CALCULATION OF NEUTRON LINE-BEAM RESPONSE FUNCTIONS WITH TWODANT

by

Thomas A. Gianakon

B.S., Kansas State University, 1987

A MASTER'S THESIS

Submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Nuclear Engineering
Kansas State University
Manhattan, Kansas

1989

Approved by:

Major Professor

LD
2668
.74
NC
1989
G53
C.2

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

In the design of radiation facilities (e.g., spent–fuel storage buildings), an important consideration is the dose rate at distances far from the radiation source. The far–field dose rate from these facilities can become significant if a sizable portion of the source radiation escapes through the roof, scatters in the air, and then returns to the earth. This phenomenon, commonly referred to as *skyshine*, is important for both gamma and neutron radiations.

The dose rate for a skyshine problem can be computed by first performing a rigorous transport calculation for the energy–dependent scalar flux density at the point of interest. The dose rate at that point is then obtained by weighting the energy–dependent scalar flux density with an appropriate response function and integrating over all energies. However, for design purposes the feasibility of repeatedly solving the transport equation is limited because the transport–based methods are calculationally expensive. Consequently, there has been considerable interest in simplified, albeit approximate, methods for the skyshine problem.

One very successful approximate method for gamma skyshine calculations is based on line–beam response functions. These response functions give the dose rate at any given source–to–detector distance for a point source emitting monoenergetic photons at a single fixed angle relative to the source–to–detector axis. Such line–beam beam response functions are used in the *SKYSHINE* and the *MicroSkyshine* codes and were obtained by fitting a simple 3–parameter formula to calculated dose rates caused by a point source emitting particles of a specified energy in one direction. [Pr76, Gr87] An entire set of line–beam response functions can be obtained by considering particles with different energies and source directions. These line–beam response functions can then be used to compute the

skyshine dose rate at a particular location from a radiation source emitting particles with an arbitrary energy and angular dependence by integrating the line–beam response functions over all source particle energies and emission directions.

Because of the success of the *SKYSHINE* and *MicroSkyshine* codes in calculating accurate dose rates for gamma photons in air, the line–beam response function technique has been extended to neutrons. [La79] The neutron skyshine problem (and consequently, the calculation of the associated line–beam response functions) differs significantly from the gamma skyshine problem. First, because of varying levels of humidity, significant changes in the neutron scattering properties of air occur, which, in turn, affect the skyshine dose rates. Second, the scattering and absorption properties of the ground can no longer be ignored (as is typically done in gamma skyshine calculations), because the energy–dependent scalar flux densities and the related dose rates are usually depressed near the air–ground interface. Finally, the energy–dependent cross sections for neutron transport calculations are not generally tabulated as continuous functions of energy (gamma photons commonly use the continuous Klein–Nishina formalism), but rather, are given in a discrete multigroup format based on low–order Legendre expansions. Even with these important caveats, a set of line–beam response functions is in use for calculating skyshine dose rates for neutrons in an infinite dry air medium. [La79] These line–beam response functions, which are based on very old cross–section data, are limited because they give the dose rates in terms of an *air dose* rather than *dose equivalent.* Thus, there is the need both to verify and to revise, if required, these old line–beam response functions.

The methods, which have been developed in the past to evaluate the skyshine line–beam response functions, have included Monte Carlo codes for both gamma photons and neutrons, and also, the orders–of–scattering technique combined with buildup factors for gamma photons. [Pr76, Sh87] The Monte Carlo method computes the dose rate by tracking particles as they leave the source, randomly scatter through the medium and are finally absorbed. While being easily implemented, the method has the disadvantage that a large number of particle histories are required to produce statistically significant results at distances far from the source. The orders–of–scattering and buildup factor method avoids the statistical problems of the Monte Carlo method by separating the flux density of particles into an uncollided and a scattered component. The uncollided component is then treated analytically and is used to generate a source of particles which lie along a line corresponding to the beam direction. The dose rate at a particular location is computed by integrating a set of buildup factors over the length of the uncollided line source (i.e., the buildup factor computes the dose from the scattered component of the flux density). This technique has proven to be an excellent approximation for gamma photons, but cannot be used for neutrons because neutron buildup factors for the skyshine geometry are not available. [Sh87]

In this study, the use of standard discrete ordinates S_n codes to evaluate neutron line–beam response functions is investigated. Standard S_n methods, however, are not well suited for treating highly singular sources such as the monodirectional point sources of the line–beam problems. In particular, the S_n method suffers nonphysical oscillations in the spatial distribution of the solution (i.e., ray effects). Recent work by R.E. Alcouffe at Los Alamos National Laboratory has investigated the use of the widely used S_n code TWODANT for

calculating the flux density from an isotropic point source in an infinite air medium. [Al89] Alcouffe found that ray–effects could be significantly reduced by utilizing the orders–of–scattering technique to compute a first collision source (i.e., treating the uncollided neutron flux density analytically and then calculating the scattered flux density numerically with TWODANT). This technique provided the motivation for the present study which investigates the ability of the orders–of–scattering technique, combined with standard S_n codes, to obtain neutron line–beam response functions for use in skyshine calculations.

This report assesses, the effectiveness of the popular production code TWODANT to generate the needed line–beam response functions for the neutron skyshine problem. [Lo84] One important characteristic of TWODANT is that source moments are required rather than angular source distributions along discrete directions. Thus, in Chapter 2 where the line–beam response function problem is formulated, techniques are also developed for describing the source moments from the orders–of–scattering technique as applied to the monoenergetic, monodirectional point source. In Chapter 3, several techniques are considered for removing negative scalar sources generated from using the Legendre expansion of scattering cross sections in the orders–of–scattering prescription. In Chapter 4, a comparison of the flux densities computed with TWODANT using the techniques of Chapters 1 and 2 is presented for several isotropically and anisotropically scattering media. Finally, in Chapter 5 the limitations of the S_n method for the calculation of skyshine line–beam response functions are discussed.

2. FIRST AND SECOND COLLISION SOURCES

The general line–beam response function for skyshine calculations is formulated by placing a monoenergetic, monodirectional point source at some distance above an air–ground interface (see Fig. 2.1) A three–dimensional transport calculation is then required to compute the detector response or dose at each location of interest. Under certain simplifying assumptions, a two–dimensional transport formulation can be used to solve this problem rather than the three–dimensional formalism. This reduction in dimensionality can significantly decrease computational effort and can allow improved accuracy. The problem can rigorously be reduced to two spatial dimensions by assuming that the air and the ground have the same scattering properties (i.e., the cross sections and atom densities are the same for both media, so that particle transport is in an infinite isotropic medium). The problem then assumes cylindrical symmetry about the source emission direction i.e., the point source is taken as the origin and the direction of particle emission as the z–axis (see Fig. 2.2). Any of the numerous solution methods for the transport equation can now be used to solve this problem. However, only a technique which combines the orders–of–scattering and the discrete ordinates methods is considered in this study.

The conventional discrete ordinates approach to this problem would be to approximate the monoenergetic, monodirectional point source as a single volumetric source in one cell of the spatial mesh used by a multigroup transport code. However when this type of source is used in TWODANT, it produces not only physically unrealistic negative flux densities, but it also fails to produce converged results for the iterated flux densities. Such problems can be attributed to the difficulty in approximating doubly singular sources with finite numbers of

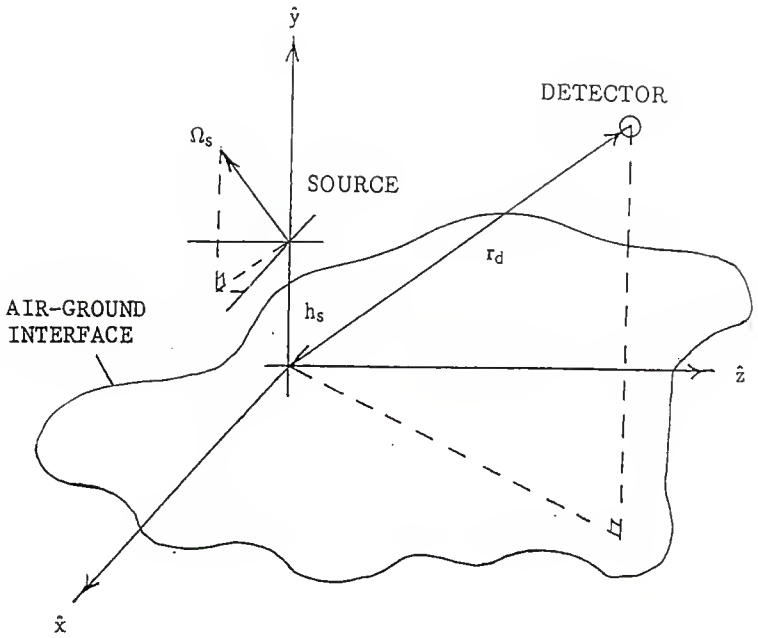


Fig. 2.1. Formulation of the three–dimensional ground–interface line–beam response function problem. The point source is located a distance h_s above the air ground interface and emits particles in direction Ω_s. A detector is located at position r_d.

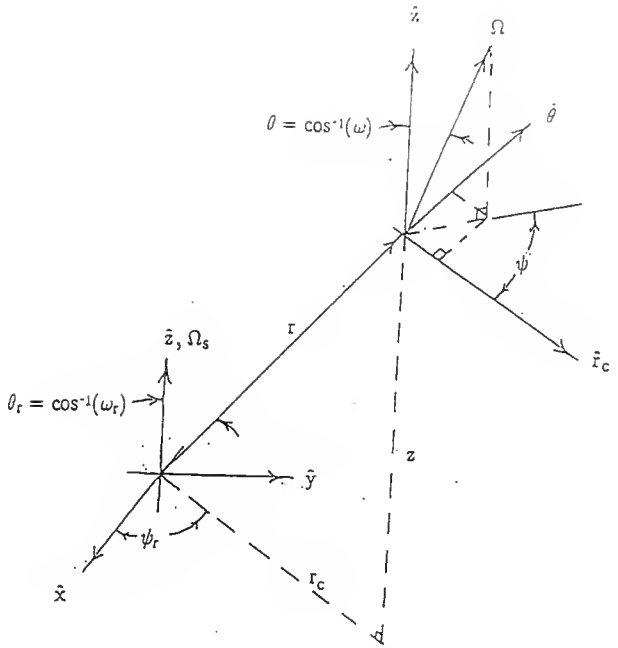


Fig. 2.2. Formulation of the two–dimensional line–beam response function problem. The point source is located at the origin and emits particles directly upward.

meshes and directions. The problem is further compounded because most of the standard discrete ordinates transport codes use spherical harmonic expansions to input the initial angular source distribution. Such expansions, particularly for angular sources which are singular, tend to generate negative angular source distributions which cause both convergence problems and negative flux densities. These problems make such a discrete ordinates approach useless for calculating line–beam response functions.

An alternate technique is to separate the total flux density into components which have scattered only a prescribed number of times, i.e., uncollided, once scattered, twice scattered, etc. The flux density for each component can then be computed separately by numerical integration over the problem geometry and all directions of particle travel. The total flux density at each point of interest is then found by adding the individual components together. Each flux density component is linked by a source based on the neutrons which have scattered from the preceding flux density component. Thus each component for the monoenergetic, monodirectional point source will be spread farther over the problem geometry then its predecessor. One consequence of this spreading is that only the uncollided flux density, which is based on the monoenergetic, monodirectional point source, and the once scattered flux density, which is based on a line source, can be computed with a minimum of computational effort. The higher order components, which are based on sources which are distributed over the entire problem geometry, require numerical evaluation of triple integrals over the problem geometry and double integrals over all directions of particle travel. Consequently, the orders–of–scattering method has only limited application to the calculation of line beam response functions.

The two techniques which are developed in this report attempt to combine both the orders–of–scattering method, which can spread the doubly singular source out over the problem geometry, with the discrete ordinates method, which works best with distributed sources. The two techniques are based on the uncollided flux density and the once scattered flux density. Both these techniques have the drawback that a preprocessor code must first be run to generate the input data for the discrete ordinates code and then a postprocessor code must be run to compute the total flux density by combining the flux density from the discrete ordinates technique with the flux densities from the orders–of–scattering technique.

The first method developed, which is based on the uncollided flux density, is referred to as the *first collision source* and is computed by first calculating the uncollided angular flux density from the monoenergetic, monodirectional point source. The uncollided angular flux density is then used to compute the first collision source (i.e., those particles which are in the process of making their first scatter since leaving the source). The angular moments of this first collision source are then used by the discrete ordinates code TWODANT to compute the flux density of particles which have scattered at least once. The total flux density is then found by adding the flux density of particles which have scattered at least once to the uncollided flux density.

The second method is an extension of this technique, which is based on the once scattered flux density. As before, the initial source distribution is used to compute the uncollided flux density, which is then used to compute the first collision source distribution. The first collision source distribution is then used to compute the once scattered flux density (i.e., those particles which have scattered only once since leaving the source). This once scattered flux density is then used

to compute a *second collision source* distribution. The angular moments of the second collision source are then used by TWODANT to calculate the flux density of particles which have scattered at least twice. Finally, the total flux density is computed by adding the flux density of particles which have scattered at least twice, with the once scattered and the uncollided flux density.

Both the first collision source and the second collision source techniques are developed in this chapter. The techniques are then implemented in the codes SRCLIN2D.FOR of Appendix A and SRC2D.FOR of Appendix B. Both codes output the source moments required by TWODANT. Also, the code SRC2D.FOR outputs the once scattered flux density. Neither code outputs the uncollided flux density since the direct component is not generally included in line–beam response functions.

2.1 Characterization of the Line–Beam Source

Once the multigroup approximation to the transport equation has been made, the normalized source term for the line–beam response functions used in skyshine calculations is given by

$$\tilde{Q}^0(r,\Omega) = \begin{cases} \delta_2(\Omega\cdot\Omega_s)\,\frac{\delta(r)}{4\pi\, r^2} \;\; ; \;\; \text{if } g = g_o \\ \\ 0 \qquad ; \;\; \text{otherwise .} \end{cases} \tag{1}$$

where r is a location vector in spherical coordinates (i.e., r, ω_r, and ψ_r) , Ω is a direction vector (i.e., ω and ψ), Ω_s is the direction source particles are being emitted, and g_o is the source energy group. For simplicity in this study, Ω_s is taken as the z–axis $\hat{z}$ so that $\Omega\cdot\Omega_s = \cos\theta = \omega$.

The δ_2 function used in Eq. (1) is related to the more conventional δ function by

$$\delta_2(\Omega\cdot\Omega_s) \;=\; \frac{\delta(1 \,-\, \Omega\cdot\Omega_s)}{2\pi} \;, \tag{2}$$

and has the following integral property

$$\iint\limits_{4\pi} d\Omega'\; f(\Omega',\Omega)\; \delta_2(\Omega'\cdot\Omega_s) \;=\; f(\Omega_s,\Omega) \;, \tag{3}$$

where $f(\Omega',\Omega)$ is an arbitrary function. [Du79]

2.2 Orders–of–Scattering/Discrete Ordinates Technique

The general multigroup transport equation which must be solved for the line beam response function problem is given by

$$\Omega\cdot\nabla\,\phi_g(r,\Omega) + \sigma_{tg}\phi_g(r,\Omega) \;=\; \tilde{Q}_g^0(r,\Omega) + \sum_{g'=1}^{G}\iint\limits_{4\pi} d\Omega'\sigma_{g'g}(\Omega\cdot\Omega')\phi_{g'}(r,\Omega') \;, \tag{4}$$

where $\phi_g(r,\Omega)$ is the angular neutron flux density for energy group g at location r and direction Ω, σ_{tg} is the total cross section for group g, $\sigma_{g'g}(\Omega\cdot\Omega')$ is the scattering transfer cross section from group g' to group g for scattering from direction Ω' to direction Ω, and $\tilde{Q}_g^0(r,\Omega)$ is the initial source distribution as defined by Eq. (1).

In the combined orders–of–scattering and discrete ordinates technique the total angular flux density is defined as

$$\phi_g(r,\Omega) = \phi_g^i(r,\Omega) + \sum_{j=0}^{i-1} \tilde{\phi}_g^j(r,\Omega) \ , \tag{5}$$

where $\phi_g^i(r,\Omega)$ is the angular flux density of neutrons which have had at least i–scatters (i.e., the discrete ordinates result) and $\tilde{\phi}_g^j(r,\Omega)$ is the angular flux density of neutrons which have had exactly j–scatters (i.e., the orders–of–scattering results). With this definition, the transport equation can be written in the form

$$\Omega\cdot\nabla\,\phi_g^i(r,\Omega) + \sigma_{tg}\phi_g^i(r,\Omega) = \tilde{Q}_g^i(r,\Omega) + \sum_{g'=1}^{G} \iint_{4\pi} d\Omega' \sigma_{g'g}(\Omega\cdot\Omega')\phi_{g'}^i(r,\Omega') \ , \tag{6}$$

where

$$\tilde{Q}_g^i(r,\Omega) = \sum_{g'=1}^{G} \iint_{4\pi} d\Omega' \ \sigma_{g'g}(\Omega\cdot\Omega') \ \tilde{\phi}_{g'}^{i-1}(r,\Omega') \ . \tag{7}$$

To compute $\tilde{\phi}_g^j(r,\Omega)$ for use in Eq. (7), the orders–of–scattering technique is applied. The angular flux density for neutrons which have undergone exactly j–scatters is given by [Du79]

$$\tilde{\phi}_g^j(r,\Omega) = \iiint_{\substack{all\\space}} d^3r_o \iint_{4\pi} d\Omega_o \, \delta_2(\Omega_o\cdot\Omega) \, \delta_2(\tilde{\Omega}(r,r_o)\cdot\Omega) \, \frac{e^{-\sigma_{tg}|r-r_o|}}{|r-r_o|^2} \, \tilde{Q}_g^j(r_o,\Omega_o) \, , \quad (8)$$

where $|r - r_o|$ represents the magnitude of the difference between the two vectors r and r_o, σ_{tg} is the total macroscopic cross section for group g, and

$$\tilde{\Omega}(r,r_o) = \frac{r - r_o}{|r - r_o|} \, . \quad (9)$$

2.3 Uncollided Angular Flux Density

The uncollided angular flux density $\tilde{\phi}_g^0(r,\Omega)$ from the source distribution of Eq. (1) can now be computed with Eq. (8). The two δ functions in the source facilitates the analytical evaluation of both the angular and the spatial integrals. The result is

$$\tilde{\phi}_g(r,\Omega) = \begin{cases} \delta_2\left(\frac{r}{|r|}\cdot\Omega_s\right) \, \delta_2(\Omega_s\cdot\Omega) \, \frac{e^{-\sigma_{tg}r}}{r^2} & ; \text{ if } g = g_o \\ 0 & ; \text{ otherwise.} \end{cases} \quad (10)$$

2.4 First Collision Source

The first collision source can now be computed by substituting Eq. (10) into Eq. (7) and performing the integration over Ω' analytically. The first collision source is

$$\tilde{Q}_g(r,\Omega) = \sigma_{g_o g}(\Omega\cdot\Omega_s)\,\delta_2\left(\frac{r}{|r|}\cdot\Omega_s\right)\frac{e^{-\sigma_{to}r}}{r^2}, \qquad (11)$$

where σ_{to} is the total macroscopic cross–section for group g_o and $\sigma_{g_o g}(\Omega\cdot\Omega_s)$ is the scattering cross–section from the source group g_o to the group g from direction Ω_s to direction Ω.

2.5 Moments of the First Collision Source

The TWODANT code package (and most other standard production discrete ordinates codes) require that the source distribution be specified by its spherical harmonics expansion (as opposed to the source along each discrete direction). To facilitate this expansion, the spherical harmonic expansion of an arbitrary function $Q(\Omega)$ (where $Q(\Omega) = Q(\omega,\psi)$) is considered. First, the function can be expressed as a sum of Legendre polynomials and associated Legendre functions, namely

$$Q(\Omega) = \sum_{l=0}^{\infty}(2l+1)\,P_l(\omega)\,Q_l + \sum_{l=0}^{\infty}\sum_{m=1}^{l}(2l+1)\,P_{lm}(\omega)\cos(m\psi)\sqrt{2\frac{(l-m)!}{(l+m)!}}\,Q^c_{lm} +$$

$$\sum_{l=0}^{\infty}\sum_{m=1}^{l}(2l+1)\,P_{lm}(\omega)\sin(m\psi)\sqrt{2\frac{(l-m)!}{(l+m)!}}\,Q^s_{lm}\ , \qquad (12)$$

where Q_l, Q^c_{lm}, and Q^s_{lm} are expansion coefficients. These coefficients can be expressed in terms of $Q(\Omega)$ with the use of the orthogonality property of the Legendre functions. The expansion coefficients are thus given by

$$Q_l = \frac{1}{4\pi} \iint_{4\pi} d\Omega \, P_l(\omega) \, Q(\Omega), \tag{13}$$

$$Q^c_{lm} = \frac{1}{4\pi} \iint_{4\pi} d\Omega \sqrt{2\frac{(l-m)!}{(l+m)!}} \, P_{lm}(\omega) \cos(m\psi) \, Q(\Omega), \tag{14}$$

and by

$$Q^s_{lm} = \frac{1}{4\pi} \iint_{4\pi} d\Omega \sqrt{2\frac{(l-m)!}{(l+m)!}} \, P_{lm}(\omega) \sin(m\psi) \, Q(\Omega). \tag{15}$$

Rather than attempting to find the above source moments by performing the appropriate integration of Eqs. (13)–(15), the moments can be obtained by expanding the scattering cross–section $\sigma_{g_o g}(\Omega \cdot \Omega_s)$ into its moments and by then comparing the result with Eq. (12). The Legendre expansion of the scattering cross–section is

$$\sigma_{g_o g}(\Omega \cdot \Omega_s) = \sum_{l=0}^{\infty} \frac{(2l+1)}{4\pi} \sigma^l_{g_o g} \, P_l(\Omega \cdot \Omega_s) \ , \tag{16}$$

where $\sigma^l_{g_o g}$ is the l–th Legendre moment of the scattering cross section for group g_o

to group g, and $P_l(\Omega\cdot\Omega_s)$ is the l-th Legendre polynomial. The Legendre polynomial is now expanded with the addition rule for Legendre polynomials to give [Ab72]

$$\sigma_{g_{og}}(\Omega\cdot\Omega_s) = \sum_{l=0}^{\infty}\frac{(2l+1)}{4\pi}\sigma^{l}_{g_{og}}\left\{P_l(\omega)\,P_l(\omega_s) + \right.$$

$$\left.\sum_{m=1}^{l} 2\frac{(l-m)!}{(l+m)!}P_{lm}(\omega)\,P_{lm}(\omega_s)\,[\cos(m\psi)\cos(m\psi_s)+\sin(m\psi)\sin(m\psi_s)]\right\}. \quad (17)$$

Substitution of Eq. (17) into Eq. (11) with i = 1 yields

$$\tilde{Q}^{1}_{g}(r,\Omega) = \sum_{l=0}^{\infty}(2l+1)\,P_l(\omega)\,\tilde{Q}^{1}_{gl}(r) +$$

$$\sum_{l=0}^{\infty}\sum_{m=1}^{l}(2l+1)\,P_{lm}(\omega)\cos(m\psi)\sqrt{2\frac{(l-m)!}{(l+m)!}}\,\tilde{Q}^{c1}_{glm}(r) +$$

$$\sum_{l=0}^{\infty}\sum_{m=1}^{l}(2l+1)\,P_{lm}(\omega)\sin(m\psi)\sqrt{2\frac{(l-m)!}{(l+m)!}}\,\tilde{Q}^{s1}_{glm}(r)\ , \quad (18)$$

where the moments of the source are given by

$$\tilde{Q}^{1}_{g l}(\mathbf{r}) = \frac{\sigma^{l}_{g_0 g}}{4\pi} \delta_2\left(\frac{\mathbf{r}}{|\mathbf{r}|}\cdot\underline{\Omega}_s\right) \frac{e^{-\sigma_{to}\mathbf{r}}}{\mathbf{r}^2} P_l(\omega_s), \tag{19}$$

$$\tilde{Q}^{c1}_{glm}(\mathbf{r}) = \frac{\sigma^{l}_{g_0 g}}{4\pi} \delta_2\left(\frac{\mathbf{r}}{|\mathbf{r}|}\cdot\underline{\Omega}_s\right) \frac{e^{-\sigma_{to}\mathbf{r}}}{\mathbf{r}^2} \sqrt{2\frac{(l-m)!}{(l+m)!}}\, P_{lm}(\omega_s)\cos(m\psi_s), \tag{20}$$

and by

$$\tilde{Q}^{s1}_{glm}(\mathbf{r}) = \frac{\sigma^{l}_{g_0 g}}{4\pi} \delta_2\left(\frac{\mathbf{r}}{|\mathbf{r}|}\cdot\underline{\Omega}_s\right) \frac{e^{-\sigma_{to}\mathbf{r}}}{\mathbf{r}^2} \sqrt{2\frac{(l-m)!}{(l+m)!}}\, P_{lm}(\omega_s)\sin(m\psi_s) \ . \tag{21}$$

By comparison of Eq. (18) to Eq. (12), it is seen that Eqs. (19)–(21) are just the moments required by TWODANT. Actually only the moments of Eq. (19) and Eq. (20) are required since TWODANT assumes that $\psi_s = 0$ for cylindrically symmetric geometries so that $\tilde{Q}^{s1}_{glm} = 0$. Additional simplification in the above moments results, for the line–beam geometry, since $\omega_s = 1$ (i.e., the particles are emitted only upward). With this value for ω_s and since $P_l(1) = 1$ and $P_{lm}(1) = 0$ for $m > 0$, the above moments reduce to

$$\tilde{Q}^{1}_{g l}(\mathbf{r}) = \frac{\sigma^{l}_{g_0 g}}{4\pi} \delta_2\left(\frac{\mathbf{r}}{|\mathbf{r}|}\cdot\Omega_s\right) \frac{e^{-\sigma_{to}\mathbf{r}}}{\mathbf{r}^2}, \tag{22}$$

and

$$\tilde{Q}^{c1}_{glm}(\mathbf{r}) = \tilde{Q}^{s1}_{glm}(\mathbf{r}) = 0. \tag{23}$$

2.6 Evaluation of the First Collision Source for TWODANT

Equation (22) describes a line source along the z–axis. However, the TWODANT code package does not explicitly treat line sources, and thus it is necessary to approximate the line source as a volumetric source about the positive z–axis (i.e., in the first set of r spatial meshes). To facilitate this approximation, Eq. (22) is first transformed to cylindrical coordinates. To perform this transformation, the unit vector $\mathbf{r}/|\mathbf{r}|$ is specified by the two spherical variables ω_r and ψ_r (see Fig. 2.2). Thus the dot product in the δ_2 function of Eq. (22) becomes

$$\frac{\mathbf{r}}{|\mathbf{r}|}\cdot\Omega_s = \omega_r\omega_s + \sqrt{1-\omega_r^2}\ \sqrt{1-\omega_s^2}\ \cos(\psi_r-\psi_s)\ . \tag{24}$$

Since $\omega_s = 1$, the dot product reduces to

$$\frac{\mathbf{r}}{|\mathbf{r}|}\cdot\Omega_s = \omega_r\ . \tag{25}$$

and Eq. (22) becomes

$$\tilde{Q}^{1}_{g,l}(\mathbf{r}) = \frac{\sigma^{l}_{g_0 g}}{4\pi}\,\delta_2(\omega_r)\,\frac{e^{-\sigma_{to}r}}{r^2}\ . \tag{26}$$

To complete the transformation of Eq. (26) to cylindrical coordinates, the following substitutions are made (see Fig. 2.2)

$$\omega_r = \frac{z}{(r_c^2 + z^2)^{0.5}} , \tag{27}$$

and

$$r^2 = r_c^2 + z^2, \tag{28}$$

where the c subscript refers to cylindrical coordinates. Additionally, since a δ function is being transformed to a new coordinate system, Eq. (26) must also be divided by the appropriate Jacobian. The Jacobian is

$$|J| = \begin{vmatrix} \frac{\delta r}{\delta r_c} & \frac{\delta \omega_r}{\delta r_c} \\ \frac{\delta r}{\delta z} & \frac{\delta \omega_r}{\delta z} \end{vmatrix} = \frac{r_c}{r^2} . \tag{29}$$

Thus the first collision source of Eq. (26) transforms in cylindrical geometry to

$$\tilde{Q}_{gl}^{1}(r) = \frac{\sigma_{g_o g}^{l}}{4\pi} \delta(1-\omega_r) \frac{e^{-\sigma_{to}(r_c^2 + z^2)^{0.5}}}{2\pi\ r_c} , \tag{30}$$

where the values of r_c and z are determined from Eq. (27) when $\omega_r = 1$. Clearly from Eqs. (27) and (28), when $\omega_r = 1$, r_c must be zero and z can be any positive value. Thus the delta function in the above expression can be replaced by $\delta(r_c)$ plus the condition $z > 0$.

The first collision source is then

$$\tilde{Q}^1_{g\ell}(r) = \begin{cases} \dfrac{\sigma^{\ell}_{g_0 g}}{4\pi}\,\delta(r_c)\,\dfrac{e^{-\sigma_{to}(r_c^2 + z^2)^{0.5}}}{2\pi\, r_c} & ; \text{if } z \geq 0 \\ 0 & ; \text{otherwise} \end{cases} \tag{31}$$

Eq. (31) can now be used to compute the volumetric source along a small cylinder about the z–axis that approximates the line source. Each cylinder cell has a radius of r_1 corresponding to the first radial mesh thickness specified by TWODANT. The top of the small cylinder is at z_{i+1} and the bottom at z_i such that the values correspond to the cell boundaries used in TWODANT. The volumetric average value is thus

$$\bar{Q}^1_{g\ell}(r) = \frac{\sigma^{\ell}_{g_0 g}}{4\pi}\,\frac{1}{\sigma_{to}}\,\frac{e^{-\sigma_{to} z_i}(1 - e^{-\sigma_{to}(z_{i+1} - z_i)})}{\pi\, r_1^2\,(z_{i+1} - z_i)}\,. \tag{32}$$

Eq. (32) is used to compute the first collision source moments in those cells directly above the initial monoenergetic, monodirectional point source. All other cells are given a source value of zero. This first collision source is implemented in the code SRCLIN2D.FOR of Appendix A. The code computes a set of spherical harmonic expansion coefficients for the first collision source, which after minor modification is suitable for use by TWODANT. (The modification entails deleting all exponential letters i.e., changing 1.2345E–08 to either the form 1.2345–08 or the form 12345–12).

2.7 Once Scattered Angular Flux Density

To compute the once scattered angular flux density, the first collision source of Eq. (11) is substituted into Eq. (8) for $\tilde{Q}_g^j(r_o,\Omega_o)$. The result, after integration over all directions Ω_o and with the spatial integral expressed in spherical coordinates, is

$$\tilde{\phi}_g^1(\mathbf{r},\Omega) = \int_0^\infty dr_o\, r_o^2 \int_0^{2\pi} d\psi_o \int_{-1}^{1} d\omega_o \frac{\delta_2(\tilde{\Omega}(\mathbf{r},\mathbf{r}_o)\cdot\Omega)}{|\mathbf{r}-\mathbf{r}_o|^2}\, \sigma_{g_og}(\Omega\cdot\Omega_s)\, \delta_2\left(\frac{\mathbf{r}}{|\mathbf{r}|}\cdot\Omega_s\right) \times$$

$$\left[e^{-\sigma_{tg}|\mathbf{r}-\mathbf{r}_o|}\, \frac{e^{-\sigma_{to}r_o}}{r_o^2}\right], \tag{33}$$

where $\tilde{\Omega}(\mathbf{r},\mathbf{r}_o)$ is defined by Eq. (9) and in Cartesian coordinates (but using spherical variables) the vectors r and r_o are

$$\mathbf{r} = r\sqrt{1-\omega_r^2}\,\cos(\psi_r)\,\hat{x} + r\sqrt{1-\omega_r^2}\,\sin(\psi_r)\,\hat{y} + r\omega_r\,\hat{z}\ , \tag{34}$$

and

$$\mathbf{r}_o = r_o\sqrt{1-\omega_o^2}\,\cos(\psi_o)\,\hat{x} + r_o\sqrt{1-\omega_o^2}\,\sin(\psi_o)\,\hat{y} + r_o\omega_o\,\hat{z}\ . \tag{35}$$

The result of integration of Eq. (33) over the variables ω_o and ψ_o with $\Omega_s = \hat{z}$ is

$$\tilde{\phi}_g^1(\mathbf{r},\Omega) = \int_0^\infty dr_o\, \delta_2(\tilde{\Omega}(\mathbf{r},\mathbf{r}_o)\cdot\Omega)\, \frac{e^{-\sigma_{to}r_o-\sigma_{tg}(\,r^2+r_o^2-\,2r\omega_r r_o)^{0.5}}}{r^2+r_o^2-2r\omega_r r_o}\, \sigma_{g_og}(\Omega\cdot\Omega_s), \tag{36}$$

where

$$\tilde{\Omega}(r,r_o) = \frac{r - r_o\hat{z}}{|r - r_o\hat{z}|}, \tag{37}$$

and r is specified by Eq. (34).

The once scattered scalar flux density at any point can now be computed by integrating Eq. (36) over Ω. The result is

$$\tilde{\phi}_g^1(r) = \int_0^\infty dr_o\, \sigma_{g_o g}(\tilde{\Omega}(r,r_o)\cdot\Omega_s) \frac{e^{-\sigma_{to}r_o}\, e^{-\sigma_{t\,g}(r^2 + r_o^2 - 2r\omega_r r_o)^{0.5}}}{r^2 + r_o^2 - 2r\omega_r r_o}, \tag{38}$$

where $\tilde{\Omega}(r,r_o)$ is defined by Eq. (37).

2.8 Second Collision Source

The once scattered angular flux density is now used to compute the second collision source. First, the once scattered angular flux density of Eq. (36) is substituted into Eq. (7) and then the integration over Ω' is completed analytically. The result is

$$\tilde{Q}_g^2(r,\Omega) = \sum_{g'=0}^{G} \int_0^\infty dr_o\, \sigma_{g'g}(\tilde{\Omega}(r,r_o)\cdot\Omega)\, \sigma_{g_o g'}(\tilde{\Omega}(r,r_o)\cdot\Omega_s) \times \left[\frac{e^{-\sigma_{to}r_o - \sigma_{t\,g}(r^2 + r_o^2 - 2r\omega_r r_o)^{0.5}}}{r^2 + r_o^2 - 2r\omega_r r_o}\right], \tag{39}$$

where $\tilde{\Omega}(r,r_o)$ is defined in Eq. (37).

Since the initial source is directed along the z–axis (i.e., $\Omega_s = \hat{z}$), Eq. (39) can be simplified to

$$\tilde{Q}_g^2(r,\Omega) = \sum_{g'=0}^{G} \int_0^{\infty} dr_o \, \sigma_{g'g}(\tilde{\Omega}(r,r_o)\cdot\Omega) \, \sigma_{g_o g'}(\omega_o) \times$$

$$\left[\frac{e^{-\sigma_{to} r_o - \sigma_t g (r^2 + r_o^2 - 2r\omega r_o)^{0.5}}}{r^2 + r_o^2 - 2r\omega_r r_o} \right] , \qquad (40)$$

where ω_o is defined as

$$\omega_o \equiv \tilde{\Omega}(r,r_o)\cdot\hat{z} = \frac{r\omega_r - r_o}{(r^2 + r_o^2 - 2r\omega_r r_o)^{0.5}} , \qquad (41)$$

and $\tilde{\Omega}(r,r_o)\cdot\Omega$ is

$$\tilde{\Omega}(r,r_o)\cdot\Omega = \omega_o\omega + \sqrt{1-\omega_o^2}\ \sqrt{1-\omega^2}\ \cos(\psi) , \qquad (42)$$

because the direction Ω in Cartesian coordinates and spherical variables is given by

$$\Omega = \sqrt{1-\omega^2}\ \cos(\psi_r + \psi)\,\hat{x} + \sqrt{1-\omega^2}\ \sin(\psi_r + \psi)\,\hat{y} + \omega\,\hat{z} . \qquad (43)$$

2.9 Moments of the Second Collision Source

To compute the moments of the second collision source, a technique similar to the first collision source is used (i.e., Eqs. (12)–(18)). First, the cross section is expanded into its Legendre moments and then the addition rule for Legendre polynomials is applied to this expansion. As before, the moments are deduced by comparing the result with Eq. (12). The moments so obtained are

$$\tilde{Q}^{c2}_{gl}(r) = \sum_{g'=0}^{G} \frac{\sigma^{l}_{g'g}}{4\pi} \int_{0}^{\infty} dr_o \, P_l(\omega_o) \, \sigma_{g_o g'}(\omega_o) \times$$

$$\left[\frac{e^{-\sigma_{to} r_o - \sigma_{t g'}(r^2 + r_o^2 - 2r\omega_r r_o)^{0.5}}}{r^2 + r_o^2 - 2r\omega_r r_o} \right], \qquad (44)$$

$$\tilde{Q}^{c2}_{glm}(r) = \sum_{g'=0}^{G} \frac{\sigma^{l}_{g'g}}{4\pi} \sqrt{2\frac{(l-m)!}{(l+m)!}} \int_{0}^{\infty} dr_o \, P_{lm}(\omega_o) \, \sigma_{g_o g'}(\omega_o) \times$$

$$\left[\frac{e^{-\sigma_{to} r_o - \sigma_{t g'}(r^2 + r_o^2 - 2r\omega_r r_o)^{0.5}}}{r^2 + r_o^2 - 2r\omega_r r_o} \right], \qquad (45)$$

and, again because of the azimuthal symmetry of the once scattered angular flux density,

$$\tilde{Q}^{s2}_{glm}(r) = 0, \qquad (46)$$

The cosine ω_o is defined by Eq. (41).

2.10 Evaluation of the Second Collision Source for TWODANT

The numerical integration of Eq. (44) and Eq. (45) is difficult to perform accurately, since the integration ranges from zero to infinity (or from the location of the initial point source to the top of the problem geometry used in the transport calculation). To reduce calculational difficulties the integration is transformed with a change of variable. First, Eqs. (44) and (45) are expressed in cylindrical coordinates by making the following substitutions

$$\omega_r = \frac{z}{(r_c^2 + z^2)^{0.5}} , \tag{47}$$

and

$$r^2 = r_c^2 + z^2 . \tag{48}$$

Thus, the non–zero source moments become

$$\tilde{Q}^2_{gl}(r) = \sum_{g'=0}^{G} \frac{\sigma^l_{g'g}}{4\pi} \int_0^\infty dr_o \, P_l(\omega_o) \, \sigma_{g_o g'}(\omega_o) \times \left[\frac{e^{-\sigma_{to} r_o - \sigma_{tg'} [r_c^2 + (r_o - z)^2]^{0.5}}}{r_c^2 + (r_o - z)^2} \right] , \tag{49}$$

and

$$\tilde{Q}^{c2}_{glm}(r) = \sum_{g'=0}^{G} \frac{\sigma^l_{g'g}}{4\pi} \sqrt{2\frac{(l-m)!}{(l+m)!}} \int_0^\infty dr_o \, P_{lm}(\omega_o) \, \sigma_{g_o g'}(\omega_o) \times \left[\frac{e^{-\sigma_{to} r_o - \sigma_{tg'} [r_c^2 + (r_o - z)^2]^{0.5}}}{r_c^2 + (r_o - z)^2} \right] . \tag{50}$$

Additionally, ω_o becomes

$$\omega_o = \frac{z - r_o}{[r_c^2 + (r_o - z)^2]^{0.5}} . \tag{51}$$

If the following substitution is now made

$$\tan(\varphi) = \frac{r_o - z}{r_c} , \tag{52}$$

then the source moments can be expressed in terms of integrals with a finite integration range, namely

$$\tilde{Q}^2_{gl}(\mathbf{r}) = \sum_{g'=0}^{G} \frac{\sigma^l_{g'g}}{4\pi} \frac{e^{-\sigma_{to}z}}{r_c} \times$$

$$\int_{\varphi_0}^{\varphi_1} d\varphi \, P_l(\omega_o) \, \sigma_{g_o g'}(\omega_o) \, e^{-\sigma_{to} r_c \tan(\varphi) - \sigma_{tg'} r_c \sec(\varphi)} , \tag{53}$$

and

$$\tilde{Q}^{c2}_{glm}(\mathbf{r}) = \sum_{g'=0}^{G} \frac{\sigma^l_{g'g}}{4\pi} \sqrt{2\frac{(l-m)!}{(l+m)!}} \frac{e^{-\sigma_{to}z}}{r_c} \times$$

$$\int_{\varphi_0}^{\varphi_1} d\varphi \, P_{lm}(\omega_o) \, \sigma_{g_o g'}(\omega_o) \, e^{-\sigma_{to} r_c \tan(\varphi) - \sigma_{tg'} r_c \sec(\varphi)} , \tag{54}$$

where

$$\omega_o = -\sin(\varphi). \tag{55}$$

The limits on the integration are

$$\varphi_0 = \tan^{-1}\left(\frac{-z}{r}\right) , \tag{56}$$

and

$$\varphi_1 = \tan^{-1}\left(\frac{z_0-z}{r}\right) , \tag{57}$$

where the distance z_0 is the distance from the source to the top of the problem geometry as defined by the transport code.

Gaussian quadrature can now be applied to evaluate accurately the above integrals. When evaluating the moments for TWODANT, the values of the moments at the center of each cell are treated as the average for that cell. Alternatively, the average could have been computed by a numerical integration of the source moments over the cell and subsequent division by the cell area. This alternate technique was not used because of the far greater computational effort required to compute such source moments for each cell. Moreover, for a sufficiently fine spatial grid, the source moments vary negligibly over an individual cell.

The second collision source technique has been implemented in the computer code SRC2D.FOR of Appendix B. This code generates source moments which are suitable with only minor modification for use by TWODANT. An additional computer code, SRC2DPT.FOR of Appendix C, has been developed to study the effect spherical harmonic expansions have on the angular source distribution for the second collision source technique. SRC2DPT.FOR will compute at a single location the angular source distribution with either a spherical harmonic expansion or by computing the source along actual directions using Eq. (40).

3. NEGATIVE SOURCE FIXUP TECHNIQUES

When calculating the second collision source, the possibility exists that in some spatial cells of the problem the scalar source (i.e., the zero–th moment of the source as defined by Eq. (53)) will be negative. Such negative values are physically impossible and arise as a direct result of the Legendre expansion of the scattering cross section (see Fig. 3.1). In addition to being physically unrealistic, severe numerical stability problems may develop when using TWODANT if a large number of cells contain negative sources. Several techniques have been developed to eliminate these negative scalar sources without the need to resort to higher order Legendre expansions which mitigate, but never eliminate negative cross section values. While these negative fixup techniques guarantee that the *scalar* sources are never negative, the possibility remains that in certain discrete directions the *angular* source may be negative. Numerical problems can again develop, if sources along many discrete directions are negative.

The first collision source does not exhibit these negative *scalar* sources which are characteristic of the second collision source. Thus no negative fixup techniques are required. However, the first collision source does suffer from negative *angular* sources which can lead to the same numerical problems found in the second collision source.

In this chapter several methods are discussed for eliminating some or all of the negative valued sources. These methods include (1) the use of an exact scattering cross section (thereby avoiding the spurious Legendre oscillations), (2) retention of the Legendre expanded cross section but all negative scalar sources and associated moments are set to zero, (3) use of the Legendre cross section expansion

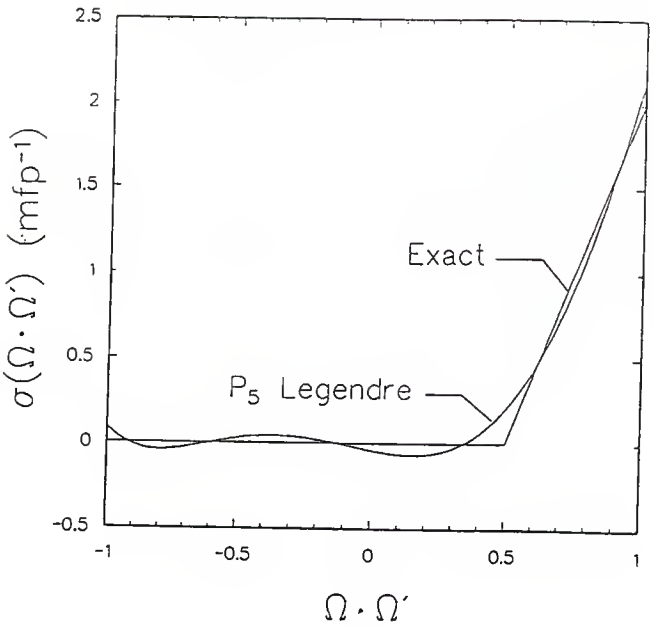


Fig. 3.1. Scattering transfer cross sections used to investigate negative flux fixup techniques. The scattering cross sections include the exact cross section and the Legendre expansion.

but all scalar sources and associated moments not under the "primary peak" are set to zero, and (4) use of a modified Legendre cross section expansion which yields non–negative cross sections.

To explore these techniques, the one–group scattering cross section of Fig 3.1 has been used to compute both scalar source distributions and angular source distributions for each of these four methods. This example scattering cross section is typical of scattering in air for source groups in the range of 2 to 3 MeV for a 47 group cross section set with a range of neutron energies from 10^{-6} eV to 17 MeV. Furthermore, this scattering cross section, which is peaked in the forward direction, in conjunction with the line–beam geometry, also parallels the worst case for transport calculations. In addition to the presentation of the scalar source distributions produced from these techniques, the resulting angular source distributions are derived to study the effect each method has on the angular source distribution under the "primary peak".

3.1 Exact Cross Section Technique

A procedure has been developed which utilizes an exact cross section for $\sigma_{g_0 g}(\Omega \cdot \Omega_s)$ to compute both the once scattered flux density and also the second collision source. The procedure utilizes both the exact scattering cross section to compute the once scattered angular flux density and also the Legendre moments of the exact scattering cross section to compute the coefficients of the spherical harmonic expansion of the second collision source.

These exact scattering cross sections are based upon kinematic relationships for energy and momentum conservation and upon minimum and maximum scattering angles permitted by the energy group structure. [Od76]

The exact cross sections generated using these techniques are generally fit to either a triangular or a trapezoidal shape based upon both the zero–th moment of the Legendre expansion and also the intercepts for minimum and maximum scattering angles for transfer between two energy groups. [Mi76] The scattering cross section generated with this method will always be non–negative for all scattering angles. Since the scattering cross section is non–negative, the scalar source terms will also necessarily be non–negative. However, since the moments of the exact scattering cross–section are still used to compute the coefficients of the spherical harmonic expansion of the second collision source, the possibility remains that the angular source distribution may contain some negative values.

One disadvantage of using these exact transfer cross sections is that they are difficult to use for mixtures and for nuclides which contain inelastic scattering components–both are important for high energy neutron transport in air. Also, this technique introduces a significant increase in calculational effort, because both the exact and the Legendre expansion for the scattering cross section are required to compute the second collision source.

The exact cross–section of Fig. 3.1 has been utilized to compute the exact scalar sources of Fig. 3.2 for a monoenergetic, monodirectional point source located at the center of a cylinder 10 mfp (mean free path) in height and with a radius of 5 mfp. One result is that the scalar source is positive in only a small region of the problem. In Fig 3.3 the associated angular distribution has been plotted for the point (r = 1.6667 mfp, z = 1.6667 mfp) using both the spherical harmonics expansion and also by actually computing the source along discrete directions using the exact scattering cross section. Likewise, the angular source distribution at (r = 1.6667 mfp, z = 3.3333 mfp) has also been plotted in Fig 3.4.

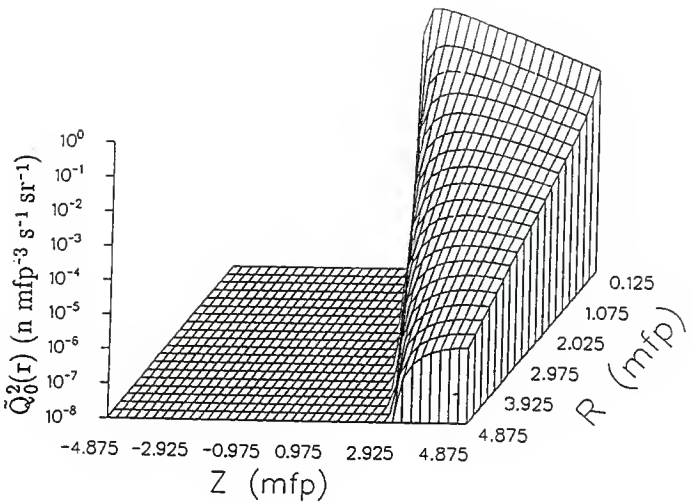
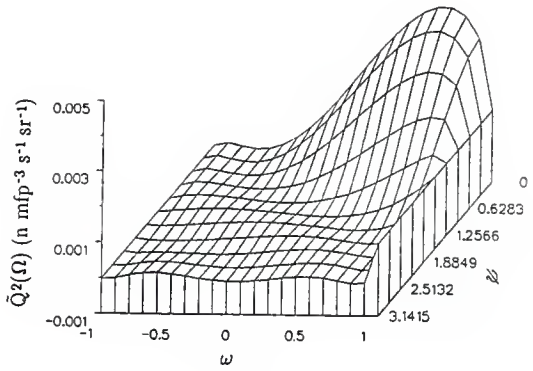
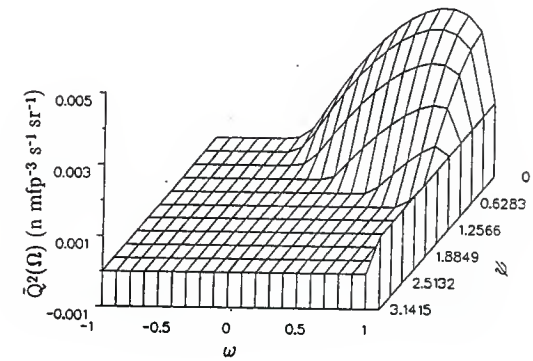


Fig. 3.2. Second collision scalar source distribution generated with the exact scattering transfer cross section for a c value of 0.5.

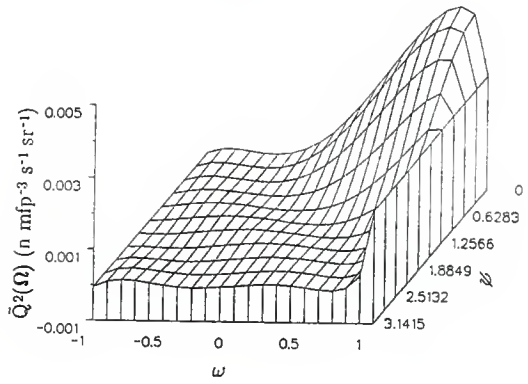


(a)

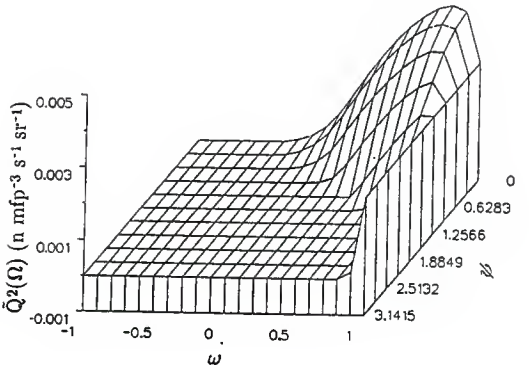


(b)

Fig. 3.3. Angular source distributions at (r = 1.667 mfp, z = 1.667 mfp) for the second collision source generated with the exact scattering transfer cross section for a c–value of 0.5. (a) Angular distribution generated by computing the spherical harmonic expansion of the second collision source. (b) Angular distribution generated by computing the second collision source along discrete directions.



(a)



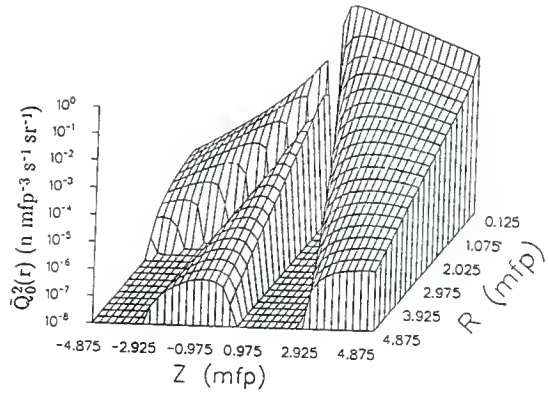
(b)

Fig. 3.4. Angular source distributions at (r = 1.667 mfp, z = 3.333 mfp) for the second collision source generated using the exact scattering transfer cross section for a c–value of 0.5. (a) Angular distribution generated by computing the spherical harmonic expansion of the second collision source. (b) Angular distribution generated by computing the second collision source along discrete directions.

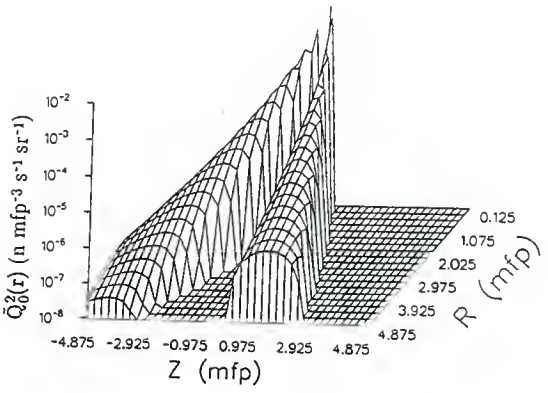
These figures demonstrate that the source intensity in some directions may be spuriously negative because of a spherical harmonic expansion of the second collision source.

3.2 Technique of Setting Negative Scalar Sources to Zero

A second technique for alleviating negative source problems in TWODANT involves using the Legendre expansion for the cross section $\sigma_{g_{o}g}(\Omega_s \cdot \Omega)$ and then generating the appropriate second collision source. In this method, all negative scalar sources and their associated higher moments are then set to zero. The result is a new source distribution which has a larger number of particles being generated in the cell volume then does the original distribution. To regain particle balance, the original scalar source and the new scalar source distributions are integrated over all cells. The ratio of these two integrations is then used to multiply each moment of the second collision source to regain particle conservation. The result is a new source distribution which has the same number of particles being emitted as the uncorrected second collision source, but has no negative scalar sources. Figure 3.5 illustrates the scalar–source spatial distribution obtained by using the Legendre expansion of the cross section and no fixup corrections. Figure 3.6 illustrates the distribution with the negative portion eliminated (i.e., similar to Fig. 3.5a). The disadvantage of this technique is that it produces positive spurious sources at locations where no actual source should exist. However, the method is still better than using an uncorrected Legendre expansion, because when the uncorrected second collision source is used by TWODANT it rapidly diverges to produce essentially infinite flux densities in all cells. When the second collision



(a)



(b)

Fig. 3.5. Second collision scalar source distribution generated with the Legendre expansion of the scattering transfer cross section for a c–value of 0.5. (a) The positive valued portion of the scalar source. (b) The negative valued portion of the scalar source.

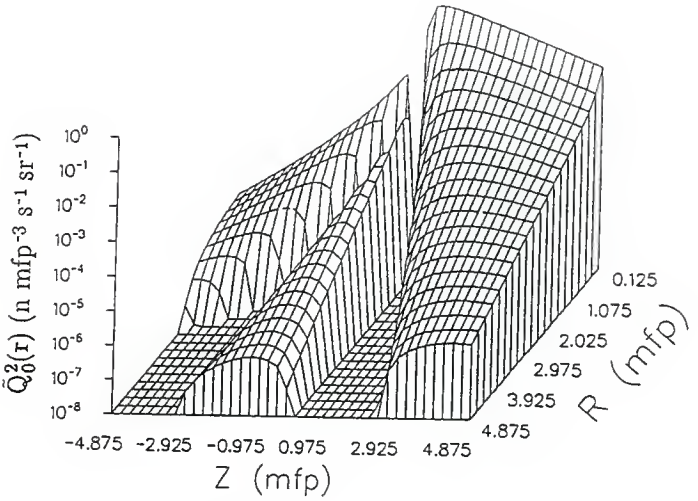


Fig. 3.6. Second collision scalar source distribution generated with the Legendre expansion of the scattering cross section for a c–value of 0.5, but with all negative scalar sources set to zero. The resultant source has been renormalized based on the original second collision source generated with the Legendre expansion.

source with the negatives set to zero is used by TWODANT, the scalar flux distribution is reasonably shaped (i.e., peaked in the forward direction) even though the flux density never fully converges.

The angular distributions of the source at the locations (r = 1.6667 mfp, z = 1.6667 mfp) and (r = 1.6667 mfp, z = 3.3333 mfp) have been plotted in Figs. 3.7 and 3.8. Again the spherical harmonic expansion of the source still produces negative sources in certain directions. The angular source distributions are not significantly different for the second collision source using the fixup technique of setting the negative scalar sources to zero to the fixup technique which uses the exact cross section.

3.3 Technique of Peak Renormalization of the Scalar Sources

The next fixup technique follows the same procedure as the previous method, but also eliminates all scalar sources and associated moments not under the "primary peak". The primary peak is easily detected because it contains the maximum scalar source and is bounded by either the problem boundary or negative scalar sources. Presently, this fixup technique, as implemented in the computer code SRC2D.FOR, is valid only for a scattering cross section peaked in the forward direction (i.e., the region along the positive z–axis is assumed to be under the "primary peak"). The procedure for finding the primary peak for the forward scattering problem involves first scanning along the top set of spatial meshes while searching for the first negative valued scalar source. Once the first negative is found, all source moments to the right (and including the negative) are set to zero. Next, a search is conducted from the top to the bottom along an r–spatial grid line (i.e., over a set of z spatial nodes). When the first negative or zero scalar source

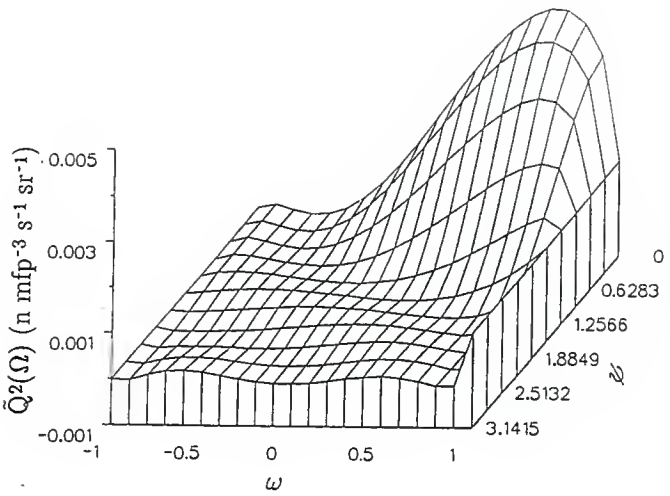


Fig. 3.7. Angular source distribution at (r = 1.667 mfp, z = 1.667 mfp) for the second collision source generated with the Legendre expansion of the scattering cross–section for a c–value of 0.5.

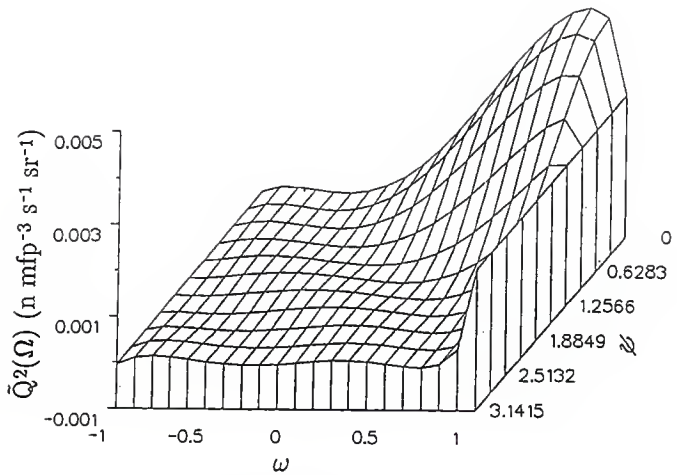


Fig. 3.8. Angular source distribution at (r = 1.667 mfp, z = 3.333 mfp) for the second collision source generated with the Legendre expansion of the scattering cross section for a c value of 0.5.

has been found, then all source moments below and including the negative source are set to zero. The procedure is then repeated for all r–spatial grid lines. As before, the new source is renormalized to regain particle balance.

In multigroup problems which use this fixup technique, only scattering which involves neutrons moving from the source group to the source group follow the above methodology. Neutrons scattering into other groups are corrected with the technique of setting negative scalar sources to zero. For instance, for a two–group problem with the monodirectional source in group one, peak renormalization would be used twice when calculating the second collision sources. First, the second collision source in group one is corrected with the technique of peak renormalization for the neutron which leaves the monodirectional point source, scatters the first time staying in group one, and then scatters a second time without leaving group one. The peak renormalization technique is used a second time to fix the component of the second collision source in group two for the neutron which leaves the monodirectional point source, scatters the first time staying in group one, and then scatters a second time and transfers to group two. The other component of the second collision source in group two for the neutron which leaves the monodirectional point source, scatters the first time and transfers into group two, and then scatters a second time and remains in group two is corrected with the technique of setting the negative scalar sources to zero.

The result of this procedure is a new scalar source distribution which closely resembles the scalar source distribution from the exact cross section technique.

Figure 3.9 illustrates the source distribution obtained with this technique for the cross section of Fig 3.1. The angular distributions will be the same as Figs. 3.7 – 3.8 except for multiplication by a renormalization factor.

3.4 A Modified Cross Section Expansion Technique

The final fixup technique explored utilizes a method which alters the moments of the Legendre cross section expansion to force the reconstituted cross section to be always non–negative. The technique is valid only if the original exact cross section data used to generate the cross section expansion was positive over the range (–1,1). If Eq. (16) represents the original (uncorrected) Legendre expansion for the scattering cross section, then the modified cross–section expansion which produces non negative values is [Sz39]

$$\sigma_{g_0g}(\Omega\cdot\Omega_s) = \sum_{l=0}^{l_{max}} \frac{(l_{max}+2-l)(l_{max}+1-l)}{(l_{max}+2)(l_{max}+1)} \left[\frac{2l+1}{4\pi}\right] \sigma^l_{g_0g} P_l(\Omega\cdot\Omega_s) \qquad (53)$$

The scalar source distribution for the second collision source produced with the modified cross section of Fig 3.10 is shown in Fig. 3.11. An apparent disadvantage of this technique is that it produces positive valued sources in locations where sources should not physically exist. These sources are a consequence of the broadening of the cross section peak by the Legendre coefficients. As before, the angular distributions at (r = 1.6667 mfp, z = 1.6667 mfp) and (r = 1.6667 mfp, z = 3.333 mfp) are plotted (see Figs 3.12 – 3.13). These plots demonstrate that the modified cross section technique also guarantees that the angular source along all discrete directions is positive, which is at the expense of a broader angular peak.

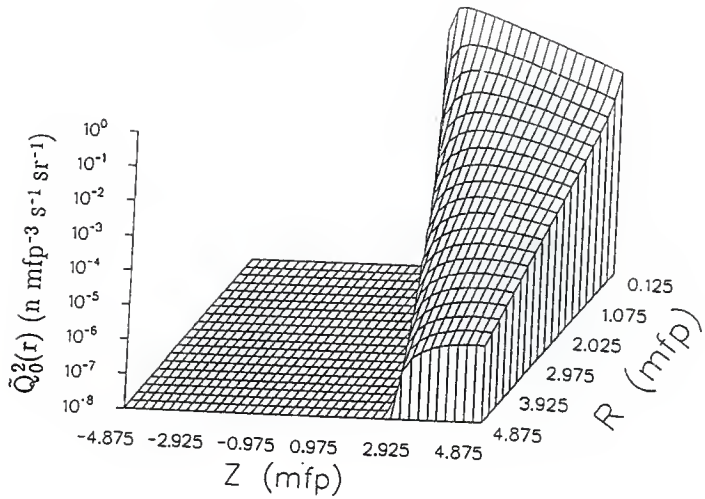


Fig. 3.9. Second collision scalar source distribution generated with the Legendre expansion of the scattering cross-section for a c value of 0.5, but with all sources not under the primary peak set to zero. The resultant source has been renormalized based on the original second collision source generated with the Legendre expansion.

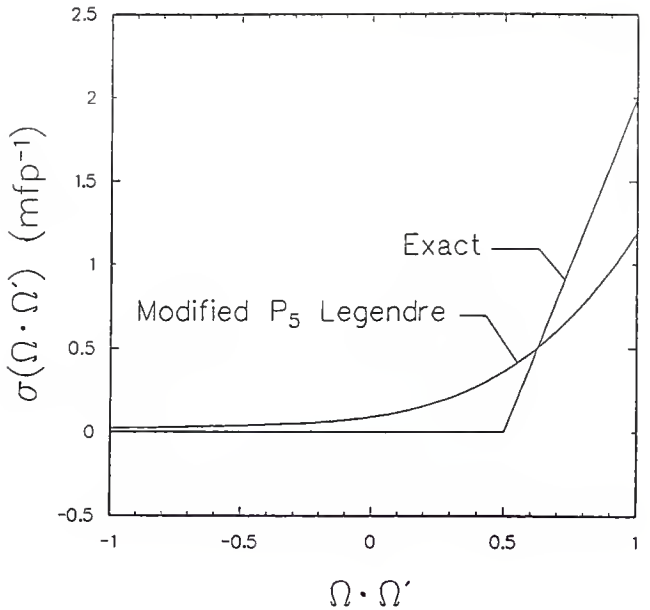


Fig. 3.10. Scattering transfer cross sections used to investigate negative source fixup techniques. The scattering cross sections include the exact and the modified Legendre expansion.

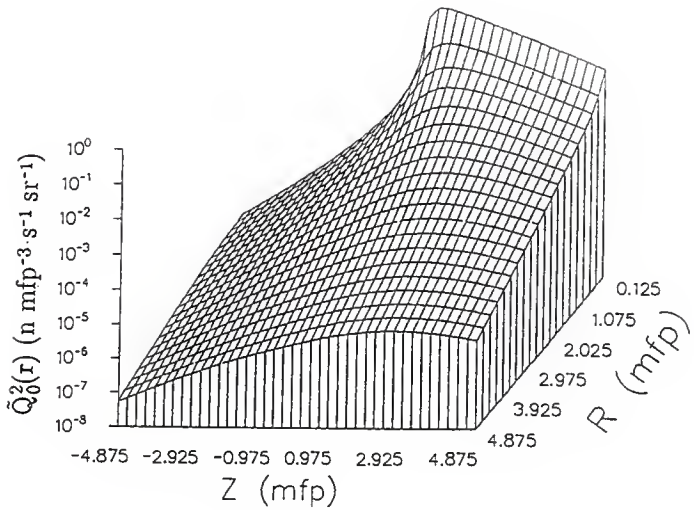


Fig. 3.11. Second collision scalar source distribution generated with the modified Legendre expansion of the scattering cross section for a c value of 0.5.

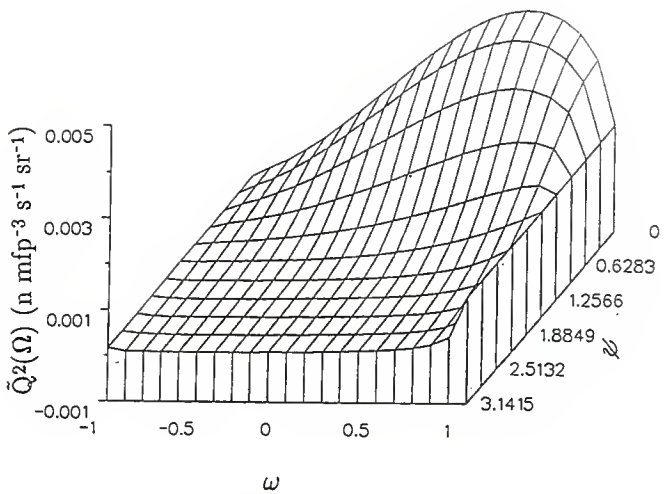


Fig. 3.12. Angular source distribution at (r = 1.667 mfp, z = 1.667 mfp) for the second collision source generated with the modified Legendre expansion of the scattering cross section for a c value of 0.5.

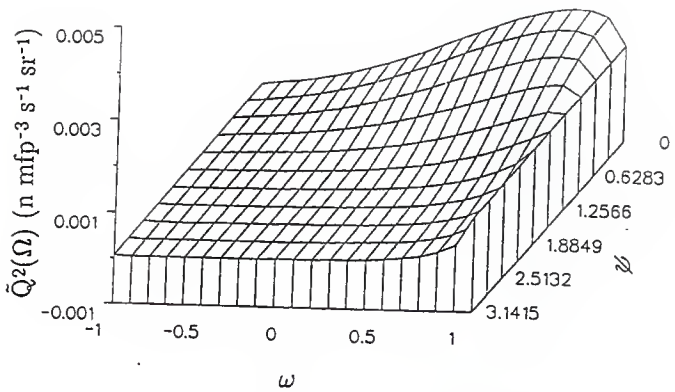


Fig. 3.13. Angular source distribution at (r = 1.667 mfp, z = 3.333 mfp) for the second collision source generated with the modified Legendre expansion of the scattering cross section for a c value of 0.5.

3.5 Comparison of Scalar Sources for Negative Fixup Methods

When comparing the scalar source distributions produced by the various second collision fixup methods, it becomes difficult to actually see all the differences between the four techniques using the three–dimensional plots. To facilitate a comparison of each of the four fixup methods and also the standard Legendre expansion, four "slices" holding r constant at 0.125, 1.625, 3.375, and 4.875 mfp have been plotted in Figs 3.14, 3.15, 3.16, and 3.17, respectively. All the methods except for the modified cross section expansion technique produce sources which are quite close to each other under the primary peak. The source distribution under the peak for the modified cross section technique starts out smaller than all the other methods at radial distances near the source, but as the radial distance increase the source becomes much larger than the other methods. Some of this effect can be neglected by noting that the magnitude of the source has also significantly decreased relative to the source at smaller radial distances. The negative portion of the scalar source distribution for the uncorrected Legendre expansion follows the same trend as the peak region for each of the methods. All four methods do a comparable job of calculating angular source distributions under the primary peak.

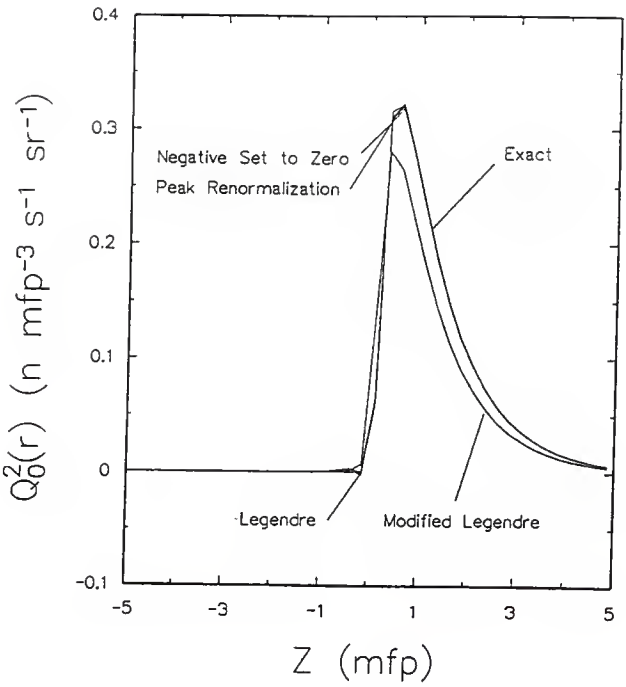


Fig. 3.14. Second collision scalar source distribution as a function of z for the radial mesh r = 0.125 mfp.

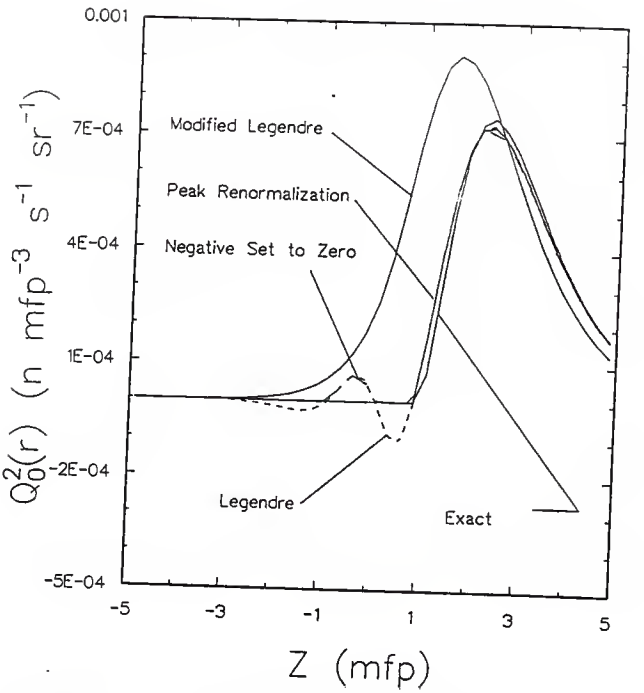


Fig. 3.15. Second collision scalar source distribution as a function of z for the radial mesh r = 1.675 mfp.

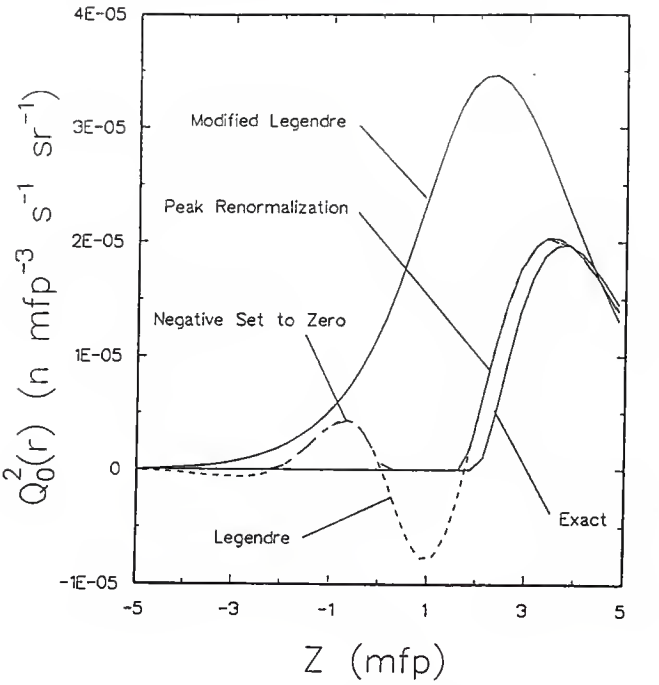


Fig. 3.16. Second collision scalar source distribution as a function of z for the radial mesh r = 3.375 mfp.

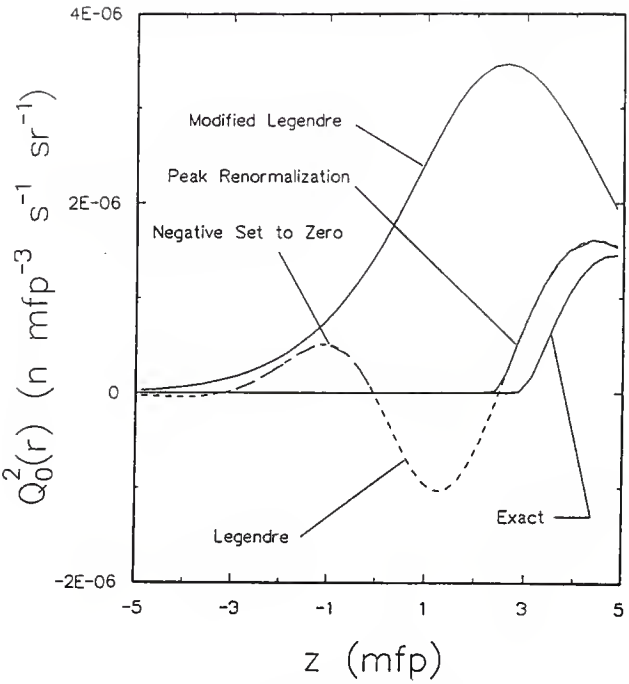


Fig. 3.17. Second collision scalar source distribution as a function of z for the radial mesh r = 4.875 mfp.

4. RESULTS

Three studies have been conducted to investigate characteristics of TWODANT for performing calculations of skyshine line–beam response functions. In the first study, the flux density in a one group isotropically scattering medium is considered for three S_n quadrature levels, for three scattering–to–total cross section ratios and for both a first and a second collision source. The study was conducted to investigate the severity of ray–effects for a highly anisotropic source (in space and direction) without the added complexities of anisotropic scattering. In the second study, the flux density in a one–group anisotropically forward scattering medium is considered using both a first and a second collision source. In this second study, the effectiveness of each of the negative source fixup techniques discussed in Chapter 3 is assessed to determine the relative merits of each for the calculation of flux densities. The study was also conducted to investigate the additional ray–effects caused by anisotropic scattering. Finally in the last study, the flux density in dry air for an energy group containing a 14 MeV monodirectional neutron source is considered. This final study is important because the problems encountered in this one group are comparable to the problems which would develop in the calculation of the flux densities for an actual line–beam response function.

4.1 Isotropic Scattering Results

An infinite, homogeneous, isotropically scattering medium was studied to investigate the severity of ray–effects for a highly anisotropic source (in space and direction) i.e., a monodirectional point source, and to see how the first and second collision sources mitigated the ray–effects. These first and second collision sources

are important because a monodirectional point source (i.e., a zero–th collision source) cannot be properly described by using a set of spherical harmonic expansion coefficients, which are the required input for TWODANT. Furthermore, if such an expansion were used by TWODANT, the reconstituted angular source distribution would contain numerous negative sources and these negative sources would produce completely erroneous results.

This phase of the report also considers the effect of S_n quadrature order and c value on ray–effects. The three S_n quadrature levels considered were 6, 8, 12, which correspond to 12, 20, and 42 directions per octant. The three different c values (i.e., the ratio of the total scattering cross section to the total cross section) used were 0.1, 0.5, and 0.9. While most skyshine calculations are considered to be the result of scattering which should correspond to large c values, the multigroup approximation tends to correspond to problems with small c values. The small c values arise because most of the neutron interactions in a group result in the scatter of the neutron into a lower energy group which then acts as an absorption for the original group. Thus, the results for the small c values are of more importance in line–beam response function calculations than are the results for the large c values. The criterion used for comparison of the various flux densities computed is the apparent reduction in ray–effects brought about by changes in the S_n quadrature level, the c value, and the use of a first or a second collision source.

The problem considered in this section of the report is based on a monoenergetic, monodirectional point source located at the center of a cylinder with a radius of 5 mfp and a height of 10 mfp. Such a size was deemed as a good approximation to an infinite medium, because few neutrons can escape from the problem boundaries (the non–leakage probability is $P_{nl} \approx 0.98$).

The spatial meshing used in TWODANT was 20 grid lines in the radial direction and 40 grid lines in the z direction. This choice was based on the need to maintain a mesh which could be solved with the limited computer resources available, while retaining a fair degree of accuracy in the calculated flux densities.

4.1.1 *Strongly Absorbing Medium* ($c = 0.1$)

The first isotropically scattering medium considered was characterized by a 10% scattering probability per interaction. Figures 4.1 – 4.3 show the spatial distribution of the scalar flux density obtained with the first collision source for three different S_n quadrature orders. Ray effects are very evident, and, in fact, they are so extreme that convergence of the problem cannot be achieved (see Tab. 4.1 for the TWODANT convergence trends). These convergence problems are probably caused by ray–effects which interfere with the finite differencing scheme by producing oscillations in the angular flux density between iterations. Figures 4.4 – 4.6 illustrate the spatial distribution of the scalar flux density obtained with the second collision source for the three S_n quadrature orders used. Almost all of the ray–effects have been eliminated by use of the second collision source and the problem no longer suffers from the convergence problems of the first collision source.

4.1.2 *Intermediate Scattering Medium* ($c = 0.5$)

The next medium considered was characterized by a 50% scattering probability per interaction. Figures 4.7 – 4.9 show the spatial distribution of the scalar flux density obtained with the first collision source for three different S_n quadrature orders. This set of figures demonstrates the ability of higher S_n

Tab. 4.1. A comparison of the number of iterations, the location of the maximum error, and the particle balance when TWODANT achieves convergence of the flux density for various isotropically scattering media. The problem geometry is a cylinder with a radius of 5 mfp and a height of 10 mfp with a monodirectional point source at the cylinder center. The source type refers to either a first collision source or a second collision source from a point source emitting particles directly upward. The spatial meshing is 20 cells in the radial direction and 40 cells in the z direction. Cells are numbered outward from the center and upward from the bottom.

S_n ORDER	c VALUE	SOURCE TYPE	# of ITERATIONS	LOCATION OF ERROR	PARTICLE BALANCE
6	0.1	FIRST	9	19,11	5.76E–05
8	0.1	FIRST	UNCONVERGED	4,36	2.91E–04
12	0.1	FIRST	UNCONVERGED	13,31	–1.54E–03
6	0.1	SECOND	4	18,13	4.57E–09
8	0.1	SECOND	4	1,2	2.41E–10
12	0.1	SECOND	4	1,1	–2.62E–10
6	0.5	FIRST	6	1,3	–2.62E–08
8	0.5	FIRST	6	19,34	2.97E–06
12	0.5	FIRST	6	17,1	5.98E–04
6	0.5	SECOND	6	19,12	–6.28E–09
8	0.5	SECOND	5	1,1	–4.84E–08
12	0.5	SECOND	6	1,9	–4.27E–08
6	0.9	FIRST	7	19,9	–4.64E–08
8	0.9	FIRST	7	16,1	–3.98E–08
12	0.9	FIRST	7	6,23	–3.43E–08
6	0.9	SECOND	6	2,22	–1.04E–07
8	0.9	SECOND	6	2,22	–8.55E–08
12	0.9	SECOND	6	2,22	–7.38E–08

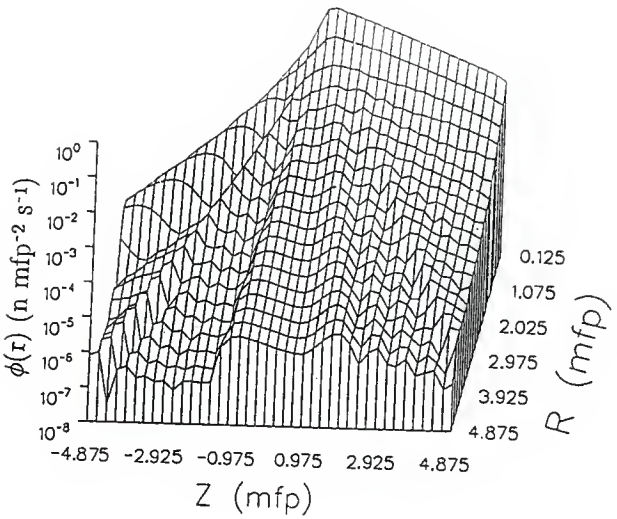


Fig. 4.1. Scalar flux density from a first collision source with an S_n level of 6 for an isotropically scattering medium with a c value of 0.1.

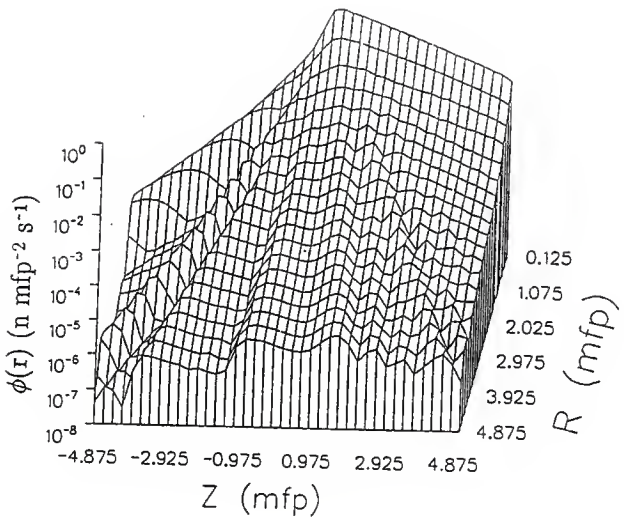


Fig. 4.2. Scalar flux density from a first collision source with an S_n level of 8 for an isotropically scattering medium with a c value of 0.1.

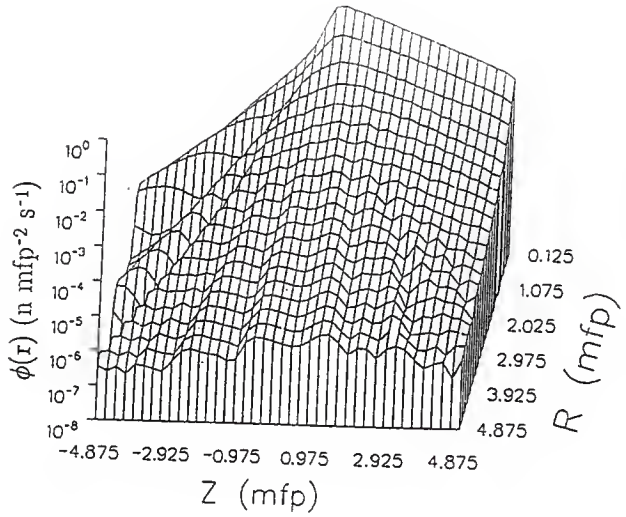


Fig. 4.3. Scalar flux density from a first collision source with an S_n level of 12 for an isotropically scattering medium with a c value of 0.1.

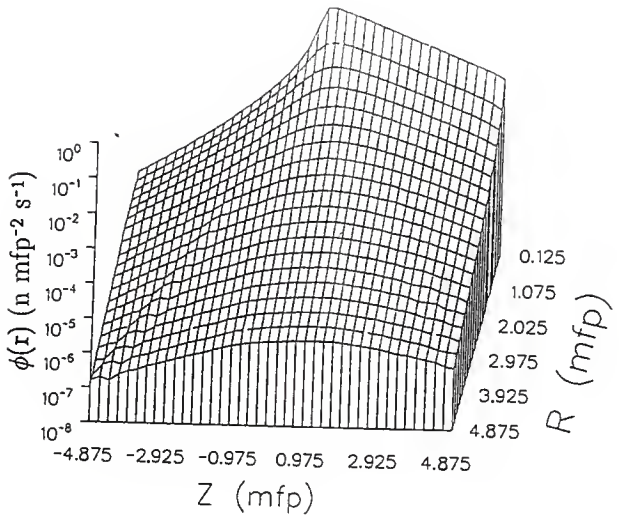


Fig. 4.4. Scalar flux density from a second collision source with an S_n level of 6 for an isotropically scattering medium with a c value of 0.1.

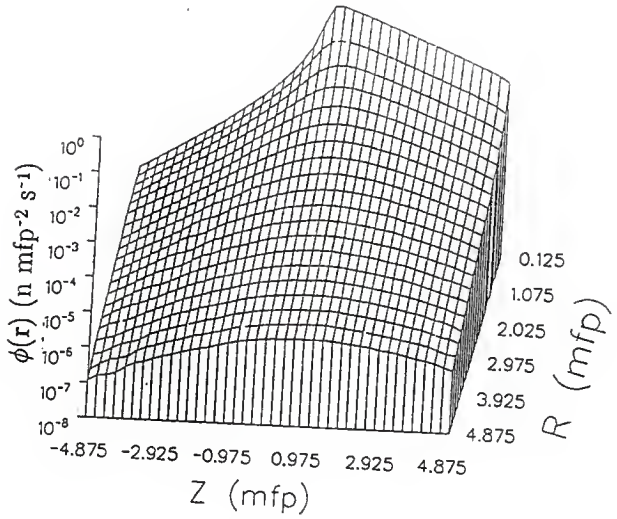


Fig. 4.5. Scalar flux density from a second collision source with an S_n level of 8 for an isotropically scattering medium with a c value of 0.1.

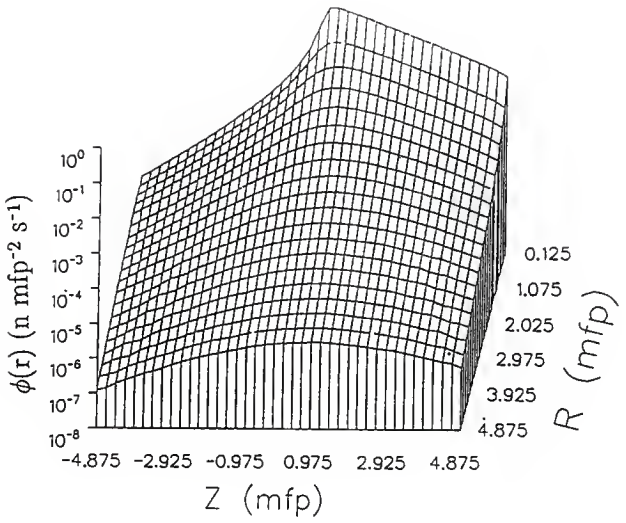


Fig. 4.6. Scalar flux density from a second collision source with an S_n level of 12 for an isotropically scattering medium with a c value of 0.1.

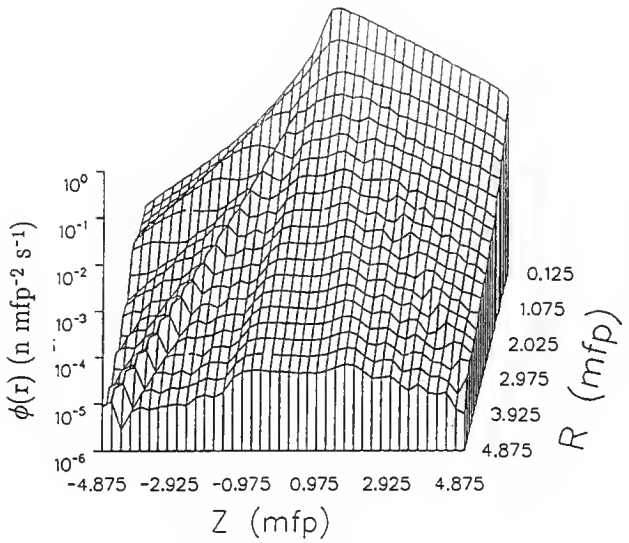


Fig. 4.7. Scalar flux density from a first collision source with an S_n level of 6 for an isotropically scattering medium with a c value of 0.5.

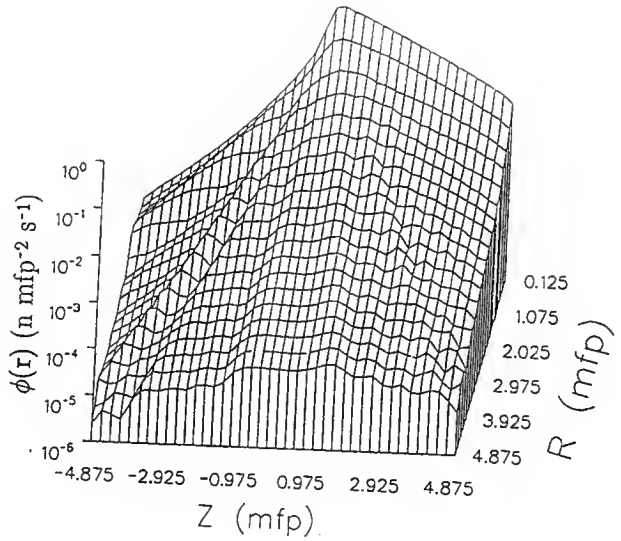


Fig. 4.8. Scalar flux density from a first collision source with an S_n level of 8 for an isotropically scattering medium with a c value of 0.5.

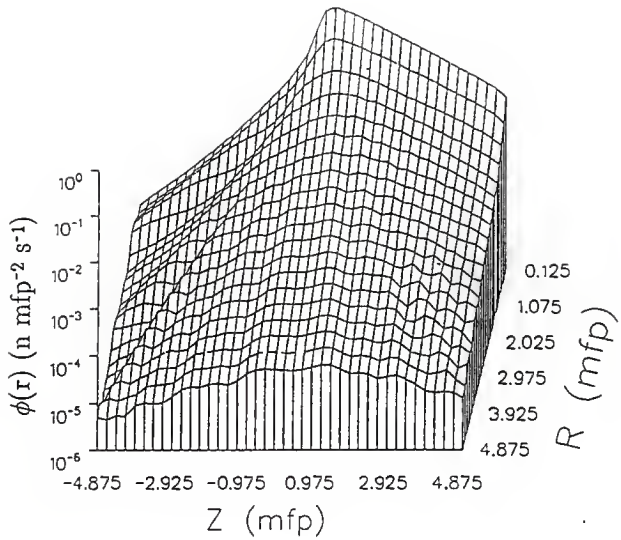


Fig. 4.9. Scalar flux density from a first collision source with an S_n level of 12 for an isotropically scattering medium with a c value of 0.5.

quadrature orders to reduce ray–effects. Figures 4.10 – 4.12 illustrate the spatial distribution of the scalar flux density obtained with the second collision source as a function of the S_n quadrature order. As before, the ray–effects have been reduced by using both the second collision source and the higher S_n quadrature orders.

Now with the increased scattering (i.e., the larger c value), the ray effects have decreased for the first collision source and increased for the second collision source. This trend in ray–effects can be attributed directly to the contributions to the total flux density from the transport result, which suffers from the ray–effects, and the orders–of–scattering results, which does not suffer from ray–effects. For the $c = 0.1$ medium and the second collision source technique almost the entire total flux density (exactly 99% for an infinite medium, since $0.99 = 1 - 0.1 \times 0.1$) is accounted for in the first two interactions which are computed with the orders–of–scattering technique. Alternatively, for the $c = 0.5$ medium and the second collision source, the orders–of–scattering component computes a smaller portion of the total flux density (exactly 75% for an infinite medium, since $0.75 = 1 - 0.5 \times 0.5$) and will subsequently suffer more ray–effects. The first collision source technique shows the opposite trend, because the orders–of–scattering technique is used to compute a much smaller portion of the total flux density (i.e., 90% for the $c = 0.1$ infinite medium and 50% for the $c = 0.5$ infinite medium).

4.1.3 *Highly Scattering Medium* ($c = 0.9$)

The final medium considered is highly scattering (i.e., few interactions involve an absorption). Figures 4.13 – 4.15 illustrate the effect S_n quadrature order has on the calculated scalar flux density obtained with the first collision source technique. With the higher S_n orders, the flux density becomes smoother

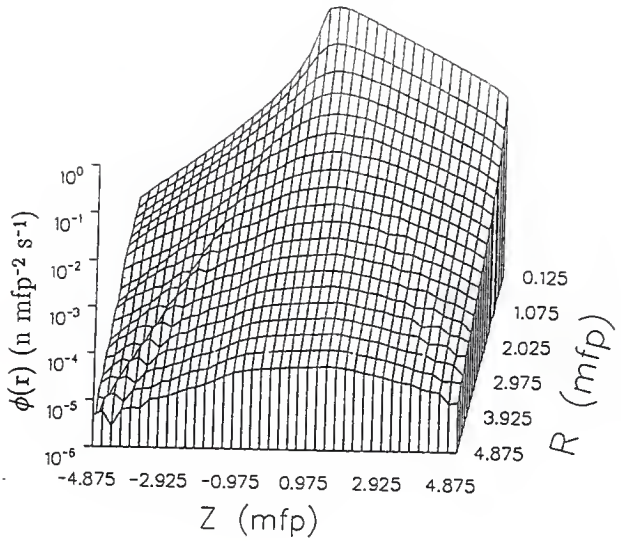


Fig. 4.10. Scalar flux density from a second collision source with an S_n level of 6 for an isotropically scattering medium with a c value of 0.5.

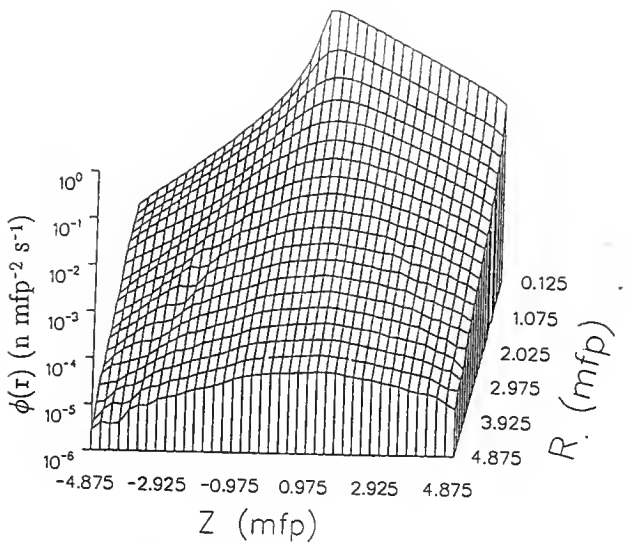


Fig. 4.11. Scalar flux density from a second collision source with an S_n level of 8 for an isotropically scattering medium with a c value of 0.5.

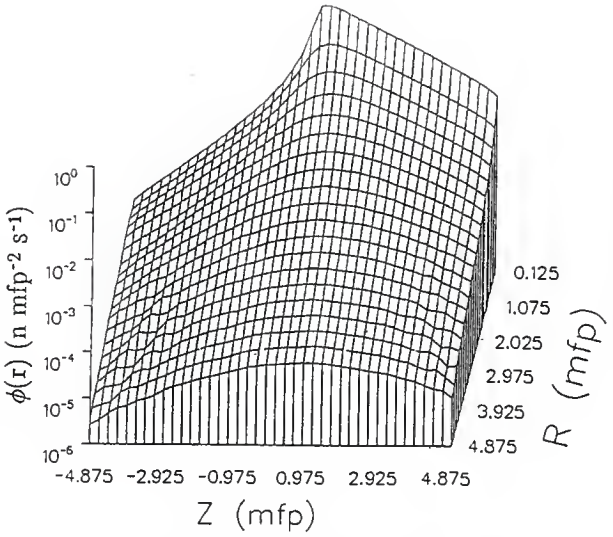


Fig. 4.12. Scalar flux density from a second collision source with an S_n level of 12 for an isotropically scattering medium with a c value of 0.5.

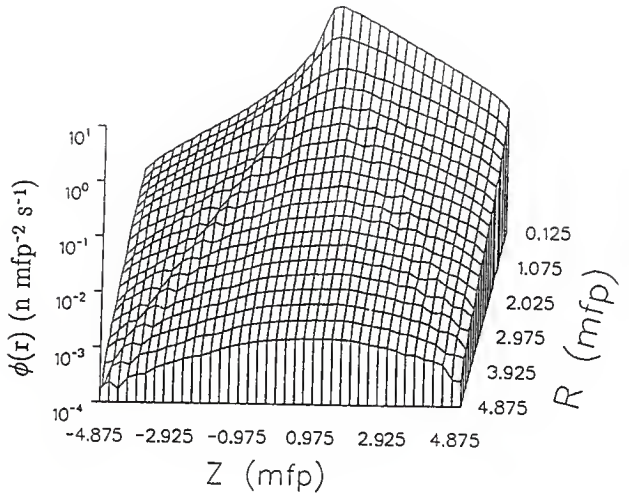


Fig. 4.13. Scalar flux density from a first collision source with an S_n level of 6 for an isotropically scattering medium with a c value of 0.9.

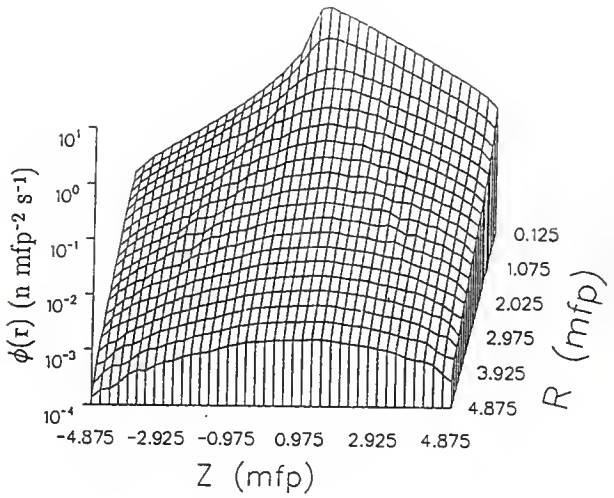


Fig. 4.14. Scalar flux density from a first collision source with an S_n level of 8 for an isotropically scattering medium with a c value of 0.9.

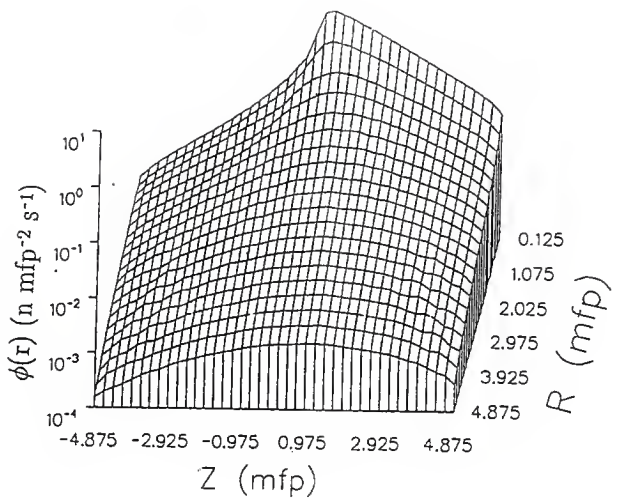


Fig. 4.15. Scalar flux density from a first collision source with an S_n level of 12 for an isotropically scattering medium with a c value of 0.9.

the effect of S_n and the ray effects are almost completely alleviated. Figures 4.16 – 4.18 illustrate quadrature order on the scalar flux density when the second collision source is used. Any ray effects which appeared when using the first collision source have been eliminated by using the second collision source. Moreover, the need for a large number of discrete directions to mitigate ray–effects is no longer necessary (as was necessary with the first collision source).

In comparison to the other two mediums, the observed ray–effects are much smaller for the spatial distributions of the scalar flux density obtained with both the first collision source and also with the second collision source. This reduction in the severity of ray–effects is expected for large c values, because the higher level of scattering insures more neutrons are redistributed (rather than absorbed) in both space and direction, thereby smoothing out the neutron distribution and reducing ray effects.

4.1.4 *General Trends for Isotropic Scattering Media*

The first general trend observed in the flux densities for the various isotropically scattering media considered indicates that ray effects are reduced for for both a first collision source and also a second collision source by increasing the S_n quadrature order. Such a trend is expected for any discrete ordinates transport code such as TWODANT. This reduction in ray effects is a consequence of the improved representation of the angular flux density in each cell, which can insure the transport of particles into a higher number of neighboring cells.

The second trend observed was the second collision source always produced a much smoother total flux density (i.e., fewer ray–effects) than did the corresponding first collision source. This is a direct result of the spreading of the

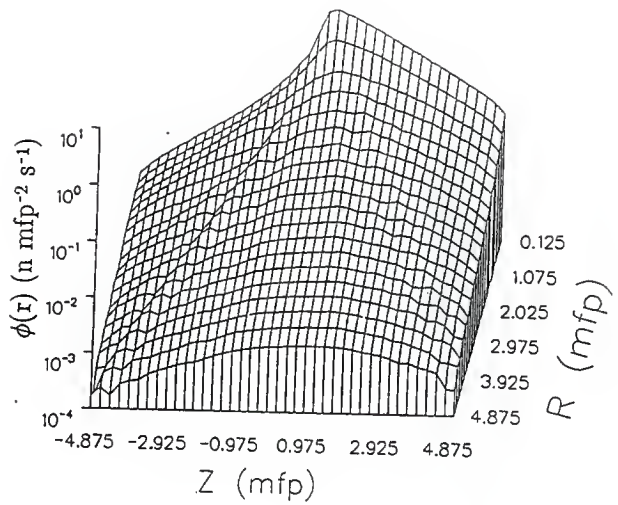


Fig. 4.16. Scalar flux density from a second collision source with an S_n level of 6 for an isotropically scattering medium with a c value of 0.9.

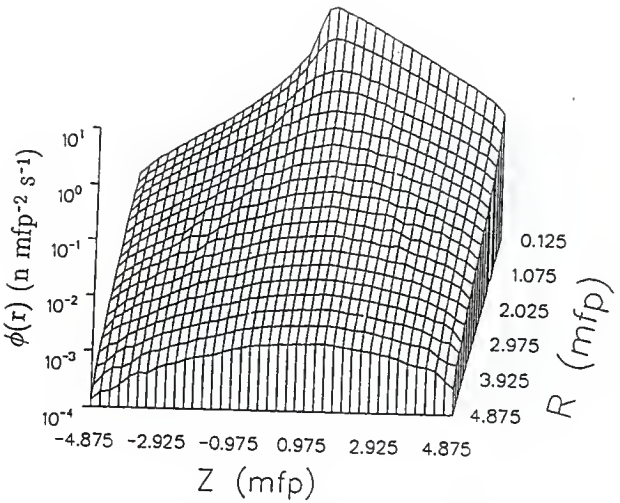


Fig. 4.17. Scalar flux density from a second collision source with an S_n level of 8 for an isotropically scattering medium with a c value of 0.9.

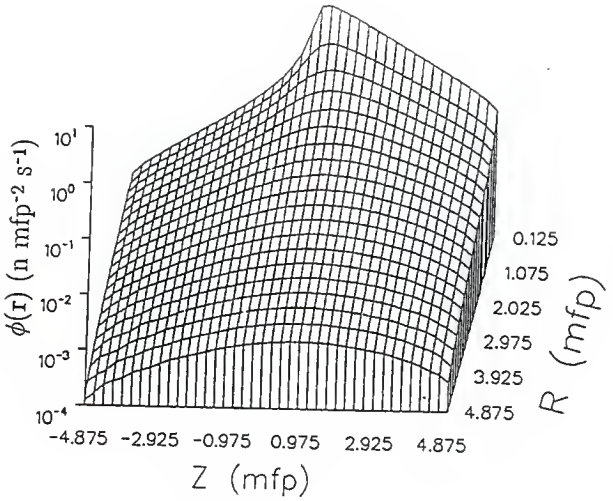


Fig. 4.18. Scalar flux density from a second collision source with an S_n level of 12 for an isotropically scattering medium with a c value of 0.9.

source over the entire medium by the orders–of–scattering technique. The first collision source is located only in cells directly above the monodirectional point source, while the second collision source is distributed into all cells of the problem.

The final trend corresponds to the ray–effects observed as the level of scattering varies. Ray–effects become worse as the amount of scattering decreases (i.e., for small values of c), when the first collision source is used in TWODANT. However, the second collision source exhibits fewer ray–effects for both large and small values of c. This reduction in ray effects for the small values of c is because the total flux density is essentially comprised of the ray–free uncollided and once scattered flux densities. With intermediate values of c, TWODANT is required to calculate a larger percentage of the total flux density and the ray–effects begin to increase. This trend eventually reverses as the the c value becomes sufficiently large enough to insure significant redistribution of neutrons throughout the medium.

4.2 Anisotropic Scattering Results

A study of a one–group anisotropically scattering medium was conducted to investigate the ability of various source techniques to generate scalar flux densities from a monodirectional point source. These techniques include (1) the first collision source, (2) the second collision source based on the exact cross section technique, (3) the second collision source using the Legendre cross section expansion, (4) the second collision source based on the Legendre cross section expansion but setting all negative scalar sources and associated moments to zero, (5) the second collision source based on the Legendre cross section expansion but

setting all scalar sources and associated moments not under the primary peak to zero, and (6) the second collision source based on the modified Legendre cross section expansion.

The problem geometry is the previously described cylinder problem with a radius of 5 mfp, a height of 10 mfp, and the monodirectional point source located at the cylinder center. The spatial mesh used by TWODANT is again 20 grid lines in the radial direction and 40 grid lines in the z direction. An S_n quadrature order of 12 is used throughout this investigation. The test scattering cross section (i.e., Fig. 3.1) used in this phase of the study is characterized by a peak in the forward direction. Other cross sections were not considered because forward scattering represents the worst case for ray effects and also because this cross section is typical of those encountered for the source group in multigroup line–beam response function problems.

Figure 4.19 illustrates the scalar flux density obtained with the first collision source technique. The flux density using this method suffers severe ray effects both in the forward direction (i.e., $z>0$) and also in the backward direction (i.e., $z<0$). The only region which does not exhibit ray effects is an area encompassing everything towards the z–axis from a line joining the origin to the top right hand corner of the problem (i.e., $r = 4.875$ mfp and $z = 4.875$ mfp). This region also corresponds to the support for the primary peak under the angular distribution of the first collision source. One difficulty encountered while using the first collision source for this problem was that convergence of the iterated TWODANT flux density was never achieved, even though the scalar flux density was not significantly changing between iterations.

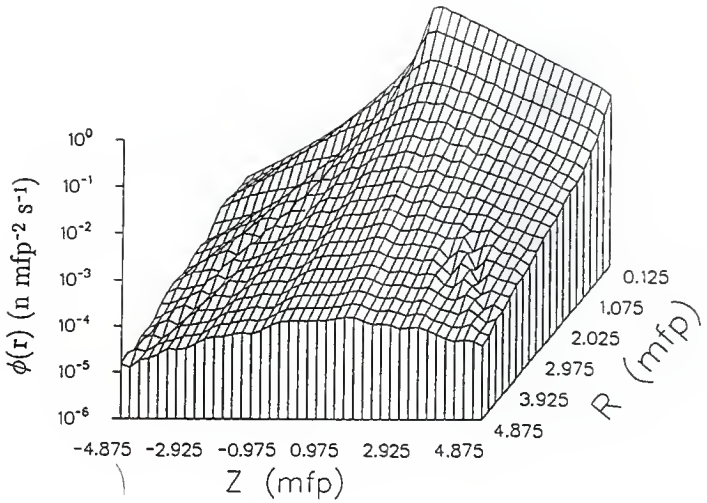


Fig. 4.19. Scalar flux density from a first collision source generated with a forward scattering transfer cross section for a c value of 0.5 and an S_n level of 12. The forward scattering cross section is zero for all cosines of the scattering angle of less than 0.5 and then is a straight ramp to 1.0. The transport calculation is unconverged, but the scalar flux density is not significantly changing between iterations.

Figure 4.20 illustrates the scalar flux density for the same problem, but instead using the second collision source based on the exact scattering cross section technique. Ray effects are now significant only in the backward direction and appear to be prominent only in a region from the z–axis out to the line joining the origin to the lower left hand corner of the problem (i.e., r = 4.875 mfp and z = –4.875 mfp). This region corresponds roughly to the maximum direction particles could be traveling after a second scatter (i.e., 120° from the initial direction.of emission from the point source). Again, with this technique TWODANT did not converge even though the scalar flux density eventually did not change significantly between iterations.

The second collision source based on a normal Legendre expansion was also used in the transport calculation, but because of the large number of cells with negative scalar sources, the iterations of the computed flux densities diverged. Figure 4.21 illustrates the flux density obtained with the second collision source based on the Legendre expansion of the cross section but with the negative scalar sources and associated moments set to zero. Figure 4.22 illustrates the flux density obtained with the second collision source based on the Legendre expansion of the cross section but with the scalar sources and associated moments not under the primary peak set to zero. Both sources, which are closely related to the second collision source based on normal Legendre expansion, produce scalar flux densities which are comparable in both magnitude and shape to the scalar flux densities obtained with the exact scattering cross section technique. These two techniques also suffered from the same convergence problems as the exact cross section technique.

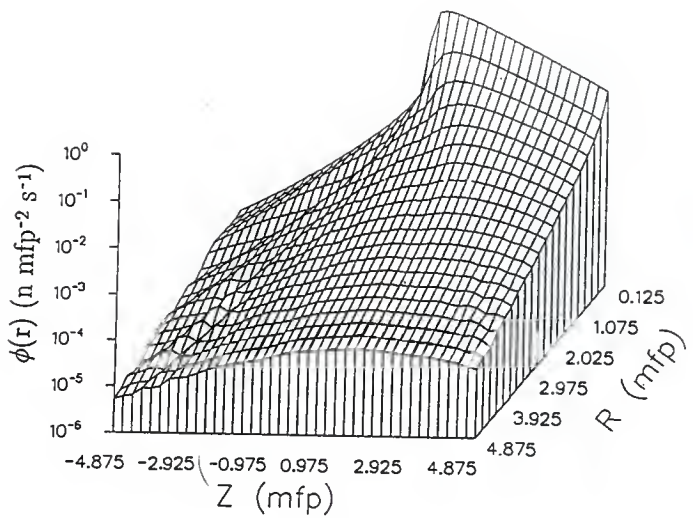


Fig. 4.20. Scalar flux density from a second collision source generated with an exact forward scattering transfer cross section for a c value of 0.5 and an S_n level of 12. The transport calculation is unconverged, but the scalar flux density is not significantly changing between iterations.

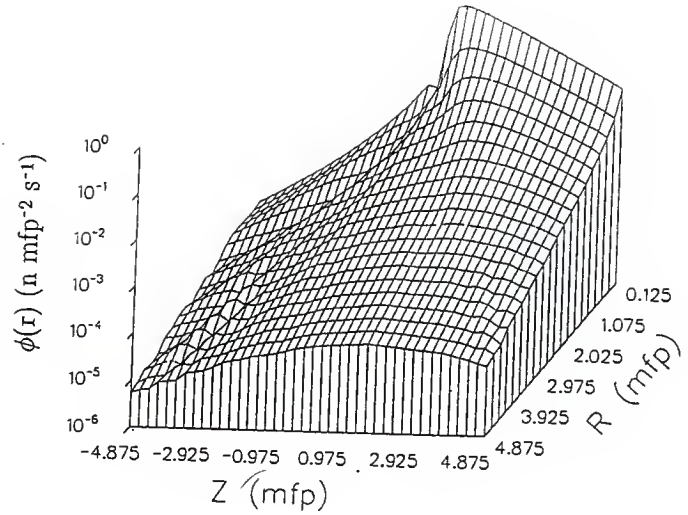


Fig. 4.21. Scalar flux density from a second collision source generated with a Legendre expansion of the scattering transfer cross section but with all negative scalar sources and associated moments set to zero. A c value of 0.5 and an S_n level of 12 are used. The transport calculation is unconverged, but the scalar flux density is not significantly changing between iterations.

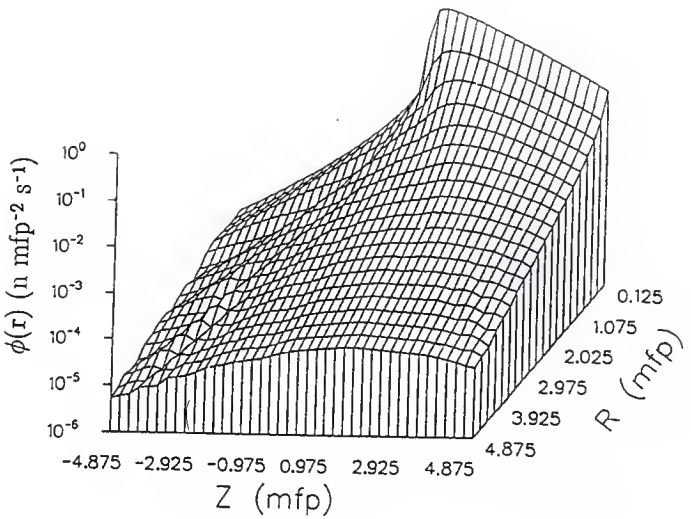


Fig. 4.22. Scalar flux density from a second collision source generated with a Legendre expansion of the scattering cross section but with all scalar sources and associated moments not under the primary peak set to zero. A c value of 0.5 and an S_n level of 12 are used. The transport calculation is unconverged, but the scalar flux density is not significantly changing between iterations.

Figure 4.23 illustrates the scalar flux density obtained with the second collision source based on the modified cross section technique. The scalar flux densities in comparison to the exact technique are slightly smaller in the forward directions and slightly larger in the backward directions. The technique still exhibits ray effects, but no longer suffers from the convergence problems of the other source techniques. This may indicate that the convergence problems of the other techniques are a direct result of the many negative angular sources generated by TWODANT from the spherical harmonic reconstitution of the source.

4.3 Air Source–Group Results

The final study considered the flux density in air computed using the second collision source for source neutrons in a 14 MeV energy group. The problem is based on a cylinder with a radius of 600 m, a height of 1200 m, and a monodirectional point source located at the center. This roughly corresponds to the previous problems which were based on a radius of 5 mfp, because the mfp length of a 14 MeV neutron is approximately 124 m. This problem size, while sufficient for describing the flux densities of this group, would not be large enough for an actual line–beam response function problem because significant neutrons are reaching the outer edges of the problem, but they would appear in lower energy groups. The cross sections for this group were extracted from energy group 4 of the ^{252}Cf weighted cross sections for ^{14}N and ^{16}O in the DABL69 RSIC DATA LIBRARY COLLECTION. [In88] The group has an energy range of 14.191 MeV to 14.840 MeV. Air was treated as being composed of 4.02E–05 atoms $cm^{-1}b^{-1}$ of ^{14}N and 1.07E–05 atoms $cm^{-1}b^{-1}$ of ^{16}O. Only the source group was considered since it serves as the primary source of neutrons for all other groups.

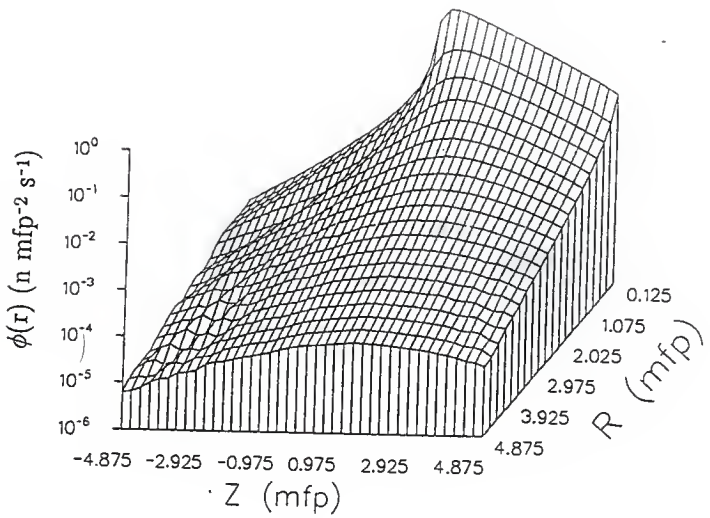


Fig. 4.23. Scalar flux density from a second collision source generated with a modified Legendre expansion of the scattering cross section. A c value of 0.5 and an S_n level of 12 are used. The transport calculation is unconverged, but the scalar flux density is not significantly changing between iterations.

Since the technique of peak renormalization of the second collision source provides the best approximation to the actual source distribution, this technique was considered first. The scalar source distribution is presented in Fig. 4.24 and the angular distributions at locations (r = 75 m, z = 225 m) and (r = 75 m, z = 435 m) are presented in Figs. 4.25 and 4.26. The large number of angular directions which contained negative sources caused the iterated flux densities of TWODANT to diverge. Subsequently, this technique failed to produce results.

In the hope of eliminating the negative sources, the second collision source based on the modified Legendre scattering cross section technique was used next. Surprisingly, this technique also generated negative scalar sources. The spatial distribution of the scalar source after setting all the scalar sources not under the primary peak to zero is presented in Fig. 4.27. Additionally, the angular distributions at locations (r = 75 m, z = 225 m) and (r = 75 m, z = 435 m) are presented in Figs. 4.28 and 4.29. As before, the iterated flux density appeared to be diverging (i.e., poor particle balances) when the code stopped at the maximum number of iterations. Also, a significant portion of the scalar flux density was negative at termination. Thus, the second collision source based on the modified Legendre scattering cross section also failed to produce results.

To determine if the cause of these failures was the spherical harmonic expansion of the source and not related to the extreme forward scattering of the cross section (see Fig. 4.30 for the macroscopic scattering cross section), all but the zero–th moment of the second collision source for the modified cross section technique were set to zero. The resulting flux density is presented in Fig. 4.31. This appears to indicate that the divergence of the other methods was directly related to negative sources in discrete directions.

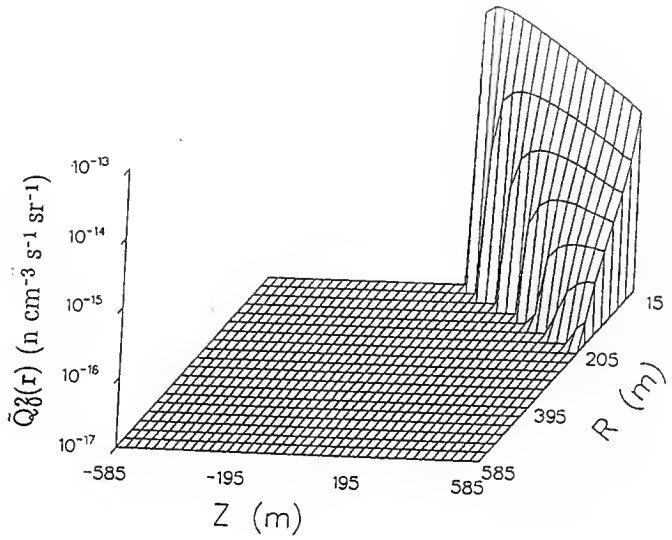


Fig. 4.24. Second collision scalar source distribution generated with a Legendre expansion of the scattering cross section for a 14 MeV energy group in air, but with all sources not under the primary peak set to zero. The resultant source is renormalized based on the original second collision source. The air cross sections are based on energy group 4 of the DLC–130/DABL69 Data Package and are composed of 4.02E–05 atoms cm^{-1} b^{-1} of ^{14}N and 1.07E–05 atoms cm^{-1} b^{-1} of ^{16}O.

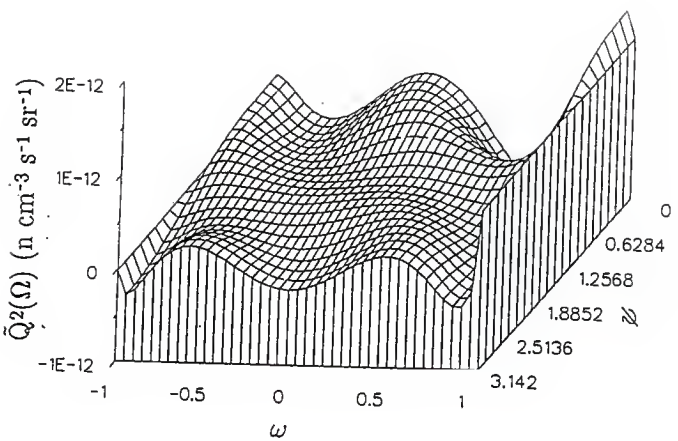


Fig. 4.25. Angular distribution at (r = 75 m, z = 225 m) for the second collision source distribution generated with a Legendre expansion of the scattering cross section for a 14 MeV energy group in air. The air cross sections are based on energy group 4 of the DLC–130/DABL69 Data Package and are composed of 4.02E–05 atoms cm^{-1} b^{-1} of ^{14}N and 1.07E–05 atoms cm^{-1} b^{-1} of ^{16}O.

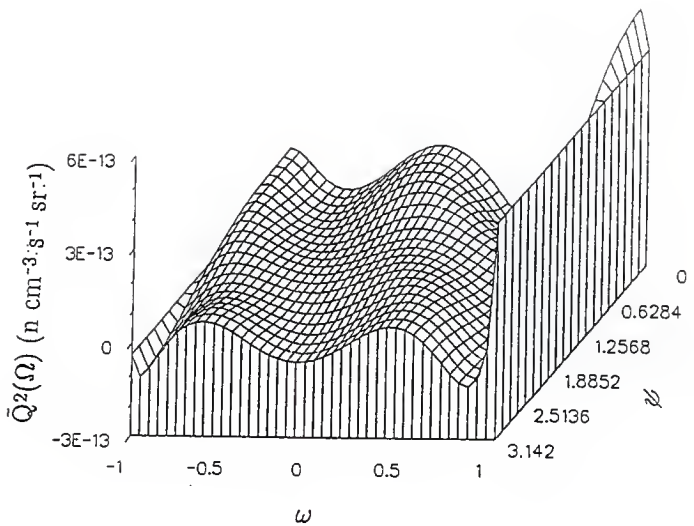


Fig. 4.26. Angular distribution at (r = 75 m, z = 435 m) for the second collision source distribution generated with a Legendre expansion of the scattering cross section for a 14 MeV energy group in air. The air cross sections are based on energy group 4 of the DLC–130/DABL69 Data Package and are composed of 4.02E–05 atoms cm^{-1} b^{-1} of ^{14}N and 1.07E–05 atoms cm^{-1} b^{-1} of ^{16}O.

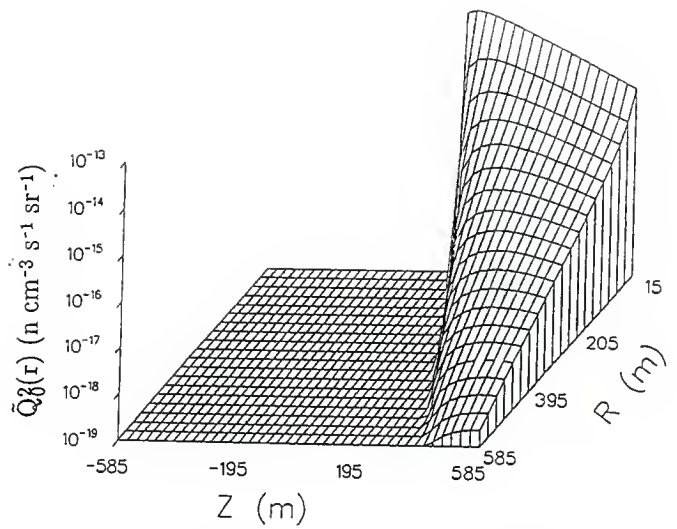


Fig. 4.27. Second collision scalar source distribution generated with a modified Legendre expansion of the scattering cross section for a 14 MeV energy group in air, but with all sources not under the primary peak set to zero. The resultant source is renormalized based on the original second collision source. The air cross sections are based on energy group 4 of the DLC–130/DABL69 Data Package and are composed of 4.02E–05 atoms cm^{-1} b^{-1} of ^{14}N and 1.07E–05 atoms cm^{-1} b^{-1} of ^{16}O.

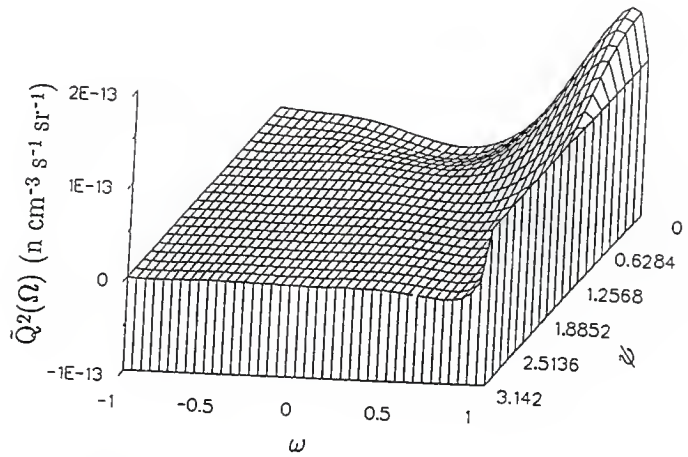


Fig. 4.28. Angular distribution at (r = 75 m, z = 225 m) for the second collision source distribution generated with a modified Legendre scattering cross section expansion for a 14 MeV energy group in air. The air cross sections are based on energy group 4 of the DLC–130/DABL69 Data Package and are composed of 4.02E–05 atoms cm^{-1} b^{-1} of ^{14}N and 1.07E–05 atoms cm^{-1} b^{-1} of ^{16}O.

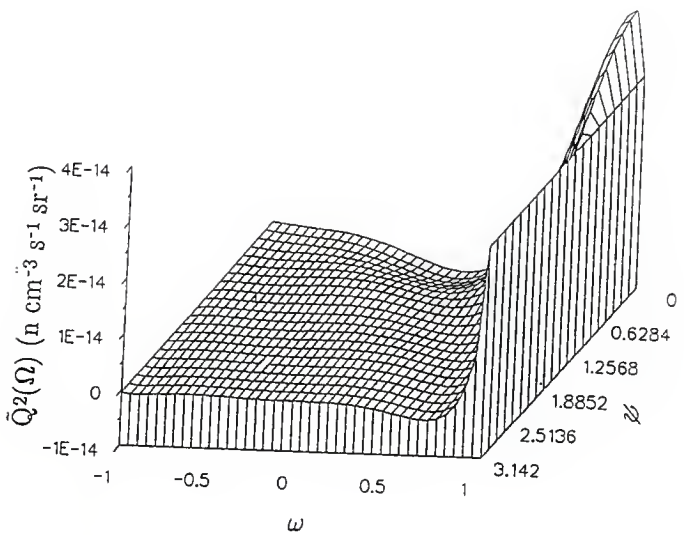


Fig. 4.29. Angular distribution at (r = 75 m, z = 435 m) for the second collision source distribution generated with a modified Legendre expansion for the scattering cross section for a 14 MeV energy group in air. The air cross sections are based on energy group 4 of the DLC–130/DABL69 Data Package and are composed of 4.02E–05 atoms cm^{-1} b^{-1} of ^{14}N and 1.07E–05 atoms cm^{-1} b^{-1} of ^{16}O.

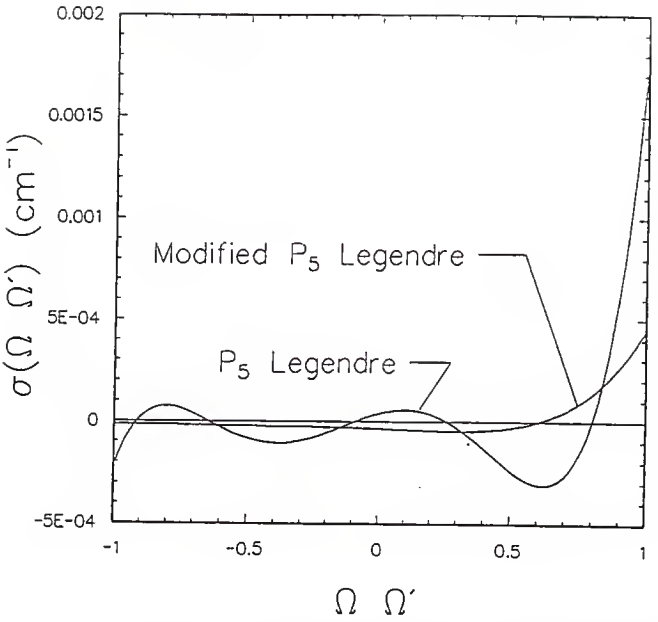


Fig. 4.30. The macroscopic scattering cross section for a 14 MeV energy group in air. The air cross sections are based on energy group 4 of the DLC–130/DABL69 Data Package and are composed of 4.02E–05 atoms cm^{-1} b^{-1} of ^{14}N and 1.07E–05 atoms cm^{-1} b^{-1} of ^{16}O.

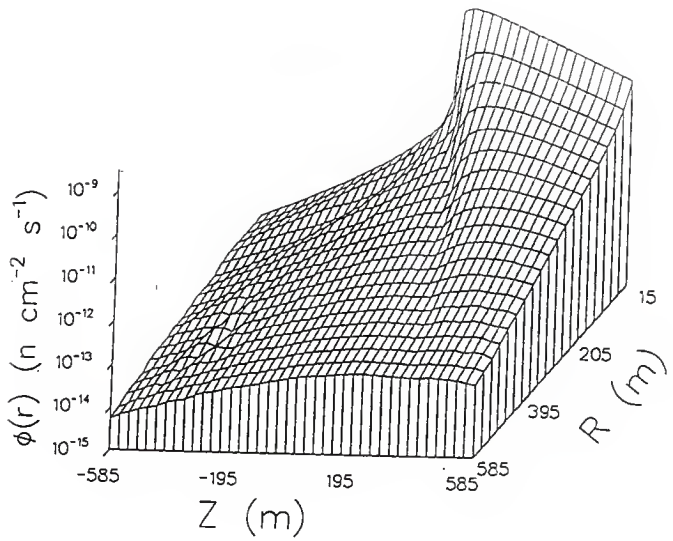


Fig. 4.31. Scalar flux density from a second collision source generated with a modified Legendre expansion of the scattering transfer cross section for a 14 MeV energy group in air, but with only the zero–th moment of the source used in the transport calculation. The air cross sections are based on energy group 4 of the DLC–130/DABL69 Data Package and are composed of 4.02E–05 atoms cm^{-1} b^{-1} of ^{14}N and 1.07E–05 atoms cm^{-1} b^{-1} of ^{16}O.

In attempt to actually solve for the flux density for this source group problem, the revised scattering cross section of Fig. 4.32 was generated by assuming that the cross section of Fig. 4.30 was the fit to a straight line with a break point at 0.8. The scattering cross section was then fit to a P_5 Legendre polynomial and renormalized so that the zero–th moment of the actual scattering cross section was equal to the zero–th moment of this revised scattering cross section. The modified cross section technique was then used to generate the scalar source distribution of Fig. 4.33 and the angular distributions of Figs. 4.34 and 4.35. The flux density for this source distribution is presented in Fig. 4.36. Ray effects remain significant in most of the backward directions, but are not apparent in the forward direction. It is not possible to tell if the flux density in the forward direction is correct in magnitude, but it is reasonable to assume it is low since the anisotropic scattering results of Section 4.2 indicate the modified cross section technique under–estimates in the forward direction and over–estimates in the backward direction.

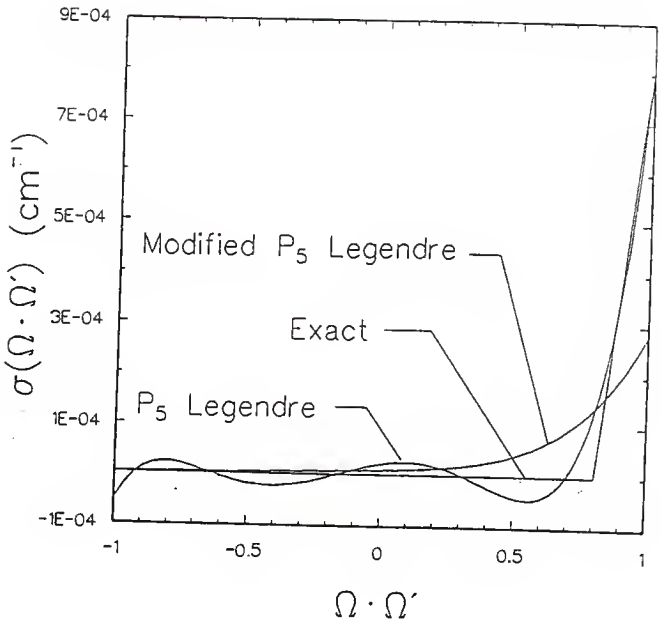


Fig. 4.32. The macroscopic scattering cross section for a 14 MeV energy group in air based on a straight line approximation. The air cross section is based on a 0.8 breakpoint and the P_0 coefficient of energy group 4 of the DLC-130/DABL69 Data Package, where air is composed of 4.02E-05 atoms cm^{-1} b^{-1} of ^{14}N and 1.07E-05 atoms cm^{-1} b^{-1} of ^{16}O.

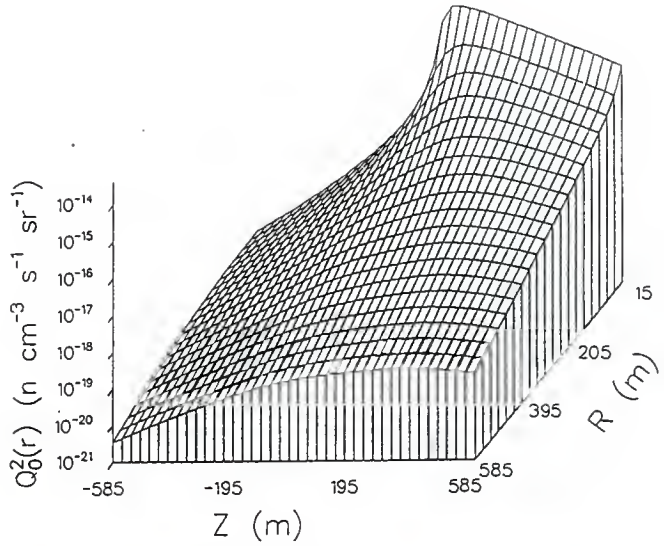


Fig. 4.33. Second collision scalar source distribution generated with a modified Legendre expansion of the scattering cross section for a straight line approximation to a 14 MeV energy group in air. The air cross section is based on a 0.8 breakpoint and the P_0 coefficient of energy group 4 of the DLC–130/DABL69 Data Package, where air is composed of 4.02E–05 atoms cm^{-1} b^{-1} of ^{14}N and 1.07E–05 atoms cm^{-1} b^{-1} of ^{16}O.

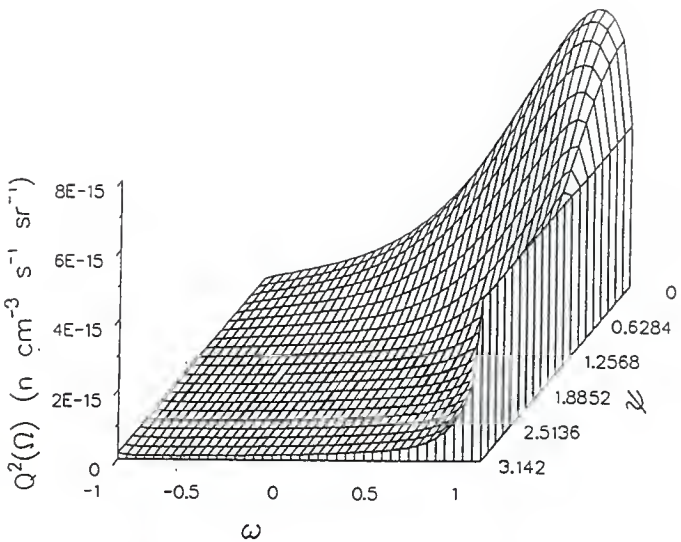


Fig. 4.34. Angular distribution at (r = 75 m, z = 225 m) for the second collision source distribution generated with a modified Legendre expansion of the scattering cross section for a straight line approximation to a 14 MeV energy group in air. The air cross section is based on a 0.8 breakpoint and the P_0 coefficient of energy group 4 of the DLC–130/DABL69 Data Package, where air is composed of 4.02E–05 atoms cm^{-1} b^{-1} of ^{14}N and 1.07E–05 atoms cm^{-1} b^{-1} of ^{16}O.

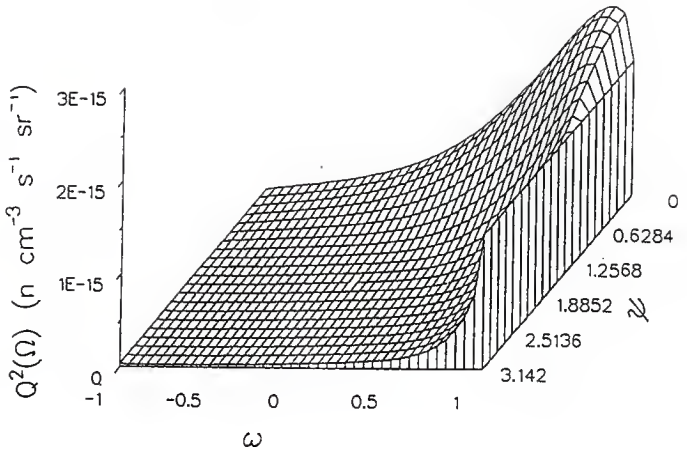


Fig. 4.35. Angular distribution at (r = 75 m, z = 435 m) for the second collision source distribution generated with a modified Legendre expansion of the scattering cross section for a straight line approximation to a 14 MeV energy group in air. The air cross section is based on a 0.8 breakpoint and the P_0 coefficient of energy group 4 of the DLC–130/DABL69 Data Package, where air is composed of 4.02E–05 atoms cm^{-1} b^{-1} of ^{14}N and 1.07E–05 atoms cm^{-1} b^{-1} of ^{16}O.

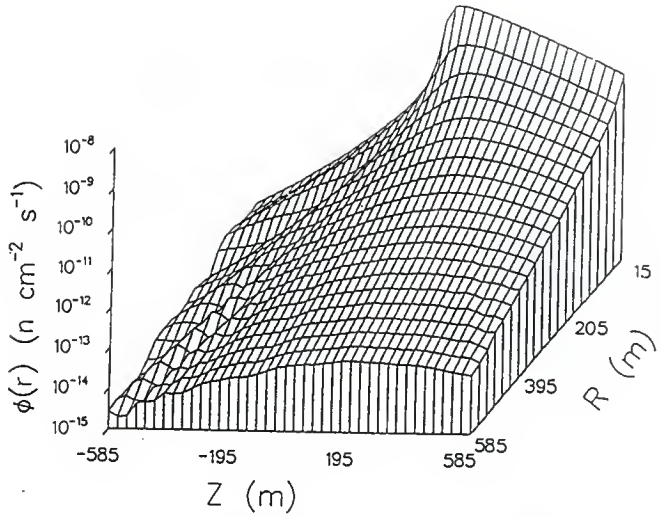


Fig. 4.36. Scalar flux density from the second collision source distribution generated with a modified Legendre expansion of the scattering cross section for a straight line approximation to a 14 MeV energy group in air. The air cross section is based on a 0.8 breakpoint and the P_0 coefficient of energy group 4 of the DLC–130/DABL69 Data Package, where air is composed of 4.02E–05 atoms cm^{-1} b^{-1} of ^{14}N and 1.07E–05 atoms cm^{-1} b^{-1} of ^{16}O.

5. CONCLUSIONS

The discrete–ordinates technique used to solve the transport equation (i.e., TWODANT) does not work well for calculating skyshine line–beam response functions. In particular, the flux density for the energy group associated with the monodirectional, monoenergetic point source suffers severely from ray effects, and, consequently, the discrete–ordinates calculation suffers from convergence problems. The ray effects are directly attributable to the lack of angular sources throughout the entire problem geometry. This can be partially remedied by using either a first or a second collision source as well as a by using a very large number of discrete angular directions. However, these techniques tend to produce negative angular sources, which, subsequently, lead to convergence problems in discrete–ordinates transport codes.

The first technique considered to solve the monodirectional point source problem was to utilize a first collision source based on the uncollided component of the angular flux density. The technique was found to be totally ineffective in reducing ray effects even for the simple isotropically scattering problems. Thus, the technique was discarded and the second collision source was developed. This technique produced relatively ray–free flux densities for all isotropic scattering cases regardless of the ratio of the scattering cross section to the total cross section. However, as the scattering became peaked in the forward direction (i.e., more closely resembling a fine group cross section) the technique became less effective in reducing ray effects.

The location of spatial mesh cells with significant ray effects appeared to be intimately related to the "sharpness" of the differential scattering cross section. For instance, for the exact scattering cross section used in Chapter 3, where the

cross section was non–zero only for scattering angles between 0° and 60° (i.e., cosines of 0.5 to 1.0), the flux density obtained with the second collision source was ray–free for roughly the region above a line joining the monodirectional point source to a point located at (r = 5 mfp and z = –2.5 mfp). The angle this line forms with the direction of the point source is 120° and is twice the range of the non–zero region of the scattering cross section. The ray–free region for the first collision source in a similar description forms an angle of 60° with the direction of the point source. The ray–free region for any other forward scattering cross section can then be determined in a like manner by considering where the scattering cross section is peaked.

The slow convergence and even divergence of the iterated angular flux densities of the source group occur because the source distribution is based on a Legendre expansion of the differential scattering cross section rather than on the exact scattering cross section. This problem occurs because the Legendre expansion of the cross section produces negative values for many scattering directions. The negative scattering cross sections can subsequently generate negative angular collision sources, which, in turn, cause the observed convergence problems. The problem is further compounded because most of the standard production codes require the source to be specified as a truncated spherical harmonic expansion. Thus, even if the exact cross section is known, its moments are still required and can produce the negative angular sources.

The one remedy which does not require modification of the transport code to eliminate these negative scalar sources is to modify the Legendre expansion coefficients used in calculating the second collision source. This technique was very effective in solving the convergence problems for any ideal differential

scattering cross section with non–negative values. However, the technique was found to be ineffective for tabulated scattering cross sections, because the original Legendre fit appeared to be based on negative values. These negative values may indicate problems exist in the techniques used to generate the Legendre expansions from real cross section data.

Another possible remedy, which would require modification of the TWODANT transport code, would be to input the actual angular source along discrete directions rather than just the source moments. However, this technique is not only computationally intensive, but also would require an exact scattering cross section (without negative values). Another remedy would be to generate the second collision source with the Legendre expansion of the scattering cross section, zero the negative scalar sources and associated moments, renormalize the spatial source distribution and then input the result into the transport code. Then when the transport code computed angular sources along discrete directions for each cell, have the transport code set any negative sources to zero and renormalize the source for that cell.

While all these proposals eliminate the convergence problems in the source group, they do not resolve the ray–effect problem. The ray–effect problem could be eliminated by considering a very large number of discrete directions, but this may result in a computationally impractical problem. An alternative is to use some other numerical technique to solve for the angular flux density in the source group and to then input this angular flux density into the discrete ordinates code. The best approach may well be to use a Monte Carlo approach. The Monte Carlo method has the advantage that it has been used in the past to solve line–beam response function problems. Another possibility is to use an integral transport

method. This represents an extension of the technique used to compute the uncollided flux density, the first collision source, the once scattered flux density and the second collision source, but would require numerical integration over the entire problem geometry.

In conclusion, a two–dimensional discrete–ordinates technique (i.e., TWODANT) cannot be easily adapted to the calculation of line–beam response functions for infinite air problems. The problems encountered also point to similar problems which can be expected in a three–dimensional, ground–interface, line–beam response function problem. Any future work in this area should concentrate on the two–dimensional line–beam response function problem with particular regard to eliminating the ray effects of the source group. Study should also be given to extending the problem size, so that line–beam response functions could be generated for distances much larger than considered in this report.

REFERENCES

Ab72 Abramowitz, M., and Stegun, I.A., *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, Dover, 1972.

Al89 Alcouffe, R.E., O'Dell, R.D., and Brinkley, F.W., "A First Collision Source Method that Satisfies Discrete S_n Transport Balance," American Nuclear Society Proceedings of a Topical Meeting on Advances in Nuclear Engineering Computation and Radiation Shielding, 1, 4:1–9 (1989).

Du79 Duderstadt, J.J., and Martin, W.R., *Transport Theory*, John Wiley & Sons, 1979.

Gr87 *Microskyshine Users' Manual* (Draft), Grove Engineering, Inc., 15215 Shady Grove Rd., Rockville, MD 20950, 1987.

La79 Lampley, C.M., *The SKYSHINE–II Procedure: Calculation of the Effects of Structure Design on Neutron, Primary Gamma–Ray, and Secondary Gamma–Ray Dose Rates in Air*, Report RRA–77901 (NUREG/CR–0781), Radiation Research Associates, Fort Worth, TX, 1979.

Lo84 *Users' Guide for TWODANT: A code Package for Two–Dimensional, Diffusion Accelerated, Neutral Particle Transport*, Los Alamos National Laboratory, Los Alamos, NM 87545, 1984.

Mi76 Mikols, W.J., *The Triangular Approximation for Highly Anisotropic Group–to Group Transfer Cross–Sections*,Report #CES–13/1, Center for Energy Studies, Kansas State University, Manhattan, KS 66506, 1976.

Ho79 Hong K.J., *Anisotropic Transport Techniques and Associated Neutron Transfer Cross Section Evaluation*, Center for Energy Studies, Kansas State University, Manhattan, KS 66506, 1979.

Pr76 Price, J.H., Collins, D.G., and Wells, M.B., *Utilization Instructions for SKYSHINE*, Report RRA–N7608, Radiation Research Associates, Fort Worth, TX, 1976.

Sh87 Shultis, J.K., and Faw, R.E., *Improved Response Functions for the MicroSkyshine Method*, Report–189, Engineering Experiment Station, Kansas State University, Manhattan, KS 66506, 1987.

Sz39 Szegö, G., *Orthogonal Polynomials*, American Mathematical Society Colloquium Publications, Vol. 23, Am. Math. Soc., New York 1939.

APPENDIX A: The Computer Program SRCLIN2D

```
CCCC  PROGRAM READS IN CROSS SECTION DATA AND COMPUTES THE FIRST
CCCC  COLLISION SOURCE MOMENTS FROM A POINT SOURCE EMITTING
CCCC  NEUTRONS IN THE UPWARD DIRECTION.  THE ORIGINAL POINT SOURCE HAS
CCCC  A NORMALIZED STRENGTH OF 4 PI.  THE 4 PI IS USED TO FACILITATE
CCCC  A SOURCE NORMALIZATION OF ONE IN THE TWODANT CODE PACKAGE.
CCCC  THE PROBLEM ASSUMES THE SOURCE IS LOCATED IN THE CENTER OF A
CCCC  CYLINDER OF RADIUS RMAX AND HALF HEIGHT ZMAX
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER ORDER,G,GO,GP
      CHARACTER*8 XSFILE,OTFILE,OTFX
      COMMON/CNSTS/PI
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(2),SIGMA(3,9,10)
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(9,235)
      COMMON/BLK2/W(32),X(32)
      DIMENSION APOLY(0:21),SUM(13)
      COMMON/SOURCE/Q(2,21,20,40),FLUX(2,20,40)
CCCC  ------------------------------------------------------------------
CCCC  GET DATA NECESSARY TO PERFORM PROGRAM RUN
      CALL INPUT(RMAX,ZMAX,IMAX,JMAX,XSFILE,OTFILE,GO,MOMENT)
      NGP=NGROUP+3
CCCC  READ IN CROSS-SECTION DATA
      CALL RDXSCT(XSFILE)
CCCC  ------------------------------------------------------------------
CCCC  GENERATE MACROSCOPIC CROSS-SECTIONS
        DO 120 IORD=1,NORD+1
          DO 119 I=1,NGP*NGROUP
            SIGTAB(IORD,I)=0.0D00
            DO 118 NNISO=1,NISO
118           SIGTAB(IORD,I)=DEN(NNISO)*SIGMA(NNISO,IORD,I)+SIGTAB(IORD,I)
119       CONTINUE
120     CONTINUE
CCCC  ------------------------------------------------------------------
      DELTAZ=2*ZMAX/JMAX
      DELTAR=RMAX/IMAX
CCCC  COMPUTE LOCATION OF TOTAL CROSS SECION FOR GROUP GO
      NGO=(GO-1)*NGP+3
CCCC  COMPUTE THE FIRST SCATTER SOURCE FOR GROUP G DO TO A SOURCE IN GO
      DO 300 G=1,NGROUP
        IF(G.LT.GO)THEN
CCCC      NO UPSCATTER INTO HIGHER ENERGY GROUPS.  ALL MOMENTS ARE ZERO
          DO 9 J=1,JMAX
            DO 8 I=1,IMAX
              DO 7 M=1,MOMENT
7               Q(G,M,I,J)=0.D00
                FLUX(G,I,J)=0.D00
8             CONTINUE
9         CONTINUE
```

```
CCCC ------------------------------------------------------------------
          ELSE
            DO 50 J=1,JMAX
              WRITE(6,*)G,J
              DO 40 I=1,IMAX
                DO 35 M=1,MOMENT
35              Q(G,M,I,J)=0.D00
40            CONTINUE
              Z1=-ZMAX+DELTAZ*J
              IF(Z1.GT.0)THEN
                Z0=Z1-DELTAZ
CCCC            CORRECT FOR A CELL WHICH ENCOMPASSES THE ORIGIN
                IF(Z0.LT.0)Z0=0
CCCC            CALCULATE LOCATION IN CROSS SECTION TABLE OF THE
CCCC            SCATTERING CROSS SECTION FOR GROUP G0 TO GROUP G
                NG=(G-1)*NGP+4+G-G0
                DUMB=DEXP(-SIGTAB(1,NG0)*Z0)*(1-DEXP(-SIGTAB(1,NG0)*
     1                    DELTAZ))/(DELTAZ*PI*DELTAR*DELTAR*SIGTAB(1,NG0))
                Q(G,1,1,J)=SIGTAB(1,NG)*DUMB
                NMQ=2
                DO 76 IL=1,ORDER
                  Q(G,NMQ,1,J)=SIGTAB(IL+1,NG)*DUMB
                  NMQ=NMQ+1
                  DO 75 IM=1,IL
75                NMQ=NMQ+1
76                CONTINUE
              END IF
80            CONTINUE
50          CONTINUE
CCCC ------------------------------------------------------------------
          END IF
300     CONTINUE
CCCC    WRITE THE OUTPUT MOMENTS TO A FILE
        CALL OUTPUT(RMAX,ZMAX,IMAX,JMAX,OTFILE,G0,MOMENT)
        WRITE(6,*)'END OF PROGRAM RUN'
        STOP
        END
```

```
      SUBROUTINE RDXSCT(XSFILE)
CCCC  ------------------------------------------------------------------
CCCC  ROUTINE READS IN CROSS-SECTION DATA FROM FILE XSFILE
CCCC  DATA MUST BE IN FIXED FEILD FIDO FORMAT
CCCC  DATA IS ASSUMED TO BE IN FORM ABSORBTION FOR G, NU*SIGMA-FISSION
CCCC  FOR G, TOTAL FOR G, G TO G, G TO G-1, ETC, AND THEN REPEATE FOR
CCCC  EACH ENERGY.  ONCE ALL ENERGIES ARE COMPLETED END TABLE WITH T
CCCC  AND THE NEXT MOMENT IS INPUT.
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*8 XSFILE,OTFILE
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(2),SIGMA(3,9,10)
      DIMENSION N(6),ND(6),IWER(6)
      CHARACTER*8 R(6),SIGN(6)
      OPEN(4,FILE=XSFILE,STATUS='OLD',ACCESS='SEQUENTIAL')
      DO 50 ISO=1,NISO
        IORD=1
2       READ(4,*)NDUMB
        K=1
CCCC    READ IN THE DATA USING FIXED-FIELD FIDO
4       READ(4,999)(N(I),R(I),ND(I),SIGN(I),IWER(I),I=1,6)
        DO 10 I=1,6
        IF(R(I).EQ.'T')THEN
          IORD=IORD+1
          GOTO 20
        ELSE IF(R(I).EQ.'R')THEN
          DO 5 J=0,N(I)-1
            IF(SIGN(I).EQ.'-')THEN
              SIGMA(ISO,IORD,K+J)=ND(I)*(10.D00**(-IWER(I)))
            ELSE
              SIGMA(ISO,IORD,K+J)=ND(I)*(10.D00**(IWER(I)))
            END IF
5         CONTINUE
          K=K+N(I)
        ELSE
          IF(SIGN(I).EQ.'-')THEN
            SIGMA(ISO,IORD,K)=ND(I)*10.D00**(-IWER(I))
          ELSE
            SIGMA(ISO,IORD,K)=ND(I)*10.D00**(IWER(I))
          END IF
          K=K+1
        END IF
10      CONTINUE
        GOTO 4
20      IF(IORD.GT.NORD+1)GOTO 50
        GOTO 2
50    CONTINUE
      CLOSE(4)
      RETURN
999   FORMAT(6(I2,A1,I6,A1,I2))
      END
```

```
      SUBROUTINE INPUT(RMX,ZMX,IMAX,JMAX,INFILE,OTFILE,GO,MOMENT)
CCCC  ------------------------------------------------------------------
CCCC  ROUTINE PROMPTS USER FOR THE REQUIRED INPUT FOR THIS CODE
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER ORDER,G,GO,GP
      CHARACTER*8 INFILE,OTFILE,OTFLX
      COMMON/CNSTS/PI
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(9,235)
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(2),SIGMA(3,9,10)
      COMMON/BLK2/W(32),X(32)
CCCC  ------------------------------------------------------------------
      PI=DACOS(-1.D00)
      WRITE(6,*)'INPUT THE NUMBER OF MATERIALS'
      READ(5,*)NISO
      WRITE(6,*)'INPUT THE NUMBER OF ENERGY GROUPS'
      READ(5,*)NGROUP
      WRITE(6,*)'INPUT THE CROSS-SECTION LEGENGRE EXPANSION ORDER'
      READ(5,*)ORDER
      NORD=ORDER
      MOMENT=(ORDER+1)*(ORDER+2)/2
      WRITE(6,*)'INPUT THE DENSITY FOR EACH OF THE ISOTOPES'
      DO 10 I=1,NISO
      WRITE(6,*)'DENSITY OF MATERIAL',I
10    READ(5,*)DEN(I)
      WRITE(6,*)'INPUT THE NAME OF THE CROSS SECTION FILE'
      READ(5,999)INFILE
      WRITE(6,*)'INPUT THE RADIAL THICKNESS'
      READ(5,*)RMX
      WRITE(6,*)'INPUT THE NUMBER OF RADIAL MESHES'
      READ(5,*)IMAX
      WRITE(6,*)'INPUT THE Z THICKNESS (SOURCE WILL BE AT MIDDLE)'
      READ(5,*)ZMX
      ZMX=ZMX/2
      WRITE(6,*)'INPUT THE NUMBER OF Z MESHES'
      READ(5,*)JMAX
      WRITE(6,*)'INPUT THE SOURCE GROUP'
      READ(5,*)GO
      WRITE(6,*)'INPUT THE NAME OF OUTPUT FILE'
      READ(5,999)OTFILE
      RETURN
999   FORMAT(A)
      END
```

```
      SUBROUTINE OUTPUT(RMAX,ZMAX,IMAX,JMAX,OTFILE,GO,MOMENT)
CCCC  ------------------------------------------------------------------
CCCC  ROUTINE OUTPUTS THE SOURCE MOMENTS WHICH ARE REQUIRED BY
CCCC  TWODANT TO DESCRIBE THE FIRST COLLISION SOURCE
CCCC  THE FIRST COLLISION SOURCE IS ASSUMED TO OCCUR ONLY IN
CCCC  THOSE CELLS ALONG THE CENTERLINE
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER ORDER,G,GO,GP
      CHARACTER*8 XSFILE,OTFILE,OTFX,DELIM
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(9,235)
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(2),SIGMA(3,9,10)
      COMMON/SOURCE/Q(2,21,20,40),FLUX(2,20,40)
      COMMON/CNSTS/PI
      DIMENSION QDM(6)
CCCC  ------------------------------------------------------------------
      DELTAR=RMAX/IMAX
      DELTAZ=2*ZMAX/JMAX
      OPEN(3,FILE=OTFILE,STATUS='NEW',ACCESS='SEQUENTIAL')
      DO 40 NMQ=1,MOMENT
        DO 30 NG=1,NGROUP
          WRITE(3,979)NMQ,NG
          FSUM=0
          IF(Q(NG,NMQ,1,JMAX/2+1).NE.0.0D00)THEN
            JQ=1
            DO 20 JT=1,JMAX
              IF(JQ.GT.3)THEN
                WRITE(3,999)QDM(1),IMAX-1,QDM(2),IMAX-1,QDM(3),IMAX-1
                JQ=1
              END IF
              FSUM=FSUM+Q(NG,NMQ,1,JT)*0.5*DELTAR
              QDM(JQ)=Q(NG,NMQ,1,JT)
              JQ=JQ+1
20          CONTINUE
            DELIM=';'
            IF((NMQ.EQ.MOMENT).AND.(NG.EQ.NGROUP))DELIM='T'
            IF(JQ.EQ.1)THEN
              WRITE(3,993)DELIM
            ELSEIF(JQ.EQ.2)THEN
              WRITE(3,994)QDM(1),IMAX-1,DELIM
            ELSE
              WRITE(3,995)QDM(1),IMAX-1,QDM(2),IMAX-1,DELIM
            END IF
```

```
      ELSE
        JQ=0
        DO 120 JT=1,JMAX
          JQ=JQ+1
          IF(JQ.GT.12)THEN
             WRITE(3,992)IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,
     1                         IMAX,IMAX,IMAX,IMAX
             JQ=1
          END IF
120       CONTINUE
        DELIM=';'
        IF((NMQ.EQ.MOMENT).AND.(NG.EQ.NGROUP))DELIM='T'
        IF(JQ.EQ.1)THEN
          WRITE(3,991)IMAX,DELIM
        ELSEIF(JQ.EQ.2)THEN
          WRITE(3,990)IMAX,IMAX,DELIM
        ELSEIF(JQ.EQ.3)THEN
          WRITE(3,989)IMAX,IMAX,IMAX,DELIM
        ELSEIF(JQ.EQ.4)THEN
          WRITE(3,988)IMAX,IMAX,IMAX,IMAX,DELIM
        ELSEIF(JQ.EQ.5)THEN
          WRITE(3,987)IMAX,IMAX,IMAX,IMAX,IMAX,DELIM
        ELSEIF(JQ.EQ.6)THEN
          WRITE(3,986)IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,DELIM
        ELSEIF(JQ.EQ.7)THEN
          WRITE(3,985)IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,DELIM
        ELSEIF(JQ.EQ.8)THEN
          WRITE(3,984)IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,IMAX
     1                ,DELIM
        ELSEIF(JQ.EQ.9)THEN
          WRITE(3,983)IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,
     1                IMAX,DELIM
        ELSEIF(JQ.EQ.10)THEN
          WRITE(3,982)IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,
     1                IMAX,IMAX,DELIM
        ELSEIF(JQ.EQ.11)THEN
          WRITE(3,981)IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,
     1                IMAX,IMAX,IMAX,DELIM
        ELSE
          WRITE(3,980)IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,IMAX,
     1                IMAX,IMAX,IMAX,IMAX,DELIM
        END IF
      END IF
      FSUM=FSUM*DELTAR*DELTAZ*2*PI
      WRITE(3,*)'/ INTEGRATED SOURCE TERM = ',FSUM
30    CONTINUE
40    CONTINUE
      CLOSE(3)
      RETURN
```

```
979   FORMAT('/ SOURCE TERM FOR SPHERICAL HARMONIC ',I2,
     1             ' AND ENERGY GROUP ',I2)
980   FORMAT(11(I3,'Z; '),I3,'Z ',A)
981   FORMAT(10(I3,'Z; '),I3,'Z ',A)
982   FORMAT(9(I3,'Z; '),I3,'Z ',A)
983   FORMAT(8(I3,'Z; '),I3,'Z ',A)
984   FORMAT(7(I3,'Z; '),I3,'Z ',A)
985   FORMAT(6(I3,'Z; '),I3,'Z ',A)
986   FORMAT(5(I3,'Z; '),I3,'Z ',A)
987   FORMAT(4(I3,'Z; '),I3,'Z ',A)
988   FORMAT(3(I3,'Z; '),I3,'Z ',A)
989   FORMAT(2(I3,'Z; '),I3,'Z ',A)
990   FORMAT(1(I3,'Z; '),I3,'Z ',A)
991   FORMAT(I3,'Z ',A)
992   FORMAT(12(I3,'Z; '))
993   FORMAT('    ',A)
994   FORMAT(E12.5,I3,'Z ',A)
995   FORMAT(E12.5,I3,'Z; ',E12.5,I3,'Z ',A)
999   FORMAT(3(E12.5,I3,'Z; '))
      END
```

APPENDIX B: The Computer Program SRC2D

```
CCCC  PROGRAM READS IN CROSS SECTION DATA AND COMPUTES THE
CCCC  SPHERICAL HARMONICS OF THE SECOND COLLISION SOURCE FOR A POINT
CCCC  SOURCE EMITTING NEUTRONS IN THE UPWARD DIRECTION.  THE ORIGINAL
CCCC  POINT SOURCE IS NORMALIZED TO 4 PI.  THE 4 PI IS USED
CCCC  TO FACILITATE A SOURCE NORMALIZATION OF ONE IN TWODANT
CCCC  THE POINT SOURCE IS ASSUMED TO BE AT THE CENTER OF A CYLINDER
CCCC  WITH A RADIUS OF RMAX AND A HALF-HEIGHT OF ZMAX
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER ORDER,G,GO,GP
      CHARACTER*8 XSFILE,OTSRC,OTFLX
      COMMON/CNSTS/PI
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(3),SIGMA(3,6,18)
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(9,18)
      COMMON/BLK2/W(32),X(32)
      DIMENSION APOLY(0:21),SUM(22)
      DIMENSION ROUT(3)
      COMMON/SOURCE/QTOT(3,21,20,40),FLUX(20,40),QPART(21,20,40)
      COMMON/FILES/OTSRC,OTFLX,XSFILE
CCCC  ------------------------------------------------------------------
CCCC  GET INPUT DATA REQUIRED FOR PROGRAM RUN
1     CALL INPUT(RMAX,ZMAX,IMAX,JMAX,NQUAD,GO,MOMENT,IXSECT,IOPT)
      IFLXOP=0
      NGP=NGROUP+3
CCCC  READ IN CROSS-SECTION DATA
      CALL RDXSCT(XSFILE)
CCCC  ------------------------------------------------------------------
CCCC  MIX CROSS-SECTIONS TO FORM MACROSCOPIC CROSS-SECTION
      DO 210 IORD=1,NORD+1
        DO 119 I=1,NGP*NGROUP
          SIGTAB(IORD,I)=0.0D00
          DO 118 NNISO=1,NISO
118       SIGTAB(IORD,I)=DEN(NNISO)*SIGMA(NNISO,IORD,I)+SIGTAB(IORD,I)
119     CONTINUE
210   CONTINUE
CCCC  ------------------------------------------------------------------
      IF(IXSECT.EQ.2)THEN
CCCC    CHANGE CROSS-SECTION COEFFICIENTS TO REFLECT THE N'TH CESARO
CCCC    MEAN OF ORDER 2.  TECHNIQUE HELPS ALLEVIATE NEGATIVES IN
CCCC    SCATTERING CROSS-SECTIONS
        IOPT=2
        DO 220 IORD=2,NORD+1
          CNST1=(NORD+3-IORD)*(NORD+2-IORD)
          CNST2=(NORD+2)*(NORD+1)
          CNST=CNST1/CNST2
          DO 219 I=1,NGP*NGROUP
            DO 218 NNISO=1,NISO
218         SIGTAB(IORD,I)=SIGTAB(IORD,I)*CNST
219       CONTINUE
220     CONTINUE
```

```
      END IF
CCCC  ------------------------------------------------------------------
CCCC  BEGIN CALCULATION OF SECOND COLLISION SOURCE MOMENTS AND
CCCC  ONCE SCATTERED FLUX DENSITY
CCCC  COMPUTE CELL SIZES
      DELTAZ=2*ZMAX/JMAX
      DELTAR=RMAX/IMAX
CCCC  COMPUTE SOURCE MOMENTS FOR GROUP G
      DO 300 G=1,NGROUP
        DO 9 J=1,JMAX
          DO 8 I=1,IMAX
CCCC        ZERO ALL SOURCE MOMENTS
            DO 7 M=1,MOMENT
7           QTOT(G,M,I,J)=0.D00
            FLUX(I,J)=0.D00
8         CONTINUE
9       CONTINUE
CCCC  ------------------------------------------------------------------
        IF(G.LT.GO)THEN
CCCC      ALL SOURCE MOMENTS REMAIN AS ZERO AND THE FLUX FOR THIS
CCCC      GROUP IS ZERO SINCE NEUTRONS DO NOT UPSCATTER BETWEEN GROUPS
CCCC      DUMP THE SCALAR FLUX TO AN OUTPUT FILE
          CALL OUTFLX(RMAX,ZMAX,IMAX,JMAX,IFLXOP,G,OTFLX)
        ELSE
CCCC      COMPUTE THE SOURCE CONTRIBUTION TO GROUP G FROM THE GROUP GP
CCCC      I.E., THE PARTICLE BEGINS IN GROUP GO SCATTERS GP AND THEN
CCCC      SCATTERS INTO GROUP G
          DO 60 GP=GO,G
            WRITE(6,*)GP,G
            DO 50 J=1,JMAX
              ZSPOT=-ZMAX+DELTAZ*(J-0.5)
              DO 40 I=1,IMAX
                RSPOT=(I-0.5)*DELTAR
CCCC            COMPUTE THE SOURCE MOMENTS AT AN I,J CORRESPONDING TO
CCCC            THE LOCATION RSPOT,ZSPOT
                CALL PTSRC(RSPOT,ZSPOT,ZMAX,GO,GP,G,NQUAD,MOMENT,IXSECT
     1                    ,SUM,XSFILE)
CCCC            ASSIGN THE COMPUTED MOMENTS TO THE APPROPRIATE PARTIAL
CCCC            SOURCE CELL I,J
                DO 30 M=1,MOMENT
30              QPART(M,I,J)=SUM(M)
40            CONTINUE
50          CONTINUE
CCCC        ------------------------------------------------------------
            IF(IOPT.NE.0)THEN
CCCC          COMPLETE NEGATIVE SCALAR SOURCE FIXUP ON THE PARITIAL
CCCC          SOURCE QPART(M,I,J)
              IOPT1=1
              IF((GP.EQ.GO).AND.(IOPT.EQ.2))IOPT1=2
              CALL NEGFIX(IMAX,JMAX,RMAX,ZMAX,MOMENT,IOPT1)
            END IF
```

```
CCCC  ------------------------------------------------------------------
CCCC        COMPUTE THE FIRST COLLIDED SCALAR FLUX DENSITY (1/4PI)
            IF(GP.EQ.G)THEN
              DO 56 J=1,JMAX
                DO 55 I=1,IMAX
55                FLUX(I,J)=QPART(1,I,J)/SIGTAB(1,(G-1)*NGP+4)
56            CONTINUE
CCCC          OUTPUT THE ONCE SCATTERED FLUX DENSITY
              CALL OUTFLX(RMAX,ZMAX,IMAX,JMAX,IFLXOP,G,OTFLX)
            END IF
CCCC  ------------------------------------------------------------------
CCCC        ADD THE PARTIAL SOURCES TO GROUP G FROM GROUP GP TO THE
CCCC        TOTAL SOURCE FOR ALL CELLS AND MOMENTS
            DO 59 M=1,MOMENT
              DO 58 J=1,JMAX
                DO 57 I=1,IMAX
57                QTOT(G,M,I,J)=QTOT(G,M,I,J)+QPART(M,I,J)
58            CONTINUE
59          CONTINUE
60        CONTINUE
CCCC  ------------------------------------------------------------------
          END IF
CCCC      REPEAT THE ABOVE PROCEDURE TO FIND CONTRIBUTION TO GROUP G
CCCC      FROM GROUP GP+1
300     CONTINUE
CCCC    CLOSE THE FLUX OUTPUT FILE AFTER ALL CALCULATIONS ARE COMPLETE
        CLOSE(2)
CCCC    WRITE SECOND COLLISION SOURCE OUPUT MOMENTS TO OUTPUT FILE
        CALL OUTSRC(RMAX,ZMAX,IMAX,JMAX,NQUAD,GO,MOMENT,OTSRC)
        WRITE(6,*)'END OF PROGRAM RUN'
        WRITE(6,*)
        WRITE(6,*)'PERFORM ANOTHER PROGRAM RUN (1=YES)'
        READ(5,*)IRUN
        IF(IRUN.EQ.1)GOTO 1
        END
```

```
      SUBRDUTINE NEGFIX(IMAX,JMAX,RMAX,ZMAX,MOMENT,IOPT1)
CCCC  ------------------------------------------------------------------
CCCC  SUBROUTINE SETS TO ZERO ALL NEGATIVE SCALAR SOURCE TERMS AND
CCCC  ASSOCIATED MOMETS FOUND IN QPART(M,I,J).
CCCC  TWD TECHNIQUES USED: 1) ELIMINATE JUST THE NEGATIVES
CCCC                       2) ELIMINATE NEGATIVES AND BACKPEAK
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,0-Z)
      INTEGER G
      COMMON/CNSTS/PI
      COMMON/SOURCE/QTOT(3,21,20,40),FLUX(20,40),QPART(21,20,40)
CCCC  ------------------------------------------------------------------
      DELTAR=RMAX/IMAX
      DELTAZ=2*ZMAX/JMAX
      FSUMP=0
      FSUMO=0
CCCC  ------------------------------------------------------------------
      IF(IOPT1.EQ.1)THEN
CCCC    ELIMINATE ONLY THE NEGATIVE SCALAR SOURCES AND ASSOCIATED MOMENTS
        DD 40 I=1,IMAX
          RSPOT=(I-0.5)*DELTAR
          DO 30 J=1,JMAX
CCCC        INTEGRATE THE POSITIVE PDRTION DF SCALAR SOURCE
            FSUMP=FSUMP+QPART(1,I,J)*RSPOT
            IF(QPART(1,I,J).LT.0)THEN
CCCC          INTEGRATE THE NEGATIVE PORTION DF SCALAR SOURCE
              FSUMO=FSUMO+QPART(1,I,J)*RSPOT
CCCC          SET THE NEGATIVE MOMENT AND HIGHER MDMENTS TO ZERD
              DO 20 NMQ=1,MOMENT
20            QPART(NMQ,I,J)=0.0D00
            END IF
30        CONTINUE
40      CONTINUE
      ELSE
CCCC  ------------------------------------------------------------------
CCCC    ELIMINATE THE NEGATIVE SCALAR SOURCES AND ANY BACK-PEAKS
CCCC    CAUSED BY OSCILLATIONS IN SCATTERING CROSS-SECTIONS
CCCC    THE TECHNIQUE IS USED ONLY FDR THE SECOND SCATTER
CCCC    COMPONENT FROM THE SOURCE GROUP FDR EACH GROUP. GO TO GO TO G
CCCC  ------------------------------------------------------------------
CCCC    FIND AND FIX THE PEAK REGION ALONG THE TOP EDGE OF PROBLEM
        DO 115 I=1,IMAX
          IF(QPART(1,I,JMAX).LT.0.0D00)THEN
CCCC        INTEGRATE NON-PEAK REGION ALONG TOP
            DD 110 II=I,IMAX
110         FSUMO=FSUMO+QPART(1,I,JMAX)*(II-0.5)*DELTAR
CCCC        SET TD ZERO NON-PEAK REGION
            QPART(1,I,JMAX)=0.0D00
          END IF
115     CONTINUE
```

```
CCCC ------------------------------------------------------------------
CCCC    FIND AND FIX PEAK REGION IN REMAINDER OF PROBLEM
        DO 250 I=1,IMAX
          RSPOT=(I-0.5)*DELTAR
          DO 220 J=JMAX,1,-1
CCCC        INTEGRATE PEAK REGION
            FSUMP=FSUMP+QPART(1,I,J)*RSPOT
            IF(QPART(1,I,J).LE.0)THEN
              DO 215 JJ=J,1,-1
CCCC            INTEGRATE THE NON-PEAK REGION
                FSUMO=FSUMO+QPART(1,I,JJ)*RSPOT
                DO 210 NMQ=1,MOMENT
210             QPART(NMQ,I,JJ)=0.0D00
215           CONTINUE
              GOTO 250
            END IF
220       CONTINUE
250     CONTINUE
      END IF
CCCC ------------------------------------------------------------------
CCCC  RENORMALIZE THE SOURCE TO REGAIN PARTICLE BALANCE
      FSUMP=FSUMP*DELTAR*DELTAZ*2*PI
      FSUMO=FSUMO*DELTAR*DELTAZ*2*PI
      FSUMO=FSUMP-FSUMO
      RENORM=FSUMP/FSUMO
      DO 380 I=1,IMAX
        DO 370 J=1,JMAX
          DO 360 NMQ=1,MOMENT
360       QPART(NMQ,I,J)=QPART(NMQ,I,J)*RENORM
370     CONTINUE
380   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE RDXSCT(XSFILE)
CCCC  ------------------------------------------------------------------
CCCC  ROUTINE READS IN CROSS-SECTION DATA FROM FILE XSFILE
CCCC  DATA MUST BE IN FIXED FEILD FIDO FORMAT
CCCC  DATA IS ASSUMED TO BE IN FORM ABSORBTION FOR G, NU*SIGMA-FISSION
CCCC  FOR G, TOTAL FOR G, G TO G, G TO G-1, ETC, AND THEN REPEATE FOR
CCCC  EACH ENERGY.  ONCE ALL ENERGIES ARE COMPLETED TABLE END WITH T
CCCC  AND THE NEXT MOMENT IS INPUT.
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*8 XSFILE
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(3),SIGMA(3,6,18)
      DIMENSION N(6),ND(6),IWER(6)
      CHARACTER*8 R(6),SIGN(6)
      OPEN(4,FILE=XSFILE,STATUS='OLD',ACCESS='SEQUENTIAL')
      DO 50 ISO=1,NISO
        IORD=1
2       READ(4,*)NDUMB
        K=1
4       READ(4,999)(N(I),R(I),ND(I),SIGN(I),IWER(I),I=1,6)
        DO 10 I=1,6
        IF(R(I).EQ.'T')THEN
          IORD=IORD+1
          GOTO 20
        ELSE IF(R(I).EQ.'R')THEN
          DO 5 J=0,N(I)-1
            IF(SIGN(I).EQ.'-')THEN
              SIGMA(ISO,IORD,K+J)=ND(I)*(10.D00**(-IWER(I)))
            ELSE
              SIGMA(ISO,IORD,K+J)=ND(I)*(10.D00**(IWER(I)))
            END IF
5         CONTINUE
          K=K+N(I)
        ELSE
          IF(SIGN(I).EQ.'-')THEN
            SIGMA(ISO,IORD,K)=ND(I)*10.D00**(-IWER(I))
          ELSE
            SIGMA(ISO,IORD,K)=ND(I)*10.D00**(IWER(I))
          END IF
          K=K+1
        END IF
10      CONTINUE
        GOTO 4
20      IF(IORD.GT.NORD+1)GOTO 50
        GOTO 2
50    CONTINUE
      CLOSE(4)
      RETURN
999   FORMAT(6(I2,A1,I6,A1,I2))
      END
```

```
      SUBROUTINE LGDRE(ITYPE,OMEGAR,NDUMB,APOLY)
CCCC  ------------------------------------------------------------------
CCCC  ROUTINE RETURNS EITHER THE VALUE OF THE LEGENDRE POLYNOMIALS AT
CCCC  OMEGAR OR EVALUATES THE SPHERICAL HARMONICS AT OMEGAR
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION APOLY(0:21)
CCCC  ------------------------------------------------------------------
      IF(ITYPE.EQ.1)THEN
CCCC    APOLY(I) IS THE EVALUATION OF THE I TH LEGENDRE POLYNOMIAL
        APOLY(0)=1
        APOLY(1)=OMEGAR
        DO 10 IP=1,NDUMB-1
10      APOLY(IP+1)=((2*IP+1)*OMEGAR*APOLY(IP)-IP*APOLY(IP-1))/(IP+1)
CCCC  ------------------------------------------------------------------
      ELSE
CCCC    APOLY(I) IS THE EVALUATION OF THE APPROPRIATE
CCCC    SPHERICAL HARMONIC OF THE SOURCE
        APOLY(0)=1
        I=1
        DO 50 L=1,NDUMB
          APOLY(I)=PLGNDR(L,0,OMEGAR)
          I=I+1
          DO 40 M=1,L
            APOLY(I)=PLGNDR(L,M,OMEGAR)*DSQRT(2*FACT(L-M)/FACT(L+M))
40          I=I+1
50      CONTINUE
CCCC  ------------------------------------------------------------------
      END IF
      RETURN
      END
```

```
      SUBROUTINE XSECT(VAL,OMEGAR,GO,GP,IXSECT,XSFILE)
CCCC  ------------------------------------------------------------------
CCCC  RETURNS THE VALUE OF THE SCATTERING CROSS-SECTION IN VAL
CCCC  FOR THE GROUP GO TO THE GROUP GP FOR THE COSINE OMEGAR
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*8 XSFILE
      INTEGER ORDER,G,GO,GP
      COMMON/CNSTS/PI
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(3),SIGMA(3,6,18)
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(9,18)
      DIMENSION APOLY(0:21)
CCCC  ------------------------------------------------------------------
CCCC  GET THE LEGENDRE MOMENTS EVALUTED AT OMEGAR
      ITYPE=1
      CALL LGDRE(ITYPE,OMEGAR,ORDER,APOLY)
CCCC  DETERMINE LOCATION OF SCATTERING CROSS-SECTION
      NLOCI=(GP-1)*NGP+4+GP-GO
      VAL=0
      IF(IXSECT.LT.4)THEN
CCCC    USE THE LEGENDRE EXPANSION
        DO 9 IVAL=0,ORDER
9       VAL=VAL+APOLY(IVAL)*SIGTAB(IVAL+1,NLOCI)*(2*IVAL+1)
        VAL=VAL*0.5
      ELSE
CCCC    USE EXACT CROSS-SECTION TECHNIQUE (VALID ONLY FOR ONE GROUP)
CCCC    REQUIRES MODIFICATION OF SOURCE ROUTINE TO DESCRIBE THE
CCCC    SCATTERING MODEL
        IF(XSFILE(5:5).EQ.'1')THEN
          VAL=0.25
        ELSEIF(XSFILE(5:5).EQ.'2')THEN
          VAL=0
          IF(OMEGAR.GT.0)VAL=0.5*OMEGAR
        ELSEIF(XSFILE(5:5).EQ.'3')THEN
          VAL=0
          IF(OMEGAR.GT.0.5)VAL=2*OMEGAR-1
        ELSEIF(XSFILE(5:5).EQ.'4')THEN
          VAL=0
          IF((OMEGAR.GT.-0.5).AND.(OMEGAR.LT.0.5))VAL=0.5-DABS(OMEGAR)
        ELSEIF(XSFILE(5:5).EQ.'5')THEN
          VAL=0
          IF(OMEGAR.LT.-0.5)VAL=-2*OMEGAR-0.5
        ELSE
          WRITE(6,*)'REWRITE XSECT SUBROUTINE FOR ALTERNATE CASES'
          STOP
        END IF
        VAL=VAL*DEN(1)
      END IF
      VAL=VAL/(2*PI)
      RETURN
      END
```

```
      SUBROUTINE PTSRC(RSPOT,ZSPOT,ZMAX,GO,GP,G,NQUAD,MOMENT,IXSECT,
     1                 SUM,XSFILE)
CCCC  ------------------------------------------------------------------
CCCC  ROUTINE COMPUTES THE MOMENTS OF THE SECOND COLLISION SOURCE FOR
CCCC  GROUP G AT LOCATION RSPOT,ZSPOT DO TO A MONO-DIRECTIONAL POINT
CCCC  SOURCE IN GROUP GO WHERE THE FIRST SCATTER PUTS THE PARTICLES
CCCC  IN GROUP GP.  THE FIRST SCATTER LIES ALONG A LINE OF LENGTH
CCCC  ZMAX WHICH IS THEN INTEGRATED OVER WITH A GAUSSIAN QUADRATURE
CCCC  OF ORDER NQUAD.  THE MOMENTS ARE RETURNED IN THE ARRAY SUM() AND
CCCC  THERE ARE "MOMENT" NUMBER OF THESE MOMENTS.  THE CROSS-SECTION
CCCC  MODEL IS BASED ON IXSECT AND XSFILE.
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*8 XSFILE
      INTEGER ORDER,G,GO,GP
      COMMON/CNSTS/PI
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(9,18)
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(3),SIGMA(3,6,18)
      DIMENSION APOLY(0:21),SUM(22)
      COMMON/BLK2/W(32),X(32)
CCCC  ------------------------------------------------------------------
CCCC  CALCULATE LOCATION IN CROSS-SECTION TABLE FOR SCATTERING GP TO G
      NG=(G-1)*NGP+4+G-GP
      DO 5 I=1,MOMENT
5     SUM(I)=0
CCCC  ------------------------------------------------------------------
CCCC  IF SCATTERING IS POSSIBLE THEN COMPUTE MOMENTS
      IF(SIGTAB(1,NG).NE.0)THEN
CCCC    CALCULATE RANGES OF INTEGRATION
        XMAX=DATAN((ZMAX-ZSPOT)/RSPOT)
        XMIN=DATAN(-ZSPOT/RSPOT)
        D1=(XMAX-XMIN)*0.5
        D2=(XMAX+XMIN)*0.5
CCCC    COMPUTE LOCATIONS OF TOTAL CROSS-SECTION FOR GROUPS GO & GP
        NU=(GO-1)*NGP+3
        NL=(GP-1)*NGP+3
CCCC    INTEGRATE THE FIRST SCATTER LINE SOURCE
        DO 20 N=1,NQUAD
          THETA=D1*X(N)+D2
          OMEGAR=-DSIN(THETA)
          CALL XSECT(VAL,OMEGAR,GO,GP,IXSECT,XSFILE)
          CALL LGDRE(2,OMEGAR,ORDER,APOLY)
          DUMB=W(N)*VAL*DEXP(-RSPOT*
     1           (SIGTAB(1,NU)*DTAN(THETA)+SIGTAB(1,NL)/DCOS(THETA)))
          DO 10 M=1,MOMENT
10          SUM(M)=SUM(M)+APOLY(M-1)*DUMB
20      CONTINUE
        DUMB=D1*DEXP(-SIGTAB(1,NU)*ZSPOT)/RSPOT
```

```
          NMQ=1
          DO 40 L=0,ORDER
            DO 30 M=0,L
            SUM(NMQ)=SUM(NMQ)*DUMB*SIGTAB(L+1,NG)
30          NMQ=NMQ+1
40        CONTINUE
        END IF
        RETURN
        END



        DOUBLE PRECISION FUNCTION FACT(NUM)
CCCC    FUNCTION RETURNS FACTORIAL OF NUM
        IMPLICIT REAL*8(A-H,O-Z)
        DUMB=1
        DO 10 I=1,NUM
10      DUMB=DUMB*I
        FACT=DUMB
        RETURN
        END
```

```
      SUBROUTINE INPUT(RMX,ZMX,IMAX,JMAX,NQUAD,GO,MOMENT,IXSECT,IOPT)
CCCC  ------------------------------------------------------------------
CCCC  ROUTINE PROMPTS USER FOR PROGRAM DATA
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER ORDER,G,GO,GP
      CHARACTER*8 XSFILE,OTSRC,OTFLX
      COMMON/CNSTS/PI
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(9,18)
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(3),SIGMA(3,6,18)
      COMMON/BLK2/W(32),X(32)
      COMMON/FILES/OTSRC,OTFLX,XSFILE
CCCC  ------------------------------------------------------------------
      PI=DACOS(-1.D00)
      WRITE(6,*)'INPUT THE NUMBER OF MATERIALS'
      READ(5,*)NISO
      WRITE(6,*)'INPUT THE NUMBER OF ENERGY GROUPS'
      READ(5,*)NGROUP
      WRITE(6,*)'INPUT THE CROSS-SECTION LEGENGRE EXPANSION ORDER'
      READ(5,*)ORDER
      NORD=ORDER
      MOMENT=(ORDER+2)*(ORDER+1)/2
      WRITE(6,*)'INPUT THE DENSITY FOR EACH (ATOM/CM-BARN)'
      DO 10 I=1,NISO
      WRITE(6,*)'DENSITY OF MATERIAL',I
10    READ(5,*)DEN(I)
      WRITE(6,*)'INPUT THE NAME OF THE CROSS SECTION FILE'
      READ(5,999)XSFILE
11    WRITE(6,*)'INPUT THE CROSS SECTION MODEL OR TECHNIQUE'
      WRITE(6,*)'  (1) LEGENDRE EXPANSION '
      WRITE(6,*)'  (2) CESARO MEAN '
      WRITE(6,*)'  (3) LEGENDRE EXPANSION BUT ZERO NEGATIVE SOURCES'
      WRITE(6,*)'  (4) EXACT '
      READ(5,*)IXSECT
      IOPT=0
      IF((IXSECT.EQ.4).AND.(NGROUP.NE.1))THEN
        WRITE(6,*)'IMPROPER CHOICE FOR CROSS-SECTION MODEL '
        WRITE(6,*)'TOO MANY GROUPS FOR EXACT TECHINQUE '
        GOTO 11
      END IF
      IF(IXSECT.EQ.3)THEN
        WRITE(6,*)'INPUT THE ZEROING MODEL TO USE '
        WRITE(6,*)'  (1) ZERO NEGATIVE SOURCES IN ALL GROUPS '
        WRITE(6,*)'  (2) ZERO NEGATIVE SOURCES IN ALL GROUPS AND '
        WRITE(6,*)'      ELIMINATE BACK-PEAKS FROM SOURCE GROUP'
        READ(5,*)IOPT
      END IF
```

```
      WRITE(6,*)'INPUT THE NAME OF SOURCE OUTPUT FILE'
      READ(5,999)OTSRC
      WRITE(6,*)'INPUT THE NAME OF 1ST SCATTER FLUX FILE'
      READ(5,999)OTFLX
      IF(IXSECT.EQ.3)IXSECT=1
      WRITE(6,*)'INPUT THE RADIAL THICKNESS (CM)'
      READ(5,*)RMX
      WRITE(6,*)'INPUT THE NUMBER OF RADIAL MESHES'
      READ(5,*)IMAX
      WRITE(6,*)'INPUT THE Z THICKNESS (SOURCE WILL BE AT MIDDLE) (CM)'
      READ(5,*)ZMX
      ZMX=ZMX/2
      WRITE(6,*)'INPUT THE NUMBER OF Z MESHES'
      READ(5,*)JMAX
      WRITE(6,*)'INPUT THE QUADRATURE INTEGRATION ORDER'
      READ(5,*)NQUAD
CCCC  READ IN THE QUADRATUE SET USED FOR INTEGRATION
      CALL GETQAD(NQUAD)
      WRITE(6,*)'INPUT THE SOURCE GROUP'
      READ(5,*)GO
      RETURN
999   FORMAT(A)
      END
```

```
      SUBROUTINE GETQAD(NQUAD)
CCCC  ------------------------------------------------------------------
CCCC  SUBROUTINE READS GAUSS QUADRATURE DATA FROM AN INPUT FILE
CCCC  THE FIRST NUMHER IS THE ORDER OF GAUSSIAN QUADRATURE SET.
CCCC  THE ABCISSAS AND WEIGHTS SHOULD BE ORDERED FROM NEGATIVE TO
CCCC  POSITIVE VALUES.
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*8,OIN
      COMMON/BLK2/W(32),X(32)
306   FORMAT(A)
      WRITE(6,*)'INPUT THE NAME OF THE QUADRATURE FILE'
      READ(5,306)OIN
      OPEN(9,FILE=OIN,STATUS='OLD',ACCESS='SEQUENTIAL')
      DO 10 I=1,NQUAD
10    READ(9,*)X(I),W(I)
      CLOSE(9)
      RETURN
      END
```

```
      SUBROUTINE OUTSRC(RMAX,ZMAX,IMAX,JMAX,NQUAD,GO,MOMENT,OTSRC)
CCCC  ------------------------------------------------------------------
CCCC  SUBROUTINE OUTPUTS THE SECOND COLLISION SOURCE MOMENTS IN A
CCCC  FORM USED BY TWODANT (EXCEPT ALL EXPONENTIAL LETTERS SHOULD
CCCC  BE ELIMINATED IN THE GENERATED FILE FOR ACTUAL USE IN TWODANT)
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER ORDER,G,GO,GP
      CHARACTER*8 OTSRC,OFLX,DELIM
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(9,18)
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(3),SIGMA(3,6,18)
      COMMON/SOURCE/QTOT(3,21,20,40),FLUX(20,40),QPART(21,20,40)
      COMMON/CNSTS/PI
      DIMENSION QDM(6)
      DIMENSION ROUT(3)
CCCC  ------------------------------------------------------------------
      WRITE(6,975)OTSRC
      DELTAR=RMAX/IMAX
      DELTAZ=2*ZMAX/JMAX
      OPEN(3,FILE=OTSRC,STATUS='NEW',ACCESS='SEQUENTIAL')
      DO 40 NMQ=1,MOMENT
        DO 30 NG=1,NGROUP
          WRITE(3,979)NMQ,NG
          FSUM=0
CCCC      ------------------------------------------------------------------
          DO 20 JT=1,JMAX
            IQ=0
            DO 10 IT=1,IMAX
              RSPOT=(IT-0.5)*DELTAR
              IQ=IQ+1
              IF(IQ.GT.6)THEN
                WRITE(3,999)QDM(1),QDM(2),QDM(3),QDM(4),QDM(5),QDM(6)
                IQ=1
              END IF
              FSUM=FSUM+QTOT(NG,NMQ,IT,JT)*RSPOT
10          QDM(IQ)=QTOT(NG,NMQ,IT,JT)
            DELIM=';'
            IF(((JT.EQ.JMAX).AND.(NG.EQ.NGROUP).AND.
     1           (NMQ.EQ.MOMENT)))DELIM='T'
```

```
            IF(IQ.EQ.1)THEN
              WRITE(3,993)(QDM(IQM),IQM=1,IQ),DELIM
            ELSEIF(IQ.EQ.2)THEN
              WRITE(3,994)(QDM(IQM),IQM=1,IQ),DELIM
            ELSEIF(IQ.EQ.3)THEN
              WRITE(3,995)(QDM(IQM),IQM=1,IQ),DELIM
            ELSEIF(IQ.EQ.4)THEN
              WRITE(3,996)(QDM(IQM),IQM=1,IQ),DELIM
            ELSEIF(IQ.EQ.5)THEN
              WRITE(3,997)(QDM(IQM),IQM=1,IQ),DELIM
            ELSE
              WRITE(3,998)(QDM(IQM),IQM=1,IQ),DELIM
            END IF
20        CONTINUE
CCCC      ------------------------------------------------------------
          FSUM=FSUM*DELTAR*DELTAZ*2*PI
          WRITE(3,980)FSUM
30      CONTINUE
40    CONTINUE
      CLOSE(3)
      RETURN
975   FORMAT('WRITING SOURCE OUTPUT FILE ',A8)
979   FORMAT('/ SOURCE TERM FOR SPHERICAL HARMONIC ',I2,
     1                ' OF ENERGY GROUP ',I2)
980   FORMAT('/ INTEGRAL SOURCE TERM = ')
993   FORMAT(' ',1(E12.5,' '),A)
994   FORMAT(' ',2(E12.5,' '),A)
995   FORMAT(' ',3(E12.5,' '),A)
996   FORMAT(' ',' ',4(E12.5,' '),A)
997   FORMAT(' ',5(E12.5,' '),A)
998   FORMAT(' ',6(E12.5,' '),A)
999   FORMAT(' ',6(E12.5,' '))
      END
```

```
      DOUBLE PRECISION FUNCTION PLGNDR(L,M,X)
CCCC  ----------------------------------------------------------------
CCCC  ROUTINE COMPUTES THE ASSOCIATED LEGENDRE POLYNOMIALS AT X
CCCC  ----------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      IF(M.LT.0.OR.M.GT.L.OR.ABS(X).GT.1)PAUSE 'BAD ARGUMENTS'
      PMM=1
      IF(M.GT.0) THEN
        SOMX2=DSQRT((1.-X)*(1.+X))
        FACT=1
        DO 11 I=1,M
          PMM=-PMM*FACT*SOMX2
          FACT=FACT+2
11      CONTINUE
      END IF
      IF(L.EQ.M)THEN
        PLGNDR=PMM
      ELSE
        PMMP1=X*(2*M+1)*PMM
        IF(L.EQ.M+1)THEN
          PLGNDR=PMMP1
        ELSE
          DO 12 LL=M+2,L
            PLL=(X*(2*LL-1)*PMMP1-(LL+M-1)*PMM)/(LL-M)
            PMM=PMMP1
            PMMP1=PLL
12        CONTINUE
          PLGNDR=PLL
        END IF
      END IF
      RETURN
      END
```

```
      SUBROUTINE OUTFLX(RMAX,ZMAX,IMAX,JMAX,IFLXOP,G,OFLX)
CCCC  ------------------------------------------------------------------
CCCC  SUBROUTINE OUTPUTS THE FIRST COLLISION FLUX DENSITY
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER ORDER,G,GO,GP
      CHARACTER*8 OSRC,OFLX,DELIM
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(9,18)
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(3),SIGMA(3,6,18)
      COMMON/SOURCE/QTOT(3,21,20,40),FLUX(20,40),QPART(21,20,40)
      COMMON/CNSTS/PI
      DIMENSION QDM(6)
      DIMENSION ROUT(3)
CCCC  ------------------------------------------------------------------
      DELTAR=RMAX/IMAX
      DELTAZ=2*ZMAX/JMAX
CCCC  FIRST TIME THROUGH, OPEN THE FLUX OUPUT FILE
      IF(IFLXOP.NE.1)THEN
        OPEN(2,FILE=OFLX,STATUS='NEW',ACCESS='SEQUENTIAL')
        WRITE(2,*)'/ SCALAR FLUX DENSITIES NORMALIZED BY 4*PI '
        IFLXOP=1
      END IF
      WRITE(2,*)'/ SCALAR FLUX DENSITY FOR ENERGY GROUP',G
      FSUM=0
      DO 60 JT=1,JMAX
        DO 50 IT=1,IMAX
          RSPOT=(IT-0.5)*DELTAR
          FSUM=FSUM+FLUX(IT,JT)*RSPOT
50      WRITE (2,999) JT,IT,FLUX(IT,JT)
60    CONTINUE
      FSUM=FSUM*DELTAR*DELTAZ*2*PI
      WRITE(2,998)FSUM
      RETURN
998   FORMAT('/ INTEGRATED SCALAR FLUX DENSITY = ')
999   FORMAT(I3,' ',I3,' ',E12.5)
      END
```

APPENDIX C: The Computer Program SRC2DPT

```
CCCC  PROGRAM READS IN CROSS SECTION DATA AND COMPUTES
CCCC  THE ANGULAR SOURCE DISTRIBUTION AT A POINT FOR THE SECOND
CCCC  COLLISION SOURCE FOR A POINT SOURCE EMITTING
CCCC  NEUTRONS IN THE UPWARD DIRECTION.  THE ORIGINAL POINT SOURCE HAS
CCCC  A NORMALIZED STRENGTH OF 4 PI.  THE 4 PI IS USED TO FACILITATE
CCCC  NORMALIZATION IN THE TWODANT CODE PACKAGE.  THE ANGULAR
CCCC  DISTRIBUTION CAN BE COMPUTED EITHER WITH A SPHERICAL HARMONIC
CCCC  EXPANSION OR BY COMPUTING THE SOURCE ALONG ACTUAL DIRECTIONS.
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER ORDER
      CHARACTER*8 XSFILE,OTFILE,OTFX
      COMMON/CNSTS/PI
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(2),SIGMA(3,17,4)
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(17,4)
      COMMON/BLK2/W(32),X(32)
      DIMENSION APOLY(0:153),SUM(154)
      COMMON/SOURCE/Q(153),FLUX
CCCC  ------------------------------------------------------------------
1     CALL INPUT(ZMX,RLOC,ZLOC,XSFILE,OTFILE,MOMENT,IXSECT,NQUAD
     1,ICALC,IMAX,JMAX,RENORM)
      NGP=NGROUP+3
      CALL RDXSCT(XSFILE)
CCCC  ------------------------------------------------------------------
      DO 120 IORD=1,NORD+1
        DO 119 I=1,4
          SIGTAB(IORD,I)=0.0D00
          DO 118 NNISO=1,NISO
118       SIGTAB(IORD,I)=DEN(NNISO)*SIGMA(NNISO,IORD,I)+SIGTAB(IORD,I)
119     CONTINUE
120   CONTINUE
CCCC  ------------------------------------------------------------------
      IF(IXSECT.EQ.2)THEN
CCCC    CHANGE CROSS SECTION COEFFICIENTS TO REFLECT THE
CCCC    N'TH CESARO MEAN OF ORDER 2
        DO 130 IORD=2,NORD+1
          CNST1=(NORD+3-IORD)*(NORD+2-IORD)
          CNST2=(NORD+2)*(NORD+1)
          CNST=CNST1/CNST2
          DO 129 I=1,4
129       SIGTAB(IORD,I)=SIGTAB(IORD,I)*CNST
130     CONTINUE
      END IF
CCCC  ------------------------------------------------------------------
```

```
      IF(ICALC.EQ.1)THEN
CCCC    COMPUTE SOURCE MOMENTS
        CALL PTSRC1(RLOC,ZLOC,ZMX,NQUAD,MOMENT,SUM,IXSECT,XSFILE)
CCCC    COMPUTE THE FIRST COLLIDED SCALAR FLUX DENSITY (1/4PI)
        FLUX=SUM(1)/SIGTAB(1,4)
        DO 36 M=1,MOMENT
36      Q(M)=SUM(M)
        CALL OUTPT1(RLOC,ZLOC,OTFILE,MOMENT,IMAX,JMAX,RENORM)
CCCC    ------------------------------------------------------------------
      ELSE
CCCC    COMPUTE SOURCE ALONG ACTUAL DIRECTIONS
        DEL1=2.D00/IMAX
        DEL2=PI/JMAX
        OPEN(3,FILE=OTFILE,STATUS='NEW',ACCESS='SEQUENTIAL')
        WRITE(3,*)'/RSPOT =',RLOC
        WRITE(3,*)'/ZSPOT =',ZLOC
        DO 180 I=0,IMAX
          CSANG1=-1+DEL1*I
          DO 170 J=0,JMAX
            ANG2=J*DEL2
            CALL PTSRC2(CSANG1,ANG2,RLOC,ZLOC,ZMX,RESULT,NQUAD,
     1                  IXSECT,XSFILE)
CCCC        4 PI FROM NORMAILIZED SOURCE STRENGTH OF 4 PI
            WRITE(3,999)CSANG1,ANG2,RESULT*4*PI*RENORM
170       CONTINUE
180     CONTINUE
        CLOSE(3)
      END IF
CCCC  ------------------------------------------------------------------
      WRITE(6,*)'END OF PROGRAM RUN'
      WRITE(6,*)'TYPE 1 TO CONTINUE'
      READ(5,*)ICONT
      IF(ICONT.EQ.1)GOTO 1
999   FORMAT(F7.3,',',F7.3,',',E12.5)
      END
```

```
      SUBROUTINE CALCRN(POLY,ANG1,ANG2,N)
CCCC  ------------------------------------------------------------------
CCCC  ROUTINE RETURNS THE POLYNOMIAL TERMS EVALUATED AT THE
CCCC  APPROPRIATE ANGLES TO DETERMINE THE SOURCE TERM
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION POLY(153)
      POLY(1)=PLGNDR(0,0,ANG1)
      NMQ=2
      DO 20 L=1,N
        POLY(NMQ)=PLGNDR(L,0,ANG1)
        NMQ=NMQ+1
        DO 10 M=1,L
          POLY(NMQ)=PLGNDR(L,M,ANG1)*DCOS(M*ANG2)*
     1              DSQRT(2*FACT(L-M)/FACT(L+M))
10      NMQ=NMQ+1
20    CONTINUE
      RETURN
      END
```

```
      DOUBLE PRECISION FUNCTION FACT(NUM)
CCCC  FUNCTION RETURNS FACTORIAL OF NUM
      IMPLICIT REAL*8(A-H,O-Z)
      DUMB=1
      DO 10 I=1,NUM
10    DUMB=DUMB*I
      FACT=DUMB
      RETURN
      END
```

```
      DOUBLE PRECISION FUNCTION PLGNDR(L,M,X)
CCCC ------------------------------------------------------------------
CCCC  COMPUTES THE ASSOCIATED LEGENDRE POLYNOMIAL
CCCC ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      IF(M.LT.0.OR.M.GT.L.OR.ABS(X).GT.1)PAUSE 'BAD ARGUMENTS'
      PMM=1
      IF(M.GT.0) THEN
        SOMX2=DSQRT((1.-X)*(1.+X))
        FACT=1
        DO 11 I=1,M
          PMM=-PMM*FACT*SOMX2
          FACT=FACT+2
11      CONTINUE
      END IF
      IF(L.EQ.M)THEN
        PLGNDR=PMM
      ELSE
        PMMP1=X*(2*M+1)*PMM
        IF(L.EQ.M+1)THEN
          PLGNDR=PMMP1
        ELSE
          DO 12 LL=M+2,L
            PLL=(X*(2*LL-1)*PMMP1-(LL+M-1)*PMM)/(LL-M)
            PMM=PMMP1
            PMMP1=PLL
12        CONTINUE
          PLGNDR=PLL
        END IF
      END IF
      RETURN
      END
```

```
      SUBROUTINE RDXSCT(XSFILE)
CCCC  ------------------------------------------------------------------
CCCC  ROUTINE READS IN CROSS-SECTION DATA FROM FILE XSFILE
CCCC  DATA MUST BE IN FIXED FEILD FIDO FORMAT
CCCC  DATA IS ASSUMED TO BE IN FORM ABSORBTION FOR G, NU*SIGMA-FISSION
CCCC  FOR G, TOTAL FOR G, G TO G, G TO G-1, ETC, AND THEN REPEATE FOR
CCCC  EACH ENERGY.  ONCE ALL ENERGIES ARE COMPLETED TABLE END WITH T
CCCC  AND THE NEXT MOMENT IS INPUT.
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*8 XSFILE,OTFILE
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(2),SIGMA(3,17,4)
      DIMENSION N(6),ND(6),IWER(6)
      CHARACTER*8 R(6),SIGN(6)
      OPEN(4,FILE=XSFILE,STATUS='OLD',ACCESS='SEQUENTIAL')
      DO 50 ISO=1,NISO
        IORD=1
2       READ(4,*)NDUMB
        K=1
4       READ(4,999)(N(I),R(I),ND(I),SIGN(I),IWER(I),I=1,6)
        DO 10 I=1,6
        IF(R(I).EQ.'T')THEN
          IORD=IORD+1
          GOTO 20
        ELSE IF(R(I).EQ.'R')THEN
          DO 5 J=0,N(I)-1
            IF(SIGN(I).EQ.'-')THEN
              SIGMA(ISO,IORD,K+J)=ND(I)*(10.D00**(-IWER(I)))
            ELSE
              SIGMA(ISO,IORD,K+J)=ND(I)*(10.D00**(IWER(I)))
            END IF
5         CONTINUE
          K=K+N(I)
        ELSE
          IF(SIGN(I).EQ.'-')THEN
            SIGMA(ISO,IORD,K)=ND(I)*10.D00**(-IWER(I))
          ELSE
            SIGMA(ISO,IORD,K)=ND(I)*10.D00**(IWER(I))
          END IF
          K=K+1
        END IF
10      CONTINUE
        GOTO 4
20      IF(IORD.GT.NORD+1)GOTO 50
        GOTO 2
50    CONTINUE
      CLOSE(4)
      RETURN
999   FORMAT(6(I2,A1,I6,A1,I2))
      END
```

```
      SUBROUTINE LGDRE(ITYPE,OMEGAR,NDUMB,APOLY)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION APOLY(0:153)
CCCC  ------------------------------------------------------------------
      IF(ITYPE.EQ.1)THEN
CCCC    APOLY(I) IS THE EVALUATION OF THE I TH LEGENDRE POLYNOMIAL
        APOLY(0)=1
        APOLY(1)=OMEGAR
        DO 10 IP=1,NDUMB-1
10      APOLY(IP+1)=((2*IP+1)*OMEGAR*APOLY(IP)-IP*APOLY(IP-1))/(IP+1)
CCCC  ------------------------------------------------------------------
      ELSE
CCCC    APOLY(I) IS THE EVALUATION OF THE APPROPRIATE MOMENT OF THE
CCCC    SOURCE
        APOLY(0)=1
        I=1
        DO 50 L=1,NDUMB
          APOLY(I)=PLGNDR(L,0,OMEGAR)
          I=I+1
          DO 40 M=1,L
            APOLY(I)=PLGNDR(L,M,OMEGAR)*DSQRT(2*FACT(L-M)/FACT(L+M))
40          I=I+1
50      CONTINUE
CCCC  ------------------------------------------------------------------
      END IF
      RETURN
      END
```

```
      SUBROUTINE XSECT(VAL,OMEGAR,IXSECT,XSFILE)
CCCC  ------------------------------------------------------------------
CCCC  RETURNS THE VALUE OF THE SCATTERING CROSS SECTION IN VAL
CCCC  FOR THE GROUP GO TO THE GROUP GP FOR THE COSINE OMEGAR
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*8 XSFILE
      INTEGER ORDER,G,GO,GP
      COMMON/CNSTS/PI
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(2),SIGMA(3,17,4)
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(17,4)
      DIMENSION APOLY(0:153)
CCCC  ------------------------------------------------------------------
      ITYPE=1
      CALL LGDRE(ITYPE,OMEGAR,ORDER,APOLY)
      VAL=0
      IF(IXSECT.LT.3)THEN
CCCC    USE THE LEGENDRE OR CESARO MEAN EXPNASION
        DO 9 IVAL=0,ORDER
9       VAL=VAL+APOLY(IVAL)*SIGTAB(IVAL+1,4)*(2*IVAL+1)
        VAL=VAL*0.5
      ELSE
CCCC    USE EXACT CROSS SECTION TECHNIQUE (VALID ONLY FOR ONE GROUP)
CCCC    REQUIRES MODIFICATION OF SOURCE ROUTINE TO DESCRIBE THE
CCCC    SCATTERING MODEL
        IF(XSFILE(5:5).EQ.'1')THEN
          VAL=0.25
        ELSEIF(XSFILE(5:5).EQ.'2')THEN
          VAL=0
          IF(OMEGAR.GT.0)VAL=0.5*OMEGAR
        ELSEIF(XSFILE(5:5).EQ.'3')THEN
          VAL=0
          IF(OMEGAR.GT.0.5)VAL=2*OMEGAR-1
        ELSEIF(XSFILE(5:5).EQ.'4')THEN
          VAL=0
          IF((OMEGAR.GT.-0.5).AND.(OMEGAR.LT.0.5))VAL=0.5-DABS(OMEGAR)
        ELSEIF(XSFILE(5:5).EQ.'5')THEN
          VAL=0
          IF(OMEGAR.LT.-0.5)VAL=-2*OMEGAR-0.5
        ELSE
          WRITE(6,*)'REWRITE XSECT SUBROUTINE FOR ALTERNATE CASES'
          STOP
        END IF
        VAL=VAL*DEN(1)
      END IF
CCCC  ------------------------------------------------------------------
      VAL=VAL/(2*PI)
      RETURN
      END
```

```
      SUBROUTINE PTSRC1(RSPOT,ZSPOT,ZMAX,NQUAD,MOMENT,SUM,IXSECT,XSFILE)
CCCC  ------------------------------------------------------------------
CCCC  ROUTINE COMPUTES THE SPHERICAL HARMONIC COEFFICIENTS NECCESSARY
CCCC  TO GENERATE THE ANGULAR DISTRIBUTION OF THE SECOND COLLISION
CCCC  SOURCE AT A PARTICULAR LOCATION
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*8 XSFILE
      INTEGER ORDER,G,GO,GP
      COMMON/CNSTS/PI
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(17,4)
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(2),SIGMA(3,17,4)
      DIMENSION APOLY(0:153),SUM(154)
      COMMON/BLK2/W(32),X(32)
CCCC  ------------------------------------------------------------------
      DO 5 I=1,MOMENT
5     SUM(I)=0
CCCC  ------------------------------------------------------------------
      XMAX=DATAN((ZMAX-ZSPOT)/RSPOT)
      XMIN=DATAN(-ZSPOT/RSPOT)
      D1=(XMAX-XMIN)*0.5
      D2=(XMAX+XMIN)*0.5
      DO 20 N=1,NQUAD
        THETA=D1*X(N)+D2
        OMEGAR=-DSIN(THETA)
        CALL XSECT(VAL,OMEGAR,IXSECT,XSFILE)
        CALL LGDRE(2,OMEGAR,ORDER,APOLY)
        DUMB=W(N)*VAL*DEXP(-RSPOT*
     1          SIGTAB(1,3)*(DTAN(THETA)+1.D00/DCOS(THETA)))
        DO 10 M=1,MOMENT
10      SUM(M)=SUM(M)+APOLY(M-1)*DUMB
20    CONTINUE
      DUMB=D1*DEXP(-SIGTAB(1,3)*ZSPOT)/RSPOT
      NMQ=1
      DO 40 L=0,ORDER
        DO 30 M=0,L
        SUM(NMQ)=SUM(NMQ)*DUMB*SIGTAB(L+1,4)
30      NMQ=NMQ+1
40    CONTINUE
CCCC  ------------------------------------------------------------------
      RETURN
      END
```

```
      SUBROUTINE INPUT(ZMX,RLOC,ZLOC,INFILE,OTFILE,MOMENT,IXSECT,
CCCC  ------------------------------------------------------------------
CCCC  ROUTINE PROMPTS USER FOR DATA NEEDED IN PROGRAM RUN
     1NQUAD,ICALC,IMAX,JMAX,RENORM)
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER ORDER
      CHARACTER*8 INFILE,OTFILE,OTFLX
      COMMON/CNSTS/PI
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(17,4)
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(2),SIGMA(3,17,4)
      COMMON/BLK2/W(32),X(32)
CCCC  ------------------------------------------------------------------
      PI=DACOS(-1.D00)
      NISO=1
      NGROUP=1
      WRITE(6,*)'INPUT THE CROSS SECTION LEGENGRE EXPANSION ORDER'
      READ(5,*)ORDER
      NORD=ORDER
      MOMENT=(ORDER+1)*(ORDER+2)/2
      WRITE(6,*)'INPUT THE DENSITY FOR EACH OF THE ISOTOPES'
      DO 10 I=1,NISO
      WRITE(6,*)'DENSITY OF MATERIAL',I
10    READ(5,*)DEN(I)
      WRITE(6,*)'INPUT THE NAME OF THE CROSS SECTION FILE'
      READ(5,999)INFILE
      WRITE(6,*)'INPUT THE CROSS SECTION MODEL '
      WRITE(6,*)'   (1) LEGENDRE EXPANSION '
      WRITE(6,*)'   (2) CESARO MEAN '
      WRITE(6,*)'   (3) EXACT (MAY REQUIRE CODE MODIFICATION)'
      READ(5,*)IXSECT
      WRITE(6,*)'INPUT THE CALCULATION TYPE'
      WRITE(6,*)'   (1) COMPUTE MOMENTS FIRST'
      WRITE(6,*)'   (2) COMPUTE SOURCE ALONG ACTUAL DIRECTIONS'
      READ(5,*)ICALC
      WRITE(6,*)'INPUT THE RENORMALIZATION FACTOR'
      READ(5,*)RENORM
      WRITE(6,*)'INPUT THE Z LOCATION'
      READ(5,*)ZLOC
      WRITE(6,*)'INPUT THE RADIAL LOCATION'
      READ(5,*)RLOC
      WRITE(6,*)'INPUT THE Z INTEGRATION LENGTH'
      READ(5,*)ZMX
```

```
      WRITE(6,*)'INPUT THE QUADRATURE INTEGRATION ORDER'
      READ(5,*)NQUAD
      CALL GETQAD(NQUAD)
      WRITE(6,*)'INPUT THE NUMBER OF ANGULAR MESHES -1 TO 1'
      READ(5,*)IMAX
      WRITE(6,*)'INPUT THE NUMBER OF ANGULAR MESHES 0 TO PI'
      READ(5,*)JMAX
      WRITE(6,*)'INPUT THE NAME OF OUTPUT FILE'
      READ(5,999)OTFILE
      RETURN
999   FORMAT(A)
      END




      SUBROUTINE GETQAD(NQUAD)
CCCC  ***********************************************************************
CCCC  SUBROUTINE READS GAUSS QUADRATURE DATA FROM AN INPUT FILE
CCCC  THE FIRST NUMBER IS THE ORDER OF GAUSSIAN QUADRATURE SET.
CCCC  THE ABCISSAS AND WEIGHTS SHOULD BE ORDERED FROM NEGATIVE TO
CCCC  POSITIVE VALUES.
CCCC  ***********************************************************************
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*8,OIN
      COMMON/BLK2/W(32),X(32)
306   FORMAT(A)
      WRITE(6,*)'INPUT THE NAME OF THE QUADRATURE FILE'
      READ(5,306)OIN
      OPEN(9,FILE=OIN,STATUS='OLD',ACCESS='SEQUENTIAL')
      DO 10 I=1,NQUAD
10    READ(9,*)X(I),W(I)
      CLOSE(9)
      RETURN
      END
```

```
      SUBROUTINE OUTPT1(RLOC,ZLOC,OTFILE,MOMENT,IMAX,JMAX,RENORM)
CCCC  ------------------------------------------------------------------
CCCC  ROUTINE COMPUTES THE ANGULAR SOURCE ALONG A PARTICULAR DIRECTION
CCCC  USING THE SPHERICAL HARMONIC EXPANSION AND THEN OUTPUTS THE
CCCC  RESULT TO A FILE
CCCC  ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER ORDER
      CHARACTER*8 OTFILE,OTFX,DELIM
      DIMENSION POLY(153)
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(17,4)
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(2),SIGMA(3,17,4)
      COMMON/SOURCE/Q(153),FLUX
      COMMON/CNSTS/PI
CCCC  ------------------------------------------------------------------
      OPEN(3,FILE=OTFILE,STATUS='NEW',ACCESS='SEQUENTIAL')
      WRITE(3,981)FLUX*RENORM
      DO 30 N=1,MOMENT
30    WRITE(3,980)N,Q(N)*RENORM
      DEL1=2.D00/IMAX
      DEL2=PI/JMAX
      WRITE(3,*)'/RSPOT =',RLOC
      WRITE(3,*)'/ZSPOT =',ZLOC
      WRITE(3,*)'/NUMBER OF MOMENTS',MOMENT
      DO 120 I=0,IMAX
        CSANG1=-1+DEL1*I
        DO 115 J=0,JMAX
          ANG2=J*DEL2
          CALL CALCRN(POLY,CSANG1,ANG2,ORDER)
          SRC=0
          NN=1
          DO 114 NJ=1,ORDER+1
            DO 113 NI=1,NJ
              SRC=SRC+(2*NJ-1)*POLY(NN)*Q(NN)
113         NN=NN+1
114       CONTINUE
115     WRITE(3,999)CSANG1,ANG2,SRC*RENORM
120   CONTINUE
      CLOSE(3)
980   FORMAT('/',I2,' ',E12.5)
981   FORMAT('/ SCALAR FLUX = ',E12.5)
999   FORMAT(F7.3,',',F7.3,',',E12.5)
      END
```

```
      SUBROUTINE PTSRC2(CSANG1,ANG2,RSPOT,ZSPOT,ZMAX,RESULT,NQUAD,
CCCC  ------------------------------------------------------------------
CCCC  ROUTINE COMPUTES THE SOURCE ALONG ACTUAL DIRECTIONS RATHER
CCCC  THAN USING A SPHERICAL HARMONICS EXPANSION
CCCC  ------------------------------------------------------------------
     1IXSECT,XSFILE)
      CHARACTER*8 XSFILE
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER ORDER
      COMMON/CNSTS/PI
      COMMON/XSEC2/ORDER,NGROUP,SIGTAB(17,4)
      COMMON/XSEC1/NGP,NORD,NISO,NDUM1,DEN(2),SIGMA(3,17,4)
      DIMENSION APOLY(0:153),SUM(154)
      COMMON/BLK2/W(32),X(32)
CCCC  ------------------------------------------------------------------
      XMAX=DATAN((ZMAX-ZSPOT)/RSPOT)
      XMIN=DATAN(-ZSPOT/RSPOT)
      D1=(XMAX-XMIN)*0.5
      D2=(XMAX+XMIN)*0.5
      RESULT=0
      DO 20 N=1,NQUAD
        THETA=D1*X(N)+D2
        OMEGAR=-DSIN(THETA)
        CALL XSECT(VAL,OMEGAR,IXSECT,XSFILE)
        WDUMB=OMEGAR*CSANG1+DSQRT((1-CSANG1*CSANG1)*(1-OMEGAR*OMEGAR))*
     1        DCOS(ANG2)
        CALL XSECT(VAL1,WDUMB,IXSECT,XSFILE)
        RESULT=RESULT+W(N)*VAL*VAL1*DEXP(-RSPOT*
     1              (SIGTAB(1,3)*DTAN(THETA)+SIGTAB(1,3)/DCOS(THETA)))
20    CONTINUE
      RESULT=RESULT*D1*DEXP(-SIGTAB(1,3)*ZSPOT)/RSPOT
      RETURN
      END
```

```
    0
    3     0     1
    1 GROUP AIR
    DELTA FUNCTION INCIDENT SOURCE AT CENTER LEFT FOR R-Z GEOMETRY.
    SOURCE IS APPROXIMATED BY THE SECOND COLLISION SOURCE SOURCE.
/ ------------------------ BLOCK I ------------------------------------
    IDIMEN=2, IGEOM=7, NGROUP=1, ISN=12 NISO=2  MT=1  IT=20  IM=1
    NZONE=1  JT=40 JM=1    T
/ ------------------------ BLOCK II (GEOMETRY) ------------------------
    XMESH=0,60000  XINTS=20   YMESH=0,120000   YINTS=40   ZONES=1  T
/ ------------------------ BLOCK III (CROSS SECTIONS) -----------------
    LIB=ODNINP
    MAXORD=5  IHM=4  IHT=3  IHS=4  IFIDO=1  ITITL=1  SAVBXS=1
    NAMES= "N-14","O-16"   T
    1     4      0    507 P0  N-14  WEIGHTED WITH CF252 SPECTRUM AT 2000M
0 +24736- 5 0 +    0+ 0 0 +15772- 4 0 +35073- 5  T
    1     4      0    508 P1  N-14  WEIGHTED WITH
3R+    0+ 0 0 +10066- 4  T
    1     4      0    509 P2  N-14  WEIGHTED WITH
3R+    0+ 0 0 +15338- 4  T
    1     4      0    510 P3  N-14  WEIGHTED WITH
3R+    0+ 0 0 +18720- 4  T
    1     4      0    511 P4  N-14  WEIGHTED WITH
3R+    0+ 0 0 +19937- 4  T
    1     4      0    512 P5  N-14  WEIGHTED WITH
3R+    0+ 0 0 +19067- 4  T
    1     4      0    513 P0  O-16  WEIGHTED WITH
0 +18763- 5 0 +    0+ 0 0 +16247- 4 0 +41412- 5  T
    1     4      0    514 P1  O-16  WEIGHTED WITH
3R+    0+ 0 0 +11855- 4  T
    1     4      0    515 P2  O-16  WEIGHTED WITH
3R+    0+ 0 0 +17976- 4  T
    1     4      0    516 P3  O-16  WEIGHTED WITH
3R+    0+ 0 0 +21781- 4  T
    1     4      0    517 P4  O-16  WEIGHTED WITH
3R+    0+ 0 0 +22980- 4  T
    1     4      0    518 P5  O-16  WEIGHTED WITH
3R+    0+ 0 0 +21737- 4  T
/ ------------------------ BLOCK IV (MIXING) --------------------------
    MATLS=AIR, "N-14" 4.02-05, "O-16" 1.07-05
    ASSIGN= MATLS  T
```

```
/ ─────────────────────────────── BLOCK V (SOLVER) ──────────────────────────────
    IEVT=0   ISCT=5   ITH=0    IBL=1    IBR=0    IBT=0    IBB=0
    IITL=100 IITM=100 EPSI=0.001
    SOURCF=
/ SOURCE TERM FOR SPHERICAL HARMONIC  1 OF ENERGY GROUP  1
 0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00 ;
 0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00 ;
 0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00 ;

         .
         .
         .

 0.13914-13  0.33827-14  0.15021-14  0.68362-15  0.34542-15  0.17807-15
 0.84428-16  0.41454-16  0.19769-16  0.80884-17  0.28739-17  0.38322-18
 0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00 ;
 0.11267-13  0.27965-14  0.13008-14  0.61623-15  0.32940-15  0.18050-15
 0.91957-16  0.49067-16  0.25946-16  0.13556-16  0.60949-17  0.26854-17
 0.87813-18  0.49567-19  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00 ;
 0.82411-14  0.22939-14  0.11092-14  0.53939-15  0.30129-15  0.17313-15
 0.92725-16  0.52405-16  0.29560-16  0.16643-16  0.84811-17  0.44366-17
 0.22154-17  0.93119-18  0.31011-18  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00 ;
 0.65436-14  0.18728-14  0.84682-15  0.46258-15  0.26776-15  0.15991-15
 0.88810-16  0.52436-16  0.31000-16  0.18363-16  1.00343-17  0.56909-17
 0.31709-17  0.17247-17  0.81721-18  0.36534-18  0.12392-18  0.00000+00
 0.00000+00  0.00000+00 ;
 0.51928-14  0.15253-14  0.69713-15  0.39095-15  0.23313-15  0.14381-15
 0.82010-16  0.50152-16  0.30767-16  0.18944-16  0.11743-16  0.64718-17
 0.38311-17  0.22464-17  0.13036-17  0.65916-18  0.33821-18  0.15658-18
 0.48232-19  0.00000+00 ;
 0.41209-14  0.12416-14  0.57008-15  0.32696-15  0.20000-15  0.12688-15
 0.73720-16  0.46417-16  0.29357-16  0.18656-16  0.11943-16  0.68410-17
 0.42200-17  0.25925-17  0.15878-17  0.88196-18  0.50786-18  0.28329-18
 0.15063-18  0.65009-19 ;
 0.32720-14  0.92167-15  0.46397-15  0.27141-15  0.16979-15  0.11043-15
 0.64936-16  0.41913-16  0.27204-16  0.17756-16  0.11680-16  0.68780-17
 0.43779-17  0.27819-17  0.17674-17  0.11258-17  0.63245-18  0.38077-18
 0.22517-18  0.13011-18 ;
```

```
 0.26008-14  0.73651-15  0.37644-15  0.22416-15  0.14312-15  0.86206-16
 0.56319-16  0.37141-16  0.24656-16  0.16472-16  0.11096-16  0.66659-17
 0.43523-17  0.28404-17  0.18558-17  0.12171-17  0.71475-18  0.44988-18
 0.28081-18  0.17368-18 ;
 0.20710-14  0.58820-15  0.30498-15  0.18464-15  0.12012-15  0.72599-16
 0.48276-16  0.32443-16  0.21969-16  0.14983-16  0.94098-17  0.62817-17
 0.41907-17  0.27964-17  0.18694-17  0.12551-17  0.75964-18  0.49308-18
 0.31866-18  0.20513-18 ;
 0.16538-14  0.47028-15  0.24726-15  0.15204-15  0.91669-16  0.60749-16
 0.41026-16  0.28034-16  0.19323-16  0.13426-16  0.85181-17  0.57907-17
 0.39361-17  0.26773-17  0.18252-17  0.12501-17  0.77317-18  0.51383-18
 0.34064-18  0.22547-18 ;
 0.13280-14  0.37804-15  0.20138-15  0.12558-15  0.75505-16  0.50640-16
 0.34660-16  0.24035-16  0.16831-16  0.11894-16  0.75930-17  0.52451-17
 0.36244-17  0.25074-17  0.17391-17  0.12121-17  0.76174-18  0.51605-18
 0.34911-18  0.23608-18 ;
 1.00173-15  0.30978-15  0.16596-15  0.95490-16  0.62273-16  0.42154-16
 0.29177-16  0.20493-16  0.14557-16  0.95477-17  0.66845-17  0.46836-17
 0.32844-17  0.23069-17  0.16251-17  0.11507-17  0.73150-18  0.50372-18
 0.34659-18  0.23854-18 ;
/ SOURCE TERM FOR SPHERICAL HARMONIC  2 OF ENERGY GROUP  1
 0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00  0.00000+00
 0.00000+00  0.00000+00 ;

    .
    .
    .

    REMAINDER OF THE SOURCE MOMENTS

    .
    .
    .
      T
/-------------------- (BLOCK V  EDITS)-------------------------------
    PTED=1   ZNED=0
    RSFE=1 T
```

APPENDIX E. Details related to the use of TWODANT

The KSU version of the TWODANT code package is based on the 4–30–89 version of TWODANT, which was put together by Los Alamos National Laboratories. Some modifications in the original code were required to make the code operational on the KSU VM/370 mainframe. The present version operates in the CMS environment and can be retrieved by linking the mini–disk 303 of the userid NECODES to a personal userid. The commands are

```
CP LINK NECODES 303 303 RR PASSWORD
ACCESS 303 P
```

The TWODANT code package can now be run by attaching the appropriate input and output files and executing the module. The commands are

```
SET LDRTBLS 20
FILEDEF 5 DISK INPUT FILE
FILEDEF 6 DISK OUTPUT LISTING
TWODANT MODULE P NOXUFLOW
```

E.1. Retrieving and modifying TWODANT for use at KSU

The Los Alamos version of TWODANT presently resides in three files on tape. Two of the files are FORTRAN coding and the third is assembly coding. The tape files are retrieved by using an OSREAD command. One of the FORTRAN files is quite large (in excess of 90000 records) and should be broken into four smaller files with the OSREAD command.

The set of OSREAD commands to be used are

```
OSREAD TWOA FORTRAN (T6250 NLABEL START 1 FOR 25075
OSREAD TWOB FORTRAN (T6250 NLABEL START 25075 FOR 24925
OSREAD TWOC FORTRAN (T6250 NLABEL START 50000 FOR 20225
OSREAD TWOD FORTRAN (T6250 NLABEL START 70225
OSREAD CHAFORTI FORTRAN (T6250 NLABEL
OSREAD CHAHANDI ASSEMBLE (T6250 NLABEL
```

After each OSREAD command, the user will be prompted to input various parameters related to the tape. These include the number of the file being retrieved from the tape, the tape label (presently FBKS), the record format (RECFM = FB), the logical record length (LRECL = 80), the block size (BLKSIZE = 3120), and finally a option code (which can be left blank). Once the files have been retrieved from tape they can be moved from the VIRTUAL READER to a mini–disk file. The best procedure is probably to create a temporary mini–disk for each file, so that sufficient memory is available for text and listing files generated during compilation.

Before the FORTRAN coding can be compiled a few minor changes are required. The first set of changes involves renaming variable names and subroutine names in the FORTRAN coding which are longer than 6 character. Most of these occur in the FORTRAN file previously defined as TWOD and are subroutine names which end in HX. If the HX is retained in any new name, then most of these subroutine names can be changed without repeating a previously defined subroutine name. The second change involves eliminating the call statement to the routine JOBID.

The final change is to alter several records (23462 to 23463) of the subroutine HEADIN in TWOA to stop the code from reading past record 80 of the input file. The FORTRAN coding should be compiled with the FORTVS compiler using an optimizer level of three.

As with the FORTRAN coding, the assembler coding must also be modified. The modifications involve eliminating a set of MACROS which are already defined in a system library. The simplest technique to find this set of MACROS is to compile the assembly code using ASMG and the BATCH option. The important MACROS occur in a set of fifteen at four locations and are each four lines long. They should simply be deleted from the assembly code. The other two errors in the assembly code can be ignored, because they do not affect the code operation.

Once the various FORTRAN codes and assembler codes have been compiled, a module can be generated by loading each of the files beginning with TWOA into memory and using the GENMOD command. Five cylinders of disk space should be allocated for storage of the module and four mega–bytes of RAM for loading the text files into system memory. The entire set of commands are

```
GLOBAL TXTLIB VFORTLIB VLNKMLIB
GLOBAL LOADLIB VFLODLIB VALTLIB
FORTVS TWOA (OPT(3)
FORTVS TWOB (OPT(3)
FORTVS TWOC (OPT(3)
FORTVS TWOD (OPT(3)
FORTVS CHAFORTI (OPT(3)
GLOBAL MACLIB
ASMG CHAHANDI (BATCH
LOAD TWOA TWOB TWOC TWOD CHAFORTI CHAHANDI
GENMOD TWODANT
```

Before eliminating the text files, all the sample files should be run to insure proper operation of the code. Not all of the sample files will run, because the TWODANT code under present use at KSU has one runtime error that has not been resolved. The error centers around the inability of the system to use random access files and can only be resolved by increasing the overall memory allocated to a job and also the values of MAXLCM and MAXSCM in BLOCK I of the input file.

E.2. Details related to the TWODANT input file

The problem to be solved by TWODANT is created in a file which is then attached to UNIT 5 with a FILEDEF command. This input file is divided into seven blocks which control various aspects of TWODANT. All input is based on the use of keywords and FIDO formatted data in these blocks. The TWODANT manual provides excellent explanation and examples of most of the options available in each block. This section attempts to rectify a few of those areas which are unexplained in any of the manuals.

In the first block of data, the general problem is defined along with controls for the other blocks. In this block, of primary importance are the variables MAXLCM and MAXSCM. A proper choice of these is required to insure TWODANT does not attempt to use random access files and subsequently crash. In close relationship to these two variables is the amount of system memory available for a computer run. Table E.1 details the memory allocations for several problems, which have been run using TWODANT.

Tab. E.1. A comparison of memory requirements for several TWODANT problems. The spatial meshing indicates first the number of radial meshes and then the number of z meshes for a cylindrical problem. The scattering order refers to the expansion order of the scattering cross section. The SCM and LCM were values that TWODANT output and indicate the minimum amount of small core memory and large core memory required to store all program variables in system memory. The system memory allocation is the amount of memory allocated to run that particuliar problem in the CMS enviroment of an IBM VM/370.

SPATIAL MESHES r,z	S_n ORDER	SCATTERING ORDER	NUMBER OF ENERGY GROUPS	SCM	LCM	SYSTEM MEMORY (Mbytes)
20,40	6	P_0	1	11490	11234	6
20,40	8	P_0	1	11568	11714	6
20,40	12	P_0	1	11740	13034	6
20,40	12	P_5	3	28509	140707	6
20,40	8	P_5	47	18558	676689	>8*
25,50	8	P_5	1	25104	46424	6

*The problem could not be run because more than 8 megabytes of system memory were required.

In the second block of the input file, the geometry of the problem is defined. Of primary concern in this block are the units on the problem edges and the locations of the cell centers. The units on the problem edges are related to the macroscopic cross–sections which are created later. The units on the problem edges are the same as one–over the units on the macroscopic cross–sections. The cell centers are located such that the edges for those cells located on the problem boundaries coincide with the boundary (i.e., the cell–centers begin half a mesh from the problem boundary).

In the third block, the microscopic cross–sections are defined. The cross–sections either appear directly after this block or can be stored in a separate file attached to UNIT 4. Of importance in this block is that the appropriate format be observed for all the cross section data. This is particularly important if fixed–field FIDO is specified. Most data in the standard cross section libraries use this format, where numbers are represented as 12345–04 (i.e., 1.2345).

In the fourth block, the cross–sections are mixed and assigned to regions in the problem geometry. Since most of the standard cross–section libraries specify cross–section units in terms of b $atom^{-1}$, any density specified for a cross–section should be in similar units (i.e. atoms b^{-1} cm^{-1}). The macroscopic cross–sections will then be in units of cm^{-1} and the problem dimensions will be in units of cm.

In the fifth block, details related to the solver module are input. Of particular importance in this section is the ordering of inhomogeneous sources. The data should be ordered starting with the bottom center of the problem geometry. The source term for the first spherical harmonic and for the first energy group is input for this cell. This source is then followed by the source term for the

first spherical harmonic and for the first energy group for the cell located in the next radial mesh out from the first cell. The process is repeated until the source at the cell on the outer radial edge of the problem is input. This cell is then followed by a semicolon and the above process is repeated for the next cell up. Once all source terms for the first spherical harmonic and for the first energy group are input, the process is repeated for the all remaining energy groups. Once the source coefficients for the first spherical harmonic and for all energy groups have been input, the above process is repeated for the second spherical harmonic coefficients and subsequently for all the remaining source moments. After all source moments have been input, the set of data is terminated with a T (i.e., no semicolon on the last set of data). All the sources moments should be in free–field format (i.e., 5.2864E–05 becomes 5.2864–5 or 52864–9).

In the sixth and final block, options for outputting the flux densities are input. Of importance in this section is the ability to output either the actual flux densities for each group or some appropriate mix of them based on a response function. Both results can be generated by using the RSFE option, and specifying different response functions separated by semicolons. Between each set of semi–colons the response functions are ordered from the first energy group to the last. To output the flux density for a particular group, set the response in the desired group to one and to zero in all others. The flux density and any other response generated following this procedure is actually $1/4\pi$ times the actual response.

CALCULATION OF NEUTRON LINE–BEAM RESPONSE FUNCTIONS WITH TWODANT

by

Thomas A. Gianakon

B.S., Kansas State University, 1987

AN ABSTRACT OF A MASTER'S THESIS

Submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Nuclear Engineering
Kansas State University
Manhattan,Kansas

1989

ABSTRACT

The purpose of this study was to investigate the ability of standard S_n transport codes to generate line–beam response functions for use in neutron skyshine calculations. These response functions give the dose rate at at any given source–to–detector distance for a point source emitting monoenergetic neutrons in an air medium at a single fixed angle relative to the source–to–detector axis. Such spatially and angularly singular sources are not easily incorporated into standard S_n codes, because these codes generally require spatially distributed sources which are only weakly anisotropic (i.e., those that can be represented by a low–order spherical harmonics expansion). If line–beam skyshine sources are used directly in an S_n code, either the calculations diverge or numerous negative values for the iterated flux density are obtained.

In this study, the first and the second collision sources are investigated to determine if these sources, with their more uniform spatial distribution and less anisotropic angular distribution, can successfully be used with the standard S_n code TWODANT to produce meaningful results for line–beam response functions. In this approach, a preprocessor code is needed to generate the data necessary to describe the spatial and angular components of the source. Also, a postprocessor code is required to compute the total flux density by adding the components of the flux density (which were used to generate the source for the S_n calculation) to the iterated flux density (which was obtained from the S_n calculation).

This study used TWODANT with a spherical harmonics expansion of either a first or a second collision source to compute the spatial distribution of the scalar flux density for several one–group problems characterized by a monodirectional

point source emitting neutrons along the axis of a large cylindrical air medium. The spherical harmonics expansion of these sources was found to generate some negative angular sources, which eventually led to convergence problems for TWODANT. Several techniques for alleviating these negative sources were developed. While these techniques aided convergence, this study found that ray effects, the bane of multidimensional S_n calculations, were still prevalent in some regions. Such ray effects limit the usefulness of TWODANT for calculating line–beam response functions to those cases with a broad energy range for the source group (i.e., a large angular support for scattering within the the source group).