

Parallel Shooting Method for Boundary-value Problems: Application to the Neutron Transport Equation

Abstract: A direct method is given for the solution of the spherical harmonics approximation to the Boltzmann equation for neutron transport in slab geometry. The roundoff instability of the problem is eliminated by performing linear transformations of the matrices involved, which ensure that the matrix columns are linearly independent. The novelty of the method lies in that only a minimum number of matrix transformations is performed, the precise number being determined dynamically, efficiently, and in a new way by the program itself in the course of the computation.

Introduction

The aim of this paper is to present a direct method for the numerical solution of two-point boundary value problems for linear systems of first-order ordinary differential equations that are subject to roundoff instability. The method is applied to the solution of the spherical harmonics approximation to the monoenergetic (one-group) neutron transport equation in slab geometry with arbitrary anisotropic scattering and source [1]. This problem is ideally suited for testing the method because the computer program can generate, from analytical formulas, finite-difference approximations to large systems of coupled differential equations having an arbitrarily high degree of roundoff instability. Furthermore, accurate answers for many of these problems are available in the literature of the nuclear reactor field. The method discussed has a much more general applicability than in the present context of the neutron transport equation. For example, the problem of the numerical solution of systems of coupled Schrödinger equations [2] has many features in common with the problem discussed here.

We first reduce the spherical harmonics equations by means of finite differences to a general algebraic problem in block form. Since the matrices of the problem are non-convergent, the roundoff errors grow exponentially with distance. This numerical instability makes the problem both difficult and interesting. Because of the instability, the most common numerical methods for the solution of

the spherical harmonics equations are iterative [1]. In general, a quantitative discussion of the instability cannot be made theoretically, because it depends on slab thickness, anisotropy, etc., and these quantities are different for each problem.

In the method proposed, the numerical instability is eliminated by performing linear transformations of the matrices of the problem, the transformations being necessary for the linear independence of the matrix columns. The new approach described here requires only a minimum of matrix transformations, the precise number being determined dynamically by the program. Thus for thin slabs, in which the algorithm does not encounter instability, the optimal direct solution of the algebraic problem is obtained without performing any matrix transformations whatever. The idea of performing linear transformations of matrices to ensure the linear independence of the columns, and thus the numerical stability of a computation, is well known [2-5]. However, both the precise way in which these transformations are done and their number are crucial for the efficiency of a numerical method. A comparison with the "stabilized march" technique, used by Lucey and Hansen for the solution of similar problems [5], shows that the present method requires one-third the computation time for each spatial grid point at which the reconditioning transformation of the matrices is performed. Furthermore, the fact that the

number of reconditioning points is kept to a minimum by the program will result in a substantial decrease of that number, typically by a factor of 3 to 10. The present method is thus computationally more efficient than the stabilized march technique.

Results of extensive critical-length computations are given for both stable and unstable problems with isotropic and anisotropic scattering. The comparison of these results with previously published work is encouraging, both as to the efficiency and the accuracy of the proposed method.

Statement of the problem

We do not repeat here the derivation of the spherical harmonics approximation to neutron transport starting with Boltzmann's equation in integral form. The interested reader can find this derivation in the review article of Gelbard [1], whose notation we follow closely. In slab geometry, Boltzmann's equation with anisotropic scattering and source is

$$\mu(\partial\varphi(x, \mu)/\partial x) + \Sigma_t(x)\varphi(x, \mu) = \int_{-1}^1 \Sigma_s(x, \mu, \mu')\varphi(x, \mu')d\mu' + S(x, \mu), \quad (1)$$

where μ is the cosine of the angle between the neutron velocity and the x axis; $\varphi(x, \mu)$ is the angular flux; $\Sigma_t(x)$ is the total cross section; $\Sigma_s(x, \mu, \mu')$ is the scattering cross section from μ' to μ ; and $S(x, \mu)$ is the anisotropic neutron source.

We assume that an approximate solution of (1) can be obtained in the form

$$\varphi(x, \mu) \cong \sum_{l=0}^L \frac{1}{2}(2l+1)f_l(x)P_l(\mu). \quad (2)$$

We also expand

$$\begin{aligned} \Sigma_s(x, \mu, \mu') &= (1/2\pi)\Sigma_s(x, \mu_0) \\ &\cong (1/2\pi) \sum_{l=0}^L \frac{1}{2}(2l+1)\Sigma_{sl}(x)P_l(\mu_0) \end{aligned} \quad (3)$$

and

$$S(x, \mu) \cong \sum_{l=0}^L \frac{1}{2}(2l+1)S_l(x)P_l(\mu), \quad (4)$$

where μ_0 is the cosine of the angle between the neutron velocities in the directions μ' and μ , and $P_l(\mu)$ are the Legendre polynomials. If we perform the necessary angular integrations using the addition theorem for the Legendre polynomials and their orthogonality properties, we obtain the classic spherical harmonics approximation, or P_L approximation

$$\frac{df_l}{dx} + \sigma_0(x)f_l(x) = S_0(x);$$

$$\frac{l+1}{2l+1} \frac{df_{l+1}(x)}{dx} + \frac{l}{2l+1} \frac{df_{l-1}(x)}{dx} + \sigma_l(x)f_l(x) = S_l(x) \quad l = 1, 2, \dots, L-1;$$

$$\frac{L}{2L+1} \frac{df_{L-1}(x)}{dx} + \sigma_L(x)f_L(x) = S_L(x). \quad (5)$$

In Eq. (5),

$$f_l(x) \equiv \int_{-1}^1 \varphi(x, \mu)P_l(\mu)d\mu;$$

$$\sigma_l(x) \equiv \Sigma_t(x) - \Sigma_{sl}(x) \equiv \Sigma_l(x)$$

$$- \int_{-1}^1 \Sigma_s(x, \mu_0)P_l(\mu_0)d\mu_0;$$

$$S_l(x) \equiv \int_{-1}^1 S_l(x, \mu)P_l(\mu)d\mu. \quad (6)$$

Equations (5) are a system of $L+1$ inhomogeneous ordinary differential equations. Their general solution has $L+1$ arbitrary constants of integration, which must be determined by imposing $L+1$ suitable boundary conditions. To fix ideas we limit our discussion to a symmetric slab problem, in which abscissas extend from the center of the slab, $x=0$, to the outer boundary, $x=R$.

• Boundary conditions

The interface and boundary conditions are discussed by Gelbard [1] and Davison [6]. In brief, for odd- L approximations, the spatial moments f_l are continuous across interfaces of different media. As the directional flux is angularly symmetric at the center of the domain, $\varphi(0, \mu) = \varphi(0, -\mu)$, we must have

$$f_{2l+1}(0) = 0 \quad 2l+1 = 1, 3, \dots, L, \quad (7)$$

exactly $\frac{1}{2}(L+1)$ left boundary conditions. The vacuum boundary condition requiring that no neutrons return from the vacuum,

$$\varphi(R, \mu) = 0 \quad \mu < 0, \quad (8)$$

cannot be satisfied exactly by any finite expansion (2). Two commonly used approximate boundary conditions due to Marshak [1] and Federighi [8] can both be written in the form

$$\mathbf{f}_0(R) = G \mathbf{f}_e(R), \quad (9)$$

where

$$\mathbf{f}_0(R) \equiv \begin{bmatrix} f_1(R) \\ f_3(R) \\ \vdots \\ f_L(R) \end{bmatrix}, \quad \mathbf{f}_e(R) \equiv \begin{bmatrix} f_0(R) \\ f_2(R) \\ \vdots \\ f_{L-1}(R) \end{bmatrix}, \quad (10)$$

and the two approximations differ in the elements of the square $\frac{1}{2}(L+1)$ -order matrix G , for which values are

available [7-9]. The relative merits of these approximations are discussed by a number of authors [8-14].

In summary, the spherical harmonics approximation is a two-point boundary value problem for a system of ordinary differential equations (5) with left boundary conditions (7) and right boundary conditions (9). The numerical solution of this problem constitutes the subject of the remaining sections of this paper.

Algebraic problem for spherical harmonics equations

Using the compact notation of matrix theory, our two-point boundary value problem can be written as follows:

$$A[df(x)/dx] + C(x)f = s(x), \quad (11)$$

$$f_0(0) = 0, \quad f_0(R) = Gf_e(R), \quad (12)$$

where the following definitions have been used,

$$f(x) \equiv \begin{bmatrix} f_0(x) \\ f_1(x) \\ \vdots \\ f_L(x) \end{bmatrix}; \quad (11a)$$

$$A \equiv \begin{bmatrix} 0 & 1 & & & \\ 1/3 & 0 & 2/3 & & \\ & 2/5 & 0 & 3/5 & 0 \\ & & \ddots & \ddots & \ddots \\ 0 & & & \frac{L-1}{2L-1} & 0 & \frac{L}{2L-1} \\ & & & & \frac{L}{2L+1} & 0 \end{bmatrix}; \quad (11b)$$

$$C(x) \equiv \begin{bmatrix} \sigma_0(x) & 0 \\ & \sigma_1(x) \\ & \vdots \\ 0 & \sigma_L(x) \end{bmatrix}; \quad (11c)$$

$$s(x) \equiv \begin{bmatrix} S_0(x) \\ S_1(x) \\ \vdots \\ S_L(x) \end{bmatrix}. \quad (11d)$$

The boundary condition matrix G in (12) is defined analytically for Marshak conditions [9]; Federighi has given in his paper [8] the matrix elements of G to six significant figures.

It is now convenient to premultiply Eq. (11) by A^{-1} so that it becomes

$$df(x)/dx + N(x)f = v(x), \quad (13)$$

where we have used the obvious definitions

$$N(x) \equiv A^{-1}C(x) \text{ and } v(x) \equiv A^{-1}s(x). \quad (13a)$$

When the cross sections σ_i [see (11c) and (6)] are constant, independent of the spatial coordinate (one homogeneous region), it is known that a direct numerical integration of (13) is unstable because the roundoff errors grow exponentially with distance. The instability is due to the fact that some of the eigenvalues of N are real and occur in positive and negative pairs [1], which explains in part the instability of the direct solution of the spherical harmonics equations for very thick regions.

In reactor problems the cross section matrix (11c) is a piecewise constant function of space. Thus, if we integrate (13) between mesh points $j-1$ and j in the finite differences approximation

$$\begin{array}{ccccccc} & & & h_{j-1} & & & \\ & & & \longleftrightarrow & & & \\ x=0 & & & & & & x=R \\ | & \dots & | & & | & \dots & | \\ 1 & & 2 & & j-1 & & j & & n-1 & & n \end{array}$$

we obtain, after using the trapezoidal rule for integration,

$$f_j - f_{j-1} + \frac{1}{2} h_{j-1} N_{j-1} (f_{j-1} + f_j) = \frac{1}{2} h_{j-1} (v_{j-1} + v_j), \quad j = 2, 3, \dots, n, \quad (14)$$

where N_{j-1} is a constant matrix involving the cross sections in the interval between mesh points $j-1$ and j . The boundary conditions (12) in discrete form are

$$(f_1)_0 = 0, \quad (f_n)_0 = G(f_n)_e. \quad (15)$$

The boundary conditions (15) separate sharply the even from the odd components of f at the two boundary points. The separation suggests that the solution vectors f_j be rearranged, putting the even and the odd components at the top and the bottom halves, respectively, throughout the grid. In fact, this rearrangement considerably simplifies the algebraic analysis of the system (14) and (15). We thus use the definitions

$$g_j \equiv \begin{bmatrix} f_e \\ f_o \end{bmatrix}_j; \quad (16)$$

$$D_{j-1,j-1} \equiv (-I + \frac{1}{2} h_{j-1} N_{j-1})', \quad D_{j-1,j} \equiv (I + \frac{1}{2} h_{j-1} N_{j-1})'; \quad (17)$$

$$w_{j-1} \equiv \frac{1}{2} h_{j-1} (v_{j-1} + v_j), \quad (18)$$

where the primes denote the reordering of the columns of the matrices inside the parentheses by parity, i.e., the first half of the columns of the D 's are the 1st, 3rd, 5th,

$\dots$, L th columns of the matrix inside the parentheses, and the second half are the 2nd, 4th, 6th, $\dots$, $(L+1)$ th columns. Using (16), (17), and (18), we write the algebraic problem (14) and (15) in full:

$$\begin{aligned} D_{11}g_1 + D_{12}g_2 &= w_1 \\ D_{22}g_2 + D_{23}g_3 &= w_2 \\ &\vdots \\ D_{n-1,n-1}g_{n-1} + D_{n-1,n}g_n &= w_{n-1} \\ g_1^b = 0, \quad g_n^b &= Gg_n^t, \end{aligned} \quad (19)$$

where the superscripts b and t designate bottom and top halves of the respective vectors, the D 's are $(L+1)$ th-order square matrices, and the g 's and w 's are $(L+1)$ component column vectors.

An attractive feature of the spherical harmonics method is that the algebraic problem (19) can be generated numerically from the analytical formulas given in the text, independently of the number of terms $L+1$ (P_l approximation) kept in the expansions (2), (3), and (4). It should also be noticed that the algebraic problem is the same whether the scattering cross-section Σ_s and the external source S (see Eq. (1)) are isotropic or anisotropic. The only change is that the elements of the D matrices and the components of the w vectors in (19) are different, but the form of the algebraic problem, and thus its numerical solution, remains the same. Finally, it is known [15] that the finite-difference approximation used is $O(h^2)$, where h is the mesh size. More precisely,

$$|g(x_j) - g_j| = O(h^2). \quad (20)$$

Direct solution of algebraic problem

In this section we present the matrix algebra analysis to implement the numerical solution of (19).

The block algebraic system of Eqs. (19) has a very simple structure that suggests an obvious direct solution by substitution. First we define the following sequence of rectangular matrices of order $(L+1) \times \frac{1}{2}(L+1)$:

$$\begin{aligned} F_1 &= \begin{bmatrix} I \\ 0 \end{bmatrix}, F_2 = -\pi_1 F_1, F_3 = \pi_2 \pi_1 F_1, \dots, F_n \\ &= (-)^{n-1} \pi_{n-1} \pi_{n-2} \dots \pi_1 F_1, \\ \pi_i &= -D_{i,i+1}^{-1} D_{ii}. \end{aligned} \quad (21)$$

In (21) the explicit form of F_1 (its upper half is the unit matrix and its lower half, the null matrix) was chosen to satisfy the left boundary conditions, whereas the other terms in the sequence are found by substitution in (19) without the inhomogeneous terms. By substitution in the full inhomogeneous system (19) we now define the sequence of vectors of $L+1$ components

$$\begin{aligned} a_1 &= 0, a_2 = D_{12}^{-1} w_1 - \pi_1 a_1, a_3 = D_{23}^{-1} w_2 - \pi_2 a_2, \dots, \\ a_n &= D_{n-1,n}^{-1} w_{n-1} - \pi_{n-1} a_{n-1}. \end{aligned} \quad (22)$$

If the vector I_1 designates the upper half of g_1 (the lower half is zero from the left boundary condition), then the exact mathematical solution of (19) is

$$g_i = F_i I_1 + a_i, \quad i = 1, 2, \dots, n, \quad (23)$$

where the $\frac{1}{2}(L+1)$ component vector I_1 is given by the solution of the inhomogeneous system of equations

$$(F_n^b - G F_n^t) I_1 = G a_n^t - a_n^b, \quad (24)$$

obtained from the right boundary condition; the superscripts t and b designate the upper and lower halves of the corresponding matrices and vectors. The method by which this exact solution is obtained is the finite-difference analogue of the well-known simple shooting technique based on initial value problems [16].

The trouble with this exact solution is that in many problems its numerical computation is unstable against the accumulation of roundoff errors. The errors result in some instances in the extreme ill-conditioning of the matrix in the left side of (24), making its numerical solution impossible. Actually, in some cases, an overflow is encountered during the computation of F_n . In a one-region problem when the P_1 approximation is used, it can be shown quite simply that the spectral radius of the π matrix in (21) is greater than unity, i.e., that the matrix π is divergent, and when computing $F_n = (-)^{n-1} \pi^{n-1} F_1$ its elements grow larger with each successive matrix product. Even if these products did not lead to overflow, the accuracy in the computation of F_n would be lost by subtraction of very large numbers, with the result that the computed F_n would have linearly dependent columns. The same numerical instability is a well-known phenomenon in the numerical solution of the Schrödinger equation, when the eigenfunctions in the classically forbidden regions are computed [2, 17, 18].

Numerical instability questions can be discussed analytically only in very simple situations, such as in one-region problems, and then only qualitatively. In an actual problem, it is imperative to "measure" the instability during the computation and to eliminate it in the most efficient way.

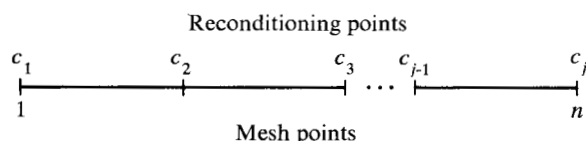
Stabilizing transformations

The concept involved in the stabilizing transformations is quite clear: The exact mathematical solution of our problem, given by (23) and (24), is to be transformed in such a way that the new solution is expressed only in terms of matrices and vectors that are arbitrarily well-conditioned for computation. Obviously, the transformed solution has to be mathematically identical to the untransformed solution.

Our general method belongs to the class of parallel (or multiple) shooting methods discussed by Keller [16] and recently reviewed by Miranker [19]. Osborne [20] has also published some interesting work along these lines.

It is difficult to give a brief and at the same time clear account of the rather complicated linear algebra manipulations involved in the stabilizing transformations. For the sake of clarity and completeness, the linear algebra analysis is expounded in considerable detail in the Appendix.

Our stabilization procedure is based on the generalization of a scheme proposed by Godunov [3] and later significantly improved by Conte [4]. In order to assure the linear independence of the columns of the matrices F_i [see (21)], both authors orthonormalize their columns by means of the Gram-Schmidt method. In the Appendix, it is shown that Conte's analysis is not restricted to orthonormal transformations, but that it is valid for arbitrary transformation matrices and vectors. This point is emphasized here, because for each problem one should choose the optimal linear transformations that offer the greatest computational efficiency. With reference to the figure below,



in which $c_1 = 1, c_2, \dots, c_j = n$ are the equally spaced conditioning points, we define the following sequence of reconditioned rectangular matrices $(L+1) \times \frac{1}{2}(L+1)$:

$$\begin{aligned} U_1 &= \begin{bmatrix} I \\ 0 \end{bmatrix}, \\ U_{c_2} &= \begin{bmatrix} I \\ (F_{c_2}^b T_1) (F_{c_2}^t T_1)^{-1} \end{bmatrix}, \\ U_{c_3} &= \begin{bmatrix} I \\ (F_{c_3}^b T_1 T_{c_2}) (F_{c_3}^t T_1 T_{c_2})^{-1} \end{bmatrix}, \\ &\vdots \\ U_n &= \begin{bmatrix} I \\ (F_n^b T_1 T_{c_2} \dots T_{c_{j-1}}) (F_n^t T_1 T_{c_2} \dots T_{c_{j-1}})^{-1} \end{bmatrix}, \end{aligned} \quad (25)$$

where the matrices F and the transformation matrices T are defined in (21) and (A13). The superscripts t and b designate the upper and lower halves of the corresponding matrix.

The vectors a_i defined by (22) must also be transformed so that the new vectors u_i are linearly independent of the columns of U_i . This can be achieved by using the reconditioned vectors

$$u_1 = 0$$

$$u_{c_k} = \begin{bmatrix} 0 \\ a_{c_k}^b - F_{c_k}^b T_1 T_{c_2} t_{c_2} - \dots - F_{c_k}^b T_1 T_{c_2} \dots T_{c_k} t_{c_k} \end{bmatrix} \quad k = 2, 3, \dots, j, \quad (26)$$

where the $\frac{1}{2}(L+1)$ -component transformation vectors t_{c_k} are defined in (A14). In-between conditioning points the matrix and vector sequences U_i and u_i are determined by substitution in (19) without and with the inhomogeneous terms, respectively [see (21) and (22)]. The t_{c_k} 's (A14) are chosen to be identical to the upper halves of the very same vectors they are to recondition and so are available without any extra computation or storage. Because the upper halves of the reconditioned matrices U_{c_k} and vectors u_{c_k} are analytically set equal to unit matrix and the null vector, respectively, they are not subject to roundoff errors. Furthermore, the columns of U_{c_k} and the vectors u_{c_k} are thus guaranteed to form a set of linearly independent vectors.

In the Appendix it is shown that the solution to (19) can be expressed in the well-conditioned form

$$\begin{aligned} g_i &= U_{c_i} l_i + u_i, & c_k \leq i < c_{k+1} \\ g_n &= U_n l_j + u_n, \end{aligned} \quad (27)$$

where the $\frac{1}{2}(L+1)$ -component vectors l_k are given by

$$\begin{aligned} l_{k-1} &= T_{c_k} (l_k - t_{c_k}), & k = 2, 3, \dots, j-1, \\ l_{j-1} &= T_n (l_j - t_n), \end{aligned} \quad (28)$$

and l_j is given by the solution of the following system of equations:

$$(U_n^b - G U_n^t) l_j = G u_n^t - u_n^b = -u_n^b. \quad (29)$$

With the choice of transformation matrices and vectors T and t given in (A13) and (A14), and an adequate frequency of conditioning, we can ensure that the system of equations (29) is arbitrarily well-conditioned. Once l_j is obtained from the solution of (29), all the other l vectors are determined recursively from the formulas (28); this means that the solution vectors g_{c_i} at all the conditioning points are obtained from (27) by solving only one system of equations, i.e., (29). Again, this method is the finite-difference analogue of the well-known multiple shooting technique for the solution of two-point boundary value problems [16].

We conclude this section with some further remarks on Conte's method and the stabilized march technique of Lucey and Hansen [5]. The transformation matrices T_{c_k} used by Conte are those needed for the Gram-Schmidt orthonormalization method [4]. The computation of these T_{c_k} 's requires approximately $2[\frac{1}{2}(L+1)]^3$ multiplications versus $[\frac{1}{2}(L+1)]^3$ multiplications for the inversion of a matrix of the same order used in our method. The evaluation of his transformation vectors t_{c_k} requires

the computation of $\frac{1}{2}(L+1)$ inner products of pairs of $\frac{1}{2}(L+1)$ -component vectors. Our choice of T_{c_k} 's and t_{c_k} 's, given in (A13) and (A14), results in a simpler and more efficient algorithm than one based on Conte's method.

Our transformation matrices T_{c_k} 's are the same forward transformation matrices used by Lucey and Hansen in their treatment of the eigenvalue problem for the transport equation [5]. However, in their stabilized march technique another matrix inversion such as that in (A13) is required to define a stabilizing transformation in a backward march; also, in order to match the forward and backward solutions, they have to solve a system such as (29) at each reconditioning point. The stabilized march technique thus requires at least three times more computation at each reconditioning point than the generalization of Conte's method used here, i.e., two matrix inversions and one linear system solution compared with one matrix inversion.

• Numerical algorithm

The first step in the algorithm to solve the block system of equations (19) is the computation of the untransformed (simple shooting) solution given by (23) and (24). To solve the system of equations (24), we use a linear system solver (LINSY1) based on Gaussian elimination with pivoting and iterative improvement of the solution, as described by Forsythe and Moler [21]. When system (24) is well-conditioned, as determined by LINSY1 [21], its solution gives us the angular flux at the slab center g_1 [see (19)], and the fluxes at the other mesh points are obtained by substitution in system (19). When marching from the slab center to the boundary, the matrix and vector sequences (21) and (22) are computed again because of storage limitations. For the homogeneous slab problems we have solved, all the π matrices in (21) are equal and thus we have to store only one matrix π . For typical reactor problems with a few material regions, we would need to store only one π matrix for each different region, and this would not pose storage problems.

When system (24) is ill-conditioned in the sense discussed in [21], LINSY1 sets a signal for the main program. Then the total spatial grid (see figure on page 358) is divided into two equal marching intervals by a reconditioning point at the midpoint c_2 . The reconditioning transformations at c_2 and at the right boundary point, $c_3 = n$, require the numerical inversion of $F_{c_2}^t$ and $F_n^t T_1 T_{c_2}$ [see (A13)]; only when the inversion of $F_{c_2}^t$ is performed by LINSY1 without any error messages is the U_{c_2} matrix [see (25)] computed and stored, and computation is continued to the slab boundary. If $F_{c_2}^t$ is instead ill-conditioned, the computation with two marching intervals is immediately stopped. The grid is then divided into four equal marching intervals by four equally

spaced conditioning points, $c_2, c_3, c_4, c_5 = n$, and the matrix inversions required at them are started [see (A13)]. Whenever the first ill-conditioned matrix is detected, the sweep is stopped immediately and the number of marching intervals doubled. The procedure is thus pursued, doubling the number of marching intervals each time until all the matrices involved in one given sweep are well-conditioned. Finally, the well-conditioned system (29) is solved and the solution vectors are obtained at all the conditioning points $c_1 = 1, c_2, \dots, c_j = n$ by means of the formulas (28) and (27). Between conditioning points, the solution is obtained by marching (substitution) through the inhomogeneous system (19). An error check is always done by comparing the solution vector g_n obtained from solving the linear system (29) [see also (27)] and that obtained by marching from the last conditioning point before the right boundary.

For the experimental homogeneous slab computations described subsequently, a maximum number of 33 conditioning points was programmed and these proved to be sufficient. The main storage requirements are quite modest, one square π matrix [see (21)], 33 rectangular matrices U_i [see (25)] and 33 square matrices T_i [see (A13)]. Since the slabs are homogeneous, the marching intervals were chosen naturally to have equal length, although it is not necessary to double at each successive sweep the number of them. Further numerical experimentation could indicate that it might be more efficient to add a fixed number of marching intervals at each successive sweep. For nonhomogeneous slabs, a more complicated strategy is required, as the length of the marching interval in each different material must be related to the corresponding neutron mean free path.

The determination of the proper number of conditioning points is of the greatest importance for the efficiency of a computer program, as most of the computation time is spent on the conditioning transformations. In this respect, two approaches are used. Lucey and Hansen [5] set the number of conditioning points in the program input. This might be quite inefficient because the number of conditioning points used might be much higher than is necessary. Compared with the other commonly used method of rescaling the matrices when the magnitude of any of their columns exceeds a certain empirical value [4, 16], the present method has the following distinct advantages:

1. No examination of matrices at each grid point is necessary; therefore, in stable problems the solution is obtained by the simple and direct shooting method without any extra computation.
2. The transformation of the matrices is carried out only when an accurate numerical solution of the problem cannot be obtained because of the ill-conditioning of the matrices, not because the values of their columns

exceed a certain empirical magnitude; this conventional criterion is not related directly to the accuracy of the numerical solution and might result in more conditioning points than are necessary.

We might conclude by stating that the efficiency of our algorithm for the more unstable problems is not as high as for the more stable ones, because of the extra matrix inversions wasted due to ill-conditioning. However, this is not a disadvantage in reactor calculations because, in the power method for criticality computations [22], it is necessary to solve system (19) many times, changing only the inhomogeneous terms on the right-hand side. Therefore, the search for the proper number of conditioning points is made only once during the solution of the first source problem. Thereafter, the same well-conditioned sequence of matrices U_i [see (25)] is used, and each new source problem requires only the computation of the vector sequence (26).

Numerical results and discussion

• Isotropic scattering problems

We now discuss the numerical results obtained with our experimental computer program for some critical-length calculations in homogeneous slabs with isotropic scattering. These calculations were first reported by Carlson and Bell [23] and have become a standard used by many in numerical neutron transport studies. The transport equation to be solved is

$$\mu[\partial\phi(x, \mu)/\partial x] + \Sigma_t \phi = \frac{1}{2}c \Sigma_t \int_{-1}^1 \phi(x, \mu') d\mu' \quad (30)$$

with the boundary conditions

$$\begin{aligned} \phi(0, \mu) &= \phi(0, -\mu) \text{ and} \\ \phi(R, \mu) &= 0 \quad -1 < \mu < 0. \end{aligned} \quad (31)$$

In (30) the total cross section Σ_t is taken as unity so that the total mean free path is also unity, and c is the average number of secondary neutrons emitted per collision. For a given slab size, the problem (30) and (31) has a solution only if we replace c by c' with

$$c' = c/\lambda, \quad \lambda \neq 1.$$

Here λ is the criticality eigenvalue. The critical length problem is to determine the slab thickness for which $\lambda = 1$. Our program searches for the critical half-thickness by computing λ for a succession of slab thicknesses after starting with an estimated value. The criticality λ for each size is obtained using the power method [22]. Linear interpolation of the successive half-thicknesses is performed to converge to the critical half-thickness with a criticality $\lambda = 1$. The error criterion used for the computed λ is $|\lambda_{\text{computed}} - 1| < 10^{-6}$.

Table 1 Slab critical half-thickness in units of mean free path by the P_3 approximation.

Secondary neutrons per collision, c	Marshak's boundary conditions		
	Lucey and Hansen ^a	This work ^b	Analytic ^c
1.02	5.6715	5.6710 ^d	5.6711
1.05	3.3067	3.3065	3.3066
1.1	2.1214	2.1213	2.1213
1.2	1.30205	1.30200	1.30200
1.4	0.75769	0.75766	0.75766
1.6	0.53839	0.53837	0.53837
1.8	0.41822	0.41821	0.41821
2.0	0.34206	0.34205	0.34205

^aWith 100 intervals, 20 reconditioning points, and single precision arithmetic [5].

^bWith 128 intervals, no reconditioning points (except where noted), and double precision arithmetic.

^cRef. 5.

^dThis was the only case in which two reconditioning points were needed.

In Table 1 we give the results obtained with the P_3 approximation and Marshak's boundary conditions for the critical half-thickness corresponding to several values of the number of secondary neutrons emitted per collision. For comparison purposes, we also give the values obtained by Lucey and Hansen [5] using the stabilized march technique, and those obtained with the analytic solution of the P_3 approximation (see also [6], p. 116). A comparison of the accuracy obtained by Lucey and Hansen with that in the present work is not meaningful because we use double precision arithmetic whereas they used single precision. Table 1 shows the power and efficiency of our method for automatically selecting the required number of reconditioning points during the computation. Only for the case $c = 1.02$ was a slight instability detected. This was eliminated by the program with only two reconditioning points. The amount of computation required by our program is thus substantially lower than in the stabilized march technique. However, this comparison is valid only if the stabilized march technique was used to solve (30) by the power method. Lucey and Hansen instead solved the eigenvalue problem (30) directly. Nevertheless, the comparison is meaningful as to the number of reconditioning points used by each method, and the amount of computation at each of these points.

In Table 2 we show the results of high order P_L computations for the thickest slab reported by Carlson and Bell. In a previous section [see discussion following Eq. (13)] we mentioned that the roundoff errors grow exponentially with distance; it is also known (see Davison, Ref. [6], Ch. X) that, because of the variation of the relaxation lengths with the order of the P_L approximation, the roundoff errors also grow for the higher P_L .

Table 2 Comparison of Federighi's with Marshak's boundary conditions for higher-order P_L calculations. Slab critical half-thickness in units of mean free path for $c = 1.02$ (secondary neutrons per collision). Exact value^d is 5.6655.

Order of Approx.	Marshak's boundary conditions		Federighi's boundary conditions	No. of recond. points	No. of inversions required ^c
	Lucey ^a	This work ^b			
P_3	5.6715	5.6710	5.6638	2	1
P_5	5.6682	5.6676	5.6647	4	2
P_7	5.6671	5.6666	5.6650	4	2
P_9	5.6667	5.6662	5.6651	8	3
P_{11}	5.6664	5.6659	5.6652	8	3
P_{13}	5.6663	5.6658	5.6654	8	3
P_{15}	5.6662	5.6656	5.6655	16	4
P_{17}	5.6661	5.6656		16	4
P_{19}		5.6655		16	4

^aWith 100 space intervals, 20 reconditioning points, and single precision arithmetic [11].

^bWith 128 space intervals and double precision arithmetic.

^cShows the computational cost of the automatic selection of the number of reconditioning points. The order of the matrices is $\frac{1}{2}(L+1)$.

^dRef. 23.

Table 3 Convergence of the spherical harmonics method to the exact values of the slab critical half-thickness in units of mean free paths, using Federighi's boundary conditions.^a

Secondary neutrons per collision, c	Order of approximation					Exact value ^b
	P_3	P_5	P_7	P_9	P_{11}	
1.02	5.6638(2)	5.6647(4)	5.6650(4)	5.6651(8)	5.6652(8)	5.6655
1.05	3.2989(0)	3.2996(2)	3.2998(4)	3.3000(4)	3.3000(4)	3.3002
1.1	2.1128(0)	2.1125(0)	2.1129(2)	2.1130(2)	2.1131(4)	2.1134
1.2	1.2920(0)	1.2886(0)	1.2888(0)	1.2891(2)	1.2892(2)	1.2893
1.4	0.7460(0)	0.7369(0)	0.7360(0)	0.7361(0)	0.7363(0)	0.7366
1.6	0.5263(0)	0.5141(0)	0.5118(0)	0.5115(0)	0.5115(0)	0.5120
1.8	0.4063(0)	0.3928(0)	0.3894(0)	0.3885(0)	0.3884(0)	0.3887
2.0	0.3307(0)	0.3166(0)	0.3124(0)	0.3111(0)	0.3107(0)	0.3108

^aAll cases with 128 equal intervals; the numbers of reconditioning points required are given in parentheses.

^bRef. 23.

calculations. The computations shown in Table 2 are thus a good test of the capability of the method to deal with strongly unstable problems. These results are further evidence that Federighi's conditions [8] give consistently better results than Marshak's conditions [1] in the context of critical-length computations. Table 2 also indicates clearly the advantages of the automatic setting of the number of reconditioning points by the program. This advantage is most significant for the more stable problems, where the number of required reconditioning points is considerably lower than that used by Lucey and Hansen. For the more unstable problems (higher P_L computations), this advantage decreases. It was considered worthwhile to record the approximate cost in computation of the automatic setting of the number of reconditioning points; this cost is due mainly to the number of ill-conditioned matrices whose inversion by LINSY1 fails, as discussed in the previous section.

In Table 3 we give the results obtained for critical length computations with several P_L approximations and Federighi's conditions. These results indicate the rapid convergence of the spherical harmonics method to the exact values when used in conjunction with Federighi's conditions. The efficiency of the automatic setting of the number of reconditioning points is illustrated in a striking way, because many of the problems were solved accurately by the simple shooting method without any reconditioning whatever.

The increased numerical instability encountered for the higher P_L calculations for isotropic problems in thick slabs has a simple physical explanation. In the limit of an infinitely thick slab with a uniform and isotropic fission source, the scalar flux f_0 [see Eqs. (5) and (6)] is constant and all higher moments vanish because the angular flux φ [see (2)] becomes isotropic. Thus in high P_L computations for the thicker slab problems the limiting

case is being approached, and as the first component of the solution vector g_i [in (23)] becomes orders of magnitude larger than the other components, the numerical instability arises.

A slightly modified version of the present program has been used recently to solve the equations of radiative transfer in a homogeneous nonabsorbing Rayleigh atmosphere [24] with an optical thickness of 16. This problem is subject to severe roundoff instability but was solved accurately with 16 conditioning points.

• Anisotropic scattering problems

We have determined the critical half-thickness of a plane homogeneous slab with anisotropic scattering. The problem solved was the following:

$$\begin{aligned} \mu \partial \varphi(x, \mu) / \partial x + \Sigma_t \varphi = & \frac{1}{2} (\Sigma_s^{\text{iso}} + \nu \Sigma_f) \int_{-1}^1 \varphi(x, \mu') d\mu' \\ & + \Sigma_s^{\text{aniso}} \sum_{n=0}^2 \frac{1}{2} (2n+1) b_n P_n(\mu) \int_{-1}^1 P_n(\mu') \varphi(x, \mu') d\mu', \end{aligned} \quad (32)$$

with the boundary conditions

$$\begin{aligned} \varphi(0, \mu) &= \varphi(0, -\mu) \text{ and} \\ \varphi(R, \mu) &= 0 \quad -1 < \mu < 0, \end{aligned} \quad (33)$$

where $x = 0$ is the slab center and $x = R$ is the right vacuum boundary. The solution of problem (32) and (33) has been reported by Lathrop and Leonard [25], who gave not only a numerical solution obtained with a S_{16} approximation using the DTF code [25], but also exact solutions obtained by the method of singular integral equations. Their notation is followed closely here, and the reader is referred to their paper for more details. Eqs. (32) and (33) were solved for different values of the secondary neutron ratio $c + c'$, where

$$c = \Sigma_s^{\text{aniso}} / \Sigma_t \quad (34)$$

is the anisotropic-scattering ratio and

$$c' = (\Sigma_s^{\text{iso}} + \nu \Sigma_f) / \Sigma_t \quad (35)$$

is the isotropic secondary-neutron ratio. For each value of $c + c'$, c is increased to permit observation of the effects of the increased anisotropy in the scattering. The values used for b_n are those of elastic hydrogen scattering

$$b_0 = 1, \quad b_1 = 2/3, \quad b_2 = 1/4, \quad b_3 = 0, \quad b_4 = -1/24. \quad (36)$$

The values (36) result from expanding the scattering kernel in a five-term Legendre polynomial series. With our method we can expand the scattering kernel unrestrictedly in a Legendre series with exactly $L + 1$ terms when performing a P_L calculation. Further, the actual

computations are independent of the number of terms kept in this expansion.

The results of the most interesting calculations reported in [25] are given in Table 4 with our P_L results obtained using Federighi's boundary conditions. In all cases, 64 spatial intervals were used.

We chose the calculations for the minimum value of secondary neutrons per collision $c + c' = 1.05$ reported in [25], because this value results in the thickest slabs, where the roundoff effects are greatest. From a computational point of view the S_{16} calculations are exactly equivalent to P_{15} calculations (i.e., both are a two-point boundary value problem for a system of 16 differential equations). The convergence of the P_L computations to the analytic results of Leonard is shown clearly in the upper portion of the table. Our P_{15} results are in closer agreement with the analytic values than are the S_{16} results.

In the other extreme we chose the maximum value, $c + c' = 1.4$ neutrons per collision reported in [25]. The angular flux is more anisotropic than for $c + c' = 1.05$, because the slabs are much thinner and thus the flux anisotropy due to boundary effects is higher. In a numerical sense, this has the effect that the higher components of the solution vector g_i [in (23)] become of the same order of magnitude as the first component (the scalar flux) and thus the numerical problem becomes much more stable. Indeed for the thinner slabs listed at the bottom of Table 4, the simple shooting method without any reconditioning points is sufficient to obtain an accurate numerical solution.

Appendix

We show here that Conte's analysis is valid for arbitrary transformation matrices and vectors. With reference to the figure on page 358 of the text, the $j - 1$ intervals defined by the conditioning points $c_1 = 1, c_2, \dots, c_j = n$ are called marching intervals. We now define the following sequence of rectangular matrices $(L + 1) \times \frac{1}{2}(L + 1)$:

$$U_i = \begin{cases} F_i T_1 = F_i I = F_i & i = 1, 2, \dots, c_2 - 1 \\ F_i T_1 T_{c_2} & i = c_2, c_2 + 1, \dots, c_3 - 1 \\ \vdots & \\ F_i T_1 T_{c_2} \cdots T_{c_k} & i = c_k, c_k + 1, \dots, c_{k+1} - 1 \\ \vdots & \\ F_i T_1 T_{c_2} \cdots T_n & i = n, \end{cases} \quad (A1)$$

where the F_i 's are defined in (21), and the arbitrary (except that $T_1 \equiv I$) square transformation matrices T_{c_k} of order $\frac{1}{2}(L + 1)$ are to be determined so as to assure that the transformed matrices U_i are well-conditioned, i.e., that their columns are linearly independent. The vectors a_i [see (22)] must also be transformed in order

Table 4 Slab critical half-thickness in units of mean free path for anisotropic scattering.

$c + c' = 1.05$					
c	S_{16}^a	P_{11}^b	P_{13}^b	P_{15}^b	Exact value ^c
0.1	3.39042	3.39025 ^d	3.39030	3.39033	3.39032
0.3	3.59724	3.59705 ^d	3.59710	3.59714	3.59714
0.5	3.85135	3.85114	3.85120	3.85125	3.85126
0.7	4.17506	4.17481	4.17489	4.17494	4.17495
0.9	4.60935	4.60905	4.60915	4.60921	4.60927

$c + c' = 1.4$				
c	S_{16}^a	P_{15}^e	Exact value ^c	
0.1	0.74532	0.74519	0.74529	
0.3	0.76381	0.76375	0.76378	
0.5	0.78393	0.78396	0.78396	
0.7	0.80595	0.80608	0.80610	
0.9	0.83022	0.83046	f	

^aDTF transport code [25].

^bWith Federighi boundary conditions, 64 intervals and eight reconditioning points except where noted.

^cResults from singular integral equation method [25].

^dWith four reconditioning points.

^eWith Federighi boundary conditions, 64 intervals and no reconditioning.

^fNot reported; see [25].

that the new vectors $\mathbf{u}_i$ be linearly independent of the columns of U_i . This can be achieved by the transformations

$$\mathbf{u}_1 = \mathbf{0},$$

$$\mathbf{u}_{c_2} = \mathbf{a}_{c_2} - F_{c_2} T_1 T_{c_2} \mathbf{t}_{c_2} = \mathbf{a}_{c_2} - U_{c_2} \mathbf{t}_{c_2},$$

$$\begin{aligned} \mathbf{u}_{c_3} &= \mathbf{a}_{c_3} - F_{c_3} T_1 T_{c_2} \mathbf{t}_{c_2} - F_{c_3} T_1 T_{c_2} T_{c_3} \mathbf{t}_{c_3} \\ &\vdots \\ &= \mathbf{a}_{c_3} - F_{c_3} T_1 T_{c_2} \mathbf{t}_{c_2} - U_{c_3} \mathbf{t}_{c_3}, \end{aligned}$$

$$\begin{aligned} \mathbf{u}_n &= \mathbf{a}_n - F_n T_1 T_{c_2} \mathbf{t}_{c_2} - F_n T_1 T_{c_2} T_{c_3} \mathbf{t}_{c_3} - \cdots - F_n T_1 T_{c_2} \\ &\quad \cdots T_{c_{j-1}} T_n \mathbf{t}_n = \mathbf{a}_n - F_n T_1 T_{c_2} \mathbf{t}_{c_2} - F_n T_1 T_{c_2} T_{c_3} \mathbf{t}_{c_3} \\ &\quad \cdots - F_n T_1 \cdots T_{c_{j-1}} \mathbf{t}_{c_{j-1}} - U_n \mathbf{t}_n, \end{aligned} \quad (\text{A2})$$

where the $\frac{1}{2}(L+1)$ -component transformation vectors $\mathbf{t}_{c_k}$ are unspecified. As already pointed out, between conditioning points the vector sequence is determined by substitution in (19) [see (22)].

As in the first marching interval the sequences U_i and $\mathbf{u}_i$ are the same as the untransformed sequences F_i and $\mathbf{a}_i$ of the exact solution given by (23) and (24), the solution can be written as

$$\mathbf{g}_i = U_i \mathbf{l}_1 - \mathbf{u}_i, \quad i = 1, 2, \dots, c_2 - 1, \quad (\text{A3})$$

where $\mathbf{l}_1$ is the $\frac{1}{2}(L+1)$ -component vector equal to the top half components of $\mathbf{g}_1$. In order to use only well-conditioned matrices in our computation, we express the solution in all the mesh points as follows:

$$\mathbf{g}_i = U_i \mathbf{l}_k + \mathbf{u}_i \quad c_k \leq i < c_{k+1},$$

$$\mathbf{g}_n = U_n \mathbf{l}_j + \mathbf{u}_n, \quad (\text{A4})$$

where the $\frac{1}{2}(L+1)$ -component vectors $\mathbf{l}_i$ are to be determined by matching the solution at the conditioning points. At the second conditioning point c_2 , the matching of the untransformed and transformed solutions gives

$$\mathbf{g}_{c_2} = F_{c_2} \mathbf{l}_1 + \mathbf{a}_{c_2} = U_{c_2} \mathbf{l}_2 + \mathbf{u}_{c_2}. \quad (\text{A5})$$

From (A5), using the definitions of U_{c_2} and $\mathbf{u}_{c_2}$ given in (A1) and (A2), we obtain

$$\mathbf{l}_1 = T_{c_2} (\mathbf{l}_2 - \mathbf{t}_{c_2}). \quad (\text{A6})$$

Thus, in the second marching interval the solution is expressed as

$$\mathbf{g}_i = U_i \mathbf{l}_2 + \mathbf{u}_i \quad i = c_2, c_2 + 1, \dots, c_3 - 1, \quad (\text{A7})$$

which is obtained by substitution in the full system (19), with the assumption that the solution at c_2 is known, as given by the right side of (A5). In terms of $\mathbf{l}_2$, the solution at c_3 is

$$\mathbf{g}_{c_3} = F_{c_3} T_1 T_{c_2} \mathbf{l}_2 + \mathbf{a}_{c_3} - F_{c_3} T_1 T_{c_2} \mathbf{t}_{c_2}, \quad (\text{A8})$$

which can be verified by substituting into (A8) the expression for $\mathbf{l}_2$ given by (A6), in which case (A8) becomes the exact (untransformed) solution (23). Matching (A8) with the newly transformed solution, we get

$$\mathbf{g}_{c_3} = F_{c_3} T_1 T_{c_2} \mathbf{l}_2 + \mathbf{a}_{c_3} - F_{c_3} T_1 T_{c_2} \mathbf{t}_{c_2} = U_{c_3} \mathbf{l}_3 + \mathbf{u}_{c_3}, \quad (\text{A9})$$

so that

$$\mathbf{l}_2 = T_{c_3} (\mathbf{l}_3 - \mathbf{t}_{c_3}). \quad (\text{A10})$$

The general formula obtained from matching at the successive conditioning points including the right boundary point is

$$\begin{aligned} \mathbf{l}_{k-1} &= T_{c_k} (\mathbf{l}_k - \mathbf{t}_{c_k}) & k = 2, 3, \dots, j-1. \\ \mathbf{l}_{j-1} &= T_n (\mathbf{l}_j - \mathbf{t}_n). \end{aligned} \quad (\text{A11})$$

We now use the second equation in (A4) and the right boundary condition in (19) to obtain the inhomogeneous system of $\frac{1}{2}(L+1)$ equations for the $\frac{1}{2}(L+1)$ components of $\mathbf{l}_j$,

$$(U_n^b - G U_n^t) \mathbf{l}_j = G \mathbf{u}_n^t - \mathbf{u}_n^b. \quad (\text{A12})$$

Therefore, the solution to our problem (19) is given by the general formulas (A4), (A11), and (A12), independently of the transformation matrices and vectors T and $\mathbf{t}$, as was to be shown.

In our work we chose the following transformation matrices and vectors:

$$\begin{aligned} T_1 &= I, \\ T_{c_2} &= (F_{c_2}^t T_1)^{-1} = (F_{c_2}^t)^{-1}, \\ T_{c_3} &= (F_{c_3}^t T_1 T_{c_2})^{-1}, \\ &\vdots \\ T_n &= (F_n^t T_1 T_{c_2} \cdots T_{c_{j-1}})^{-1}, \end{aligned} \quad (\text{A13})$$

and

$$\begin{aligned} \mathbf{t}_1 &= \mathbf{0}, \\ \mathbf{t}_{c_2} &= (\mathbf{a}_{c_2} - F_{c_2} T_1 \mathbf{t}_1)^t = \mathbf{a}_{c_2}^t, \\ \mathbf{t}_{c_3} &= (\mathbf{a}_{c_3} - F_{c_3} T_1 T_{c_2} \mathbf{t}_{c_2})^t, \\ \mathbf{t}_n &= (\mathbf{a}_n - F_n T_1 T_{c_2} \mathbf{t}_{c_2} - \cdots - F_n T_1 T_{c_2} \cdots T_{c_{j-1}} \mathbf{t}_{c_{j-1}})^t. \end{aligned} \quad (\text{A14})$$

Substituting (A13) and (A14) into (A1) and (A2) and carrying out the matrix and vector products, we obtain the reconditioned sequences U_i and $\mathbf{u}_i$ given explicitly in the text [(25) and (26)].

References

1. E. M. Gelbard, "Spherical Harmonics Methods: P_L and Double- P_L Approximations", in *Computing Methods in Reactor Physics*, H. Greenspan, C. N. Kelber and D. Okrent, Eds., Gordon and Breach, Science Publishers, New York, 1968.
2. R. G. Gordon, *J. Chem. Phys.* **51**, 14 (1969).
3. S. Godunov, *Uspekhi Mat. Nauk* **16**, 171 (1961).
4. S. D. Conte, *SIAM Review* **8**, 309 (1966).
5. J. W. Lucey and K. F. Hansen, *Nucl. Sci. Eng.* **33**, 327 (1968).
6. B. Davison, *Neutron Transport Theory*, Oxford University Press, London, 1958.
7. A. Erdelyi, et al., *Higher Transcendental Functions*, Vols. I and II, McGraw-Hill Publishing Co., Inc., New York, 1953.
8. F. D. Federighi, *Nukleonik* **6**, 277 (1964).
9. J. Canosa and H. R. Penafiel, "Numerical Solution of the Spherical Harmonics Approximation to the Neutron Transport Equation", Report 320-3295, IBM Scientific Center, Palo Alto, California, 1971.
10. G. C. Pomraning, *Nucl. Sci. Eng.* **18**, 528 (1964).
11. J. Canosa, *Nucl. Sci. Eng.* **43**, 349 (1971).
12. G. C. Pomraning, *Nucl. Sci. Eng.* **22**, 259 (1965).
13. H. Amster, *Nucl. Sci. Eng.* **22**, 254, 262 (1965).
14. J. A. Davis, *Nucl. Sci. Eng.* **25**, 189 (1966).
15. P. Henrici, *Discrete Variable Methods in Ordinary Differential Equations*, John Wiley and Sons, New York, 1962, Ch. III.
16. H. B. Keller, *Numerical Methods for Two-Point Boundary-Value Problems*, Ginn-Blaisdell, Waltham, Mass., 1968, Ch. II.
17. J. Canosa and R. G. deOliveira, *J. Comp. Phys.* **5**, 188 (1970).
18. J. Canosa, *J. Comp. Phys.* **7**, 255 (1971).
19. W. L. Miranker, *SIAM Review* **13**, 524 (1971).
20. M. R. Osborne, *J. Math. Anal. Appl.* **27**, 417 (1969).
21. G. Forsythe and C. B. Moler, *Computer Solution of Linear Algebraic Systems*, p. 58, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1967.
22. R. S. Varga, "Numerical Methods for Solving Multi-Dimensional Multigroup Diffusion Equations", in *Proc. XI Symp. Appl. Math.*, G. Birkhoff and E. P. Wigner, Eds., American Mathematical Society, Providence, Rhode Island, 1961.
23. B. G. Carlson and G. I. Bell, "Solutions of the Transport Equation by S_n Method", in *Proc. Int. Conf. Peaceful Uses Atomic Energy*, Geneva **16**, 535 (1958).
24. J. V. Dave and W. H. Walker, *Astrophys. J.* **144**, 798 (1966).
25. K. D. Lathrop and A. Leonard, *Nucl. Sci. Eng.* **22**, 115 (1965).

Received November 17, 1971

The authors are located at the IBM Data Processing Division Scientific Center, 2670 Hanover Street, Palo Alto, California 94304.