle, a wide variety of quadrature types can be used with the method of characteristics and simple formulations such as equal angles or equal weights are often employed in lattice physics codes. In MOCK-3D, level symmetric quadrature is utilized [23], with the user free to choose the order of approximation – S_4, S_6, S_{10}, S_{12}, and S_{16} are presently available. Level symmetric quadrature exhibits octant symmetry and track lengths need be calculated for one octant only. The results are applied via rotation to each of the seven remaining octants, further reducing the memory requirements of ray-tracing.

The MOCK-3D ray-tracing procedure for a representative 3D Cartesian cell – summarized pictorially in Fig. 2 – requires as input three sets of user-specified parameters: x-, y-, and z-dimensions of the Cartesian cell, order of approximation for the quadrature set, and desired track separation in the azimuthal and polar planes. User-specified track separations and quadrature must be modified during ray-tracing to enforce periodicity. Ray-tracing proceeds by transforming the 3D problem into a coupled set of 2D problems, each of which is amenable to pre-existing 2D ray-tracing procedures. The first step is to define an azimuthal cross-section of the 3D cell, as shown in Fig. 2a. Application of 2D ray-tracing procedures to this plane yields the set of azimuthal tracks shown in Fig. 2b. In a 2D method of characteristics, these would correspond to the final track set. In 3D, it is assumed that each of these azimuthal tracks will correspond to a line of intersection between the azimuthal plane and one in a set of polar planar sections, as illustrated in Fig. 2c. These polar planar sections must be combined in sequence to generate a periodic polar unit, described by Fig. 2d, to which a modified set of 2D ray-tracing procedures is applied, yielding the final set of tracks, as shown in Fig. 2e.

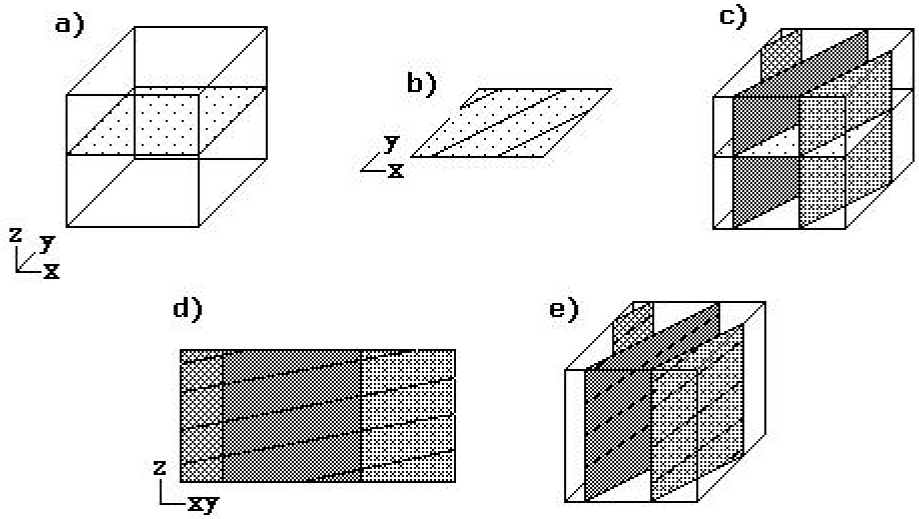

Figure 2: Overview of the MOCK-3D Ray-Tracing Methodology.

The 2D ray-tracing algorithm that is applied to the azimuthal cross-section of the 3D cell, as in Fig. 2b, is essentially identical to the CACTUS-like procedures that are implemented in most lattice physics codes. This algorithm, the details of which have been well-documented elsewhere [8,9], proceeds in four steps: (1) the azimuthal angles and azimuthal track separation are corrected to enforce periodicity; (2) the entry location for each track in the azimuthal set is determined; (3) the length of each azimuthal track is computed; and (4) a set coupling relationships to connect adjacent tracks across cell boundaries is derived. Each of these steps is completed via execution of simple linear geometric algorithms that are facilitated by the absence of mixed-geometry features.

Tracks segments that are calculated for the azimuthal plane are assumed to be lines of intersection between the azimuthal plane and a series of polar planar sections, as depicted in Fig. 2c. The final 3D track segments will all lie within one of these polar planes; however, the planar sections will have variable width and are, thus, not inherently periodic. Before this stage of ray-tracing can be performed, the polar planar sections must be combined into a periodic unit, as demonstrated in Fig. 2d. The sequence in which these planes are connected is determined by the coupling information as defined during ray-tracing for the azimuthal plane. Taken as a whole, this polar unit will be periodic and the 2D azimuthal ray-tracing algorithms can be applied in an identical manner. Ray-tracing in the polar plane is thus performed in six steps: (1) polar planar sections are arranged to form the polar periodic unit; (2) the polar angles and polar track separation are corrected to enforce periodicity; (3) the entry location for each periodic track in the polar periodic unit is determined; (4) the length of each periodic track is computed; (5) the periodic tracks are subdivided to yield the track length of all track segments in each polar planar section; and (6) a coupling relationship to connect adjacent 3D tracks across cell boundaries is derived. In the tracking file and during the ensuing method of characteristics computation, each track is identified by a set of four indices: i, which identifies the cell in which a track is located; m, which identifies the direction; n, which identifies the polar planar section in which a track is located; and p, which identifies the specific track in that plane. Bounds for these indices are stored to the tracking file for use by MOCK-3D in executing looped procedures.

The ray-tracing procedures described above have been implemented into the stand-alone computational module RayMonde, which loads cell dimensions, track spacing, and quadrature information as input, performs ray-tracing for a representative 3D Cartesian cell, and generates a tracking file that will be passed as input to the primary MOCK-3D characteristics solver. The RayMonde tracking file will contain modified quadrature information and track spacing, the length of each track in the representative cell, and information regarding the coupling of tracks across adjacent cell boundaries. By limiting ray-tracing requirements to 1/8 of the total directions in the quadrature set and to a single representative cell in the system model, the memory requirements for storage of this tracking file are greatly minimized and, in fact, are rendered trivial in comparison to the expense associated with other aspects of the 3D method of characteristics calculation. Tracking files generated during the tests described in Section 3, for example, do not exceed 150 kB in size.

2.3 Derivation and Implementation of Steady-State Step Characteristics in MOCK-3D

The method of characteristics provides a means to solve the steady-state neutron transport equation, or Boltzmann equation,

$$\mathbf{\Omega}_m \cdot \nabla \Phi_m^{i,g} + \Sigma_{tr}^{i,g} \Phi_m^{i,g} = \frac{1}{4\pi} \left[\sum_{g^{'}=1}^{G} \Sigma_s^{i,g' \to g} \phi^{i,g} + \frac{\chi^g}{k_{eff}} \sum_{g^{'}=1}^{G} \nu \Sigma_f^{i,g'} \phi^{i,g'} \right], \qquad (2)$$

where the angular flux has been discretized in energy, angle, and space. The source term on the right-hand-side of Equation (2) has been derived using the assumption of isotropic scattering and the transport approximation has been applied. This form of the Boltzmann equation presents neutron transport in integro-differential form, with neutron motion viewed from a neutral, fixed point of reference. If neutron motion is instead represented in Lagrangian coordinates, the Boltzmann equation can be rewritten in characteristic form,

$$\frac{d\Phi_m^{i,g}(s)}{ds} + \Sigma_{tr}^{i,g}(s) \Phi_m^{i,g}(s) = Q_m^{i,g}(s), \qquad (3)$$

where

$$Q_m^{i,g}\left(s\right) \equiv \frac{1}{4\pi}\left[\sum_{g'=1}^{G}\Sigma_s^{i,g'\to g}\left(s\right)\phi^{i,g}\left(s\right)+\frac{\chi^g}{k_{eff}}\sum_{g'=1}^{G}\nu\Sigma_f^{i,g'}\left(s\right)\phi^{i,g'}\left(s\right)\right]. \tag{4}$$

In the step-characteristics method it is assumed that material properties, $\Sigma_x^{i,g}$, and total neutron source, $Q_m^{i,g}$, are constant within each cell, so that

$$\frac{d\Phi_m^{i,g}(s)}{ds}+\overline{\Sigma}_{tr}^{i,g}\Phi_m^{i,g}(s)=\overline{Q}^{i,g}, \tag{5}$$

where

$$\overline{Q}^{i,g} \equiv \frac{1}{4\pi}\left[\sum_{g'=1}^{G}\overline{\Sigma}_s^{i,g'\to g}\overline{\phi}^{i,g}+\frac{\chi^g}{k_{eff}}\sum_{g'=1}^{G}\overline{\nu\Sigma}_f^{i,g'}\overline{\phi}^{i,g'}\right]. \tag{6}$$

The step characteristic equation, Equation (5), can be solved explicitly by integrating along the streaming path, yielding

$$\Phi_{m,n,p}^{i,g}\Big|_{out}=\Phi_{m,n,p}^{i,g}\Big|_{in}e^{-\overline{\Sigma}_{tr}^{i,g}\Delta s_{m,n,p}}+\frac{\overline{Q}^{i,g}}{\overline{\Sigma}_{tr}^{i,g}}\left(1-e^{-\overline{\Sigma}_{tr}^{i,g}\Delta s_{m,n,p}}\right). \tag{7}$$

Equation (7) relates the value of the angular flux at the point where a track exits a cell, $\Phi_{m,n,p}^{i,g}\Big|_{out}$, to the value of the angular flux at the entrant face, $\Phi_{m,n,p}^{i,g}\Big|_{in}$. By solving this equation for every track in the system model, a complete angular flux distribution can be generated. The scalar neutron flux distribution, which is typically of greater interest, can be calculated via

$$\phi^{i,g}=\int_{4\pi}\overline{\Phi}^{i,g}\left(\Omega\right)d\Omega=4\pi\sum_m\overline{\Phi}_m^{i,g}\omega_m, \tag{8}$$

where

$$\overline{\Phi}_{m,n}^{i,g}=\frac{\overline{Q}_m^{i,g}}{\overline{\Sigma}_{tr}^{i,g}}+\frac{\Phi_{m,n,p}^{i,g}\Big|_{in}-\Phi_{m,n,p}^{i,g}\Big|_{out}}{\overline{\Sigma}_{tr}^{i,g}\Delta s_{m,n,p}} \tag{9}$$

and

$$\overline{\Phi}_m^{i,g} = \frac{\sum_{n=1}^{N} \overline{\Phi}_{m,n}^{i,g} \Delta s_{m,n,p} \delta A_m}{\sum_{n=1}^{N} \Delta s_{m,n,p} \delta A_m} . \qquad (10)$$

The main calculation sequence in MOCK-3D utilizes a typical steady-state power iteration scheme [24], which separates the total neutron source term into fission and scattering components and computes the angular and scalar neutron flux distributions via nested inner and outer iterations. The computational sequence in MOCK-3D is outlined in Fig. 3.

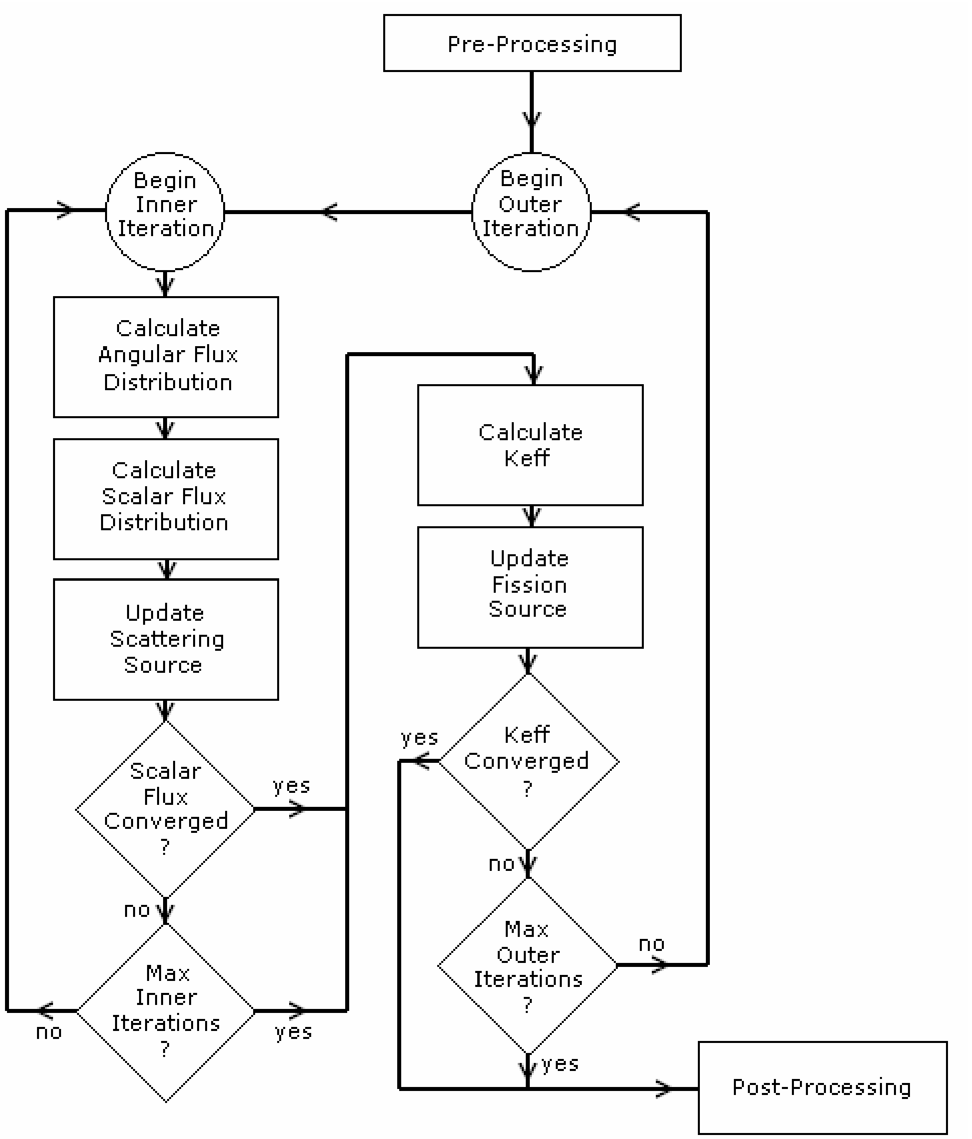

Figure 3: Computational Flow of Mock-3D.

During the inner iterate, Equation (7) is solved for each track in the spatial discretization via a Gauss-Seidel track sweep algorithm, as illustrated in Fig. 4.

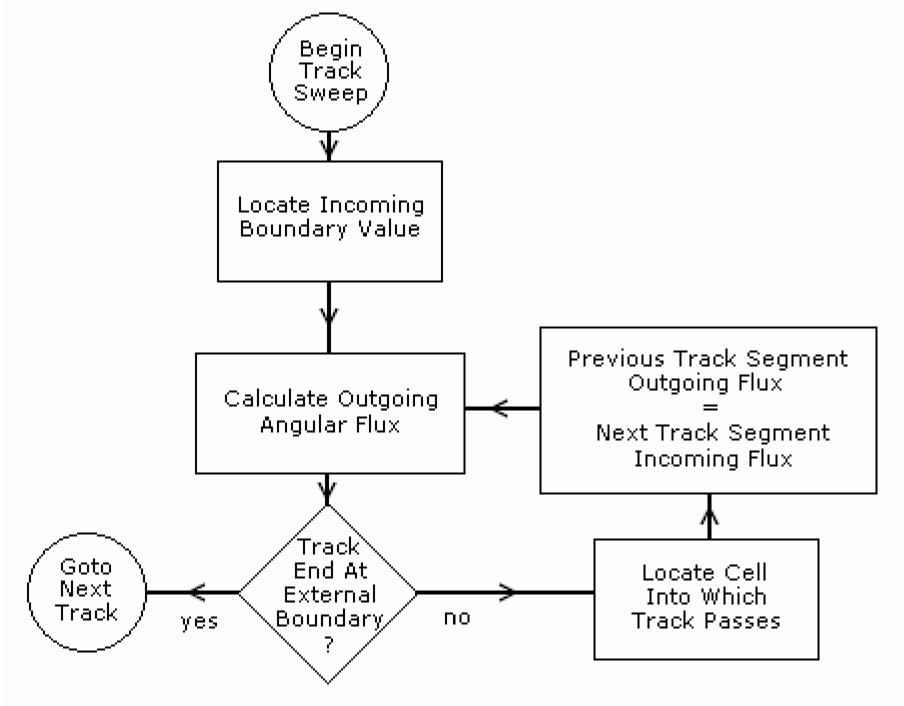
Begin Track Sweep
Locate Incoming Boundary Value
Calculate Outgoing Angular Flux
Previous Track Segment Outgoing Flux = Next Track Segment Incoming Flux
Goto Next Track
yes
Track End At External Boundary ?
no
Locate Cell Into Which Track Passes

Figure 4: Track Sweep Procedure in Mock-3D.

Most transport solvers incorporate a cell-sweep, which would calculate all relevant parameters in a single cell before proceeding to the next cell. In the track-sweep, the path of a single characteristic is traced across the entire system model, thereby crossing multiple cells, before proceeding to the next characteristic. Continuity of the angular flux at all internal cell boundaries and direct connection of tracks at external system boundaries are enforced, with leakage at external boundaries governed by user-specified albedoes. The main advantage to a track-sweep is that it is consistent with techniques for parallelization of the method of characteristics. While parallel computation procedures are not included in MOCK-3D at this time, such an advance is anticipated as a future development to further reduce computational expense. Other forms of computational acceleration, such as the coarse-mesh rebalance method that is frequently employed in lattice physics codes, also have not been implemented in this development version of MOCK-3D.

3. TEST PROBLEM: OECD/NEA 3D NEUTRON TRANSPORT BENCHMARK

The 3D Neutron Transport Benchmark Problem was conducted by researchers from Osaka University from 1988 through 1991 under the auspices of the OECD/NEA Committee on Reactor Physics and is commonly referred to as the "Takeda Benchmark" after its principle investigator. Since the publication of final results in 1991 [7], this problem has been used as a standard test problem for various 3D neutron transport codes. The Takeda benchmark includes four distinct problems, three of which (Models 2 through 4) involve large and spatially complex calculations for a fast breeder reactor. Model 1 is a simple problem that involves a small solid-moderated research reactor and is more useful for performing developmental assessment of the methods presented above.

The reactor core simulated by Takeda Model 1 is based on the Kyoto University Critical Assembly (KUCA) solid-moderated test reactor, in operation at the Kyoto University Research Reactor Institute since 1974. The simulated cubic reactor configuration has total dimensions of 50cm x 50cm x 50cm and consists of a 30cm x 30cm x 30cm cubic reactor core that is surrounded on all sides by reflector material. The fuel is comprised of a combination of 93 w/o enriched U-Al alloy plates and natural uranium plates, with a 9.6 w/o core-averaged ^{235}U enrichment. The core is moderated by polyethelene with a moderator volume to fuel volume ratio of 1.5. Two full-length axial control rods are located within the reflector region, immediately adjacent to the core, at the positions indicated in Fig. 5, which models the reactor core with 1/8 core symmetry and 1.0 cm cubic cells.

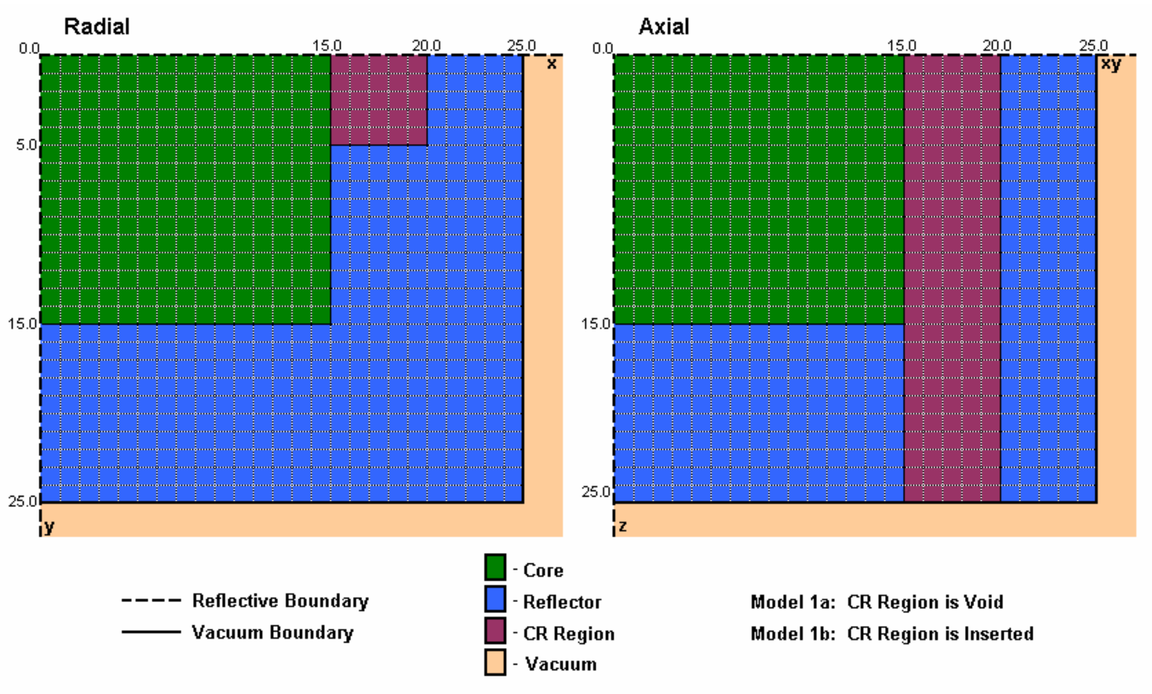

Figure 5. Problem Schematic for Takeda Model 1

Specifications for this problem call for two calculations to be performed. The first calculation, Model 1a, assumes that the control rod is totally withdrawn and the material in the control rod region is modeled as a void. In Model 1b the control rod is fully inserted. Model 1 includes three material regions for which four different sets of material properties must be specified: (1) the core region, with material properties specified in Table I; (2) the reflector region, with material properties specified in Table II; (3) the control rod region with the rod fully withdrawn (i.e., void), with properties given in Table III; and (4) the control rod region with the rod fully inserted, with properties given in Table IV.

Table I: Material Properties for Takeda Model 1, Core Region

Energy Group	$\Sigma_{tr,g}$ (cm^{-1})	$\Sigma_{a,g}$ (cm^{-1})	$\nu\Sigma_{f,g}$ (cm^{-1})	$\Sigma_{s,1\rightarrow g}$ (cm^{-1})	$\Sigma_{tr,2\rightarrow g}$ (cm^{-1})
1	2.23775E-01	8.52709E-03	9.09319E-03	1.92423E-01	0.00000E+00
2	1.03864E+00	1.58196E-01	2.90183E-01	2.28253E-02	8.80439E-01

Table II: Material Properties for Takeda Model 1, Reflector Region

Energy Group	$\Sigma_{tr,g}$ (cm^{-1})	$\Sigma_{a,g}$ (cm^{-1})	$\nu\Sigma_{f,g}$ (cm^{-1})	$\Sigma_{s,1\rightarrow g}$ (cm^{-1})	$\Sigma_{tr,2\rightarrow g}$ (cm^{-1})
1	2.50367E-01	4.16392E-04	0.0000E+00	1.93446E-01	0.00000E+00
2	1.64482E+00	2.02999E-02	0.0000E+00	5.65042E-02	1.62452E+00

Table III: Material Properties for Takeda Model 1 Control Rod Region (Withdrawn)

Energy Group	$\Sigma_{tr,g}$ (cm^{-1})	$\Sigma_{a,g}$ (cm^{-1})	$\nu\Sigma_{f,g}$ (cm^{-1})	$\Sigma_{s,1\rightarrow g}$ (cm^{-1})	$\Sigma_{tr,2\rightarrow g}$ (cm^{-1})
1	1.28407E-02	4.65132E-05	0.0000E+00	1.27700E-02	0.0000E+00
2	1.20676E-02	1.32890E-03	0.0000E+00	2.40997E-05	1.07387E-02

Table IV: Material Properties for Takeda Model 1, Control Rod Region (Inserted)

Energy Group	$\Sigma_{tr,g}$ (cm^{-1})	$\Sigma_{a,g}$ (cm^{-1})	$\nu\Sigma_{f,g}$ (cm^{-1})	$\Sigma_{s,1\rightarrow g}$ (cm^{-1})	$\Sigma_{tr,2\rightarrow g}$ (cm^{-1})
1	8.52325E-02	1.74439E-02	0.0000E+00	6.77241E-02	0.0000E+00
2	2.17460E-01	1.82224E-01	0.0000E+00	6.45461E-05	3.52358E-02

Although Takeda Model 1 simulates an operable test reactor, experimental data was not provided to evaluate the participants' results. Instead, a reference solution was derived from results that were submitted as part of the original benchmark study. Participants in the benchmark used methods that include perturbed and unperturbed Monte Carlo methods, P_N methods, S_N methods, collision probability methods, and others. The reference solution was generated by averaging the results for all of the unperturbed Monte Carlo methods, yielding the so-called "exact Monte Carlo" solution. Relevant reference data is included with the MOCK-3D results below.

The spatial model for calculation of this problem via MOCK-3D discretizes space into homogeneous 1.0 cm cubic cells. Uncorrected track separation is 0.2 cm in the azimuthal and polar directions and S_4 level symmetric quadrature is utilized. Convergence is required to satisfy criteria of 1.0x10^{-6} for k_{eff} during the outer iterate, and 1.0x10^{-6} for the scalar neutron flux during the inner iterate. Computations were performed on the LION-XO cluster, which is operated by the High Performance Computing Group of The Pennsylvania State University Academic Services and Emerging Technologies Department. This multi-processor PC cluster uses the linux operating system and is designed for parallel or serial computations that have large memory requirements.

Three types of results, suggested in the benchmark specifications, are presented for this problem. Table V lists k_{eff} for each case as well as total control rod worth, which is calculated according to:

$$Worth = \frac{1}{k_{eff}^{in}} - \frac{1}{k_{eff}^{out}} \,. \tag{11}$$

Table V: Comparison of k_{eff} and Rod Worth from MOCK-3D to Reference

	Reference		**MOCK-3D**	
	Value	**Std. Dev (%)**	**Value**	**% Diff.**
k_{eff} Rod Out	0.9780	0.06	0.9757	-0.2
k_{eff} Rod In	0.9624	0.06	0.9596	-0.3
Worth	1.66E-02	5.0	1.73E-02	4.1

Next, the scalar neutron flux is averaged over all cells in each material region. These region-averaged scalar neutron flux values are provided in Table VI for Model 1a, where the control rod is withdrawn, and in Table VII for Model 1b, where the rod is inserted.

Table VI: Region-Averaged Scalar Flux – Rod Withdrawn

Energy Group	**Core Region**			**Reflector Region**			**Void Region**		
	Ref.	**MOCK-3D**	**% Diff**	**Ref.**	**MOCK-3D**	**% Diff**	**Ref.**	**MOCK-3D**	**% Diff**
1	4.75E-03	4.72E-03	-0.57	5.93E-04	6.00E-04	1.25	1.45E-03	1.49E-03	2.83
2	8.70E-04	8.73E-04	0.35	9.14E-04	8.28E-04	-9.46	9.74E-04	9.19E-04	-5.62

Table VII: Region-Averaged Scalar Flux – Rod Inserted

Energy Group	Core Region			Reflector Region			Void Region		
	Ref.	MOCK-3D	% Diff	Ref.	MOCK-3D	% Diff	Ref.	MOCK-3D	% Diff
1	4.91E-03	4.86E-03	-0.98	5.91E-04	5.96E-04	0.83	1.22E-03	1.26E-03	2.63
2	8.69E-04	8.69E-04	-0.07	8.79E-04	7.92E-04	-9.87	2.46E-04	2.66E-04	8.00

Finally, a series of 2D scalar neutron flux distributions have been extracted to illustrate the nature of the 3D flux distribution. These profiles are generated at each of the three planes of symmetry and are provided in Appendix A, Figs. 6 through 17.

Table V shows that the accuracy of k_{eff} values for the MOCK-3D calculations are a few tenths of one percent lower than the reference, while the rod worth is over predicted by approximately 4%. While this degree of error in the prediction of k_{eff} is larger than would be sought for comparable transport methods, the error associated with rod worth is less than the standard deviation for the reference solution. The region-averaged scalar neutron flux data reveals potential sources for this error. MOCK-3D is generally accurate in the core region, with errors increasing in the reflector and control rod regions. Errors are also larger for the rod-in case than with the rod withdrawn. These observations suggest that MOCK-3D is failing in the presence of the strongly absorbing control rod. The most likely source for this effect is specification of the model geometry, including such parameters as cell size, track spacing, and the number of directions in the quadrature set. For example, the 1cm cell size used in these calculations is approximately equal to the transport mean free path. As a general rule, a fine-mesh transport model should be specified so that cell dimensions are less than ~½ of the transport mean free path; thus, a cubic cell size of no greater than 0.5 cm would be preferred.

In addition to the results shown here, a series of sensitivities have been conducted to assess this issue more thoroughly. As expected, these studies have shown that the accuracy of the MOCK-3D computation is extremely sensitive to geometry and choosing a small cell size is especially important. The 1.0 cm cubic cell dimensions used above are recommended in the benchmark specifications and appropriate for comparison to the reference results. If cell size is reduced to 0.5 cm cubic, the errors on k_{eff} are reduced to –0.04% for both rod-in and rod-out cases, while rod worth error is reduced to –0.2%. This accuracy is much more acceptable, but comes at the expense of longer runtimes. The problem of CPU expense is exacerbated by the absence of computational acceleration in MOCK-3D. Runtimes for various cell sizes are provided in Table VIII to illustrate this concern. The introduction of computational acceleration and/or parallel processing into more sophisticated computational packages would be expected to reduce these runtimes significantly.

Table VIII: Effect of Cell Size of MOCK-3D Runtime

Cubic Cell Dimension (cm)	Rod In		Rod Out	
	Runtime (hh:mm:ss)	# Outer Iterations	Runtime (hh:mm:ss)	# Outer Iterations
5.000	00:01:09	12	00:00:52	9
2.500	00:06:16	15	00:06:53	15
1.667	00:23:23	17	00:24:26	17
1.250	01:49:38	18	01:34:58	18
1.000	03:55:29	18	03:58:20	18
0.625	12:04:16	16	12:04:22	16
0.500	12:11:20	17	12:11:51	17

In spite of the concerns raised by quantitative analysis of the MOCK-3D results, qualitative analysis of the 2D scalar neutron flux profiles provided in Appendix A shows that MOCK-3D does correctly capture the basic physics of this problem. Insertion of the control rod, for example, causes a significant depression in the scalar flux in the control rod region, as can be clearly seen in Figs. 9 and 17. When the rod is withdrawn, this region is void and peaks in the thermal scalar flux are observed in Figs. 7 and 15. Absence of the absorber also yields an increase in the fast scalar flux, as seen in Figs. 6 and 14. Fig. 6 demonstrates that this effect is strongest near the center of the reactor. The latter effect appears to be more significant than would be expected, however, as neutron streaming through a void would intuitively indicate a flat flux rather than the observed spikes. Reflection of fast and thermal neutrons from the reflector region into the void is one potential cause for this phenomenon. Unfortunately, the benchmark results do not provide comparable 2D flux profiles, nor do other analyses in the literature provide such information, so a more thorough examination of these issues is not possible at this time.

4. CONCLUSIONS

The purpose of the developments presented here is to demonstrate the feasibility of a 3D steady-state method of characteristics for eventual use within a 3D reactor kinetics methodology. To be feasible, the method should provide reasonably accurate results at a practical computational expense. Both of these aspects can be observed in results from MOCK-3D for the Takeda problem. While some loss in accuracy has been noted, this can be corrected by further refinements to the spatial model. Computational inefficiency can be reduced via the introduction of computational acceleration and parallel processing to the MOCK-3D computational methods. More importantly, the inherent expense associated with 3D ray-tracing, which has hindered practical development of a 3D lattice physics application for the method of characteristics, has been effectively eliminated.

Having completed the development of a 3D steady-state method of characteristics, two of the four primary components of a complete 3D reactor kinetics methodology based on the method of characteristics are available: the steady-state method and an efficient ray-tracing procedure. The third component of this methodology – a time-dependent method of characteristics – is the focus of ongoing research. The final component regards the capability to generate suitable material properties for use in these calculations and has not been addressed. Similarity of the reactor kinetics methods to cross-section generation methods, both of which could employ the method of characteristics, could ultimately lead to a practical technique for on-line cross-section generation; however, such a prospect lies beyond the scope of the present research.

REFERENCES

1. S. GOLUOGLU, H. L. DODDS, *Nucl. Sci. Eng.*, **139**, 248 (2001).
2. A. PAUTZ, A. BIRKHOFER, *Nucl. Sci. Eng.*, **145**, 299 (2003).
3. G. C. POMRANING, *Nukleonik*, **6**, 348 (1964).
4. A. RINEISKI, J. Y. DORIATH, *Proc. Joint Intl. Conf. Mathematical Methods and Supercomputing for Nucear Applications*, Saratoga Springs, NY, **2**, 1661 (1997).
5. E. M. GELBARD, WAPD-BT-20 (1960).
6. E. W. LARSEN, J. E. MOREL, J. M. MCGHEE, *Nucl. Sci. Eng.*, **123**, 328 (1996).
7. T. TAKEDA, H, IKEDA, *3-D Neutron Transport Benchmarks*, NEACRP-L-330, OECD/NEA NEACRP (1991)
8. M. J. HALSALL, AEEW-R 1291, UKAEA, Winfrith, UK (1980).
9. D. KNOTT, Ph.D. Thesis, The Pennsylvania State University (1990).
10. V. S. VLADIMIROV, Ph.D. Thesis, V. A. Steclov Mathematics Institute (1959).
11. J. R. ASKEW, AEEW-M 1108, UKAEA, Winfrith, UK (1972).
12. D. KNOTT, M. EDENIUS, *Proc. Int. Conf. Mathematical Methods and Supercomputing in Nuclear Applications*, Karlsruhe, Germany, **2**, 547 (1993).
13. J. R. ASKEW, F. J. KAYERS, P. B. KEMSHELL, *J. Br. Nucl. Energy Soc.*, **5**, 564 (1966).
14. K.-S. KIM, H. Y. KIM, S. Q. ZEE, *Trans. Am. Nucl. Soc.*, **86**, 369 (2002).
15. D. KNOTT, V. W. MILLS, E. WEHLAGE, *Mathematics and Computation, Supercomputing, Reactor Physics and Nuclear and Biological Applications*, Palais des Papes, Avignon, France, on CD-ROM (2005).
16. S. KOSAKA, E. SAJI, *J. Nucl. Sci. Tech.*, **37**, 1015 (2000).
17. N. Z. CHO, G. S. LEE, C. J. PARK, *Trans. Am. Nucl. Soc.*, **86**, 322 (2002).
18. T. TAKEDA, et al, *Mathematics and Computation, Supercomputing, Reactor Physics and Nuclear and Biological Applications*, Palais des Papes, Avignon, France, on CD-ROM (2005).
19. S. KOSAKA, T. TAKEDA, *J. Nucl. Sci. Tech.*, **41**, 645 (2004).
20. I. R. SUSLOV, *Proc. Int. Conf. Mathematical Methods and Supercomputing in Nuclear Applications*, Karlsruhe, Germany, **1**, 752 (1993).
21. G. J. WU and R. ROY, *Ann. Nucl. Energy*, **30**, 1 (2003).
22. M. DAHMANI, R. ROY, *Nucl. Sci. Eng.*, **150**, 1 (2002).
23. E. E. LEWIS, W. F. MILLER, *Computational Methods of Neutron Transport*, American Nuclear Society, La Grange Park, Illinois (1993).
24. G. BELL, S. GLASSTONE, *Nuclear Reactor Theory*, Robert E. Krieger Publishing, Malabar, Florida (1970).

APPENDIX A

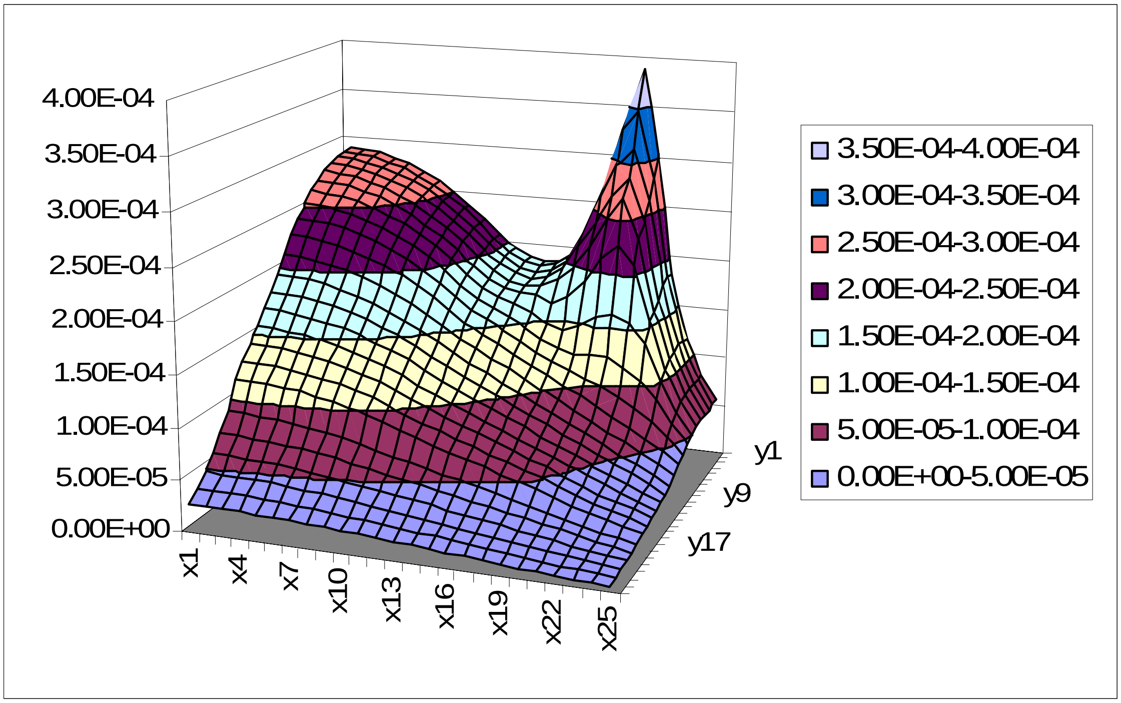

Figure 6: Normalized Scalar Flux, Rod-Out, Group 1, Z-Axis Plane of Symmetry

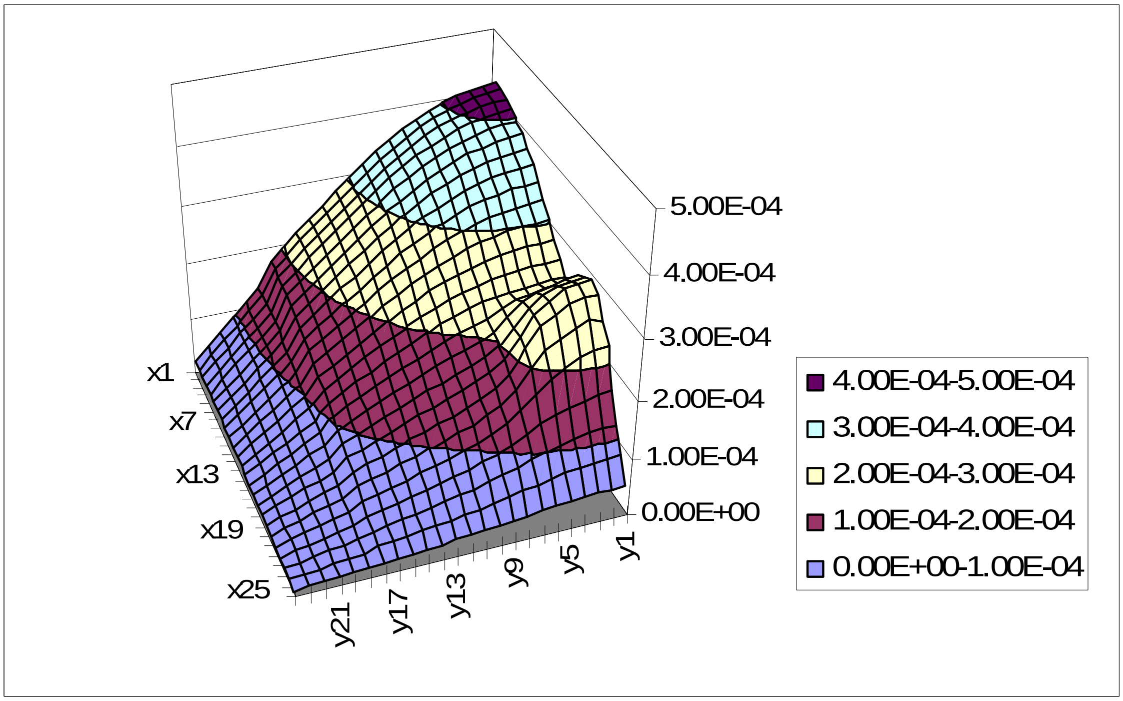

Figure 7: Normalized Scalar Flux, Rod-Out, Group 2, Z-Axis Plane of Symmetry

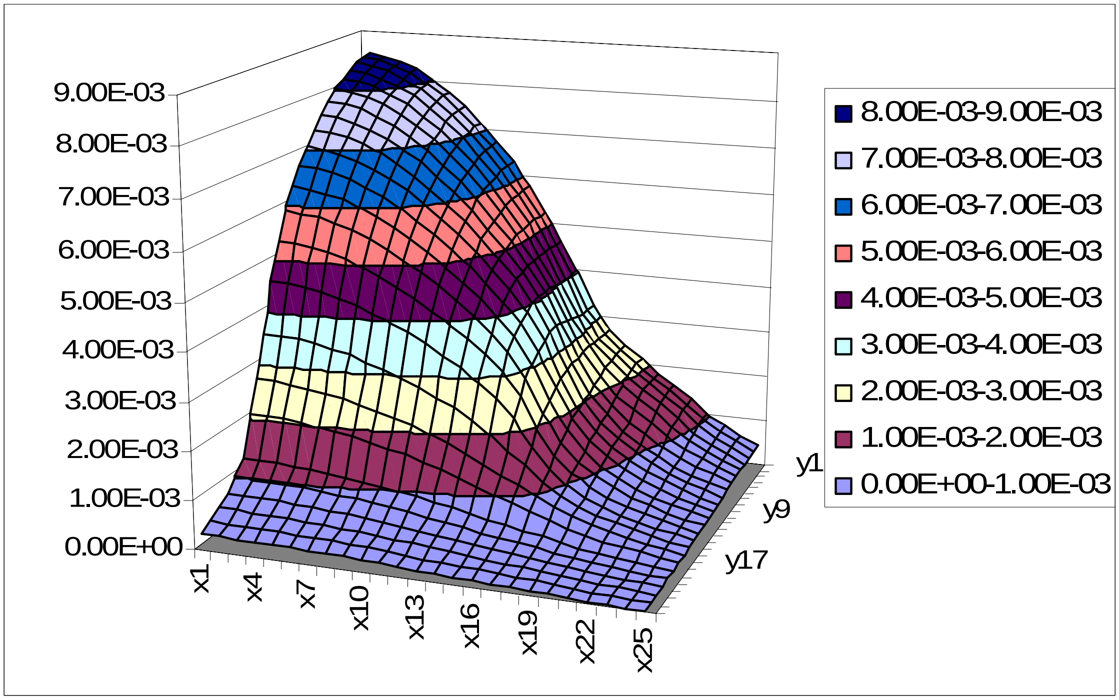

Figure 8: Normalized Scalar Flux, Rod-In, Group 1, Z-Axis Plane of Symmetry

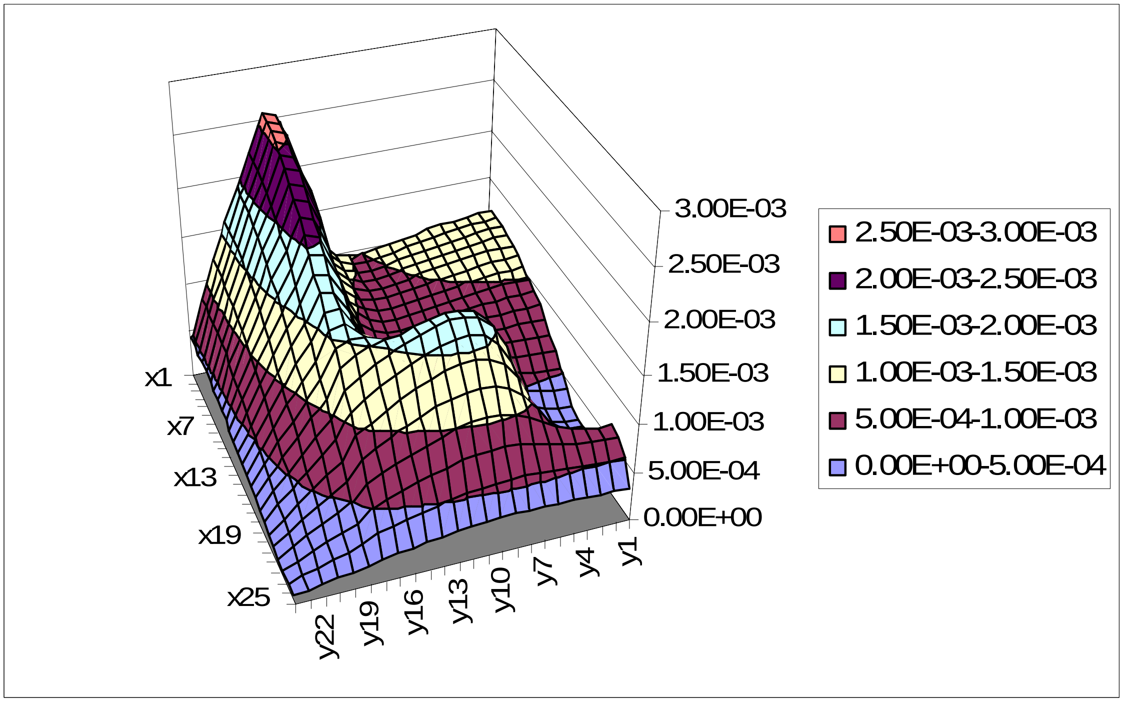

Figure 9: Normalized Scalar Flux, Rod-In, Group 2, Z-Axis Plane of Symmetry

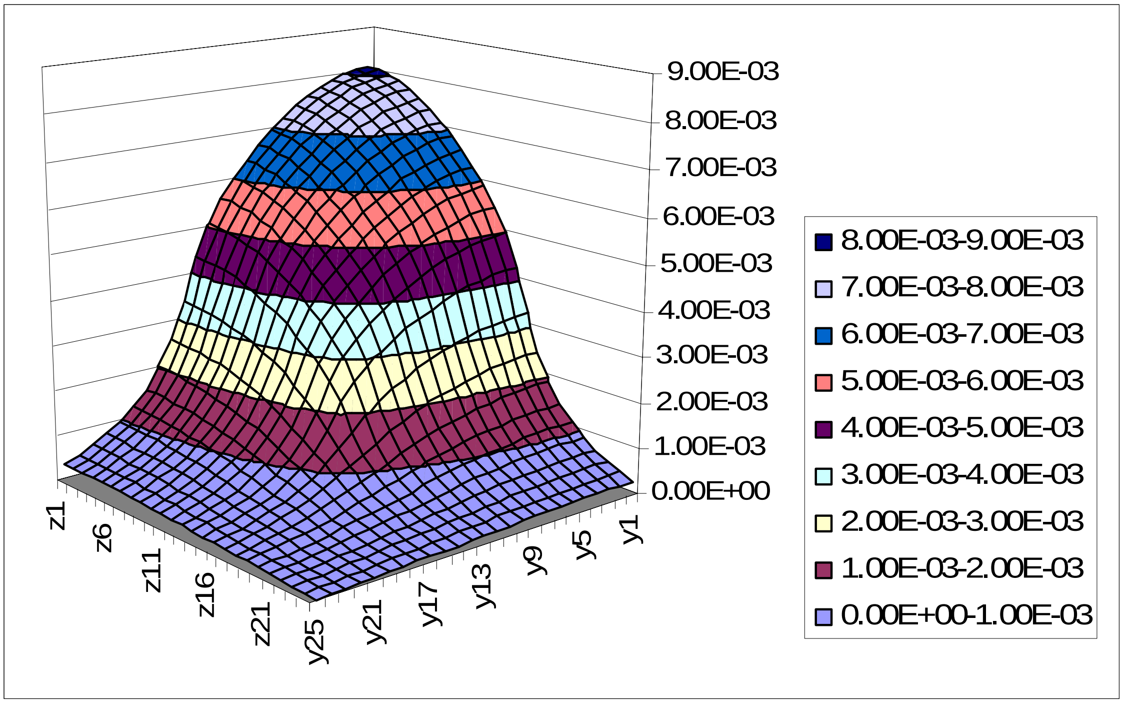

Figure 10: Normalized Scalar Flux, Rod-Out, Group 1, X-Axis Plane of Symmetry

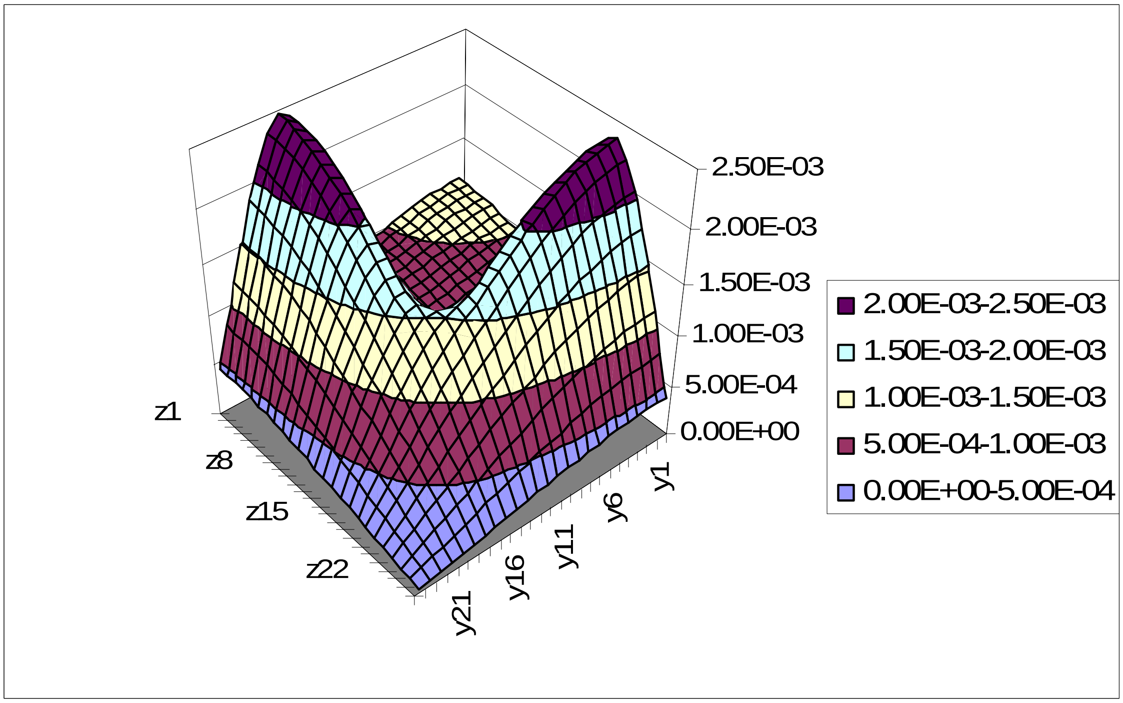

Figure 11: Normalized Scalar Flux, Rod-Out, Group 2, X-Axis Plane of Symmetry

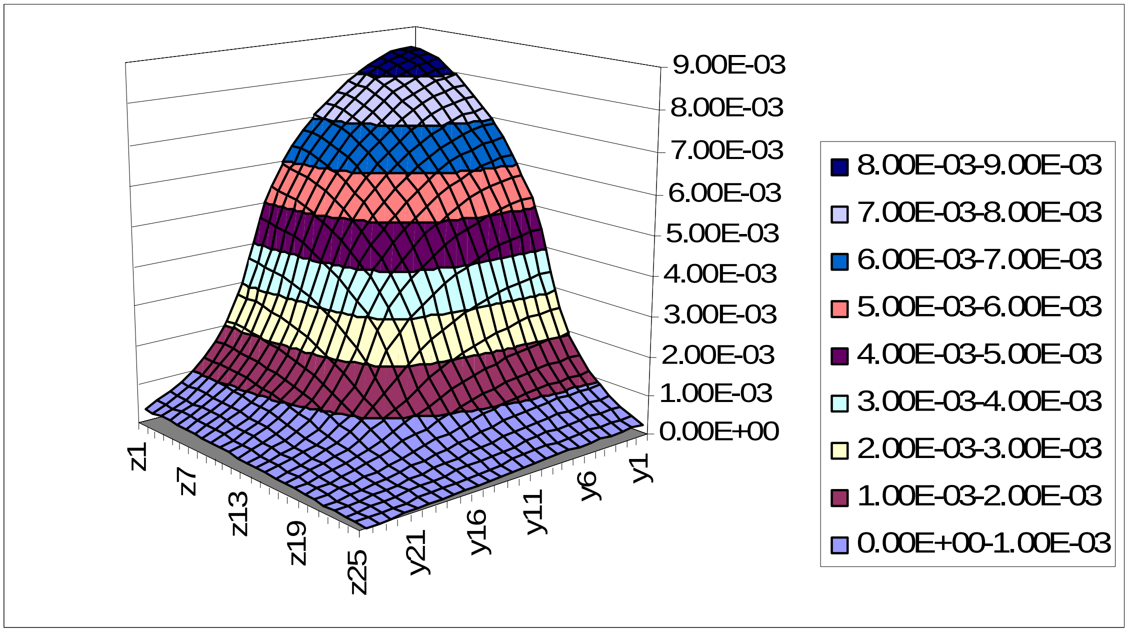

Figure 12: Normalized Scalar Flux, Rod-In, Group 1, X-Axis Plane of Symmetry

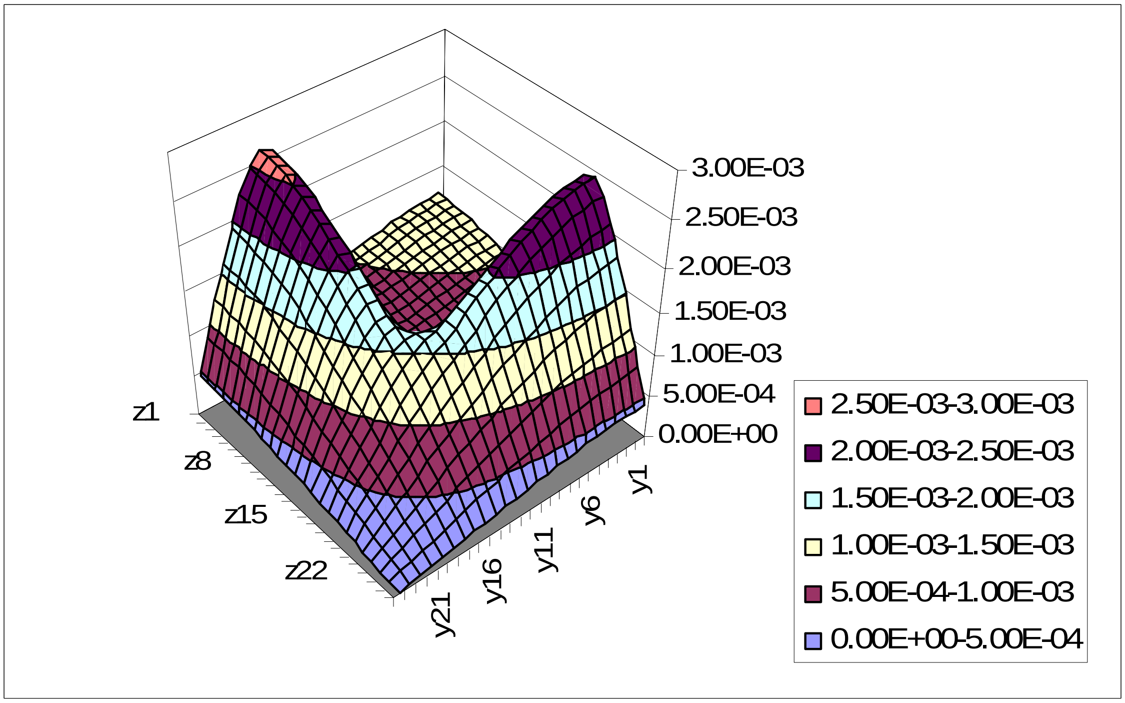

Figure 13: Normalized Scalar Flux, Rod-In, Group 2, X-Axis Plane of Symmetry

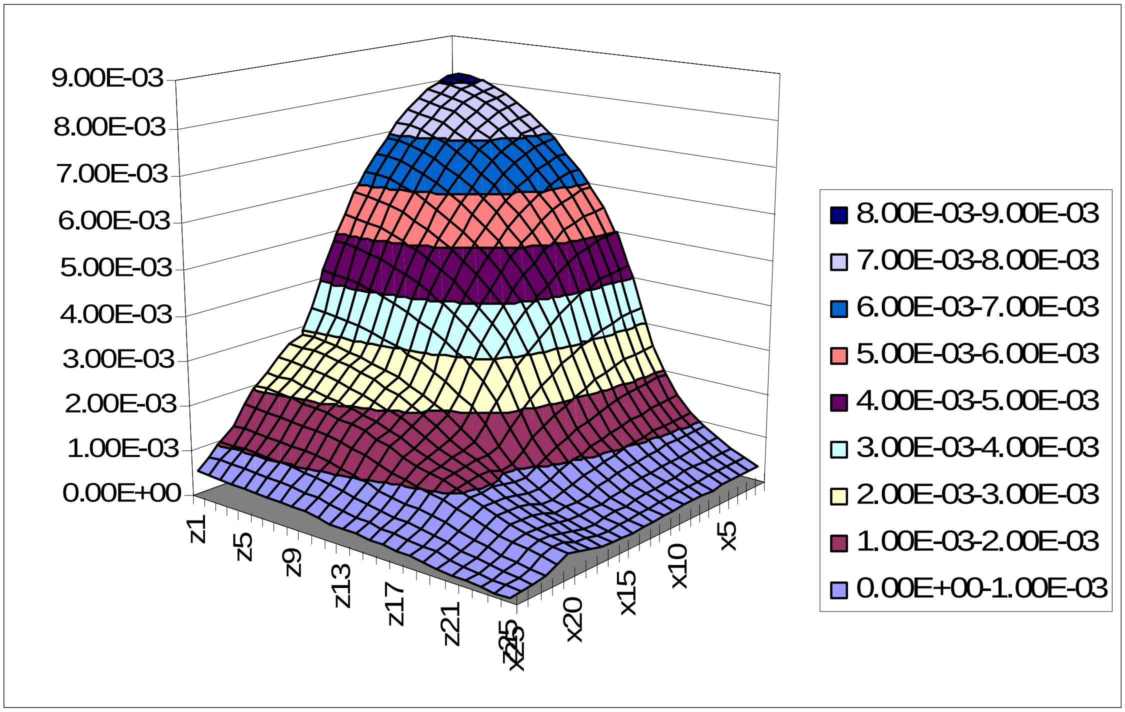

Figure 14: Normalized Scalar Flux, Rod-Out, Group 1, Y-Axis Plane of Symmetry

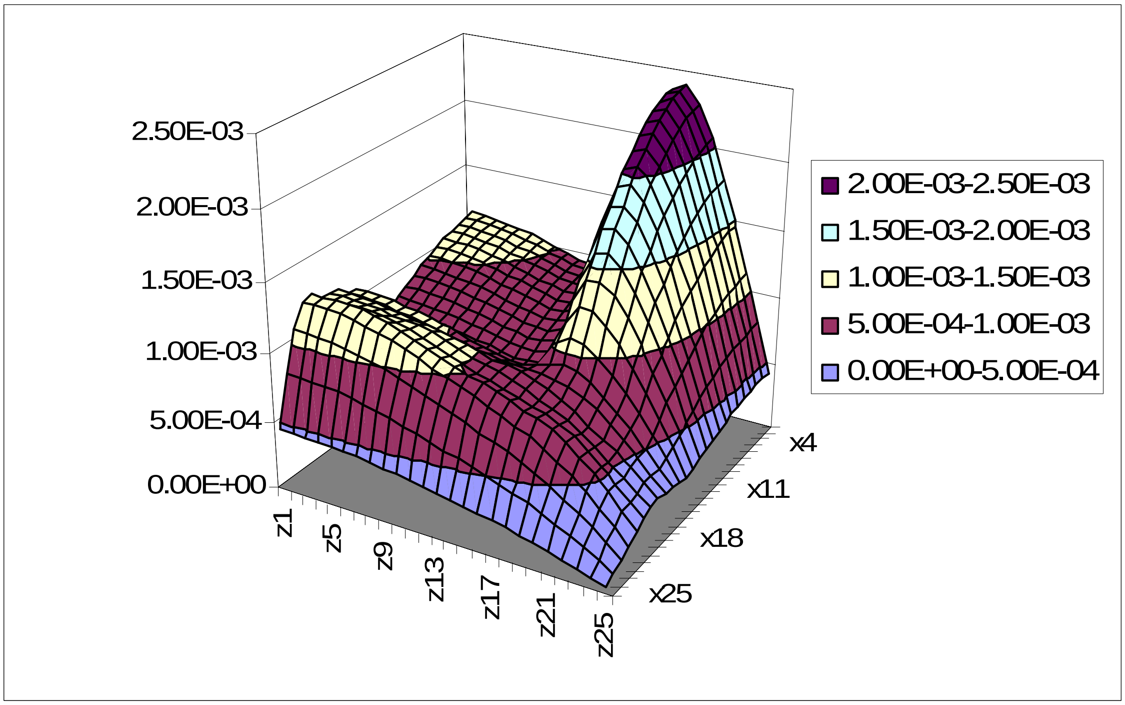

Figure 15: Normalized Scalar Flux, Rod-Out, Group 2, Y-Axis Plane of Symmetry

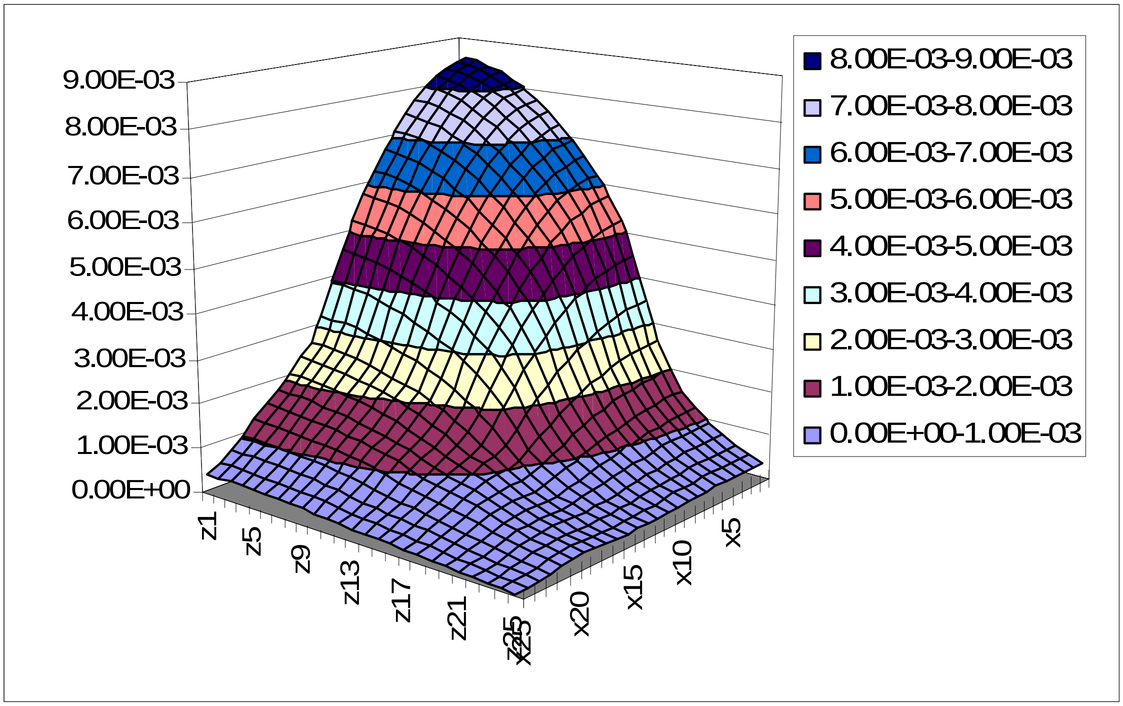

Figure 16: Normalized Scalar Flux, Rod-In, Group 1, Y-Axis Plane of Symmetry

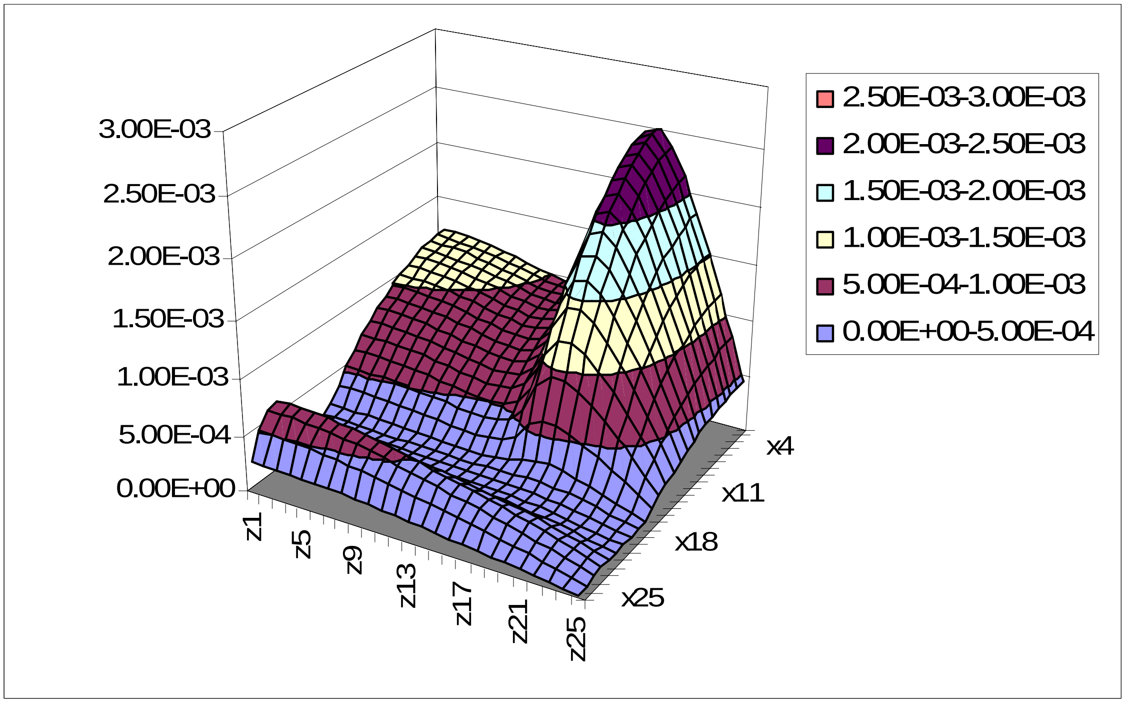

Figure 17: Normalized Scalar Flux, Rod-In, Group 2, Y-Axis Plane of Symmetry