

# NE255 Final Report:

## Deterministic solution of TE in 1-D spherical coordinates

Milos Atz

December 13, 2016

### 1 Introduction

For a geologic repository containing fissile materials, a criticality safety assessment (CSA) is necessary to ensure subcriticality. The CSA should cover (1) the canister environment, (2) the near field, which includes the engineered barriers, and (3) the far field, where fissile material from multiple canisters can accumulate in geologic formations after canister failure and subsurface transport [1]. The far-field case is complicated by long timescales and uncertainties regarding material compositions and geometry. One way to assess the far-field criticality risk is by using conservative neutronic analysis alongside deterministic transport models [1]. Understanding how the multiplication factor and critical mass of a deposition of fissile material behave under variable conditions can allow insights into mitigating criticality risk by controlling fuel cycle management, waste processing, and repository design.

In previous work, criticality calculations have been carried out using MCNP v.6.1 [2] [3]. While MCNP can produce good estimates of the multiplication factor, the calculations are expensive when performing parametric study and iterating to find the minimum critical mass. A deterministic model can give results more quickly than MCNP. At the very least, such a model can be used to validate findings in MCNP. If expanded with an extensive

data library, it could grow into more. In this work, a 1-D discrete ordinates transport solver for a two-region spherical geometry in curvilinear coordinates is developed. Applied to criticality safety, the model represents a water-saturated deposition of fissile material in host rock with a reflector of the host rock material.

## 2 Mathematics

Using curvilinear coordinates in the discrete ordinates method allows for straightforward treatment of spherical systems. The geometry can be considered 1-D. In 1-D curvilinear coordinates, the angular flux,  $\psi(\rho, \mu)$  is dependent on only  $\rho$ , the distance from the origin, and  $\mu$ , the directional cosine of  $\hat{\Omega}$  with respect to the radial direction at  $\rho$ . The transport equation in spherical coordinate is shown below [4].

$$\mu \frac{\partial}{\partial \rho} \psi(\rho, \mu) + \frac{(1 - \mu^2)}{\rho} \frac{\partial}{\partial \mu} \psi(\rho, \mu) + \sigma(\rho) \psi(\rho, \mu) = q(\rho, \mu) \quad (1)$$

In Equation 1, the first two terms represent streaming with respect to position and angle. The second term is the total interaction term, where  $\sigma$  is the total cross section. On the right hand side,  $q$  is this emission density [4]:

$$q(\rho, \mu) = \sum_{l=0}^L (2l + 1) \sigma_l(\rho) P_l(\mu) \phi_l(\rho) + s(\rho, \mu) \quad (2)$$

In Equation 2, the  $\phi_l$  are the Legendre moments of the angular flux. To ensure that any approximations to the transport equation retains its particle conservation properties, it is often written in conservation form, which requires rewriting the streaming operator [4]:

$$\frac{\mu}{\rho^2} \frac{\partial}{\partial \rho} \rho^2 \psi(\rho, \mu) + \frac{1}{\rho} \frac{\partial}{\partial \mu} (1 - \mu^2) \psi(\rho, \mu) + \sigma(\rho) \psi(\rho, \mu) = q(\rho, \mu) \quad (3)$$

In discrete ordinates, angle and space must be discretized, and curvilinear coordinates is no exception. However, the discretization is different than for Cartesian coordinates. As a particle streams through the system, its position and its angle change relative to the reference point (the origin of the sphere). Thus, two layers of discretization are necessary

and the equations must be treated in a special way. In Appendix A, the discretization in angle and space are derived in full; derivations are sourced entirely based on from the "Computational Methods of Neutron Transport" by E. E. Lewis and W. F. Miller [4].

A radial mesh is imposed on the sphere dividing it into concentric cells with centers at positions  $\rho_i$ . The inner and outer bounds of the cells are  $\rho_{i-1/2}$  and  $\rho_{i+1/2}$  respectively. The cell average flux is defined as follows, where  $V_i$  is the volume of the cell.

$$\psi_{ni} = \frac{1}{V_i} 4\pi \int_{\rho_{i-1/2}}^{\rho_{i+1/2}} d\rho \rho^2 \psi_n(\rho) \quad (4)$$

Angle is discretized by expanding the angular flux in the first  $N + 1$  Legendre polynomials  $P_N$ . Gaussian quadrature is used so that each  $\mu_n$  in the set of  $N$   $\mu$  are taken to be the roots of  $P_N(\mu_n) = 0, n = 1, 2, \dots, N$ , where  $P_N$  is . The quadrature weights are determined such that the quadrature correctly integrates the Legendre polynomials  $P_0(\mu)$  through  $P_{N-1}(\mu)$ .

$$\frac{1}{2} \sum_{n=1}^N w_n P_l(\mu_n) = \delta_{l0}, \quad l = 0, 1, \dots, N - 1 \quad (5)$$

The treatment of angle in curvilinear coordinates is tricky. Angular differencing coefficients  $\alpha_{n\pm 1/2}$  are required to difference in angle while continuing to use commonly used quadratures like Gaussian quadrature. These coefficients are appear in the equation used to solve for cell average flux in each cell.

$$\begin{aligned} \psi_{ni} = & \left[ 2|\mu_n| A_{i\mp 1/2} + \frac{4}{w_n} (A_{i+1/2} - A_{i-1/2}) \alpha_{n+1/2} + V_i \sigma_i \right]^{-1} \\ & \times \left( |\mu_n| (A_{i+1/2} - A_{i-1/2}) \psi_{n,i\pm 1/2} + \right. \\ & \left. \frac{2}{w_n} (A_{i+1/2} - A_{i-1/2}) (\alpha_{n+1/2} - \alpha_{n-1/2}) \psi_{n-1/2,i} + V_i Q_{ni} \right), \end{aligned} \quad (6)$$

$\mu \leq 0$

There are three terms in Equation 6, each divided by the same denominator (the first line). The first term represents particles incoming from the preceding cell in space  $i \pm 1/2$  into the current cell  $i$ . The second term represents particles incoming from the previous

half-angle  $n - 1/2$  into the current angle  $n$ . The third term is particles that are born in cell  $i$ . The denominator accounts for particles leaving the cell and particles that are absorbed in the cell. This equation, solved alongside the diamond difference relationships in space and angle in Equations 7 and 8, is used to solve for the flux in the sphere.

$$\psi_{n+1/2,i} = 2\psi_{n,i} - \psi_{n-1/2,i} \quad (7)$$

$$\psi_{n,i\pm 1/2} = 2\psi_{n,i} - \psi_{n,i\mp 1/2} \quad (8)$$

### 3 Algorithms

The space-angle sweep in curvilinear coordinates is similar to that in Cartesian coordinates. In spherical geometry, the starting angle is radially inward,  $\mu = -1$  and starting spatial mesh cell is the outermost cell. Starting with  $\psi_{1/2,I+1/2}$ , the march proceeds inward (decreasing  $i$ ), sequentially solving for each  $\psi_{1/2,i}$  using Equation 9.

$$\psi_{1/2,i} = \frac{2\psi_{1/2,i+1/2} + (\rho_{i+1/2} - \rho_{i-1/2}) Q_{ni}}{2 + \sigma_i (\rho_{i+1/2} - \rho_{i-1/2})} \quad (9)$$

Then, the  $\psi_{1,i}$  are calculated, again starting from the outermost cell. This goes on until all angular fluxes for  $\mu_n < 0$ , ( $n \leq N/2$ ) are known. Then, the starting fluxes at  $i = 1/2$  for  $n > N/2$  are determined and the process is repeated, except this time with  $\mu > 0$ . These fluxes are combined according to the quadrature with the fluxes from the inward sweep and are used to update the scattering source for the next iteration. Figure 3 demonstrates the process graphically for  $N = 4$  [4].

### 4 Code Use

Using the code requires the user to have Python and numerous packages, all available and automatically installed with Anaconda. The code is written in a detailed and commented Jupyter notebook so that the user can see inside the algorithm and implementation.

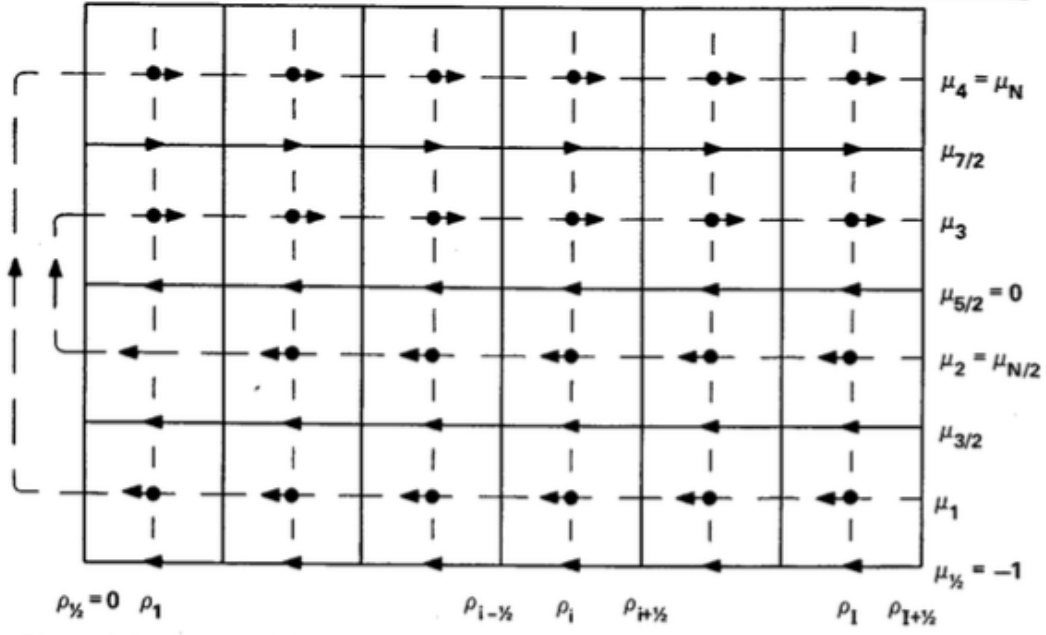


Figure 1: Space-angle mesh sweep for 1-D spherical geometry [4].

Compilation of the code has been started but not completed because the code isn't yet fully functional (teaser, more details to come). Readers are encouraged to view the notebook to understand the details and subroutines.

Currently, the user supplies many inputs that specify various aspects of the problem.

- Radius of the core
- Number of cells in the core
- Degree of Gaussian quadrature ( $N$ )
- Order of  $P_N$  expansion of scattering ( $L$ )
- The incoming flux at the sphere edge
- $\Sigma_{tot}, \Sigma_s$

The user also specifies the type of source, either a point source or a uniform source. If a point source is selected, the user must identify where the point source is located. If the user desires a reflecting region around the core, they must also specify the following for the reflector:

- Thickness of the reflector
- Number of cells in the reflector
- $\Sigma_{tot}, \Sigma_s$

## 5 Test Problems and Results

In this section, we will come to the unfortunate realization that the code doesn't work and will discuss some reasons for why that might be. To try and simplify that discussion, the reflector is omitted in all tests, so the results presented here are for a bare spherical system. Three test problems are identified to check the performance of the code without a reflector. All are easy to solve either conceptually or with diffusion theory. These solutions are implemented in code in an accompanying Jupyter notebook.

1. Bare sphere, uniform, isotropic source in near-infinite medium, no scattering
2. Bare sphere, isotropic point source at the center of the sphere
3. Bare sphere, uniform external source
4. Two region bare sphere, inner region has uniform external source

The solution to the near-infinite system with uniform, isotropic source is a constant with respect to position and angle:  $\phi = S/\sigma$ . If the system is large enough, this result should be obtainable by the code. This is also the case used by Lewis and Miller [4] to refine the angular differencing in the derivation of the discretization. Figure 2 shows the

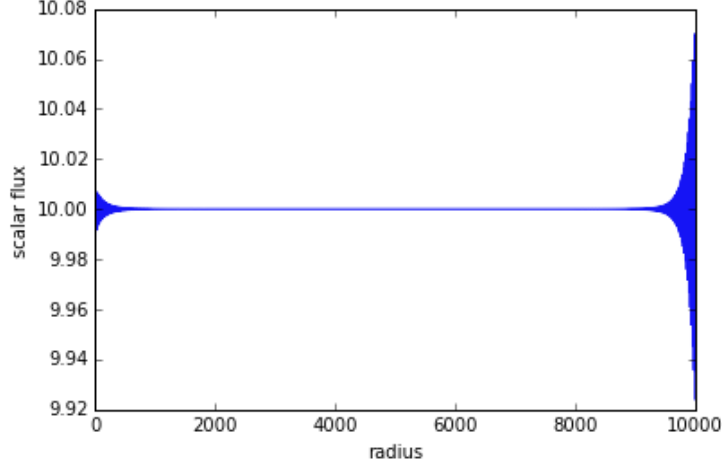


Figure 2: Solution for flux in a bare sphere with uniform isotropic source with no scattering. In this system the flux equals the source (40.0) divided by the cross section (4.0).

flux as a function of radius in the sphere with isotropic external source  $s_o = 40.0$  and total cross section  $\sigma = 4.0$ . For this figure, Gaussian quadrature of degree  $N = 4$  is used. The result shows flux constant across space with the exception of oscillations that arise near the origin and by the boundary. These are the focus of later discussions because the quality of the result suffers near the origin and by the boundary, in some problems to the point that the answer is clearly wrong. More on that later.

The second test problem involves introducing scattering. The diffusion solution to this equation is straightforward and is shown in Equation 10.

$$\phi(r) = \frac{s_o \sinh((R - r)/L)}{4\pi D \sinh(R/L)} \quad (10)$$

In the transport solution, scattering is expanded in Legendre polynomials according to the  $P_N$  method with order  $L = 4$ . A source is placed in the first cell. In the result, plotted in Figure 3, the degree of Gaussian quadrature is  $N = 2$  and 20 radial mesh cells of uniform width are used. The diffusion result is plotted in the same figure, according to the equation for flux. Both the diffusion and code solutions used source values of  $s_o = 10.0$ ,  $\sigma_{tot} = 4.0$ ,

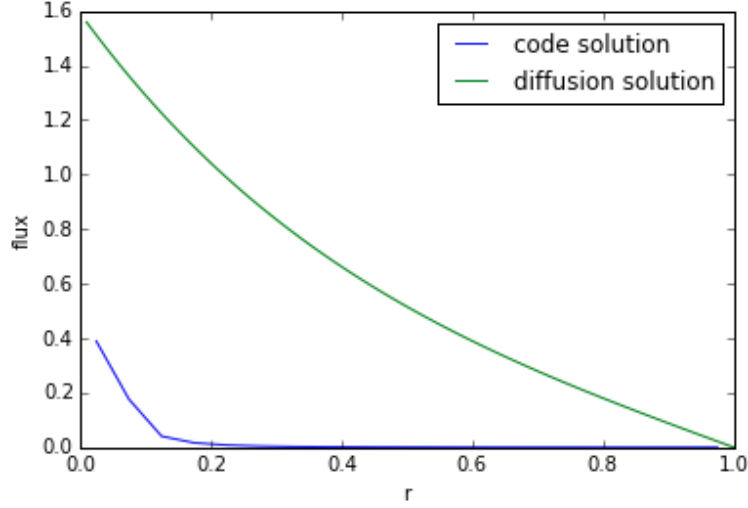


Figure 3: Comparison of diffusion and code solutions for point source at center of sphere.

$\sigma_s = 2.0$ . This is an unfortunate result. Though the code solution appears to behave somewhat properly, the magnitude of the difference between the two solutions is high. If the diffusion solution is taken to be “more correct”, it would seem as though the code is underpredicting the scattering contribution to the flux.

Things get worse when assessing the third test problem, a bare sphere with uniform distributed source with scattering. The solution by the diffusion approximation is given by Equation 11 [5].

$$\phi(r) = \frac{-s_o(R+a)}{\Sigma_a \sinh((R+a)/L)} \frac{1}{r} \sinh(r/L) + \frac{s_o}{\Sigma_a} \quad (11)$$

The solution is obtained from the code using 200 uniformly distributed radial meshes, Gaussian quadrature of degree  $N = 2$  and scattering expansion with  $L = 4$ . The source in both diffusion and code solutions is  $s_o = 0.5$  and cross sections are  $\sigma_{tot} = 2.5$ ,  $\sigma_s = 0.5$ . Ignoring the behavior at the origin, the code performs reasonably well compared to the diffusion solution.

The behavior at the origin is difficult to assess, but we can get better insight into the



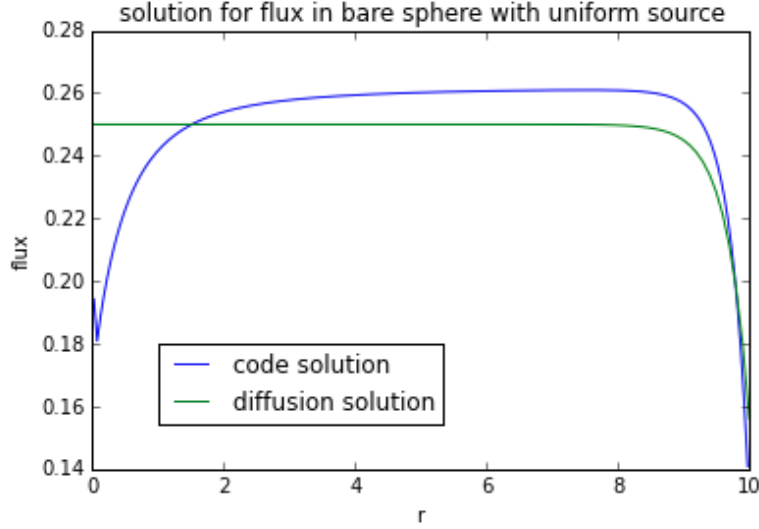


Figure 4: Comparison of the code and diffusion solutions for flux in a bare sphere with uniform source.

behavior of the code by examining the angular flux and half-angle fluxes at convergence. Figure 5 shows the same code solution but includes the angular and half-angle fluxes in the  $\pm\mu$  directions. Because  $N = 2$ , the blue dots mark the "starting flux", when  $n = 1/2$ . Based on this half-angle flux, the angular flux represented by the blue line is calculated for  $n = 1$ . Similarly, the red dots mark the half-angle flux where  $n = 3/2$  and the red line is the angular flux for  $n = 2$ . The green line is the scalar flux. The sharp dip in angular flux for  $n = 2$  at the origin is due to the difference between the incoming flux at the origin, which converges to 0.2537 in this example, and the low half-angle flux (red dots).

At this time, these issues have not been fully diagnosed. The fluxes behave appropriately at the boundary and give reasonable results through the majority of the sphere. I suspect that there are issues with the treatment of flux at the center and the integration of the angular fluxes. However, despite many frustrating hours, I have not been able to find these issues in my algorithm.

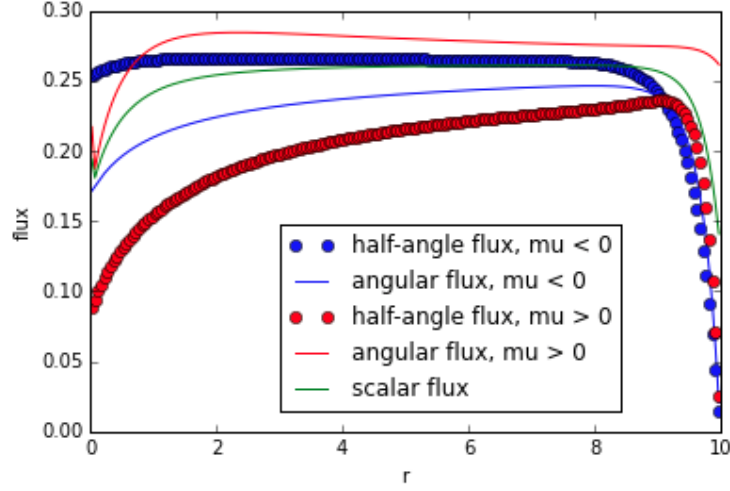


Figure 5: Breakdown of the half angle and angular fluxes that comprise the code solution for uniform source in bare sphere.

Finally, the fourth test problem demonstrates the use of the reflector. The diffusion solution for a two region bare sphere in which the inner sphere has a uniform source is given by Equations 12 and 13. The constants are given in 14 and 15;  $A$ ,  $B$ ,  $C$ , and  $D$  are constants whose values can be found along with the full derivation in a supplemental Jupyter notebook [5].

$$\phi(r) = \frac{2C_1}{r} \sinh(r/L) + \frac{s_o}{\Sigma_{a,1}}, \quad 0 \leq r < R_1 \quad (12)$$

$$\phi(r) = \frac{C_3}{r} e^{\frac{r}{L_2}} - \frac{C_3}{r} e^{2\frac{R_2+a}{L_2}} e^{-\frac{r}{L_2}}, \quad R_1 \leq r < R_2 \quad (13)$$

$$C_1 = \left( \frac{C_3}{R_1} \left[ e^{\frac{R_1}{L_2}} - A e^{-\frac{R_1}{L_2}} \right] - \frac{s_o}{\Sigma_{a,1}} \right) \quad (14)$$

$$C_3 = \frac{2D_1 s_o}{\Sigma_{a,1}} \left[ \frac{2D_1}{R_1} \left( e^{\frac{R_1}{L_2}} - A e^{-\frac{R_1}{L_2}} \right) - \frac{DB}{C} \right]^{-1} \quad (15)$$

The diffusion solution and the code solution are plotted in Figure 6 for the case where the radius of the core and reflector thickness are  $R_1 = 5.0$  and  $R_t = 10.0$ , respectively. The total and scattering cross sections are set to 2.0 and 1.0, respectively, in the core, and 5.0

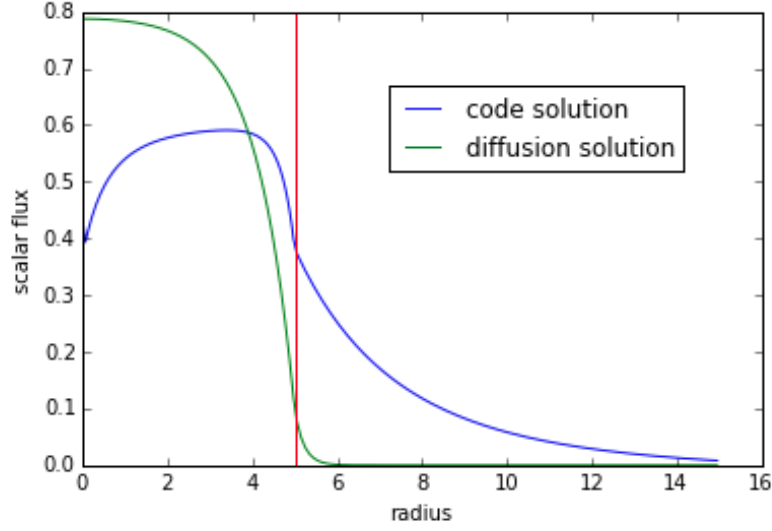


Figure 6: Comparison of code output with diffusion result for two region sphere; the red line marks the interface between the two regions

and 2.0, respectively, in the reflector. The uniform flux in the core is  $s_o = 0.8$ . The code solution used quadrature degree  $N = 2$  and scattering expansion order  $L = 4$ .

The code results matches up somewhat reasonably with the diffusion result, again with the exception of the behavior at the origin.

## 6 Summary

In this work, we covered the implementation of a solver for the neutron transport equation in 1-D spherical coordinates in a variety of settings: with uniform and point sources, bare and reflected. The derivation and explanations of the special treatment required to handle the change in reference position and angle as the particle streams are discussed (in detail in Appendix A). The code result is shown to produce reasonable results for simple problems but shows erratic behavior at the origin, as evidenced when comparing it against

diffusion approximation results for similar systems. More work is required to diagnose and address this issue. Future work can and should involve compiling and packaging the code into a nice, user-friendly product.

## References

- [1] Joonhong Ahn. *Criticality Safety of Geologic Disposal for High-Level Radioactive Wastes* 2006: Nuclear Engineering and Technology v38:6.
- [2] Xudong Liu, Joonhong Ahn, Fumio Hirano. *Conditions for criticality by uranium deposition in water-saturated geological formations* 2014: Journal of Nuclear Science and Technology v.52
- [3] D.B. Pelowitz, ed. *MCNP6TM User's Manual* 2011, Los Alamos.
- [4] E.E. Lewis and W.F. Miller. *Computational Methods of Neutron Transport* 1984: John Wiley & Sons.
- [5] E.E. Lewis. *Fundamentals of Nuclear Reactor Physics* 2008: Academic Press.

## A Discretization of the TE in 1-D spherical coordinates

### A.1 Angular Discretization

In curvilinear coordinates, we can attempt to represent angle as a set of discrete ordinates equations, analogous to the treatment in slab geometry. We evaluate the equation at the same  $\mu_n$  as in the slab geometry. However, because of the different nature of the curvilinear equations, this procedure can no longer be considered as tracing neutrons along discrete paths through space. If we want to utilize desirable quadrature sets (e.g. Gauss and double Gauss quadrature), the angular redistribution effects caused by curvilinear

coordinates must be taken into account. Suppose, for example, a diamond differencing relationship like the one in Equation 16 is applied directly to Equation 3.

$$\psi(\rho, \mu_n) \approx \frac{1}{2} [\psi(\rho, \mu_{n+1/2}) + \psi(\rho, \mu_{n-1/2})] \quad (16)$$

This simple method places too many restrictions on the ordinates for even simple problems. The solution to Equation 3 for a uniform isotropic source in an infinite medium is just a constant,  $\psi = S/\sigma$ . In that case, the sum of the two streaming terms in Equation 3 should vanish. For this to be true,  $\psi(\rho, \mu_{n\pm 1/2}) = \psi(\rho, \mu_n) = \phi$ . Using the approximation in Equation 16 requires  $\mu_n$  to be the average of  $\mu_{n+1/2}$  and  $\mu_{n-1/2}$ . Gauss and double Gauss quadrature do not meet this constraint even when  $N = 2$ . Instead, angular differencing coefficients are introduced into the conservation form of the transport equation in spherical geometry in Equation 3. Here,  $n$  represents an angle from the quadrature set, and  $\psi_n(\rho) = \psi(\rho, \mu_n)$  for  $\mu_1 < \mu_2 < \dots < \mu_N$  with  $\mu_{1/2} \equiv -1$  and  $\mu_{N+1/2} = 1$ .

$$\frac{\mu_n}{\rho^2} \frac{\partial}{\partial \rho} \rho^2 \psi_n(\rho) + \frac{2}{\rho w_n} [\alpha_{n+1/2} \psi_{n+1/2}(\rho) - \alpha_{n-1/2} \psi_{n-1/2}(\rho)] + \sigma(\rho) \psi_n(\rho) = q_n(\rho) \quad (17)$$

This method can account for the disappearance of the streaming term required to correctly solve for a uniform isotropic flux in an infinite medium. In order to ensure this result, each  $\alpha_{n+1/2}$  is defined recursively. If the value of  $\alpha_{1/2}$  is known, all  $\alpha_{n+1/2}$  can be determined uniquely in terms of quadrature parameters.

$$\alpha_{n+1/2} = \alpha_{n-1/2} - \mu_n w_n \quad (18)$$

In addition, Equation 17 must obey the neutron balance condition, which is obtained by integrating Equation 3, where  $\phi$  is the scalar flux,  $S$  is the source,  $J$  is the current, and  $\sigma_r = \sigma - \sigma_0$ .

$$\frac{1}{\rho^2} \frac{\partial}{\partial \rho} \rho^2 J(\rho) + \sigma_r(\rho) \phi(\rho) = S(\rho) \quad (19)$$

Expressed in terms of the discrete ordinates, Equation 19 becomes the following:

$$\frac{1}{\rho^2} \frac{\partial}{\partial \rho} \rho^2 \frac{1}{2} \sum_{n=1}^N w_n \mu_n \psi_n(\rho) + \sigma_r(\rho) \frac{1}{2} \sum_{n=1}^N w_n \psi_n(\rho) = S(\rho) \quad (20)$$

Equation 17 meets this condition when weighted by  $w_n$  and summed of  $n$  if:

$$\sum_{n=1}^N [\alpha_{n+1/2}\psi_{n+1/2}(\rho) - \alpha_{n-1/2}\psi_{n-1/2}(\rho)] = \alpha_{1/2}\psi_{1/2}(\rho) - \alpha_{N+1/2}\psi_{N+1/2}(\rho) = 0 \quad (21)$$

Any quadrature set that is even in  $\mu$  will produce  $\alpha_{N+1/2} = 0$  if  $\alpha_{1/2} = 0$ , thus satisfying the condition. With these equations, we can create a set of coupled equations using diamond differencing in angle. Solving  $\psi_n(\rho) = \frac{1}{2} [\psi_{n+1/2}(\rho) + \psi_{n-1/2}(\rho)]$  for  $\psi_{n+1/2}$  and substituting into equation 17 gives the following:

$$\begin{aligned} \frac{\mu_n}{\rho^2} \frac{\partial}{\partial \rho} \rho^2 \psi_n(\rho) + \frac{2}{\rho w_n} [2\alpha_{n+1/2}\psi_n(\rho) - (\alpha_{n+1/2} + \alpha_{n-1/2})\psi_{n-1/2}(\rho)] + \\ \sigma(\rho)\psi_n(\rho) = q_n(\rho) \end{aligned} \quad (22)$$

The diamond difference relationship and Equation 22 together can be solved successively for  $\psi_1, \psi_{3/2}, \psi_2, \dots, \psi_N$ . To find  $\psi_{1/2}$ ,  $\mu_{1/2} = 1$  is defined as the radially inward direction.

## A.2 Spatial Discretization

As in Cartesian spatial differencing, finite volume method is used to discretize space. Equation 22 is first integrated over incremental volume  $V_i$  with spherical shell surface area  $A_i$  and a differencing relationship is applied. The radial mesh is defined in terms of  $\rho$  and cross sections are assumed constant in each region. The cell averaged flux and source are defined as follows:

$$\psi_{ni} = \frac{1}{V_i} 4\pi \int_{\rho_{i-1/2}}^{\rho_{i+1/2}} d\rho \rho^2 \psi_n(\rho) \quad (23)$$

$$Q_{ni} = \frac{1}{V_i} 4\pi \int_{\rho_{i-1/2}}^{\rho_{i+1/2}} d\rho \rho^2 q_n(\rho) \quad (24)$$

We also make the approximation

$$4\pi \int_{\rho_{i-1/2}}^{\rho_{i+1/2}} d\rho \rho \psi_n(\rho) = (A_{i+1/2} - A_{i-1/2}) \psi_{ni} \quad (25)$$

The spatial mesh is applied to Equation 22. Integrating over  $V_i$ , applying the above cell-average approximations, and dividing by  $V_i$  yields the equations required to sweep through

the spatial mesh. For  $\mu \leq 0$ .

$$\psi_{n,i\mp 1/2} = 2\psi_{ni} - \psi_{n,i\pm 1/2} \quad (26)$$

$$\psi_{ni} = \frac{|\mu_n| (A_{i+1/2} - A_{i-1/2}) \psi_{n,i\pm 1/2} + \frac{2}{w_n} (A_{i+1/2} - A_{i-1/2}) (\alpha_{n+1/2} - \alpha_{n-1/2}) \psi_{n-1/2,i} + V_i Q_{ni}}{2|\mu_n| A_{i\mp 1/2} + \frac{4}{w_n} (A_{i+1/2} - A_{i-1/2}) \alpha_{n+1/2} + V_i \sigma_i} \quad (27)$$

The starting direction is approximated by the following, which is solved for  $\psi_{1/2,i}$  and plugged into Equation 27.

$$-\frac{(\psi_{1/2,i+1/2} - \psi_{1/2,i-1/2})}{\rho_{i+1/2} - \rho_{i-1/2}} + \sigma_i \psi_{1/2,i} = Q_{ni} \quad (28)$$

The symmetry at the center of the sphere (in this case, the origin) is most often approximated by  $\psi_{N+1-n,1/2} = \psi_{n,1/2}$  for  $n = 1, 2, \dots, N/2$ . However, due to truncation error, a small nonphysical dip in spatial flux can occur at the origin. To improve accuracy numerically, the condition that is actually implemented is  $\psi_{n-1/2,1/2} = \psi_{n+1/2,1/2}$  for  $n = 1, 2, \dots, N$ . The starting direction calculation determines  $\psi_{1/2,1/2}$  at the origin.