LA-4848-MS

INFORMAL REPORT
UC-32
ISSUED: July 1973

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87544

TWOTRAN-II: An Interfaced, Exportable Version of the TWOTRAN Code for Two-Dimensional Transport

by

K. D. Lathrop
F. W. Brinkley

Partial support for this work was received from the AEC Division of Reactor Research and Development.

This report revises LA-4432.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

CONTENTS

Page

(CONTENTS - continued) Page

FIGURES Page

TABLES

TWOTRAN-II, AN INTERFACED, EXPORTABLE VERSION OF THE TWOTRAN CODE FOR TWO-DIMENSIONAL TRANSPORT

by

K. D. Lathrop

F. W. Brinkley

ABSTRACT

1. Program Identification: TWOTRAN-II.
2. Computer for which program is designed and others on which it is operable: CDC-7600, CDC-6600, IBM 360/195.
3. Description of Function: TWOTRAN-II solves the two-dimensional multigroup transport equation in (x,y), (r,θ), and (r,z) geometries. Both regular and adjoint, inhomogeneous and homogeneous (k_{eff} and eigenvalue searches) problems subject to vacuum, reflective, periodic, white or input specified boundary flux conditions are solved. General anisotropic scattering is allowed and anisotropic inhomogeneous sources are permitted.
4. Method of Solution: The discrete ordinates approximation for the angular variable is used in finite difference form which is solved with the central (diamond) difference approximation. Negative fluxes are eliminated by a local set-to-zero and correct algorithm. Standard inner (within-group) and outer iterative cycles are accelerated by coarse-mesh rebalancing on a coarse mesh which may be independent of the material mesh.
5. Restrictions: Variable dimensioning is used so that any combination of problem parameters leading to a container array less than MAXLEN can be accomodated. On the CDC machines MAXLEN can be slightly greater than 40,000 words and peripheral storage is used for most group-dependent data. On IBM machines, TWOTRAN-II will execute in the 4-byte mode so that MAXLEN can be several hundred thousand and most problems can be core-contained.
6. Running Time: A six-group, S_4, 42 x 42 mesh point, k_{eff} calculation of an EBR-II model requires about 3.9 minutes of CDC-7600 time.
7. Unusual Features of the Program: Provision is made for creation of standard interface output files for S_n constants, angle-integrated fluxes and angular fluxes. Standard interface input files for sources, fluxes, cross sections and S_n constants may be read. All binary operations are localized in subroutines called REED and RITE. Detailed edit options, including angular fluxes, dumps and restart capability are provided. Optional use of an arbitrary rebalance mesh independent of the material mesh is allowed.
8. Machine Requirements: Five output units, five interface units (use of interface units is optional) and two system input/output units are required. A large bulk memory is desirable, but it can be replaced by disk, drum or tape storage.
9. Related Programs: TWOTRAN-II is an improved version of the TWOTRAN program. Many comment cards were added and much simplifying programming was performed to make TWOTRAN-II as easy to understand as possible.
10. Material Available: Source deck, test problems, results of executed test problems and this report will be available from the Argonne Code Center.

I. INTRODUCTION

TWOTRAN-II is an improved version of the TWOTRAN program[1] for two-dimensional transport calculations in (x,y) planar, (r,θ) cylindrical planar, and (r,z) cylindrical geometries. The program solves both regular and adjoint, homogeneous or inhomogeneous, time-independent problems subject to vacuum, reflective, periodic, white or input specified boundary flux conditions.

TWOTRAN-II was created to become the main frame for the TRIPLET (triangular mesh, planar, explicit, transport) code[2] and as such it was designed to be as easy as possible to make operational on computers other than CDC manufactured machines. A version of TWOTRAN-II successfully compiled and executed test problems (in the 4-byte mode) on the first try on the Argonne National Laboratory IBM 360/195 computer. TWOTRAN-II also was modified to include some of the standard interface input and output data files.[3] The code now optionally creates output files for S_n constants, angle-integrated fluxes, and angular fluxes and reads standard input data files for cross sections, inhomogeneous sources, fluxes, and S_n constants. In addition to dump restructuring, elimination of installation dependent routines, localization of binary operations and other clean-up operations associated with making TWOTRAN-II easily exportable, several functional improvements were made in the TWOTRAN code. These include:

1. the optional separation of the rebalance mesh from the material mesh to save storage in problems with many hundreds of different material regions,

2. the optional printing and/or writing on an output file of cell-boundary angular fluxes,

3. the use of an effective absorption implied by the input cross sections,

4. the option of specifying boundary sources for all types of boundary conditions,

5. the ability to suppress output printing for input fluxes, final fluxes, cross sections, fission rates, sources and coarse mesh balance tables. If an extra outer iteration is not required for printing, it is not taken, saving computation time.

6. the option of specifying a separable pointwise density for cross section spatial dependence,

7. the addition of a parametric value of k_{eff} in alpha searches,

8. the option of reading cross sections in FIDO[4] or Los Alamos formats as well as from the standard interface file.

Details of these features as well as a complete description of all options of TWOTRAN-II are given in Section III.

Because the manual[1] describing TWOTRAN is out of print, much of the material in that report is reproduced here. The contents of this report follow the guidelines for the documentation of digital computer programs[5] proposed as an American Nuclear Society standard. In the next section, difference equations and solution algorithms are discussed. Section III is a guide for user application and Section IV contains information relevant to the programming of TWOTRAN-II.

II. DIFFERENCE EQUATIONS AND SOLUTION ALGORITHMS

A. Analytic Forms of the Transport Equation

The time-independent transport equation is written

$$\nabla\cdot(\underline{\Omega}\psi) + \sigma_t\psi(\underline{r},E,\underline{\Omega}) = \iiint dE'd\Omega'\ \psi(\underline{r},E',\underline{\Omega}')\sigma_s(E' \rightarrow E,\ \underline{\Omega}'\cdot\underline{\Omega}) \qquad (1)$$
$$+\ \chi(E)\iiint dE'd\Omega'\ \psi(\underline{r},E',\underline{\Omega}')\ \nu\sigma_f(E')/4\pi + Q(\underline{r},E,\underline{\Omega})\ ,$$

in which ψ is the particle flux (number density of particles times their speed) defined such that $\psi dVdEd\Omega$ is the flux of particles in the volume element dV about $\underline{r}$, in the element of solid angle dΩ about $\underline{\Omega}$, in the energy range dE about E. Similarly, $QdVdEd\Omega$ is the number of particles in the same element of phase space emitted by sources independent of ψ. The macroscopic total interaction cross section is denoted by σ_t, the macroscopic scattering transfer probability (from energy E' to E through an angle with cosine $\underline{\Omega}'\cdot\underline{\Omega}$) by σ_s, and the macroscopic fission cross section by σ_f. All of these quantities may depend on $\underline{r}$, but we have omitted this argument for simplicity. The number of particles emitted isotropically (1/4π) per fission is ν, and the fraction of these liberated in the range dE about E is χ(E). This fraction may actually depend on both $\underline{r}$ and the fissioning species, but such possibilities are not admitted in the TWOTRAN-II program.

1. Particular Forms of the Divergence Operator

The form of $\nabla\cdot\underline{\Omega}\psi$ for the three geometries treated by TWOTRAN-II is given in Table I in terms of the variables defined in the coordinate systems sketched in Figs. 1-3.

In all of these geometries we choose φ to be the angle of rotation about the μ axis such that $d\Omega = d\mu d\varphi$, and

$$\xi = (1 - \mu^2)^{\frac{1}{2}} \sin\varphi \quad ,$$
$$\eta = (1 - \mu^2)^{\frac{1}{2}} \cos\varphi \quad . \qquad (2)$$

In all of these geometries ψ is symmetric in φ, and only a hemisphere of angular directions need be considered. In the planar geometries [(x,y) and (r,θ)]

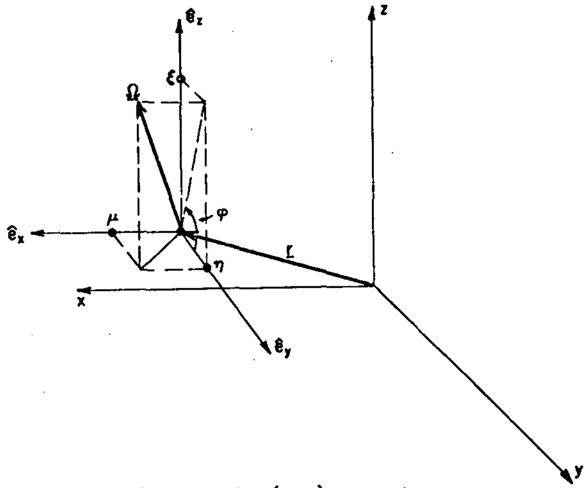


Fig. 1. Coordinates in (x,y) geometry.

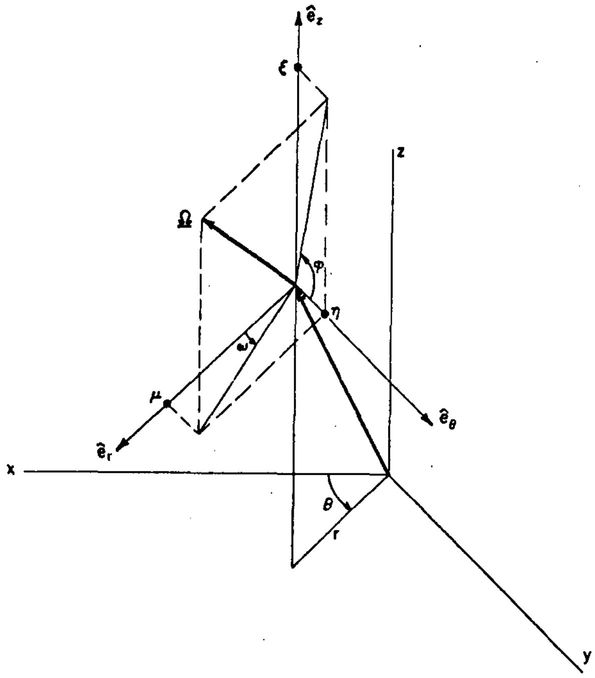


Fig. 2. Coordinates in (r,θ) geometry.

TABLE I

FORMS OF $\nabla\cdot\underline{\Omega}\psi$

Geometry	Dependence of ψ	Definition of Variables	$\nabla\cdot\underline{\Omega}\psi$
rectangular planar	$\psi(x,y,\mu,\eta)$	$\mu = \hat{e}_x \cdot \underline{\Omega}$ $\eta = \hat{e}_y \cdot \underline{\Omega}$	$\mu \frac{\partial\psi}{\partial x} + \eta \frac{\partial\psi}{\partial y}$
cylindrical planar	$\psi(r,\theta,\mu,\varphi)$	$\mu = \hat{e}_r \cdot \underline{\Omega}$ $\eta = \hat{e}_\theta \cdot \underline{\Omega}$ $\xi = \hat{e}_z \cdot \underline{\Omega}$ $\mu = (1 - \xi^2)^{\frac{1}{2}} \cos\omega$ $\eta = (1 - \xi^2)^{\frac{1}{2}} \sin\omega$	$\frac{\mu}{r}\frac{\partial(r\psi)}{\partial r} - \frac{1}{r}\frac{\partial(\eta\psi)}{\partial\omega} + \frac{\eta}{r}\frac{\partial\psi}{\partial\theta}$
cylindrical finite	$\psi(r,z,\mu,\varphi)$	$\mu = \hat{e}_r \cdot \underline{\Omega}$ $\eta = \hat{e}_z \cdot \underline{\Omega}$ $\xi = -\hat{e}_\theta \cdot \underline{\Omega}$ $\mu = (1 - \eta^2)^{\frac{1}{2}} \cos\omega$ $\xi = (1 - \eta^2)^{\frac{1}{2}} \sin\omega$	$\frac{\mu}{r}\frac{\partial(r\psi)}{\partial r} - \frac{1}{r}\frac{\partial(\xi\psi)}{\partial\omega} + \eta\frac{\partial\psi}{\partial z}$

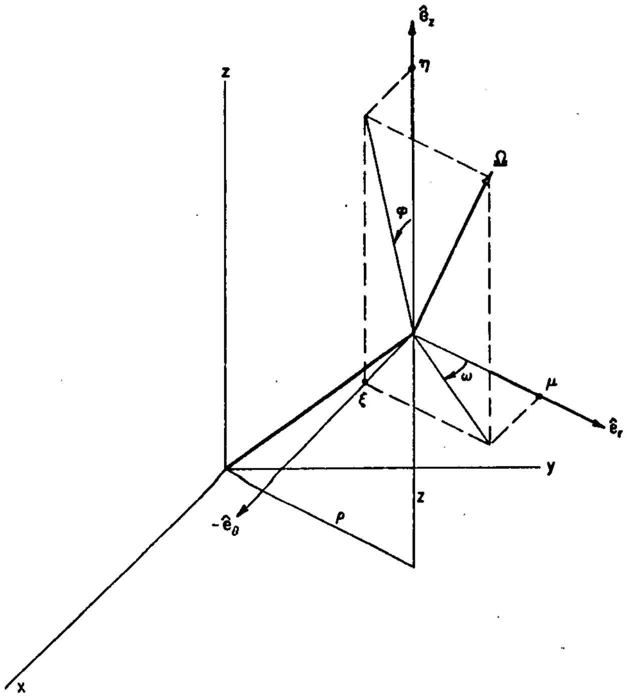


Fig. 3. Coordinates in (r,z) geometry.

the north pole of the hemisphere points in the $\hat{e}_z$ direction; in (r,z) geometry the north pole points in the $-\hat{e}_\theta$ direction.

2. Spherical Harmonic Expansion of the Source Terms

In the TWOTRAN-II program, the scattering transfer probability is assumed to be represented by a finite Legendre polynomial expansion:

$$\sigma_s(E' \rightarrow E, \mu_o) = \sum_{n=0}^{ISCT} \frac{2n+1}{4\pi} P_n(\mu_o)\sigma_{sn}(E' \rightarrow E) \quad ,$$

where $\mu_o = \underline{\Omega}\cdot\underline{\Omega}' = \mu\mu' + (1 - \mu^2)^{\frac{1}{2}} (1 - \mu'^2)^{\frac{1}{2}} \cos(\varphi - \varphi')$. If this expansion is inserted in Eq. (1), and if the addition theorem is used to expand $P_n(\mu_o)$, we can write

$$\iint d\Omega'\psi(\underline{r},E',\underline{\Omega}')\sigma_s(E' \rightarrow E, \mu_o) = \sum_{n=0}^{ISCT} \frac{2n+1}{2\pi} \sigma_{sn}(E' \rightarrow E) \sum_{k=0}^{n} R_n^k(\mu,\varphi) \int_{-1}^{1} d\mu' \int_{0}^{\pi} d\varphi' R_n^k(\mu',\varphi')\psi \quad . \tag{3}$$

In deriving this expression, the φ symmetry of ψ is used, reducing the domain of Ω to a hemisphere and eliminating expansion terms odd in φ. The functions R_n^k are defined by

$$R_n^k = \left[\frac{(2-\delta_{ko})(n-k)!}{(n+k)!}\right]^{\frac{1}{2}} P_n^k(\mu) \cos k\varphi \quad , \tag{4}$$

where δ_{ko} is the Kronecker delta (equal to 1 when k = 0, and vanishing otherwise) and the R_n^k are the associated Legendre polynomials. These functions are orthogonal; i.e.,

$$\int_{-1}^{1} d\mu \int_{0}^{\pi} d\varphi R_n^k(\mu,\varphi)R_m^\ell(\mu,\varphi) = \frac{2\pi}{2n+1} \delta_{nm}\delta_{k\ell} \quad . \tag{5}$$

Hence, if the angular flux is expanded in a series of these functions,

$$\psi = \sum_{n=0}^{\infty} \left(2n+1\right) \sum_{k=0}^{n} R_n^k\psi_n^k \quad , \tag{6}$$

the expansion coefficients are given by

$$\psi_n^k = \int_{-1}^{1} d\mu \int_{0}^{\pi} d\varphi R_n^k\psi/2\pi \quad . \tag{7}$$

Using this formula, we can rewrite Eq. (1) as

$$\nabla\cdot(\underline{\Omega}\psi) + \sigma_t\psi =$$

$$\int_0^\infty dE' \sum_{n=0}^{ISCT} (2n+1)\sigma_{sn}(E' \rightarrow E) \sum_{k=0}^{n} R_n^k(\mu,\varphi)\psi_n^k \tag{8}$$

$$+ \chi(E) \int_0^\infty dE' \nu\sigma_f(E')\psi_o^o + \sum_{n=0}^{IQAN} \left(2n+1\right) \sum_{k=0}^{n} R_n^k Q_n^k \quad .$$

As implied by the sum in this equation, we have assumed that the source Q is representable by IQAN+1 terms of an expansion like Eq. (6).

B. Multigroup Equations

The energy domain of interest is assumed to be partitioned into IGM intervals of width ΔE_g, g =

1, 2, ..., IGM. By convention, increasing g represents decreasing energy. If we multiply Eq. (8) by ΔE_g and integrate we can write

$$\nabla\cdot(\underline{\Omega}\psi_g) + \sigma_{tg}\psi_g =$$

$$\sum_{h=1}^{IGM}\sum_{n=0}^{ISCT}\left(2n+1\right)\sigma_{snh\to g}\sum_{k=0}^{n}R_n^k\psi_{nh}^k \qquad (9)$$

$$+\ \chi_g\sum_{h=1}^{IGM}\nu\sigma_{fh}\,\psi_{oh}^o + \sum_{n=0}^{IQAN}\left(2n+1\right)\sum_{k=0}^{n}R_n^kQ_{ng}^k \quad .$$

Here, the flux for group g,

$$\psi_g = \int_{\Delta E_g}\psi dE \quad , \qquad (10)$$

is no longer a distribution in energy, but is the total number of particles in the energy interval. For this reason, when group structures are changed, the effect on results must be evaluated by comparing $\psi_g/\Delta E_g$. Because of Eq. (10), energy integrals in TWOTRAN-II are evaluated by simple sums.

The cross sections subscripted with g are averages, e.g.,

$$\sigma_{tg} = \int_{\Delta E_g}\sigma_t\psi dE\Bigg/\int_{\Delta E_g}\psi dE \quad , \qquad (11)$$

but, of course, ψ is not known and must be approximated by some means. If in Eq. (11) the angular dependence of ψ is nonseparable, then σ_{tg} will depend on angle. No provision for such dependence is made in TWOTRAN-II. Recipes for taking this dependence into account, as well as for improving the averages $\sigma_{snh\to g}$ when scattering is severely anisotropic, are given by Bell and Hansen.[6]

C. Difference Equations

To concentrate on the finite difference representation of the streaming operator, we write Eq. (9) as

$$\nabla\cdot\underline{\Omega}\psi + \sigma_t\psi = S \quad , \qquad (12)$$

omitting the group subscript. Below, whenever we specialize to a particular geometry, we use (r,θ) cylindrical geometry. All of the results follow in the other geometries when the appropriate variables and definitions are used.

We first partition phase space into IT intervals in the r (or x) direction, JT intervals in the θ (y or z) direction, and MT solid angle intervals such that

$$\begin{aligned} &r_{i-\frac{1}{2}} < r_i < r_{i+\frac{1}{2}} && i = 1,\ 2,\ \ldots,\ IT \quad , \\ &\theta_{j-\frac{1}{2}} < \theta_j < \theta_{j+\frac{1}{2}} && j = 1,\ 2,\ \ldots,\ JT \quad , \\ &\underline{\Omega}_{m-\frac{1}{2}} < \underline{\Omega}_m < \underline{\Omega}_{m+\frac{1}{2}} && m = 1,\ 2,\ \ldots,\ MT \quad , \end{aligned} \qquad (13)$$

In this notation, all cell boundaries are indicated with a ± 1/2 subscript, but we deliberately refrain from specifying Ω-boundaries. We also use $d\Omega = d\xi d\omega/2\pi$ or $d\Omega = d\mu d\varphi/2\pi$, whichever is convenient. Here, the 2π arises from our convention of measuring solid angle in units of 2π so that angular integrals are normalized to unity.

We orient our geometries so that x, r, and r are in the i-direction while y, θ, and z are in the j-direction. We refer to j = JT as the top of the system, to j = 1 as the bottom, to i = 1 as the inside or left side, and to i = IT as the outside or right side of the system. Although not necessary in (x,y) geometry, in the cylindrical geometries we must have $r_1 = 0$, because we have made no provision for a cavity boundary condition.

We next multiply Eq. (12) by the phase space volume $(dVd\Omega)$ and integrate over one cell of our partitioning. For all geometries we represent the result by

$$\begin{aligned} &w_m\mu_m(A_{i+\frac{1}{2},j}N_{i+\frac{1}{2},j,m} - A_{i-\frac{1}{2},j}N_{i-\frac{1}{2},j,m}) \\ &\quad + (A_{i+\frac{1}{2},j} - A_{i-\frac{1}{2},j})(\alpha_{m+\frac{1}{2}}N_{i,j,m+\frac{1}{2}} - \alpha_{m-\frac{1}{2}}N_{i,j,m-\frac{1}{2}}) \\ &\quad + w_m\eta_m(B_{i,j+\frac{1}{2}}N_{i,j+\frac{1}{2},m} - B_{i,j-\frac{1}{2}}N_{i,j-\frac{1}{2},m}) \\ &\quad + \sigma_t w_m V_{ij}N_{ijm} = w_m V_{ij}S_{ijm} \quad , \end{aligned} \qquad (14)$$

and define

$$\text{volume} \quad V_{ij} = \iint_{ij} dV \equiv \iint_{ij} dAdB \quad ,$$

$$\begin{matrix}\text{i-direction}\\ \text{surface area}\end{matrix} \quad A_{i+\frac{1}{2},j} = \int_j dA_{i+\frac{1}{2}} \quad ,$$

j-direction surface area $B_{i,j+\frac{1}{2}} = \int_i dB_{j+\frac{1}{2}}$,

quadrature weight $w_m = \iint_m d\Omega$. (15)

The N's that appear in Eq. (14) are certain averages of ψ over the phase space cell or its boundaries. These averages and the truncation error associated with the difference equations in Eq. (14) are discussed in Ref. 1. The values of V, A, and B for the specific geometries of TWOTRAN-II are given in Table II. Note in (x,y) geometry in which there is no angular derivative that $A_{i+\frac{1}{2},j} = A_{i-\frac{1}{2},j}$ so that the term involving α vanishes from Eq. (14).

In the analytic form of the transport equation, when the flux is a constant, the terms analagous to those involving A in Eq. (14) vanish. To preserve this property in the finite difference form we require

$$\alpha_{m+\frac{1}{2}} - \alpha_{m-\frac{1}{2}} = -w_m\mu_m \qquad (16)$$

and use this relation to define the α coefficients recursively. As will be seen subsequently, direction cosines in standard sets are ordered on latitudes so that for each η level there will be several μ values. Defining the first value of α on each η level to be zero initializes the recursion. For the standard sets which are symmetric in μ, the last value of α for each η level will also be zero. This ensures that if Eq. (14) is summed over all directions, the terms in α will vanish. That is, there will be no net particle loss or gain due to angular redistribution, a property of the analytic equation.

The following conventions are used in TWOTRAN-II:

1. The angle theta is measured in revolutions,
2. Solid angles are measured in units of 2π,
3. Direction cosines are independent of space,
4. The α coefficients are defined recursively such that $\alpha_{m+\frac{1}{2}} - \alpha_{m-\frac{1}{2}} = -w_m\mu_m$ with first and last α values vanishing on each η level,
5. Cross sections are constant in each space cell. We actually assume that cross sections are regionwise constant, but a region may be $\geq$ a single cell.

On the right-hand side of Eq. (9) is the term

$$\psi^k_{nh} = \int_{-1}^{1} d\mu \int_0^{\pi} d\varphi \; R^k_n(u,\varphi)\psi_h/2\pi \quad .$$

For each group we approximate this integral by the sum

$$N^k_{nhij} = \sum_{m=1}^{MT} w_m R^k_n(\mu_m,\varphi_m)N_{hijm} \quad , \qquad (17)$$

where φ_m is defined in the program by

$$\begin{aligned} \varphi_m &= \tan^{-1} (1 - \mu^2_m - \eta^2_m)^{\frac{1}{2}}/\eta_m \qquad \eta_m > 0 \quad , \\ &= \tan^{-1} (1 - \mu^2_m - \eta^2_m)^{\frac{1}{2}}/\eta_m + \pi \quad \eta_m < 0 \quad . \end{aligned} \qquad (18)$$

TABLE II

AREA AND VOLUME ELEMENTS

Geometry	dV	dA	dB	$A_{i+\frac{1}{2},j}$	$B_{i,j+\frac{1}{2}}$	V_{ij}	$\frac{A_{i+\frac{1}{2},j}-A_{i-\frac{1}{2},j}}{V_{ij}}$	$\frac{A_{i+\frac{1}{2},j}+A_{i-\frac{1}{2},j}}{V_{ij}}$	$\frac{B_{i,j+\frac{1}{2}}+B_{i,j-\frac{1}{2}}}{V_{ij}}$
(x,y)	dxdy	dy	dx	Δy_j	Δx_i	$\Delta x_i \Delta y_j$	0	$2/\Delta x_i$	$2/\Delta y_j$
$(r,\theta)^a$	$rd\theta dr$	$rd\theta$	dr	$2\pi r_{i+\frac{1}{2}}\Delta\theta_j$	Δr_i	$2\pi\bar{r}\Delta r_i\Delta\theta_j$	$1/\bar{r}$	$2/\Delta r_i$	$2/2\pi\bar{r}\Delta\theta_j$
(r,z)	$2\pi rdrdz$	$2\pi rdz$	$2\pi rdr$	$2\pi r_{i+\frac{1}{2}}\Delta z_j$	$2\pi\bar{r}\Delta r_i$	$2\pi r\Delta r_i\Delta z_j$	$1/\bar{r}$	$2/\Delta r_i$	$2/\Delta z_j$

[a] θ measured in revolutions

$\bar{r} \equiv (r_{i+\frac{1}{2}} + r_{i-\frac{1}{2}})/2$.

Because we customarily deal with only one quadrant of directions, the program uses four functions for R^j_n:

$$
\begin{aligned}
R^j_n(\mu_m,\varphi_m) \equiv R^j_{nm} &= P1^j_{nm} \qquad \mu < 0,\ \eta < 0\ , \\
&= P2^j_{nm} \qquad \mu > 0,\ \eta < 0\ , \\
&= P3^j_{nm} \qquad \mu < 0,\ \eta > 0\ , \\
&= P4^j_{nm} \qquad \mu > 0,\ \eta > 0\ .
\end{aligned} \tag{19}
$$

with this notation we have the right side of Eq. (14), S_{gijm}, given by the sum of

$$(SS)_{gijm} = (\text{Scatter Source})_{gijm} =$$

$$\sum_{h=1}^{IGM} \sum_{n=0}^{ISCT} (2n+1)\sigma_{snh\to g} \sum_{k=0}^{n} R^k_{nm} N^k_{nhij}\ , \tag{20}$$

$$(FS)_{gijm} = (\text{Fission Source})_{gijm} =$$

$$\chi_g \sum_{h=1}^{IGM} \nu\sigma_{fh} N^o_{ohij}\ , \tag{21}$$

$$(IS)_{gijm} = (\text{Inhomogeneous Source})_{gijm} =$$

$$\sum_{n=0}^{IQAN} (2n+1) \sum_{k=0}^{n} R^k_{nm} Q^k_{ngij}\ . \tag{22}$$

D. Solution Algorithms

In the following material we adopt the convention that centered subscripts (i,j,m,g) are omitted except to indicate sums or to clarify equations. We write our finite difference equations for group g as

$$
\begin{aligned}
&\mu\,(A_{i+\frac{1}{2}}N_{i+\frac{1}{2}} - A_{i-\frac{1}{2}}N_{i-\frac{1}{2}}) \\
&\quad + (A_{i+\frac{1}{2}} - A_{i-\frac{1}{2}})\,(\alpha_{m+\frac{1}{2}}N_{m+\frac{1}{2}} - \alpha_{m-\frac{1}{2}}N_{m-\frac{1}{2}})\,/w \\
&\quad + \eta B\,(N_{j+\frac{1}{2}} - N_{j-\frac{1}{2}}) + \sigma_t VN = VS = V(SS + FS + IS)\ .
\end{aligned} \tag{23}
$$

Here we have indicated that B is independent of j in the three geometries considered. This equation is to be solved for MT directions defined by ordered pairs (μ_m,η_m), for IT x JT space cells, and for IGM energy groups.

1. Boundary Conditions

Equation (23) contains seven N's. Information about boundary flux values can be specified by the TWOTRAN-II user by designating the following options:

a. Vacuum Boundary Condition

The value of the angular flux N on the boundary is set to zero for all incoming directions.

b. Reflective Boundary Condition

The value of the flux on the boundary for incoming directions is set equal, in detail, to the value of the outgoing flux in the direction corresponding to specular reflection. For example, at an i-direction outer boundary,

$$\psi_{\text{incoming}}(-\mu,\eta) = \psi_{\text{outgoing}}(\mu,\eta) \qquad \mu > 0\ . \tag{24}$$

c. White Boundary Condition

The values of incoming boundary fluxes are all set equal to the same value. This value is the average of the outgoing fluxes such that the net flow through the boundary is zero. In particular, for an i-direction boundary, this value is given by

$$\psi_{\text{incoming}} = \frac{\sum_m w_m\mu_m\,\psi(\mu_m,\eta_m)_{\text{outgoing}}}{\sum_m w_m\mu_m}\ , \tag{25}$$

where the sums range over outgoing directions. This condition is meaningful for cell calculations in cylindrical geometries where it is applied to the outside radial boundary.

d. Periodic Boundary Condition

The values of incoming fluxes at a boundary are set equal, in detail, to the values of the outgoing fluxes on the opposite boundary. This condition is allowed only on the top and bottom (j-direction) boundaries. Then, the incoming flux on the bottom is set equal to outgoing flux on the top, and the incoming flux at the top to the outgoing flux on the bottom. This condition must be used to represent any full circle in (r,θ) geometry in which the system is not homogeneous in θ.

e. Boundary Sources

On the top, bottom, and right sides of the domain, the user may specify the incoming angular flux in each direction. These inflows represent a source of particles and are treated as such by the code. For example, the number of particles incoming from the right-boundary source, QR_{mj}, is given by

$$\text{Number} = \sum_{m,j} w_m \mu_m QR_{mj} A_{IT+\frac{1}{2},j} \quad , \tag{26}$$

where the sum is over incoming directions.

2. Arithmetic Mean (Diamond) Difference Approximation

To solve Eq. (23) we need additional relations among the fluxes. To second-order accuracy, we have

$$2N = N_{i+\frac{1}{2}} + N_{i-\frac{1}{2}} \quad , \tag{27}$$

$$2N = N_{j+\frac{1}{2}} + N_{j-\frac{1}{2}} \quad , \tag{28}$$

$$2N = N_{m+\frac{1}{2}} + N_{m-\frac{1}{2}} \quad . \tag{29}$$

These relations are usually called the "Diamond" difference equations. If we assume that in Eq. (23) we know three fluxes, say $N_{i+\frac{1}{2}}$, $N_{j+\frac{1}{2}}$, and $N_{m-\frac{1}{2}}$, and that we want to calculate $N_{i-\frac{1}{2}}$, $N_{j-\frac{1}{2}}$, and $N_{m+\frac{1}{2}}$ [implying we are moving inward ($\mu < 0$), downward ($\eta < 0$), and in order of increasing angle subscript], we can use Eqs. (27), (28), and (29) to eliminate the fluxes we are going to solve for from Eq. (23), giving a single equation for N,

$$N = \frac{|\mu|(A_{i+\frac{1}{2}}+A_{i-\frac{1}{2}})N_{i+\frac{1}{2}} + 2|\eta|BN_{j+\frac{1}{2}} + (A_{i+\frac{1}{2}}-A_{i-\frac{1}{2}})(\alpha_{m+\frac{1}{2}}+\alpha_{m-\frac{1}{2}})N_{m-\frac{1}{2}}/w + SV}{|\mu|(A_{i+\frac{1}{2}}+A_{i-\frac{1}{2}}) + 2|\eta|B + (A_{i+\frac{1}{2}}-A_{i-\frac{1}{2}})(\alpha_{m+\frac{1}{2}}+\alpha_{m-\frac{1}{2}})/w + \sigma_t V} \quad . \tag{30}$$

In writing this equation we used Eq. (16) to arrange the denominator so that the coefficients are the same as those of the numerator. Using this result we can use Eqs. (27), (28), and (29) to solve for $N_{i-\frac{1}{2}}$, $N_{j-\frac{1}{2}}$, and $N_{m+\frac{1}{2}}$. Equation (30) applies for outward progress as well if $N_{i+\frac{1}{2}}$ is replaced with $N_{i-\frac{1}{2}}$. Similarly, when moving upward, we use $N_{j-\frac{1}{2}}$ instead of $N_{j+\frac{1}{2}}$. We always move in order of increasing angle subscript. Thus Eq. (30) can always be used when we know three fluxes on the boundary of our phase space cell. To initialize the spatial boundary fluxes, we can start applying Eq. (30) at the outer boundaries for incoming directions. To initialize $N_{m-\frac{1}{2}}$, we use the step function approximation for the first incoming μ direction on each η level, setting $N_{m+\frac{1}{2}} = N$. This assumption leads to an equation for N exactly like Eq. (30) if we assume that $(\alpha_{m+\frac{1}{2}} + \alpha_{m-\frac{1}{2}})/w$ is zero. That is, these terms do not appear in the formula. This is an important difference of TWOTRAN-II compared to other programs: TWOTRAN-II does not use special zero-weighted directions to initialize the angular flux $N_{m-\frac{1}{2}}$. Therefore, for a given S_n order, fewer directions are used.

3. Progression Through the Space-Angle Mesh

The unknowns N are ordered so that the difference scheme is stable and so that the coefficient matrix is lower triangular. Physically this corresponds to proceeding in the direction of particle flow. We begin the description of the progression through the space-angle mesh by defining the angular mesh used in TWOTRAN-II.

a. S_n Constants and Angular Orientation

For a quadrature order ISN, TWOTRAN-II requires MT/4 = MM = (ISN)*(ISN+2)/8 positive pairs (μ,η) of direction cosines in all three geometries. These may be supplied by the user or, for ISN up to 16, obtained from the subroutine SNCON. If input by the user, the numbers should be supplied first for the largest η (one value of μ), then for the next largest η (two values of μ, the largest first), then for the third largest η (three values of μ, in order of decreasing magnitude), etc. Only one quadrant of directions is used, and a 90° rotational symmetry is assumed to perform calculations in the other three quadrants.

In Fig. 4 we show the ordering of the direction cosines for ISN=4 and ISN=8. The numbers in parentheses for ISN=4 indicate the actual numbering 1, 2, ..., MM, of the direction cosines. The other numbers indicate the order of progression through the angular mesh both for ISN=4 and ISN=8. In the figure we have shown directions lying on η levels, although this is not strictly necessary.

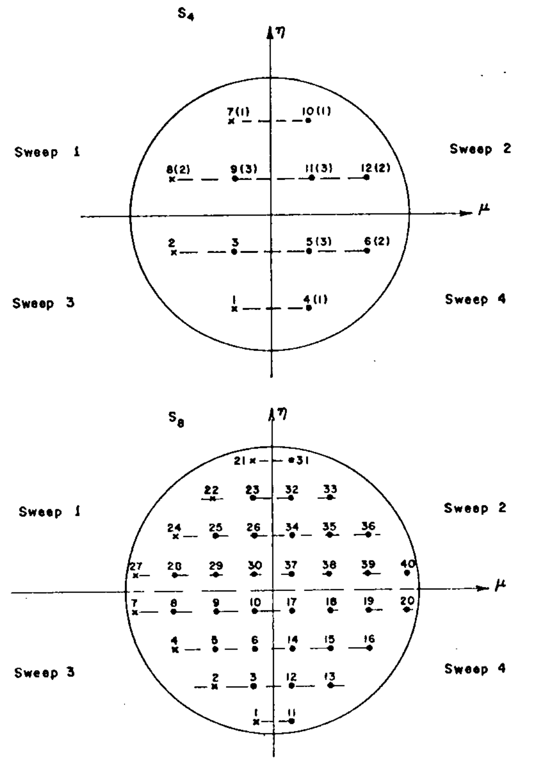


Fig. 4. Ordering of S_n directions.

It does, however, reduce the ω derivative to a purely μ derivative in (r,z) and (r,θ) geometries. Actually, there are no restrictions on the direction sets other than the number of directions. For example, for ISN=8 one could have for the quadrant, say, a 1,1,2,6 arrangement on η levels rather than the 1,2,3,4 arrangement shown. We do require ISN/2 η levels, however. Because of Eq. (18), the user should avoid the use of precisely zero values of η. The program does also check that the quadrature weights associated with the directions sum to one-fourth (because only one quadrant of directions is entered).

In Figs. 5, 6, and 7 we show the orientation of the hemisphere of directions in each of the geometries. In these figures, as in the projection of Fig. 4, the directions in which the step function scheme is used in the angle extrapolation are marked with a cross.

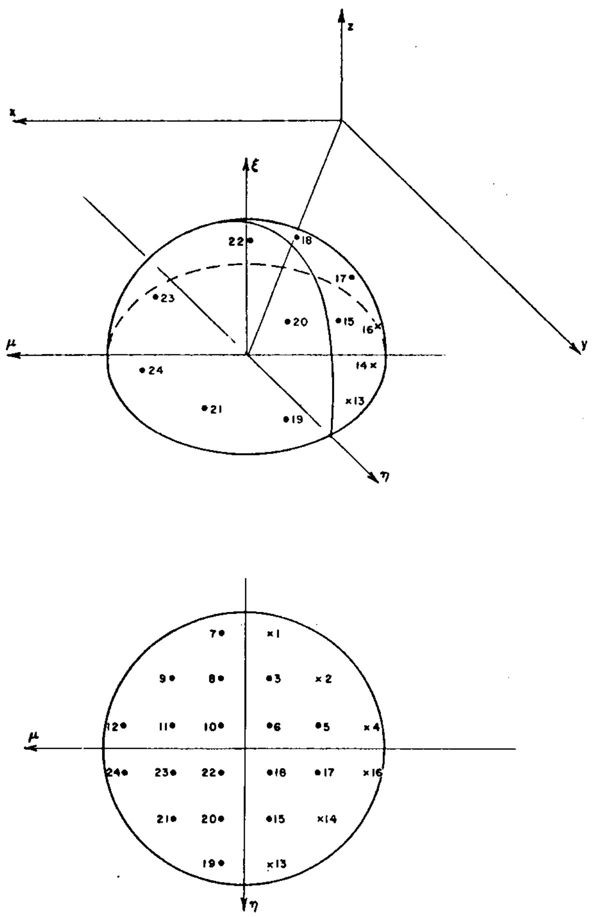


Fig. 5. Orientation of S_6 directions in (x,y) geometry. Directions in the negative quadrants are not shown in the upper sketch.

b. Sweeps of the Space-Angle Mesh.

In subroutine INNER, the NM group-total source components are calculated in the array SOURCE(N,I,J), and the NM components of the angular flux are stored in FLUX(N,I,J). The angular flux is not stored in its entirety unless requested by the user; local values of the angular fluxes are stored in the arrays:

$N_{i+\frac{1}{2},j,m}$ in BR1(J,M) in-down and out-down sweeps
in BR2(J,M) in-up and out-up sweeps
in TI temporarily

$N_{i,j+\frac{1}{2},m}$ in BT1(I,M) in-down and in-up sweeps
in BT2(I,M) out-down and out-up sweeps
in TJ temporarily

$N_{i,j,m+\frac{1}{2}}$ in ALFL(K,I) all sweeps
in TM temporarily

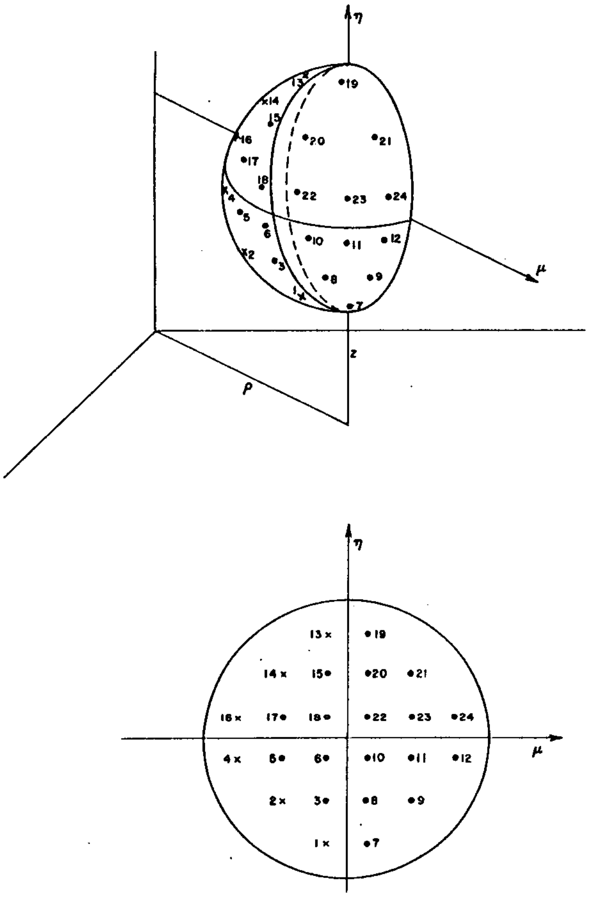


Fig. 6. Orientation of S_6 directions in (r,z) geometry.

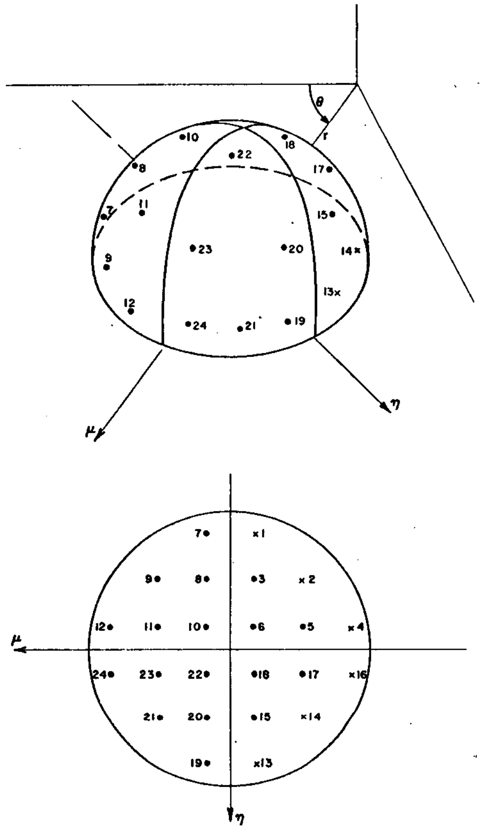


Fig. 7. Orientation of S_6 directions in (r,θ) geometry. Directions in the $\mu < 0$, $\eta < 0$ quadrants are not shown in the upper sketch.

where the range of subscripts is

I = 1,2,..., IT
J = 1,2,..., JT
M = 1,2,..., MM = MT/4
K = 1,2,..., ISN/2
N = 1,2,..., NM = (ISCT+1)*(ISCT+2)/2 .

Computation begins for the $\mu < 0$, $\eta < 0$ quadrant of directions at the top, right corner of the system. We refer to the traverse through the entire space mesh for these directions as Sweep 1. The sweep begins with the storing of the right boundary condition in BR1 for all J and M and in BT1 for all I and M. (Because it is convenient, we also set the boundary condition in BT2 and BR2 for use in other sweeps.) After the boundary condition is set, the coarse-mesh flow through the boundary is computed. For these directions, this is FL and FD. Then Sweep 1 begins first for J = JT, i.e., along the top boundary of the system. For this value of J, I runs from the right: IT, IT-1, ..., 1. For each I, M runs in the sequence shown in Fig. 4 before the next value of I is taken. For each M, before the flux calculation, the angular source is generated by performing the sum

$$QT_{ijm} = \sum_{N=1}^{NM} \text{SOURCE(N,I,J)*P1(N,M)} \quad ,$$

where P1 is used because we are in Sweep 1. Given the source, QT, we compute N_{ijm} (stored in T) from

$$T = \frac{\text{AA*BR1(J,M)+BB*BT1(I,M)+CC*ALFL(K,I)+QT}}{\text{AA+BB+CC+CT}} \qquad (31)$$

where CT is the total cross section for the cell times the I-dependent portion of the volume. (The I-dependent portion of the volume is included in SOURCE before any sweeps begin.) AA, BB, and CC are the products of angle and areas shown in Eq. (30) except we find it convenient to divide numerator and denominator of Eq. (30) by the J-dependent portion of the volume.

If we use Eq. (31) we extrapolate

$$\begin{aligned} TI &= 2T - BR1(J,M) \quad , \\ TJ &= 2T - BT1(I,M) \quad , \qquad (32) \\ TM &= 2T - ALFL(K,I) \quad . \end{aligned}$$

If we are in a direction for which we have no value for ALFL(K,I) (the directions indicated with crosses in Fig. 4), we use Eq. (31) with CC = 0 and set TM = T. If any of the T values in Eq. (32) are negative, we call subroutine FIXUP (see the next subsection) and compute new T values, otherwise we store Tl in BRl, TJ in BTl, and TM in ALFL. We then use T to accumulate the sum

$$FLUX(N,I,J) = \sum_{M=1}^{MM} WGT(M)*P1(N,M)*T \quad . \qquad (33)$$

This same sum accumulates in all sweeps with Pl changing to P2, etc. If we cross a coarse-mesh boundary, we also compute the contribution to the partial flow. We decide whether we have crossed a coarse-mesh boundary by testing for a change in the IDX(I) and IDYA(J) arrays which contain the number of the coarse-mesh i or j zone for each fine-mesh interval. These arrays are surrounded by zeroes in core storage so that we can also identify the first and last coarse-mesh zone.

When the flux computation has been completed for all values of M, we move to the next I level to the left, for the same J and repeat the process. The flux extrapolated in the previous cell is now properly in BRl, just as if we were at the right boundary. When we reach the inside boundary, we have computed the flux for all downward pointing incoming directions along the top interval of the system. We have saved the value of the extrapolated flux ALFL for each space cell for each η level (K = 1, 2, ..., ISN) corresponding to the $\mu = 0$ value. These values initialize ALFL for Sweep 2 which now begins for the same (top) J level. The left boundary condition is set (in BRl), and Sweep 2 proceeds along the top interval of the system, from left to right for all downward-outward directions. When we reach the right boundary we begin Sweep 1 for the next lower J interval. We now have in BTl the fluxes calculated in the cell above (for outward directions, the cell top edge fluxes are in BT2), and computation proceeds just as it did for the top interval. We thus proceed inward, then outward, for each J level for decreasing values of J until we reach the bottom of the system. We then set the bottom boundary condition (in BTl and BT2) and start moving upward in J, first inward (Sweep 3), then outward (Sweep 4), on each J level.

When we reach the top, we have generated all the new flux components, FLUX(N,I,J), and partial flows needed. At this point we have completed the space-angle traverse. We next rebalance the fluxes, perform convergence tests, and, if they fail, recompute the source for the next traverse.

4. Negative Flux Control

Although all the coefficients of Eq. (30) are positive, extrapolations of the form $N_{i+\frac{1}{2}} = 2N - N_{i-\frac{1}{2}}$ can lead to negative values for fluxes. When sources are positive, we attempt to prevent negative fluxes by using a "set-flux-to-zero-and-correct" recipe. After much experimentation with other difference schemes and recipes,[7] including their use as fixups, we have decided that the set-to-zero scheme is the most accurate while being computationally no more expensive. It is most accurate because it is the minimum deviation from the second-order accuracy of the diamond difference equations. All linear difference schemes which lead to positive coefficients in the extrapolations like Eqs. (27) - (29) are first-order accurate,[7] and all other than the set-to-zero scheme lead to positive extrapolated fluxes and hence, to a greater deviation from the negative diamond result. Secondly, the set-to-zero scheme is more selective, not only because it is used only when a flux is negative, unlike a weighted diamond scheme, but also because it is applied to the extrapolations individually, unlike a scheme in which all fluxes are recomputed with, say, a step function approximation.

The logic of the set-to-zero fixup is as follows. If any one flux is negative, it is set to zero and the cell-centered flux, N, is recomputed assuming that particular flux is zero. (See formulas below). If this N should be negative, the fixup attempt is aborted. If not, the other two fluxes are re-extrapolated. If either one of these is negative, N is recomputed assuming two fluxes are negative. If this N is positive, the remaining flux is extrapolated. If it should be negative, then N is recomputed assuming all three boundary fluxes are zero (provided the total cross section is not zero). If any center N is negative (implying a negative source or boundary flux) or if any fixup leads to a division by zero (e.g., in voids), the fixup attempt is aborted. One cannot exclude negative sources because the finite Legendre polynomial expansion of anisotropic scattering may produce negative numbers in particular directions.

Formulas for fixup are derived as follows. Suppose we are attempting to calculate $N_{i-\frac{1}{2}}$ and find it negative. Then we use Eqs. (28) and (29) in Eq. (23) where $N_{i-\frac{1}{2}}$ is taken to be zero. We have then

$$N = \frac{\left[|\mu|A_{i+\frac{1}{2}}N_{i+\frac{1}{2}} + 2\ |\eta|BN_{j+\frac{1}{2}} + (A_{i+\frac{1}{2}}-A_{i-\frac{1}{2}})(\alpha_{m+\frac{1}{2}}+\alpha_{m-\frac{1}{2}})N_{m-\frac{1}{2}}/w+SV\right]}{2|\eta|B+2(A_{i+\frac{1}{2}}-A_{i-\frac{1}{2}})\alpha_{m+\frac{1}{2}}/w+\sigma_t V} \quad . \tag{34}$$

This formula, of course, lacks the symmetry of Eq. (30). If we had been trying to compute $N_{i+\frac{1}{2}}$ ($\mu > 0$) and found it negative, then Eq. (34) would be altered by using $A_{i-\frac{1}{2}}N_{i-\frac{1}{2}}$ instead of $A_{i+\frac{1}{2}}N_{i+\frac{1}{2}}$. In either case the negative flux is set to zero, and, provided N itself is positive, the remaining two fluxes, $N_{j-\frac{1}{2}}$ and $N_{m+\frac{1}{2}}$ are computed from Eqs. (29) and (29). All other formulas needed (six more) can be generated in the same fashion. These appear explicitly in subroutine FIXUP in the program.

Controlling negative fluxes in this fashion seems to be more compatible with the coarse-mesh rebalancing acceleration procedure (see Section E below) than any other fixup we examined.

5. Adjoint Problems

The TWOTRAN-II program solves the adjoint transport equation by transposing the scattering and fission matrices and inverting the group order of the problem. The solution of the resulting problem in the direction $\underline{\Omega}$ is then identified with the adjoint solution in direction $-\underline{\Omega}$.[8] Transposition of the scattering matrix converts the normal, predominantly downscattering problem to an upscattering problem. Group order inversion restores this downscattering dominance and eliminates unnecessary upscattering iteration.

E. Iterative Processes and Convergence Acceleration

The source in Eq. (23) usually depends on the fluxes being computed, through the within-group scattering and/or through fission or upscattering. We use an iterative process in which the source is updated when new flux information is available. Within each energy group we update the portion of the source due to within-group scattering ($\sigma_{sng\to g}$) after each pass through the space-angle mesh. This iterative process, including iteration for implied boundary conditions (e.g., a reflective outer boundary condition), is called inner iteration and is performed by subroutine INNER. The iterative process involving the updating of sources due to fission and upscatter after a progression through all groups is called outer iteration and is performed by subroutine OUTER. Both of these processes can be accelerated by coarse-mesh rebalancing. In this section we describe the theory of rebalancing.

1. Inner Iteration Coarse-Mesh Rebalancing

When Eq. (23) is multiplied by w and summed over all directions, the resulting equation,

$$\begin{aligned} &A_{i+\frac{1}{2}}I_{i+\frac{1}{2}} - A_{i-\frac{1}{2}}I_{i-\frac{1}{2}} + B(J_{j+\frac{1}{2}} - J_{j-\frac{1}{2}}) \\ &\quad + \sigma_t V\phi = V\sigma_{sog\to g}\phi^P + VQ \quad , \end{aligned} \tag{35}$$

is a statement of particle balance. In this equation the i-direction currents, I, are given by

$$I_{i\pm\frac{1}{2},j} = \sum_{m=1}^{MT} w_m \mu_m N_{i\pm\frac{1}{2},j,m} \equiv N^0_{1,i\pm\frac{1}{2},j} \quad , \tag{36}$$

and the j-direction currents, J, by

$$J_{i,j\pm\frac{1}{2}} = \sum_{m=1}^{MT} w_m \eta_m N_{i,j\pm\frac{1}{2},m} \equiv N^1_{1,i,j\pm\frac{1}{2}} \quad . \tag{37}$$

When multiplied by the surface areas as in Eq. (35), these terms account for the net flow through the cell surface. The angle-integrated, or scalar, flux, ϕ, is defined by

$$\phi_{ij} = \sum_{m=1}^{MT} w_m N_{ijm} \equiv N^0_{0ij} \quad . \tag{38}$$

On the right side of Eq. (35) we have split the angle-integrated portion of VS into a part due to isotropic within-group scattering ($\sigma_{sog\to g}$) and a remainder, Q. All anisotropic sources vanish when the m sum is performed, because of the orthogonality of the spherical harmonic functions R^k_n, *provided that the quadrature set used correctly integrates these functions*.

On the right of Eq. (35) we have indicated that the scattering source depends on the flux from a previous iteration by labeling this flux with a superscript p. The particle balance equation is satisfied only when $\phi \equiv \phi^p$. It has long been realized[9] that enforcing balance accelerates iterative convergence. DTF-IV[10] and earlier Los Alamos codes use a system-wide within-group renormalization. Here we describe a variation of the technique which enforces particle balance in each zone of a coarse mesh.

The object of the technique is to find a factor by which all fluxes in a zone may be multiplied to insure that leakage plus absorption equals sources in the zone. Usually, application of the factor quickly brings all flux amplitudes within the zone close to their final amplitude, and subsequent iteration refines the flux shapes in the zone.

To describe the terms entering the rebalance equation, we superpose a coarse mesh on the fine space mesh defined by Eq. (13), letting [again using (r,θ) geometry to illustrate]

$$r_{k-\frac{1}{2}} < r_i < r_{k+\frac{1}{2}} \qquad k = 1,\ 2,\ \ldots,\ IM \tag{39}$$

$$\theta_{\ell-\frac{1}{2}} < \theta_j < \theta_{\ell+\frac{1}{2}} \qquad \ell = 1,\ 2,\ \ldots,\ JM \tag{40}$$

define the coarse mesh.

For each zone (k,ℓ) we compute the upward, downward, rightward, and leftward flows,

$$FU_{k,\ell\pm\frac{1}{2}} = \sum_{i\epsilon k} B_i \sum_{\eta_m>0} w_m \eta_m N_{i,\ell\pm\frac{1}{2},m} \quad , \tag{41}$$

$$FD_{k,\ell\pm\frac{1}{2}} = \sum_{i\epsilon k} B_i \sum_{\eta_m<0} w_m \eta_m N_{i,\ell\pm\frac{1}{2},m} \quad , \tag{42}$$

$$FR_{k\pm\frac{1}{2},\ell} = \sum_{j\epsilon\ell} \sum_{\mu_m>0} w_m \mu_m A_{k\pm\frac{1}{2},j} N_{k\pm\frac{1}{2},j,m} \quad , \tag{43}$$

$$FL_{k\pm\frac{1}{2},\ell} = \sum_{j\epsilon\ell} \sum_{\mu_m<0} w_m \mu_m A_{k\pm\frac{1}{2},j} N_{k\pm\frac{1}{2},j,m} \quad ; \tag{44}$$

the zone effective absorption,

$$AB_{k\ell} = \sum_{i\epsilon k} \sum_{j\epsilon\ell} V_{ij} (\sigma_t - \sigma_{sog\to g}) \sum_{m=1}^{MT} w_m N_{ijm} \quad ; \tag{45}$$

and the isotropic component of the zone source,

$$QQ_{k\ell} = \sum_{i\epsilon k} \sum_{j\epsilon\ell} V_{ij} Q_{ijo} \quad , \tag{46}$$

where Q is the nonself-scatter portion of S used in Eq. (35), i.e., the portion that does not change during iteration. If boundary-flux sources occur, they are placed in $QQ_{k\ell}$ in the zones along the boundaries.

If now all fluxes are multiplied by the factor $f_{k\ell}$, we obtain the equation (written with our subscripting convention)

$$\begin{aligned} f_{k\ell}(FR_{k+\frac{1}{2}} + FL_{k-\frac{1}{2}} + FU_{\ell+\frac{1}{2}} + FD_{\ell-\frac{1}{2}} + AB) = \\ QQ + f_{k+1,\ell} FL_{k+\frac{1}{2}} + f_{k-1,\ell} FR_{k-\frac{1}{2}} \\ + f_{k,\ell+1} FD_{\ell+\frac{1}{2}} + f_{k,\ell-1} FU_{\ell-\frac{1}{2}} \quad , \end{aligned} \tag{47}$$

by equating losses in the cell (outflows plus absorption) to the sources (true source plus inflows from adjoining cells). This equation may also be deduced from a variational method using discontinuous (zone step functions) trial functions. Other trial functions lead to more complicated equations.[8]

With boundary conditions, Eq. (47) can be solved for the scale factors, $f_{k\ell}$. If the outer boundary condition is a vacuum condition, then the corresponding incoming flow is zero, eliminating the unknown f from the equation. For reflecting and white boundary conditions, we set the factor outside the boundary equal the factor just inside the boundary. For example, suppose the right boundary is reflecting. Then, at the boundary we set $f_{k+1,\ell} = f_{k\ell}$. Then, subtracting $f_{k\ell}FL_{k+\frac{1}{2}}$ from both sides, we have

$$f_{k\ell}(FR_{k+\frac{1}{2}} - FL_{k+\frac{1}{2}} + FL_{k-\frac{1}{2}} + FU_{\ell+\frac{1}{2}} + FD_{\ell-\frac{1}{2}} + AB)$$
$$= QQ + f_{k-1,\ell}FR_{k-\frac{1}{2}} + f_{k,\ell+1}FD_{\ell+\frac{1}{2}}$$
$$+ f_{k,\ell-1}FU_{\ell-\frac{1}{2}} \qquad (48)$$

The term $FR_{k+\frac{1}{2}} - FL_{k+\frac{1}{2}}$ is the net flow through the boundary and should vanish when the reflecting condition is satisfied. Such conditions are identically satisfied at the nonimplicit left and bottom boundaries. With a periodic boundary condition, the outgoing flux on, say, the bottom is used as the incoming flux on the top. Thus the factor $f_{k,\ell+1}$ for the top boundary zone is replaced with $f_{k\ell}$ from the bottom boundary zone. A similar replacement is made for the bottom periodic boundary condition.

Equation (47) is solved iteratively, using a forward elimination, backward substitution on each k level. Iteration, including that for any periodic boundary condition, is performed in the ℓ direction. If the iteration fails to converge after 200 iterations, if any factor becomes negative, or, if any denominator becomes zero, we set all $f_{k\ell}$ equal to the single-system factor given by

$$f = \frac{\sum\limits_{k\ell} QQ_{k\ell}}{NL + \sum\limits_{k\ell} AB_{k\ell}} \quad , \qquad (49)$$

where NL is the system net leakage determined from the boundary net flows. All of these operations are performed by subroutine REBAL.

2. Outer Iteration Coarse-Mesh Rebalancing

To accelerate the convergence of the outer iteration process, we determine a different set of scale factors, f. We collapse the entire group structure to a single group, accumulating the coarse-mesh flows from each group. We also calculate the total absorption,

$$ABT = \sum_{i\epsilon k} \sum_{j\epsilon \ell} \sum_{g=1}^{IGM} \sigma_{ag} V_{ij} \sum_{m=1}^{MT} w_m N_{gijm} \quad . \qquad (50)$$

To insure consistency, we define an effective σ_{ag} from the input cross sections by

$$(\sigma_{ag})_{eff} = \sigma_{tg} - \sum_{h=1}^{IGM} \sigma_{sog\rightarrow h} \quad . \qquad (51)$$

When we are performing an adjoint calculation we must, because of the cross-section transposition, calculate a different effective absorption,

$$(\sigma_{ag})_{eff} = \sigma_{tg} - \sum_{h=1}^{IGM} \sigma_{soh\rightarrow g} \quad . \qquad (52)$$

The source for over-all-group rebalance consists of the group sum of the inhomogeneous source (if any) and the group sum of the fission source (if any). If there is a fission source, we perform a source iteration to determine the rebalance factors. If there is no inhomogeneous source, this iteration can also be used to estimate an eigenvalue, say k_{eff}. This can be visualized by replacing QQ in Eq. (48) by $f_{k\ell}FT_{k\ell}/k_{eff}$ where $FT_{k\ell}$ is the total fission source. We choose to do this only for the first two outer iterations, thereafter considering the fission source to be a known inhomogeneous source. All of these iterations are controlled by subroutine TESTS.

The outer iteration rebalance process is advantageous because it accelerates all types of problems, e.g., inhomogeneous source problems with upscatter and/or fission, or eigenvalue problems with or without upscatter.

In both the within-group and over-all-group rebalancing calculations, the scale factors all usually approach unity in a few inner or outer iterations.

3. Convergence Tests

There are three levels of iterative processes in the TWOTRAN-II program: (1) the inner iteration, in which the within-group scattering source or the boundary flux at an implicit boundary (right or top) changes, (2) the outer iteration in which the fission or upscattering source changes or which is caused by artificial inner iteration limitation (usually in inhomogeneous source problems), and (3) the parametric eigenvalue search iteration in which, after a converged outer iteration, the value of a coarse-mesh boundary, a material concentration, or a time absorption is changed. Two additional iterations are required for the calculation of coarse-mesh rebalance factors, one for the factors themselves and one for the coarse-mesh rebalance eigenvalue.

Two convergence precisions are input: EPS and XLAX. In a parametric eigenvalue search, two values of (r indicates iteration number)

$$\lambda = \frac{\text{Fission Source}^{r} + \text{Inhomogeneous Source}}{\text{Fission Source}^{r-1} + \text{Inhomogeneous Source}} \qquad (53)$$

are required to differ by less than XLAX before a new eigenvalue guess is computed. All other processes are tested against precisions derived from EPS. These precisions are

EPSO = EPS	outer iteration convergence
EPSI = EPS	inner iteration convergence
EPSR = EPS/2	coarse-mesh rebalance numbers
EPSX = EPS	collapsed group coarse-mesh convergence, both factors (5*EPSX) and eigenvalue (EPSX).

In the inner iteration process, we require

$$\underset{ij}{\text{Max}}\left|1 - \frac{\left(\phi^{r-1}_{ij}+\phi^{r-1}_{i-1,j}+\phi^{r-1}_{i,j-1}+\phi^{r-1}_{i-1,j-1}\right)}{\left(\phi^{r}_{ij}+\phi^{r}_{i-1,j}+\phi^{r}_{i,j-1}+\phi^{r}_{i-1,j-1}\right)}\right| < \text{EPSI} \qquad (54)$$

for the r^{th} iteration where ϕ is the scalar flux after application of rebalance factors. If the system is one interval wide in either direction, we require

$$\underset{ij}{\text{Max}}\left|1 - \phi^{r-1}_{ij}/\phi^{r}_{ij}\right| < \text{EPSI} \quad , \qquad (55)$$

instead. If the denominator in either Eq. (54) or Eq. (55) is zero, we increment the quantity which is compared to EPSI by the numerator divided by TS, the total volume integrated source to the group. This allows both the numerator and denominator to be zero. If the number of iterations is greater than IITL, we stop inner iteration even if Eq. (54) or Eq. (55) is not satisfied.

For the iterative computation of the coarse-mesh scale factors in the inner iteration, we require

$$\underset{k,\ell}{\text{Max}}\left|1 - f^{r}_{k\ell}/f^{r-1}_{k\ell}\right| < \text{EPSR} \quad . \qquad (56)$$

If $f^{r-1}_{k\ell}$ is zero we compute, noniteratively, a single factor, avoiding Eq. (56). No iteration is required if the system is only one coarse-mesh interval high.

In the outer process, if both

$$|\lambda - 1| < \text{EPSO and } \underset{k\ell}{\text{Max}}|1 - f_{k\ell}| < 5{*}\text{EPSX} \quad , \qquad (57)$$

the problem is terminated after one final outer iteration. In the coarse-mesh iterative process (denoted with iterative superscript r) we require

$$|(\lambda)^{r}_{\text{Coarse Mesh}} - 1| < \text{EPSX} \quad , \qquad (58)$$

if IEVT=1 (k_{eff} calculations) and

$$\left|1 - (\lambda)^{r}_{CM}/(\lambda)^{r-1}_{CM}\right| < \text{EPSX} \quad , \qquad (59)$$

if IEVT ≠ 1 (parametric eigenvalue or inhomogeneous source plus fission problems).

III. A GUIDE TO USER APPLICATION

In this section we provide information needed by the user to understand TWOTRAN-II options and to prepare input for the code.

A. Overall Program Flow

A schematic flow chart for TWOTRAN-II is given in Fig. 8. A more detailed flow chart is provided below in the section on programming information.

B. Details of Program Options

1. Cross Sections

a. Input Formats

The TWOTRAN-II program accepts cross sections either from the standard file ISOTXS,[3] in FIDO format,[4] or in the standard Los Alamos format. In upscattering problems, the program does not need the special σ^{up} cross section which is required in earlier Los Alamos programs.[10] In TWOTRAN-II, it is assumed that σ^{up} is present, and σ^{up} is automatically removed from card input cross-sections sets unless the user tags the input number IHT with a minus sign. Cross sections read with the FIDO format may not contain σ^{up}.

The Los Alamos cross-section format assumes that each nuclide is described by a block of cross sections of IHM rows for IGM group columns. The row position of cross sections is specified relative to the total cross section, σ_t (row IHT), and

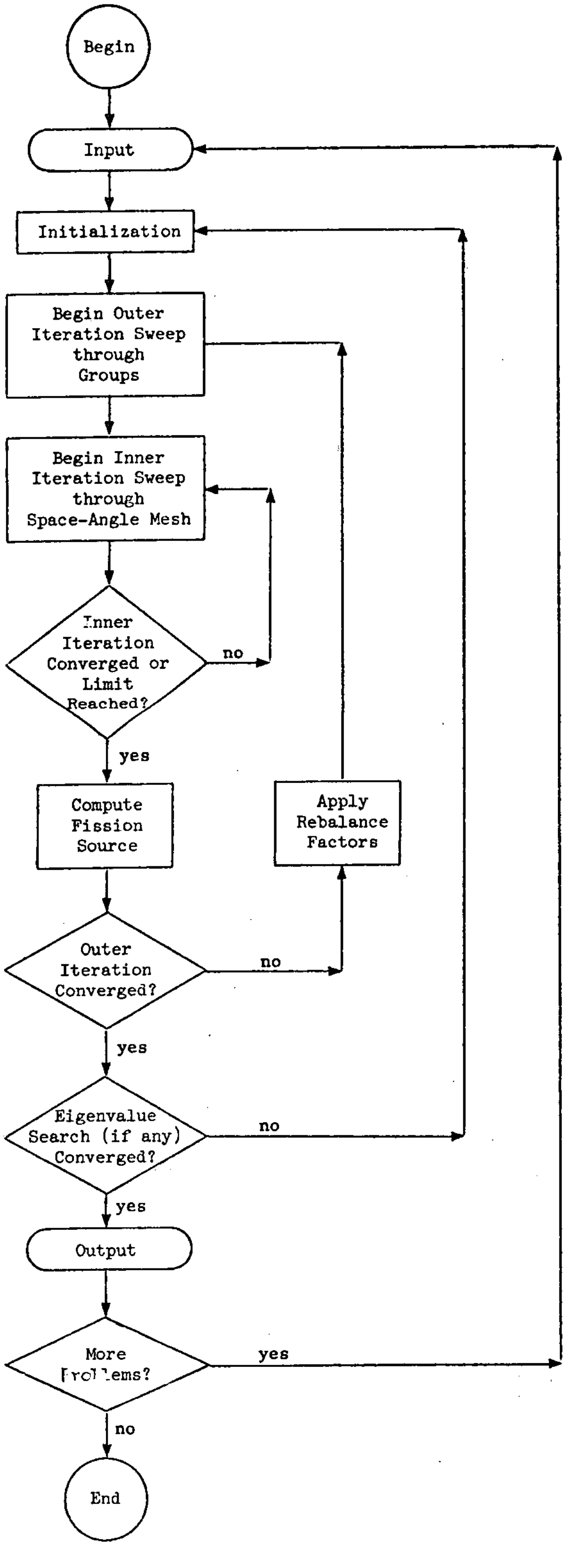

Begin
Input
Initialization
Begin Outer Iteration Sweep through Groups
Begin Inner Iteration Sweep through Space-Angle Mesh
Inner Iteration Converged or Limit Reached?
no
yes
Compute Fission Source
Apply Rebalance Factors
Outer Iteration Converged?
no
yes
Eigenvalue Search (if any) Converged?
no
yes
Output
More Problems?
yes
no
End

Fig. 8. Simplified logical flow diagram for TWOTRAN-II.

the within-group scattering cross section, $\sigma_{s,g\to g}$, (row IHS). It is assumed that the row order of the cross sections is as follows:

	Row	Cross-Section Type-Group g
	.	
	.	
	.	
	IHT-4	$\sigma_{n,2n}$
	IHT-3	σ_{tr}
	IHT-2	σ_a
	IHT-1	$\nu\sigma_f$
	IHT	σ_t
	IHT+1	$\sigma_{s,g+N\to g}$
IHM	.	
	.	
	.	
	IHS-2	$\sigma_{s,g+2\to g}$
	IHS-1	$\sigma_{s,g+1\to g}$
	IHS	$\sigma_{s,g\to g}$
	IHS+1	$\sigma_{s,g-1\to g}$
	IHS+2	$\sigma_{s,g-2\to g}$
	.	
	.	
	.	
	IHS+M	$\sigma_{s,g-M\to g}$

In this format, group g+1 corresponds to a group of lower energy than group g. The symbol $\sigma_{s,g-2\to g}$ denotes the scattering transfer probability from group g-2 to group g. The format allows N groups of upscatter and M groups of downscatter; i.e., the scattering matrix need not be symmetric. However, all cross-section blocks must have the same values for IHM, IHS, and IHT. The fission cross section, σ_f, times the mean number of neutrons per fission, ν, must be located in row IHT-1, and the absorption cross section, σ_a, must be entered in row IHT-2. If a scattering matrix is used to represent (n,2n) reactions, that is, if $\sigma_{s,g\to h}$

contains 2 times the (n,2n) transfer probability $\sigma_{n,2n;g\to h}$, then the value,

$$\sigma_{n,2n} = \sum_{\text{all } h} \sigma_{n,2n;g\to h} \quad ,$$

must be entered in IHT-4. The user is free to enter additional cross sections at the top of the format. These extra cross sections are not used in the calculation, but are available for reaction-rate computations after the particle flux is obtained. It is sometimes convenient to locate the transport cross section, σ_{tr}, in IHT-3 and use this cross section instead of σ_t in the calculation of buckling corrections.

b. Cross Section Mixing

The user is free, in TWOTRAN-II, to enter macroscopic cross sections and bypass the mixing algorithms; specification of the input value MS = 0 is all that is required for this. If MS ≠ 0, the user must provide three sets of MS numbers which are stored in the vectors MIXNUM, MIXCOM, and MIXDEN. These numbers are used in the following algorithm to manipulate cross section blocks:

```
      DO  315  M = 1, MS
      N = MIXNUM(M)
      L = MIXCOM(M)
      AD = MIXDEN(M)
      DO  315  I = 1, IHM
      IF(L.EQ.0) GO TO 310
      IF((AD.EQ.0.0).AND.(IEVT.EQ.3)) GO TO 313
      C(I,N) = C(I,N) + AD*C(I,L)
      GO TO 315
313   C(I,N) = EV*C(I,N)
      GO TO 315
310   C(I,N) = AD*C(I,N)
315   CONTINUE
```

In this algorithm, cross section block N is created or altered by adding multiples of block L or by multiplying the block N by a factor. Let us consider some examples.

Suppose we have entered 45 cross sections as input. Then any mixtures that are made must be given block numbers higher* than 45. Suppose we enter:

MIXNUM (N)	MIXCOM (L)	MIXDEN (AD)
46	0	0.0
46	1	0.0478
46	20	0.0333
47	0	0.0
47	2	0.75
47	3	0.25
47	0	0.1179
48	0	0.0
48	15	0.0049
48	14	0.0078
48	48	0.0
49	0	0.0
49	33	0.5
49	34	0.5
49	0	0.187
49	49	0.0
49	46	0.1

For this example we have MS = 17 instructions. In the first three instructions, block 46 is cleared (set to zero) and then made up of 0.0478 parts of block 1 and 0.0333 parts of block 20. If blocks 1 and 20 are microscopic cross sections in barns, then 0.0333 and 0.0478 times 10^{24} are the atomic densities. In the second set of instructions, block 47 is cleared and then made up of 0.1179 times the result of adding three-fourths of block 2 to one-fourth of block 3. In the next set of instructions, block 48 is cleared and made up of portions of blocks 15 and 14. If IEVT (the input eigenvalue type option) is 3, then the resulting block 48 is multiplied by EV (the input eigenvalue guess). In this type of problem the program attempts to find a value of EV such that the resulting concentration of block 48 renders the system critical. If IEVT ≠ 3, the line of instructions 48, 48, 0.0 would not alter the composition of block 48. In the final sequence, block 49 is made up of 0.187 times one-half of block 33 and block 34, and provision is made to search for the concentration of this portion of 49 to which is always added 0.1 of the previously mixed block 46. It should be clear that there are many possibilities not covered in these examples, but by examining the Fortran instructions above, the user should be able to prepare his own sets of mixture instructions.

*To preserve the input values. If these need not be saved, mixtures can be created in lower block numbers.

c. Anisotropic Cross Sections

In the TWOTRAN-II program it is assumed that the scattering transfer probability can be represented by a finite Legendre polynomial expansion; i.e., that

$$\sigma_s(E' \to E,\mu_o) = \sum_{n=0}^{ISCT} \frac{2n+1}{4\pi} P_n(\mu_o)\sigma_{sn}(E' \to E) \quad , \qquad (60)$$

where ISCT is an input control integer. Thus if ISCT > 0, additional blocks of scattering transfer cross sections must be entered for those nuclides for which anisotropic scattering sources are to be computed. In these blocks, the rows 1 through IHT are zero, and $\sigma_{sn,g\to h}$ (the energy average of $\sigma_{sn}(E' \to E)$ in groups g and h) is entered as for the isotropic component of the cross section. It is assumed in TWOTRAN-II that blocks of anisotropic cross sections which are used in the calculation have block numbers in ascending sequence, starting with the isotropic cross-section block. For example, suppose that block 50 is the isotropic cross-section block for hydrogen and that ISCT = 3. Then, block 51 must be σ_{s1} for hydrogen, block 52 must be σ_{s2} and block 53 must be σ_{s3}. If a material is made by mixing two anisotropic scatters, then the anisotropic blocks must also be mixed with the same densities to form anisotropic blocks for the material.

Anisotropic scattering sources may be computed selectively within each zone, but in all zones in which such sources occur, the number of anisotropic scattering blocks (ISCT) must be the same.

d. Adjoint Cross Sections

In adjoint calculations, cross sections are entered just as for a direct calculation. The program then transposes the scattering matrices and, because this usually changes a downscattering problem to an upscattering problem, reverses the group order of the blocks. Further, the effective absorption in an adjoint calculation is not simply related to σ_a. That is, the effective absorption is normally

$$(\sigma_a)_{eff} = \sigma_t - \sum_{\text{all } h} \sigma_{so,g\to h} \quad . \qquad (61)$$

But when the scattering matrix has been transposed, the effective absorption is

$$(\sigma_a)_{eff} = \sigma_t - \sum_{\text{all } h} \sigma_{so,h\to g} \quad . \qquad (62)$$

In regular and adjoint problems, these quantities are calculated and processed in mixing operations for use in the rebalancing algorithms.

e. Cross Section Checking

As input cross sections are processed, Eq. (61) is computed and compared to the input value of σ_a. If the relative difference between the input total cross section and the computed total cross section exceeds EPS (outer convergence precision), the user is so informed.

f. Pointwise Spatial Variation of Cross Sections

If the user so desires, the macroscopic cross section at point (I,J), say C, can be modified by multiplication by the input separable factors XDF(I) and YDF(J) to give a pointwise spatial variation of cross sections, C(I,J) = C*XDF(I)*YDF(J). All cross sections actually used in the calculation are so modified. Changing the program to use a more general modifying factor, say XYDF(I,J) would be a simple matter, but would require more storage space.

2. Geometry and Boundary Condition Specification

a. Material Mesh and Rebalance Mesh Coincide

To specify the domain of the problem, the user supplies IM+1 coarse-mesh i-boundaries (defining IM intervals) and JM+1 coarse-mesh j-boundaries. Except for the first radius in cylindrical geometries, these sets need not begin at 0.0, but both sets must form a monotone increasing sequence. The user also supplies IM integers which indicate how many fine-mesh i intervals are in each coarse-mesh interval, k, and JM integers to indicate the number of fine-mesh j intervals in each coarse-mesh interval, ℓ. The fine-mesh spacing is uniform between the coarse-mesh boundaries. Finally, the user specifies the boundary condition on each of the four (left, right, bottom, top) boundaries.

The coarse-mesh boundaries define IMxJM zones. The user must supply a number for each of these zones to designate which cross-section block belongs in the zone. If anisotropic scattering is present in any zone, the block number for the zone is made negative. This indicates (as noted above) that the next ISCT blocks in numerical sequence contain the anisotropic cross sections for this zone.

All of the above information is converted by subroutine MAPPER to a pictorial description of the system. A sample is shown for (x,y) geometry.

```
Materials by broad zone.  Origin at lower left.  M
is number of fine intervals/broad row (column).

           Y

      ROW   M

      10.9000   1000000000000
                1     *     0
        2    4  1  -5 *  -7 0
                1     *     0
       6.5400   1***********0
                1     *     0
        1    6  1  -5 *  -5 0
                1     *     0
       0.0000   1111111111111

        X       0.    6.  10.

                0   5400 9000

        M          6    4

      COLUMN       1    2
```

In the problem pictured, the left and bottom boundaries are reflecting (1's) and the right and top boundaries are vacuum boundaries (0's). The square system extends from 0.0 to 10.9 in x with a coarse-mesh boundary at 6.54. In the first coarse x interval there are six fine intervals; in the second, four. The same spacing is used in the y direction. In three zones, cross-section block 5 is used, and, as indicated by the negative tag, anisotropic scattering is to be computed in these zones. Similarly, block 7 and anisotropic scattering are used in the remaining zone. (One can deduce that the scattering is linearly anisotropic (ISCT=1) because block 6 must be an anisotropic block and block 7 is used as an isotropic block). If one desires to assess the effect of anisotropic scattering in the zone with material 7, the only input change necessary is the change of -7 to +7.

Although zones defined by coarse-mesh boundaries normally coincide with material discontinuities, it may sometimes accelerate convergence if artificial boundaries are introduced. This is so because of the rebalancing algorithm which enforces particle conservation within each zone as the iteration proceeds.

The above example is for (x,y) geometry. The same procedures are followed in the other geometries in which r plays the role of x, and θ or z play the role of y. In (r,θ) geometry the entire left boundary of the figure drawn by MAPPER corresponds to a single point, the origin.

b. Material Mesh Does Not Coincide With Rebalance Mesh

If the user desires, he may specify a material mesh and a separate rebalance mesh. This is useful in two classes of problems. In the first class, there are so many different materials used in the calculation that a large (several hundred or more) number of coarse-mesh rebalance cells would be required. In this case, using a smaller rebalance mesh can result in a significant storage reduction. In one problem, reducing the number of coarse-mesh cells from 459 to 64 saved 3146 words of core storage and 9816 words of peripheral storage. In the second class of problems, it may be desirable to use a variable fine-mesh spacing near a material interface to allow for a rapidly changing flux. This too might require a large number of coarse-mesh cells, but it also might lead to an instability in the rebalance algorithm because each fine-mesh cell would coincide with a rebalance cell in an area of rapidly changing flux. Generally speaking, the rebalance algorithm is the more stable, the coarser the rebalance mesh. However, it is usually stable and the more effective in accelerating convergence the finer the rebalance mesh.

To specify a separate material mesh the user enters an additional set of mesh boundaries (IMC+1 radial and JMC+1 axial) and additional sets of integers (IMC radial and JMC axial) indicating how many fine mesh intervals are in each material coarse mesh. The extra boundaries and integers then describe the material mesh for which the user must provide IMC x JMC numbers to specify the material used in each zone. The mesh boundaries and integers normally supplied are used to define the IM x JM rebalance zones. There are two restrictions on the generality of these two meshes. The rebalance coarse-mesh boundaries must coincide with some material-mesh boundary, and the number of fine mesh intervals defined by both meshes must be the same.

If an edit is required when the optional separation of the material mesh and rebalance mesh is specified, then angular fluxes are stored to generate edit quantities. Some of the storage gained in separating the meshes may then be lost. This is particularly true for peripheral storage because two entire angular flux arrays are stored.

3. Source Options

The TWOTRAN-II user may specify an anisotropic distributed source or the boundary flux at the top and/or bottom and/or right boundaries of the system. The inhomogeneous distributed source must be represented by the finite spherical harmonic expansion

$$Q_g(\underline{r},\mu,\eta) = \sum_{n=0}^{IQAN} 2n+1 \sum_{k=0}^{n} R_n^k(\mu,\eta)Q_{gn}^k(\underline{r}) \quad , \qquad (63)$$

in which IQAN is an input number designating the order of anisotropy of the source, and R_n^k is the spherical harmonic defined in Eq. (4) above. In these terms,

$$Q_{gn}^k = \frac{1}{4\pi} \int_{-1}^{1} d\mu \int_{0}^{2\pi} d\varphi Q_g(\underline{r},\mu,\eta)R_n^k(\mu,\eta) \quad . \qquad (64)$$

We have written the integral in this equation over the entire range of φ and used a 4π normalization of the integral, but it should be clear that Q_g must be symmetric in $\varphi[Q_g(\varphi) = Q_g(-\varphi)]$, and this we have noted by writing Q_g as a function of μ and η alone.

In addition to specifying IQAN, the user must enter [(IQAN + 1)(IQAN + 2)]/2 components of Q_g multiplied by (2n + 1); that is, the user must enter

$$(2n + 1)Q_{gn}^k \qquad \begin{array}{l} k = 0, 1, \ldots, n \\ n = 0, 1, \ldots, IQAN \end{array} \qquad (65)$$

in the order shown in Table III. When using the anisotropic distributed source option, the order of anisotropic scattering, ISCT, must be at least as large as IQAN so that the requisite number of scattering coefficients R_n^k are computed.

The user is also allowed to specify the value of the incoming flux along the top and/or right and/or bottom boundaries. Because of the quadrant organization of TWOTRAN-II, the user must supply two blocks of MM values at each point along the boundary. For example, at each point along the right boundary the user must supply MM boundary fluxes (QR1) for the in-down directions ($\mu < 0$, $\eta < 0$) and MM values (QR2) for the in-up directions ($\mu < 0$, $\eta > 0$). The ordering of these values in direction corresponds directly to the ordering of direction cosines given in Fig. 4. The detailed input ordering is described in the next section.

TABLE III

ORDERING OF ANISOTROPIC
DISTRIBUTED SOURCE COMPONENTS

Component Number	n	k
1.	0	0
2.	1	0
3.	1	1
4.	2	0
5.	2	1
6.	2	2
7.	3	0
8.	3	1
9.	3	2
10.	3	3
.		
.		
.		
etc.		

An incoming flux on the boundary corresponds to a source of particles which is computed from

$$\text{Source} = \sum_{j=1}^{JT} \sum_{m=1}^{MM} w_m\mu_m(QR1_{m,j} + QR2_{m,j})A_{IT+\frac{1}{2},j} \qquad (66)$$

for the right boundary and from

$$\text{Source} = \sum_{i=1}^{IT} \sum_{m=1}^{MM} w_m\eta_m(QT1_{m,i} + QT2_{m,i})B_i \qquad (67)$$

$$\text{Source} = \sum_{i=1}^{IT} \sum_{m=1}^{MM} w_m\eta_m(QB1_{m,i} + QB2_{m,i})B_i \qquad (68)$$

for the top and bottom boundaries. The sum of the values of these sources for all groups is added to the value (if any) of the volume-energy integrated, distributed sources for the purpose of normalization of all inhomogeneous sources.

4. Source and Flux Input Options

a. Source Input Options

If a source of anisotropy IQAN is designated, the NMQ = [(IQAN+1)(IQAN+2)]/2 components of the source must be entered for each group in the order listed above. This may be accomplished with the use of six options designated by the input value of IQOPT. For IQOPT=0, a value of zero is automatically entered for all elements of the distributed source, and then input boundary fluxes are read. In the remaining options, the various arrays described below are supplied for each group, first for group 1, then for group 2, and so forth, ending with group IGM (the lowest energy group if the Los Alamos cross-section format is used). For each group the boundary sources (if any) are entered after the distributed source. If distributed or boundary sources are specified in any group, values must be entered in all groups. The ordering of source input within each group is

1. $[(Q_{gijn}, i = 1, IT), j = 1, JT]$ for n = 1, NMQ
2. Optionally: a. $[(QR1_{gjm}, j = 1, JT), m = 1,MM]$, then
   b. $[(QR2_{gjm}, j = 1, JT), m = 1,MM]$
3. Optionally: a. $[(QB1_{gim}, i = 1, IT), m = 1,MM]$ then
   b. $[(QB2_{gim}, i = 1, IT), m = 1,MM]$
4. Optionally: a. $[(QT1_{gim}, i = 1, IT), m = 1,MM]$ then
   b. $[(QT2_{gim}, i = 1, IT), m = 1,MM]$.

The above notation is that of standard Fortran. For example, the source Q is entered in NMQ blocks of ITxJT numbers, in a continuous stream, the first IT numbers for j = 1, the next IT numbers for j = 2, etc., all for n = 1. Then, for n = 2, IT numbers are entered for j = 1, and so on. For the boundary fluxes, the QR1 block corresponds to the in-down directions and the QR2 block to the in-up directions. Each of these blocks is entered by supplying JT numbers for m = 1, then JT numbers for m = 2, etc. The top boundary flux block, QT1, is for fluxes in the in-down direction, and the block QT2 is for the out-down directions. The block QB1 is for in-up directions, and QB2 is for out-up directions.

In addition to IQOPT=0 in which Q_{gijn} is automatically set to zero, there are five other options to simplify the reading of Q_{gijn}. These options are all executed for each anisotropic component of Q, i.e., for n = 1, 2, ..., NMQ. These options are:

IQOPT=1 Enter an energy spectrum $(GR_{g,n}$, g=1, IGM) for each n. Then

$Q_{gijn} = GR_{g,n}$ for all i and j for each g and n.

IQOPT=2 Enter the complete array Q_{gijn} as described above.

IQOPT=3 Enter first a spectrum $(GR_{g,n}$, g=1, IGM) as in option 1. Then enter a shape

$[(F_{ijn}, i = 1, IT), j = 1, JT]$ in a continuous block of ITxJT numbers. Then the source is made by

$Q_{gijn} = GR_{g,n}F_{ijn}$ for all i and j for each g and n.

IQOPT=4 Enter a spectrum $(GR_{g,n}$, g=1, IGM), an i-directional spatial shape $(X_{i,n}$ i=1, IT), and a j-direction spatial shape $(Y_{j,n}$, j=1, JT). The source is then given by $Q_{jijn} = GR_{g,n}X_{i,n} Y_{j,n}$ for all i and j for each g and n.

IQOPT=5 The entire source is read from a standard interface file FIXSRC mounted on unit IFIXSR.

b. Flux Input Options

Options for reading an input flux guess are similar to those for reading an input source. If ISCT is the order of anisotropic scattering, then there are NM = (ISCT+1)(ISCT+2)/2 spherical harmonic components of the angular flux.* Options for entering these components are selected by the input value of the integer ISTART. A negative value for ISTART indicates that only the angle-integrated or scalar flux is to be read. Allowed values of ISTART are:

*These components are ordered exactly as are those of anisotropic sources. See Table III.

ISTART OPTION

-5 An entire scalar flux guess from standard interface file RTFLUX or ATFLUX is read from unit ITFLUX.

-4 Same as option 4 except that only the first (isotropic) component of the NM components is entered.

-3 Same as option 3 except isotropic components only.

-2 Same as option 2 except isotropic components only.

-1 Same as option 1 except isotropic components only.

0 No flux guess required, but a fission guess (unity in every mesh cell) is automatically supplied.

1 As in source option 1, a spectrum is supplied so that

$FLUX_{gijn} = GR_{gn}$.

2 The entire array $FLUX_{gijn}$ is entered in blocks of ITxJT continuous numbers first for the first component, then the second, up to NM components. Then this process is repeated for the next group. Imagine that fluxes are read by:

```
   DØ 10 G = 1, IGM
   DØ 10 N = 1, NM
 10 READ ((FLUX(G,I,J,N), I = 1, IT),
         J = 1, JT).
```

3 Just as in source option 3, a spectrum and an (i,j) shape are entered. Then the flux is given by

$FLUX_{gijn} = GR_{g,n}F_{ijn}$.

4 Just as in source option 4, a spectrum, an i shape, and a j shape are read so that the flux is

$FLUX_{gijn} = GR_{g,n}X_{i,n}Y_{j,n}$.

6 A problem-restart dump is read from unit NDUMP1.

5. Flux Dumps and Restart Procedures

Three types of dumps are taken and each dump has the same form and may be used to restart a problem. A periodic dump is taken every M minutes where M is a program variable which can be set to meet particular installation requirements. A final dump is always taken after the successful completion of a problem, and a time limit dump is taken after a user-specified period of time. Dumps are written alternately on units NDUMP1 and NDUMP2 depending on which is free; an output message is written to indicate which unit contains the latest dump.

When problem execution is continued using a restart dump, certain input parameters can be changed and edit specifications can be added or modified. It is even possible to use the program to edit a final dump. However, if this option is selected and more information is required to perform the edit, one more outer iteration may be required before the edit is performed. An example of this situation is the editing of a problem for which the material mesh and rebalance mesh differ and for which neither angular flux storage nor an edit was requested until the restart using the final dump. Then an extra outer iteration is required to store the angular fluxes to be used during the edit. Similarly, if a restart of a final dump is performed to create a standard interface angular flux file, then another outer iteration is required to create these fluxes unless the problem had already required their storage.

To restart a problem, a special problem input deck, consisting of three sections, is required. The first section is the same as the normal problem integer input with the value of ISTART set to six. During restart all other integer values are ignored. The second section of restart input makes use of the namelist feature standard to FORTRAN to permit the user to change certain input parameters (those listed below) and to enter only those he wants to change. If no changes are desired, this section is omitted.

The special character for beginning and terminating a namelist block may vary from machine to machine. On CDC computers the dollar character is used. The first column of namelist cards is ignored. Columns two through eight must contain $TWOINP, and column nine must contain a blank for the first card. Using entries of the free-field form name = value, the user defines his changes, separating different entries by commas and ending the last entry with a dollar sign. Continuation cards are permitted provided that the last entry of the preceding card ends with a comma. The integers which may be changed are:

1. IITL Maximum number of inner iterations.
2. ITLIM Time limit.

3. IEDOPT Edit options.
4. I2 Final flux print indicator.
5. I4 Final fission print indicator.
6. I6 Coarse-mesh balance table print indicator.
7. IANG Angular flux storage indicator.
8. IFO Interface file output indicator.

The floating point values which may be changed are:

1. EV Eigenvalue guess.
2. EVM Eigenvalue modifier.
3. PV Parametric value of k_{eff} or alpha.
4. XLAL Search lambda lower limit.
5. XLAH Search lambda upper limit.
6. XLAX Fine-mesh search precision.

The third section of restart input is the edit section. The composition of the edit input section is determined by the value of IEDOPT after the namelist section is read. If the original problem had IEDOPT > 0 and if IEDOPT is not changed to zero or minus five on restart, then the edit input must be re-entered. Upon restart the sign of IEDOPT is examined. If the sign is positive, the problem proceeds normally to convergence. If the sign of IEDOPT is negative and all information required for an edit exists, the final output portion of the program is executed at once with no inner iterations. If the sign of IEDOPT is negative and all the information required for an edit does not exist, a final computational pass is made through all groups prior to execution of the final output portion of the program. A pseudo edit option of IEDOPT equal minus five is provided which requires no edit input to gain entry into the final output portion (e.g., to create an interface file).

6. Coarse-Mesh Rebalancing

All of the operations of the coarse-mesh rebalancing algorithms are automatic. The user, however, can select the zones in which particle balance is to be enforced. The zones needed to describe material discontinuities normally provide all the zones necessary for rebalance, but in homogeneous or nearly homogeneous systems it is frequently advantageous to add zones to accelerate convergence. Experience has shown that there is a problem-dependent optimum number of zones which most rapidly accelerates convergence. This number is close to the limit in which the coarse mesh coincides with the fine mesh. However, too many coarse-mesh zones waste core storage and computing time.

The user should keep in mind that the solution of the coarse-mesh equations is iterative. A single line inversion in the i-direction is made, starting at the bottom j-level and moving to the top. If the system is a single coarse-mesh zone high, then there is no iteration, but in a large many-zoned system (particularly a thermal system) the inversion of the five-point rebalance equations may become time-consuming.

A zone-dependent balance edit is automatically made for each of the rebalance regions. Inflows, outflows, absorptions, and sources are given for each zone so that it is sometimes convenient to construct a zone simply to obtain this edit.

7. Eigenvalue Searches

It is possible in TWOTRAN-II to adjust nuclide concentrations, system dimensions, or the value of the time absorption to achieve a desired value of k_{eff}. This value is taken to be unity (criticality) unless the parametric eigenvalue trigger (IPVT) is set to unity. In this case, the parametric value of k_{eff} is entered as an input number. If IPVT=2, a 1/v absorber of value PV is added to the problem in each space cell.

The modification of cross-section concentrations takes place as indicated in Section III.B.1.b. This type of problem is run when the eigenvalue type indicator (IEVT) is 3. If IEVT is 2 (time absorption computation), the value EV/v_g is added to the absorption and total cross sections in each group. Here v_g is the speed associated with energy group g. If IEVT = 4, the coarse-mesh boundaries can be modified selectively. Either the i-boundaries (IXM = 1) or the j-boundaries (IYM = 1), or both, can be altered. In each case the modified boundaries (XRADA or YRADA) are calculated from the initial boundaries (XRAD or YRAD) by

$$XRADA_{k+1} = XRADA_k + (XRAD_{k+1} - XRAD_k)*(1 + EV*XM_k)$$
$$k = 1, 2, \ldots, IMC \quad , \qquad (69)$$

$$YRADA_{\ell+1} = YRADA_\ell + (YRAD_{\ell+1} - YRAD_\ell)*(1 + EV*YM_\ell)$$
$$\ell = 1, 2, \ldots, JMC \quad , \qquad (70)$$

where IMC and JMC are the number of material mesh intervals.

In these equations, XM and YM are zone modification coefficients that must be entered by the user. Clearly, if one of these numbers is zero, that particular zone is not altered. If all XM are unity, the i-dimension is uniformly expanded or contracted. Many sophisticated changes can be made. For example, an interface between two zones can be moved while the rest of the system is left unchanged.

Regardless of the parameter being adjusted, the search is executed by performing a sequence of k_{eff} calculations, each for a different value of the parameter being treated as the eigenvalue. Each of the successive k_{eff} calculations is accelerated by coarse-mesh rebalancing, but the search for the desired value of k_{eff} is conducted by subroutine NEWPAR. Regardless of the nature of the problem, the search is for a value of the parameter which makes the value of λ defined in Eq. (53) unity.

In the following description of NEWPAR, it is helpful to refer to Fig. 9 in which the deviation of λ from unity is plotted against outer iteration number.

For the initial system, NEWPAR continues outer iteration until two successive values of λ differ by less than EPSO. For subsequent sequences of λ values, a different convergence precision, XLAX, is used. After the first converged λ sequence is obtained, the initial value of the eigenvalue (EV) is altered by EVM, an input value. If $\lambda > 1$ (multiplying system), the new eigenvalue is equal to EV + EVM; if $\lambda < 1$ (decaying system), the new value is EV - EVM. These alterations correspond to the addition or the subtraction of an absorption, e.g., as in a time-absorption search or a poison-concentration search. However, certain system changes act as negative absorptions, and the user must adjust the sign of EVM accordingly; otherwise the modification made will take the system away from the desired value of k_{eff} and the search will take longer. For example, if all i-dimensions are altered uniformly, an expansion acts as a negative absorption, and thus a negative value of EVM should be used.

Basically, after two values of k_{eff} (λ) are obtained for two different system configurations, subroutine NEWPAR attempts to fit a curve through the most recent values to extrapolate or interpolate to a value of unity. Depending on the amount of information available and the size of $|1 - \lambda|$, this fit proceeds in different ways. A parabolic fit cannot be made until three converged values of λ are available, and is not attempted unless $|1 - \lambda|$ is greater than an input-search lower limit (XLAL) and less than an input-search upper limit (XLAH).

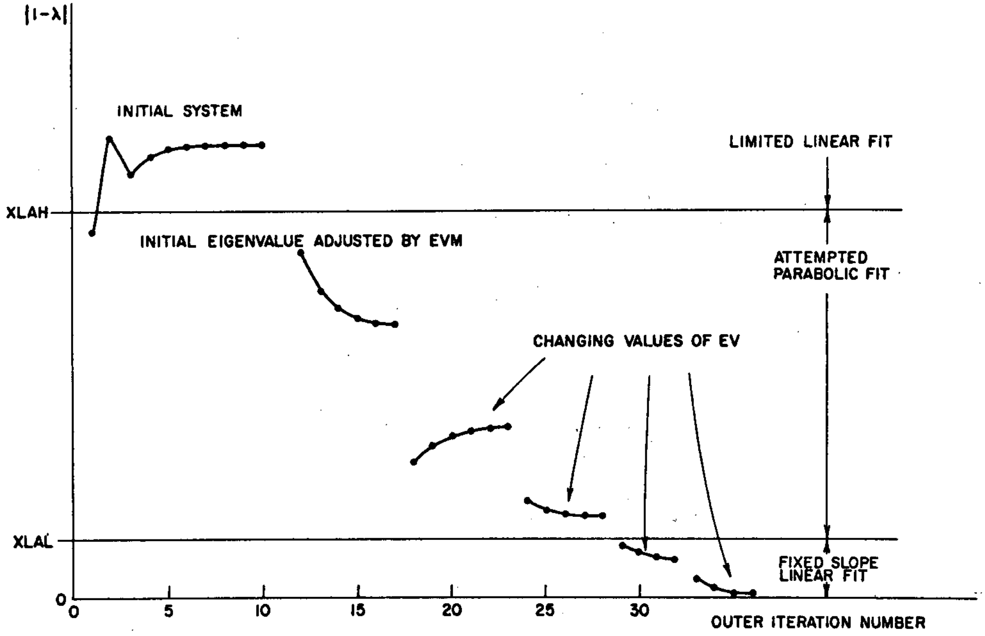


Fig. 9. Variation of λ during a hypothetical eigenvalue search.

If a parabolic fit is tried and the roots are imaginary, a straight-line fit is used. If the roots are not imaginary, the closest root is used as the new value of EV. Once a bracket is obtained (change of sign of $\lambda - 1$), the fit procedure is not allowed to move outside the region of the bracket. Should a parabolic fit select an eigenvalue outside the bracket region, this value is rejected and the new value is taken to be one-half the sum of the previous value and the value previous to that.

Whenever the parabolic fit is not used, a linear fit is used and the new eigenvalue is computed from

$$(EV)_{new} = (EV)_{old} + POD*EQ*(1 - \lambda) \quad , \qquad (71)$$

where POD is an input "parameter oscillation damper" which may be used to restrict the amount of change in the eigenvalue. In Eq. (71), EQ is a measure of the slope of the curve. When $|1 - \lambda| >$ XLAH, $(1 - \lambda)$ in Eq. (71) is replaced by XLAH (with the correct sign) to prevent too large a change in EV. After $|1 - \lambda| <$ XLAL, the value of EQ is fixed and kept constant until convergence to prevent numerical difficulty in the approximation of the derivative when λ is close to unity.

Because parametric search problems represent sequences of k_{eff} calculations, it behooves the user to study the use of subroutine NEWPAR in order to optimize his calculations. It also behooves the user to pose soluble problems. That is, there are many problems, especially concentration searches, for which solutions are not possible, and discovering this by trial and error is the hard way. Ideally, the user will have some estimate of the critical parameter available from a lower order computation.

Convergence in time-absorption calculations is typically one-sided. That is, if EV is negative, then there is a possibility that the denominator of Eq. (8) will become negative. If this happens, the automatic search procedure fails (dramatically). For this reason POD = 0.5 or less is frequently used in such searches.

8. Adjoint Computations

The TWOTRAN-II program solves the adjoint transport equation by transposing the matrices of scattering coefficients and inverting the group order of the problem. The solution of the resulting problem in direction $\vec{\Omega}$ is then identified with the solution of the adjoint equation in direction $-\vec{\Omega}$.[11]

The inversion of the group order is made because the transposition of the scattering matrices usually converts a downscattering problem to an upscattering problem. Because of the inversion, the user must:

(a) Enter any inhomogeneous sources, including boundary fluxes, in inverse group order,

(b) Enter any flux guess in inverse group order, and

(c) Remember that any output is in inverse group order, i.e., that groups labeled 1, 2, ..., are really group IGM, IGM - 1, etc.

Similarly, the output flux from an adjoint problem must be inverted before insertion in a direct problem. On the other hand, an output flux from one adjoint problem is in the proper group order for use in another adjoint problem.

The group order of the group speeds and the fission spectrum is inverted by the program.

9. Edit Options

During an optional final outer iteration, a balance table can be prepared for each coarse-mesh zone in each group. This table lists the absorption, source, inflows, and outflows for each coarse-mesh interval. The absorption in this table is actually the removal rate used in the coarse-mesh rebalance. An identical balance table is also prepared for the sum of the quantities over all groups. In this table the group sum of the true absorption appears.

In addition to these balance tables, the user of TWOTRAN-II may also edit the output in a variety of ways. First of all the user may obtain a macroscopic activity edit. In this edit, the activity of cross sections 1 through IHT is computed for each group for the macroscopic cross sections used in the problem. These activities are computed for edit zones defined by the user by entering sets of numbers which identify each material mesh cell with an edit zone. As many such sets of numbers may be used as desired. That is, the user may perform the edit calculation for as many zone configurations as he chooses. An edit zone is simply a collection of material mesh cells with the same zone number. The material mesh cells of a zone need not be contiguous. In addition to the macroscopic activity edit, the user is also given the zone right, left, top, bottom, and net leakage, the zone source integral, and the zone flux integral. The zone absorption is given by the macroscopic activity for the cross section IHT-2.

The user may also obtain all the above information plus a microscopic activity edit. For this edit, the user enters the number of microscopic activities he desires and the cross-section block numbers of the microscopic cross sections he is interested in. Cross-section blocks for this edit must appear in the MIXNUM array; i.e., they must be part of the mixture specifications. If the microscopic cross section appears in several mixtures, the total density of the material appearing in the table is used in calculating the activity. To edit a material which is not actually part of the problem, the user may simply add a mixture instruction to the mixture tables; or, if interested in only a few cross sections, he may add these cross sections to other blocks in rows IHT-5, IHT-6, etc.

For either the microscopic or the macroscopic edit, the user may also obtain a zone relative power density. In this calculation, the group sum of the zone volume integral of the fission rate (times ν, the number of secondaries per fission) is divided by the zone volume, for all zones. These numbers are then normalized to (divided by) the value of this quantity for a user-designated zone. If the user selects zone zero, the normalization is to the whole system power density.

All of the edit calculations are performed in subroutine EDITØR. Certain rearrangements of core storage are made by subroutine EDCALL, and a picture of the edit zones is drawn by EDMAP. Edit input must be entered for any problem restart, and thus can be changed at this point. A Final Dump tape may also be edited by using it as a restart dump for the same problem.

Any of the normal edit options may be performed immediately, i.e., without further iterative calculation, from any dump tape by tagging the edit option input number IEDOPT negative. The input required for the edit is that required for a normal problem restart. If angular flux information is required for such an edit, one complete outer iteration is performed to provide it. The negative edit option may be used to repeatedly edit, in different ways, a final dump tape.

A special edit option of IEDOPT = -5 provides a means of restarting a problem and going directly to the output section, say to print output previously suppressed or to create a standard interface file.

C. Data Input Rules

Except for the control parameters, cross sections, and edit parameters, all floating-point numbers and integers are read into TWOTRAN-II in special formats. These formats are [6,(I1,I2,E9.4)] for reading floating-point numbers and [6,(I1,I2,I9)] for integers. In each word of both of these formats, the first integer field, I1, designates the options listed below. The second integer field, I2, controls the execution of the option, and the remainder of the field, I9 or E9.4, is for the input data. All data blocks read with these formats must be ended with a 3 in the I1 field after the last word of the block. The available options are given in Table V.

TABLE V

OPTIONS FOR SPECIAL READ FORMATS

Value of I1	Nature of Option
0 or blank	No action
1	Repeat data word in 9 field number of times indicated in I2 field.
2	Place number of linear interpolants indicated in I2 field between data word in 9 field and data word in next 9 field. <u>Not allowed for integers</u>.
3	Terminate reading of data block. <u>A 3 must follow last data word of all blocks</u>.
4	Fill remainder of block with data word in 9 field. This operation must be followed by a terminate (3).
5	Repeat data word in 9 field 10 times the value in the I2 field.
9	Skip to the next data card.

Five illustrations of the use of the special formats are given below. These illustrate:

1 - Zero is repeated 47 times.

2 - Zero is repeated 470 times.

3 - Four interpolants are inserted between 0.0 and 5.0 giving six data numbers: 0.0, 1.0, 2.0, 3.0, 4.0, 5.0.

4 - Four interpolants are inserted between 0.0 and 5.0, two between 5.0 and 7.0, and 7.0 is repeated 10 times.

5 - After reading 0 and 4 we skip to next card and read 7.

I1	I2		I1	I2		I1	I2		I1	I2	
I	NN		I	NN		I	NN		I	NN	
1	47	0.0	3								EXAMPLE 1
5	47	0.0	3								EXAMPLE 2
2	4	0.0			5.0	3					EXAMPLE 3
2	4	0.0	2	2	5.0	1	10	7.0	3		EXAMPLE 4
		0			4	9					EXAMPLE 5
		7	3								EXAMPLE 5

A special routine, WRITE, is used to print most of the two- and three-dimensional arrays that occur in the program. This routine can be used for one-, two-, or three-dimensional arrays and has an option for printing a portion of an array, e.g., the mixed cross-section blocks, if any. When fluxes, sources, and fission rates are printed, they are oriented with the origin at the lower left of the page, corresponding to the picture drawn by MAPPER.

D. Description of Input Data

In the following pages the input data for TWOTRAN-II are listed in exactly the order in which they are entered in the code. The data are divided into three categories: (1) job title cards, (2) control integers on cards 1 through 3 and control floating-point numbers on cards 4 and 5, and (3) problem-dependent data on subsequent cards. Categories (1) and (2) can be entered on the form shown in Fig. 10.

1. Job Title Cards

The user begins by indicating on a card in an I6 format the number of title or job description cards he wants to use. He then enters the descriptive material on these cards which are read with a 18A4 format.

2. Input of Control Numbers

On cards 1 through 3, the user enters the following control integers which are read in a 12I6 format, except as noted:

Number of Word on Card	Name of Variable	Comments
CONTROL INTEGERS (12I6)		-- CARD 1
1	ITH	0/1 (direct/adjoint) type of calculation to be performed.
2	ISCT	0/N (isotropic/Nth-order anisotropic) order of scattering calculation. There are NM = (ISCT + 1)(ISCT + 2)/2 spherical harmonics flux components computed. These are not used to compute a scattering source unless some zone material identification number is negative. See IDCS below.
3	ISN	S_n Order. Even integer only. If negative, quadrature coefficients are taken from interface file SNCONS. Otherwise (for ISN = 2 through 16) built-in constants are used.
4	IGM	Number of groups.
5	IM	Number of rebalance coarse-mesh intervals in the i-direction. See entries IMC and XRAD below.
6	JM	Number of rebalance coarse-mesh intervals in the j-direction. See entries JMC and YRAD below.
7	IBL	Left boundary condition: 0/1 vacuum/reflective.
8	IBR	Right boundary condition: 0/1/2 vacuum/reflective/white.
9	IBB	Bottom boundary condition: 0/1/2/3 vacuum/reflective/white/periodic.
10	IBT	Top boundary condition: 0/1/2/3 vacuum/reflective/white/periodic.
11	IEVT	Eigenvalue type - 0/1/2/3/4 - inhomogeneous source (Q)/k_{eff} calculation/time absorption (alpha) search/nuclide concentration (C) search/zone thickness (delta) search.

12 ISTART Input flux guess and starting options - -5/-4/-3/-2/-1/0/1/2/3/4/6. See Section III.B.4.

CONTROL INTEGERS (I6, 2I3, 6I6, 3I2, 3I6) -- CARD 2

1 (I6) MT Total number of materials (cross-section blocks including anisotropic cross sections) in the problem.

2 (I3) MTPS Number of input material sets from the interface file ISOTXS. Caution: each material set from this file yields ISCT+1 materials. See IDLIB below.

3 (I3) MCR Number of input materials from the code dependent input file. If this number is negative, FIDO format cross sections are read.

4 (I6) MS Number of mixture instructions. See Section III.B.1 and items MIXNUM, MIXCOM, and MIXDEN below.

5 (I6) IHT Row of total cross section in the cross-section format. If IHT < 0, code assumes that there is no σ^{up} in cross-section table.

6 (I6) IHS Row of within-group scattering cross section in the cross-section format.

7 (I6) IHM Total number of rows in the cross-section format.

8 (I6) IQOPT 0/1/2/3/4/5 options for input of inhomogeneous source. See Section III.B.4.

9 (I6) IQAN Order of anisotropy of inhomogeneous distributed source.

10 (I2) IQR Right boundary source to be specified as input (0/1-no/yes). The source is the value of the incoming flux on the right boundary. See QR1 and QR2 below.

11 (I2) IQB Bottom boundary source to be specified as input (0/1-no/yes). The source is the value of the incoming flux on the bottom boundary. See QB1 and QB2 below.

12 (I2) IQT Top boundary source to be specified as input (0/1-no/yes). The source is the value of the incoming flux on the top boundary. See QT1 and QT2 below.

13 (I6) IPVT 0/1/2 none/k_{eff}/alpha parametric eigenvalue entered. See entry PV below.

14 (I6) IITL Maximum number of inner iterations allowed per group.

15 (I6) IXM 0/1 (no/yes) Are the i-direction zone thicknesses to be modified? See entry XM below.

CONTROL INTEGERS (5I6, 6I1, 6I6) -- CARD 3

1 (I6) IYM 0/1 (no/yes) Are the j-direction zone thicknesses to be modified? See entry YM below.

2 (I6) ITLIM 0/seconds If an integer number of seconds is entered, a restart dump is taken after this number of seconds and the problem is terminated.

3 (I6) IGEOM 1/2/3 (x,y)/(r,z)/(r,θ) geometry.

4 (I6) IEDOPT 0/1/2/3/4 (none/option) Edit options. Option 1 is a macroscopic edit, Option 2 is a macroscopic plus microscopic edit. Options 3 and 4 give the information of options 1 and 2 (respectively) plus a zone relative power density edit.
If IEDOPT is -1, -2, -3, or -4, an edit will be performed immediately if all necessary data is present. If additional data is needed (e.g. angular fluxes), one outer iteration is performed and then an edit is performed. If IEDOPT = -5, direct access to the program output section is provided, say to create an output interface file from a final dump.

5 (I6) ISDF 0/1 no/yes density factor input indicator. See entries XDF and YDF below.

6 (I1) I1 0/1 no/yes full input flux print suppression indicator.

7 (I1) I2 0/1/2 all/isotropic/none final flux print indicator.

8 (I1) I3 0/1/2 all/mixed/none cross section print indicator.

9 (I1) I4 0/1 yes/no final fission print indicator.

10 (I1) I5 0/1/2/3 all/input/normalized/none source input print indicator.

11 (I1) I6 0/1 yes/no prepare and print coarse-mesh balance tables. The tables are for the rebalance mesh when the rebalance mesh and material mesh are different. Caution: The preparation of these tables requires an additional outer iteration after problem convergence.

12 (I6) IANG -1/0/1 print and store/no/store angular flux indicator. The preparation of angular fluxes requires an additional outer iteration after problem convergence as well as additional storage.

13 (I6) IMC 0/N no/number of material coarse-mesh intervals in the i-direction. When this value is non-zero the rebalance coarse-mesh as given by items IM and JM is not the same as the material coarse-mesh. The material coarse-mesh is the same as the mesh upon which all edits are done. When edits are requested and IMC ≠ 0, angular fluxes must be stored. See entries IDCS, XM, IHXC, and XRAD below.

14 (I6) JMC 0/N no/number of material coarse-mesh intervals in the j-direction. When this value is non-zero the rebalance coarse-mesh as given by items IM and JM is not the same as the material coarse-mesh.

When the rebalance mesh and material mesh are no longer identical, rebalance zones may include several different material zones subject to the restriction that rebalance boundaries must coincide with some material boundaries.

15 (I6) IFO 0/1 no/yes interface file output is created. Total (angle-integrated) flux and SNCONS files are always created. Angular flux file is created only if IANG ≠ 0.

CONTROL FLOATING-POINT DATA (6E12.4) -- CARD 4

1 EV Eigenvalue guess. It is satisfactory to enter 1.0 for IEVT = 3 and 0.0 for all other problems.

2 EVM Eigenvalue modifier used only if IEVT > 1. See Section III.B.7 above.

3 PV Parametric value of k_{eff} for subcritical or supercritical systems or for 1/v absorption. See Section III.B.7 above.

4 XLAL Lambda lower limit for eigenvalue searches. See Section III.B.7 above.

5 XLAH Search lambda upper limit.

6 XLAX Search lambda convergence precision for second and subsequent values of the eigenvalue.

CONTROL FLOATING-POINT DATA (6E12.4) -- CARD 5

1 EPS Convergence precision.

2 NORM Normalization factor. Total number of particles in system normalized to this number if it is nonzero. No normalization if NORM is zero.

3 POD Parameter oscillation damper used in eigenvalue searches. See Section III.B.7.

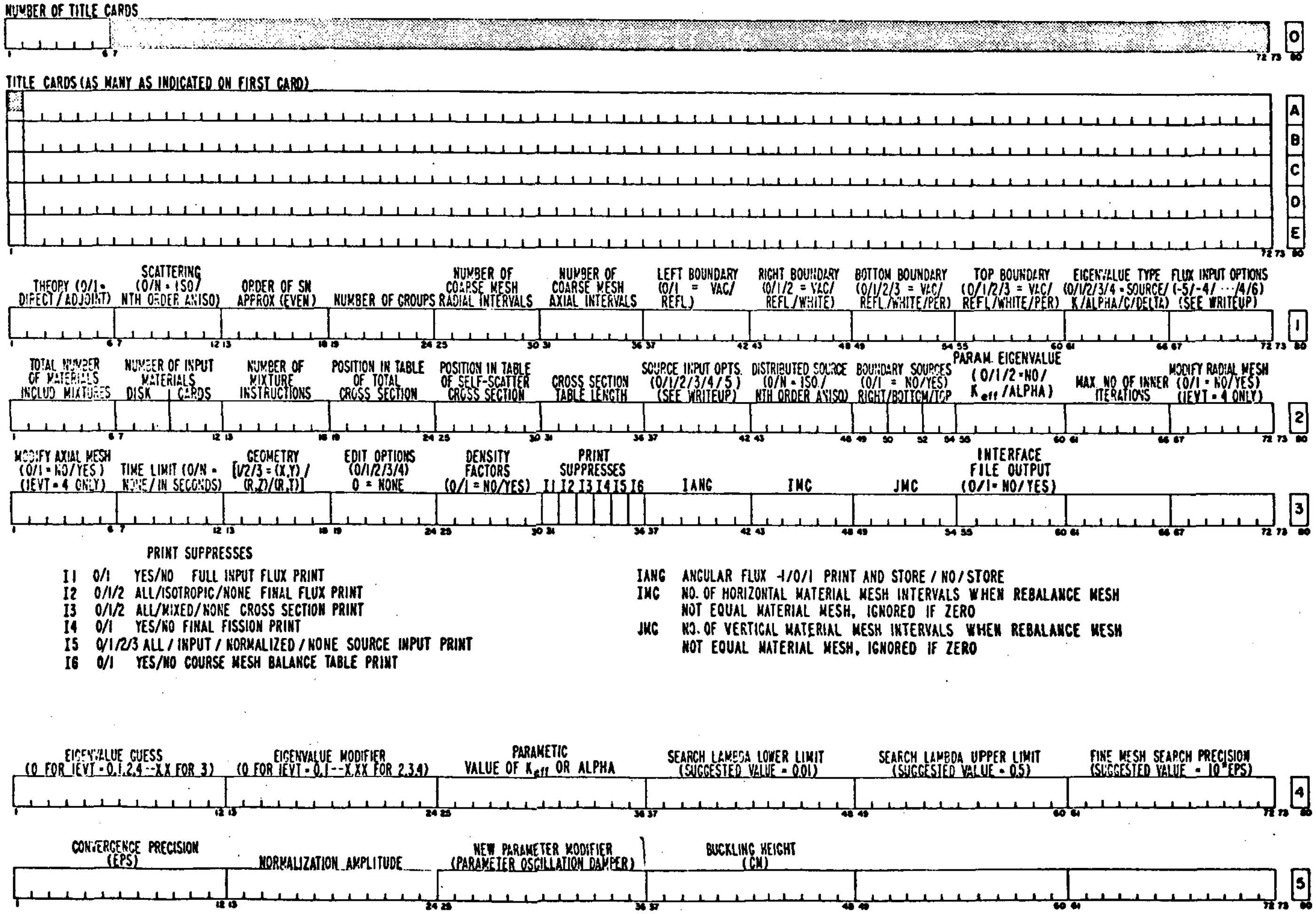
4 BHGT Buckling height (in cm if cross sections are in barns) used to simulate z-dimension of system by adding an absorption given by

$$\sigma_{a,BHGT} = \frac{\sigma_t}{3} (\pi/(BHGT*\sigma_t + 1.4209))^2 \quad .$$

Here 1.4209 is twice the Milne problem extrapolation distance, and σ_t is the total cross section. Used in (r,θ) and (x,y) geometry only.

TWOTRAN - II

Programmer________________________ Page ______



NUMBER OF TITLE CARDS

			0
1–6	7–72	73–80	

TITLE CARDS (AS MANY AS INDICATED ON FIRST CARD)

		A
		B
		C
		D
		E
1–72	73–80	

THEORY (0/1= DIRECT/ADJOINT)	SCATTERING (0/N = ISO/ NTH ORDER ANISO)	ORDER OF SN APPROX (EVEN)	NUMBER OF GROUPS	NUMBER OF COARSE MESH RADIAL INTERVALS	NUMBER OF COARSE MESH AXIAL INTERVALS	LEFT BOUNDARY (0/1 = VAC/ REFL)	RIGHT BOUNDARY (0/1/2 = VAC/ REFL/WHITE)	BOTTOM BOUNDARY (0/1/2/3 = VAC/ REFL/WHITE/PER)	TOP BOUNDARY (0/1/2/3 = VAC/ REFL/WHITE/PER)	EIGENVALUE TYPE (0/1/2/3/4 = SOURCE/ K/ALPHA/C/DELTA)	FLUX INPUT OPTIONS (-5/-4/ ···/4/6) (SEE WRITEUP)		
													1
1–6	7–12	13–18	19–24	25–30	31–36	37–42	43–48	49–54	55–60	61–66	67–72	73–80	

TOTAL NUMBER OF MATERIALS INCLUD. MIXTURES	NUMBER OF INPUT MATERIALS DISK	CARDS	NUMBER OF MIXTURE INSTRUCTIONS	POSITION IN TABLE OF TOTAL CROSS SECTION	POSITION IN TABLE OF SELF-SCATTER CROSS SECTION	CROSS SECTION TABLE LENGTH	SOURCE INPUT OPTS. (0/1/2/3/4/5) (SEE WRITEUP)	DISTRIBUTED SOURCE (0/N = ISO./ NTH ORDER ANISO)	BOUNDARY SOURCES (0/1 = NO/YES) RIGHT	BOTTOM	TOP	PARAM. EIGENVALUE (0/1/2 = NO/ K_{eff} /ALPHA)	MAX. NO OF INNER ITERATIONS	MODIFY RADIAL MESH (0/1 = NO/YES) (IEVT = 4 ONLY)		
																2
1–6	7–	–12	13–18	19–24	25–30	31–36	37–42	43–48	49–50	–52	–54	55–60	61–66	67–72	73–80	

MODIFY AXIAL MESH (0/1 = NO/YES) (IEVT = 4 ONLY)	TIME LIMIT (0/N = NONE/IN SECONDS)	GEOMETRY [1/2/3 = (X,Y)/ (R,Z)/(R,T)]	EDIT OPTIONS (0/1/2/3/4) 0 = NONE	DENSITY FACTORS (0/1 = NO/YES)	PRINT SUPPRESSES I1	I2	I3	I4	I5	I6	IANG	IMC	JMC	INTERFACE FILE OUTPUT (0/1= NO/YES)				
																		3
1–6	7–12	13–18	19–24	25–30	31					–36	37–42	43–48	49–54	55–60	61–66	67–72	73–80	

PRINT SUPPRESSES

I1	0/1	YES/NO FULL INPUT FLUX PRINT
I2	0/1/2	ALL/ISOTROPIC/NONE FINAL FLUX PRINT
I3	0/1/2	ALL/MIXED/NONE CROSS SECTION PRINT
I4	0/1	YES/NO FINAL FISSION PRINT
I5	0/1/2/3	ALL / INPUT / NORMALIZED / NONE SOURCE INPUT PRINT
I6	0/1	YES/NO COURSE MESH BALANCE TABLE PRINT

IANG ANGULAR FLUX -1/0/1 PRINT AND STORE / NO / STORE

IMC NO. OF HORIZONTAL MATERIAL MESH INTERVALS WHEN REBALANCE MESH NOT EQUAL MATERIAL MESH, IGNORED IF ZERO

JMC NO. OF VERTICAL MATERIAL MESH INTERVALS WHEN REBALANCE MESH NOT EQUAL MATERIAL MESH, IGNORED IF ZERO

EIGENVALUE GUESS (0 FOR IEVT = 0,1,2,4 --X.X FOR 3)	EIGENVALUE MODIFIER (0 FOR IEVT = 0,1 --X.XX FOR 2,3,4)	PARAMETIC VALUE OF K_{eff} OR ALPHA	SEARCH LAMBDA LOWER LIMIT (SUGGESTED VALUE = 0.01)	SEARCH LAMBDA UPPER LIMIT (SUGGESTED VALUE = 0.5)	FINE MESH SEARCH PRECISION (SUGGESTED VALUE = 10*EPS)		
							4
1–12	13–24	25–36	37–48	49–60	61–72	73–80	

CONVERGENCE PRECISION (EPS)	NORMALIZATION AMPLITUDE	NEW PARAMETER MODIFIER (PARAMETER OSCILLATION DAMPER)	BUCKLING HEIGHT (CM)				
							5
1–12	13–24	25–36	37–48	49–60	61–72	73–80	

Fig. 10. Input Data Sheet

3. Input of Remaining Data

In the input data listed below, all the items are dimensionless except for the source, flux, velocities, mesh boundary values, cross sections, and mixture densities. The dimensions of these quantities are arbitrary in the following sense. Macroscopic cross sections define a unit of inverse length (usually cm^{-1} but occasionally km^{-1}) in which the mesh boundary values are measured. The one exception is the θ variable which is measured in fractions of revolutions. For source problems, the flux will have the dimensions of source/cross section where cross section is the quantity used in the calculation. Normally sources are in units of particles/length^3/solid angle/sec (the energy dependence is removed by the multigroup approximation, i.e., $\int QdE$ is used, see Section II.B.), microscopic cross sections are in units of barns x length^2/cm^2, nuclide number densities in units of 10^{24} x number/length^3, and velocities in length/sec, although Los Alamos velocities are habitually measured in units of length/10^{-8} sec.

With the exception of the cross sections from the code dependent input file and three edit parameter specifications, all the following data is loaded by the LASL block loader using the special formats described in Section III.D. We denote these formats by S(I) for integers and S(E) for floating point numbers.

The number of entries for certain data such as block IDCS is dependent on whether the rebalance mesh and the material coarse-mesh are the same. We use the parameter MESH to indicate this difference. We take MESH.EQ.1 when IMC.GT.0 and JMC.GT.0, otherwise MESH.EQ.0.

Block Name and Dimension	Format	Number of Entries	Comments
IHX(IM)	S(I)	IM	Integers defining the number of fine mesh i-intervals in each coarse-mesh k-interval. This vector is used for both the rebalance mesh and material mesh when MESH.EQ.0 and only for the rebalance mesh when MESH.NE.0.
IHY(JM)	S(I)	JM	Integers defining the number of fine mesh j-intervals in each coarse-mesh ℓ-interval. This vector is used for both the rebalance mesh and material mesh when MESH.EQ.0 and only for the rebalance mesh when MESH.NE.0.
C(IHM,IGM,MIN)			Three options are available for reading cross sections. The LASL input format may not be mixed with the FIDO format. 1. LASL INPUT If MCR.GT.0, MCR blocks of IHM*IGM numbers are read in a 6E12.5 format. Each block is preceded by an identification card read in a 18A4 format. MIN = MCR + MTPS*(ISCT+1) 2. FIDO INPUT If MCR.LT.0, MCR blocks of data are created from FIDO input. FIDO input data must be preceded by a 14* (floating point block number 14) loading card when an IBM 360 computer is used, but no card is required on CDC computers. 3. INTERFACE FILE ISOTXS When MTPS.GT.0, MTPS material sets are read from standard file ISOTXS. On this file each material set consists of ISCT+1 cross section blocks for the isotropic and ISCT anisotropic cross sections. The first component of the first material is stored in cross-section block MCR+1, the first component of the second material is stored in cross-section block MCR+ISCT+2, etc. Should the ISOTXS file not contain ISCT anisotropic components, zeroes are supplied for the components not present. If the ISOTXS file contains more components than needed, only the first ISCT+1 components are read.

IDLIB(MTPS)	S(I)	MTPS	Position numbers of material sets to be read from ISOTXS. Do not enter unless MTPS.GT.0. The material sets are read in the order specified in this entry, and this order need not be in order of increasing set identification number.
Input FLUX Guess FLUX(NM,IT,JT)	S(E)		Number of entries depends on option. See Section III.B.4.b. Option Number -5 Input from RTFLUX or ATFLUX standard file -4 IGM+IT+JT -3 IGM+IT*JT -2 IGM blocks of IT*JT -1 IGM 0 None 1 NM sets of IGM 2 IGM groups of NM sets of IT*JT 3 NM sets of IGM+IT*JT 4 NM sets of IGM+IT+JT
Input Source Q(NMQ,IT,JT)	S(E)		Number of entries depends on option. See Section III.B.4.a. Option Number 0 None 1 NMQ sets of IGM 2 IGM groups of NMQ blocks of IT*JT 3 NMQ sets of IGM+IT*JT 4 NMQ sets of IGM+IT+JT 5 Input from standard file FIXSRC
QR1(JT,MM)	S(E)	JT*MM	Right boundary source (flux) in the in-down directions. Do not enter unless IQR = 1.
QR2(JT,MM)	S(E)	JT*MM	Right boundary source (flux) in the in-up directions. Do not enter unless IQR = 1.
QB1(IT,MM)	S(E)	IT*MM	Bottom boundary source (flux) in the in-up directions. Do not enter unless IQB = 1.
QB2(IT,MM)	S(E)	IT*MM	Bottom boundary source (flux) in the out-up directions. Do not enter unless IQB = 1.
QT1(IT,MM)	S(E)	IT*MM	Top boundary source (flux) in the in-down directions. Do not enter unless IQT = 1.
QT2(IT,MM)	S(E)	IT*MM	Top boundary source (flux) in the out-down directions. Do not enter unless IQT = 1.
XRAD(K)	S(E)	K	Coarse k-mesh boundaries. Must form increasing sequence. When MESH.EQ.0, K = IM+1, and entries are used for both the rebalance and material mesh. When MESH.NE.0, K = IMC+1 and entries are used for rebalance mesh only. However, each rebalance mesh boundary must then coincide with some material mesh boundary.
YRAD(L)	S(E)	L	Coarse ℓ-mesh boundaries. Must form increasing sequence. When MESH.EQ.0, L = JM+1, and entries are used for both the rebalance and material mesh. When MESH.NE.0, L = JMC+1 and entries are used for rebalance mesh only. However, each rebalance mesh boundary must coincide with some material mesh boundary.
IDCS(IC)	S(I)	IC	Cross-section zone identification numbers. Number of entries IC = IM*JM if MESH.EQ.0 and IC = IMC*JMC if MESH.NE.0. These numbers assign a cross section block to each zone defined by the material

			coarse mesh. If these numbers are negative, an anisotropic scattering source is calculated in the zone; but the numbers need not be negative when ISCT > 0.
CHI(IGP)	S(E)	IGM	Fission fractions. Fraction of fission yield emerging in each group.
VEL(IGP)	S(E)	IGM	Group speeds. Used only in time absorption calculations.
MIXNUM(MS)	S(I)	MS	Numbers identifying cross-section block being mixed. See Section III.B.1.b. Do not enter if MS = 0.
MIXCOM(MS)	S(I)	MS	Numbers controlling cross-section mixture process. See Section III.B.1.b. Do not enter if MS = 0.
MIXDEN(MS)	S(E)	MS	Mixture densities. See Section III.B.1.b. Do not enter if MS = 0.
XM(I)	S(E)	I	Material i-mesh modification factors. See Section III.B.7. The number of entries, I, is IM when MESH.EQ.0 and IMC when MESH.NE.0. Do not enter unless IEVT=4 and IXM > 0.
YM(J)	S(E)	J	Material j-mesh modification factors. See Section III.B.7. Number of entries, J, is JM when MESH.EQ.0 and JMC when MESH.NE.0. Do not enter unless IEVT=4 and IYM > 0.
XDF(IT)	S(E)	IT	Radial fine-mesh density factors. Do not enter if ISDF.EQ.0.
YDF(JT)	S(E)	JT	Axial fine-mesh density factors. Do not enter if ISDF.EQ.0. The effective cross section at fine mesh point (I,J) is cross section for the point defined by the IDCS array multiplied by the factor XDF(I)*YDF(J).
IHXC(IMC)	S(I)	IMC	Integers defining the number of fine mesh i-intervals in each material coarse-mesh k-interval. Caution: the sum of this vector must be the same as the sum of the IHX vector. Do not enter unless MESH.NE.0.
IHYC(JMC)	S(I)	JMC	Integers defining the number of fine mesh j-intervals in each material coarse mesh ℓ-interval. Caution: the sum of this vector must be the same as the sum of the IHY vector. Do not enter unless MESH.NE.0.
XRADA(IMC+1)	S(E)	IMC	Coarse k-mesh material boundaries. Must form increasing sequence. Do not enter unless MESH.NE.0.
YRADA(JMC+1)	S(E)	JMC+1	Coarse ℓ-mesh material boundaries. Must form increasing sequence. Do not enter unless MESH.NE.0.
NEDS	I6	1	Integer defining number of edits to be performed. Do not enter unless 0<|IEDOPT|<5.
MN	I6	1	Integer defining number of microscopic activities to be computed. See Section III.B.9. Do not enter unless |IEDOPT|=2 or 4.
MICID(MN)	S(I)	MN	Integers defining material blocks for which microscopic edit is to be made. IEDOPT = 2 and 4 only. See Section III.B.9.
NZ,NORMZ	2I6	2	The integer NZ is the number of edit zones. The integer NORMZ is the zone to which the power density is normalized (NORMZ is not used unless IEDOPT = 3 or 4). See Section III.B.9.
NEDZ(IZ)	S(I)	IZ	Integers defining which edit zone each coarse mesh material zone is in. Number of entries, IZ, is IM*JM when MESH.EQ.0 and IMC*JMC when MESH.NE.0. Caution: The edit blocks beginning with NZ,NORMZ must be repeated NEDS times. See Section III.B.9. Do not enter unless 0<|IEDOPT|<5.

E. Output Description for a Test Problem

The TWOTRAN-II program comes with a set of 25 test problems. The problem output presented in the following pages is from the second variation of the fifteenth problem of this set, as identified by the title cards on the first page of the output listing. Each page of the output is numbered, and we refer to these numbers in the text below.

As can be seen from the first page, the problem is a S_6, linearly anisotropic delta search problem with a 2x2 rebalance mesh, a 4x4 material mesh and a complete edit. Because the material mesh and rebalance mesh do not coincide and an edit is requested, the angular flux must be stored. In the first variation of the fifteenth problem, both meshes coincide and no angular fluxes need be stored. The edit results from the first variation problem are used to check those from the second.

Immediately following the integer and floating point input is the input of the number of fine mesh intervals per rebalance interval. Ten radial and ten axial intervals are used to describe the finite test cylinder, making the problem cheap to execute, despite all the options. The remaining problem input is listed on pages 2 and 3. Note that flux input option ISTART = -1 was used to set all values of the isotropic component of the flux to unity. The next two pages contain a picture of the rebalance mesh and the material mesh. In the rebalance map the mesh cells are simply numbered, but in the material map the identification number of the cross section belonging to each zone is listed. Note that material 5 (which must be a mixture because only four cross sections were read as input) is tagged with a minus, indicating anisotropic scattering. Only isotropic scattering is computed for material 7 (material 6 is the linearly anisotropic part of material 5).

From the maps we see that the system is 10.9 units (here cm) in radius and 21.8 cm in height, but that only the top 10.9 cm of the system are calculated by using a reflecting bottom boundary condition. This, of course, assumes that the system is symmetric about the cylinder midplane.

After the maps, the edit input is printed, then the cross section mixture instructions, the mixed cross sections, the fission spectrum and the velocities.

Beginning on page 6 is the monitor of the progress of the calculation following a summary of convergence precisions. The column headed "rebalance convergence" contains the maximum deviation from one of any rebalance factor for the rebalance performed after each outer iteration. If the number of inner iterations in a group exceeds IITL, we print the maximum flux error at that time immediately to the right of the number of inner iterations. In this problem this happens only three times in the second group.

The monitor lines display a typical sequence of events in an eigenvalue search. Here, both radial and axial meshes are modified, with all the radial coarse-mesh zones modified equally, but with only the second and fourth (from z = 0) axial zones modified, and these only half as much as the radial zones. We see that for zero eigenvalue, lambda converges to 0.877289834 (this is k_{eff} for the unaltered system), then for an eigenvalue of 0.1 (the input value of EVM), lambda converges to 0.921909614, and so on until convergence at an eigenvalue of about 0.3065.

Following the monitor lines, balance tables are printed. On page 9 the final monitor line is printed followed by a summary of the original and final coarse-mesh boundaries. Then, isotropic fluxes and currents are printed for each group, and the fine-mesh fission rate is listed. This is the end of output unless there is an edit.

On page 14, the edit zones for the first edit are displayed, including the changed zone boundaries. Here, the edit zones correspond to the locations of materials 5 and 7. On page 15 in the microscopic edits, two items should be noted. First, the reaction rates for material 2 are zero. This is because material two (used to make 6, the anisotropic part of 5) does not appear as a constituent of either 5 or 7, and the edit routine therefore finds no indication that 2 is in the system. That is, only constituents of materials with numbers appearing in the system map can be edited. Second, the reaction rates for material 4 are nonzero only for zone 2 which is the only place material 7 occurs. To edit 4 over the whole system, it would have been necessary to add 4 to 5 in the same manner as it was added to 7.

After page 15, we have omitted the edits for group two and the group sums. On page 16, we show a second edit that can be performed, this time for three zones. Edits can be performed in this problem for up to 16 zones for any arrangement of zone numbers desired.

```
                                                                                                          (1)
   THIS CASE WAS PROCESSED BY THE TWOTRAN-II CODE OF 04/24/73 ON   05/23/73
  EDIT OPTION PROB DELTA CALC  R,Z MODIFIED, BUT NOT EQUALLY
TWOTRAN TEST CASE NUMBER 15  ---VARIATION 2

  0 ITH     0/1 DIRECT/ADJOINT
  1 ISCT    0/N ISOTROPIC/NTH ORDER ANISOTROPIC
  6 ISN     SN ORDER ( +/- BUILT-IN/STANDARD INTERFACE FILE )
  2 IGM     NO. GROUPS
  2 IM      NO. COARSE MESH R INTERVALS
  2 JM      NO. COARSE MESH Z INTERVALS
1 0 1 0     LEFT,RIGHT,BOTTOM,TOP BOUNDARY CONDITION 0/1/2/3 VACUUM/REFLECTIVE/WHITE/PERODIC
  4 IEVT    0/1/2/3/4  Q/K/ALPHA/C/DELTA  CALCULATION
 -1 ISTAR!  -5/-4/-3/-2/-1/0/1/2/3/4/6 STARTING OPTIONS ( MINUS FOR ISOTROPIC COMPONENT ONLY ) SEE MANUAL FOR DETAILS

  7 MT      TOTAL NO. OF MATERIALS
-0   4      STANDARD INTERFACE/CARD INPUT MATERIALS
 10 MS      NO. OF MIXTURE INSTRUCTIONS
  5 IHT     ROW OF TOTAL CROSS SECTION
  6 IHS     ROW OF SELF SCATTER CROSS SECTION
  7 IHM     LAST ROW OF CROSS SECTION TABLE
  0 IQOPT   0/1/2/3/4/5 NONE/SAME AS ISTART FOR SOURCE DISTRIBUTION
  0 IQAN    0/N  ISOTROPIC/NTH ORDER ANISOTROPIC SOURCE
0 0 0       IQR/IQB/IQT RIGHT,BOTTOM,TOP BOUNDARY SOURCE ( 0/1 NO/YES )
  0 IPVT    0/1/2 NONE/K/ALPHA  PARAMETRIC EIGENVALUE TYPE
 10 IITL    MAX NO. INNER ITERATIONS
  1 IXM     0/1 NO/YES MODIFY R RADII (IEVT=4 ONLY)

  1 IYM     0/1 NO/YES MODIFY Z RADII (IEVT=4 ONLY)
240 ITLIM   0/N NONE/SECONDS TIME LIMIT
  2 IGEOM   1/2/3 (X,Y)/(R,Z)/(R,T) GEOMETRY
  4 IEDOP!  0/1/2/3/4=NONE/N EDIT OPTIONS SEE MANUAL FOR DETAILS
 -0 ISDF    0/1 NO/YES INPUT FINE MESH DENSITY FACTORS
 -0 I1      0/1 YES/NO FULL INPUT FLUX PRINT
 -0 I2      0/1/2 ALL/ISOTROPIC/NONE FINAL FLUX PRINT
 -0 I3      0/1/2 ALL/MIXED/NONE CROSS SECTION PRINT
 -0 I4      0/1 YES/NO FINAL FISSION PRINT
 -0 I5      0/1/2/3 ALL/READUF/NORMALIZED/NONE INPUT SOURCE PRINT
 -0 I6      0/1 YES/NO COARSE MESH BALANCE TABLE PRINT
  1 IANG    -1/0/1 PRINT AND STORE/NO/STORE ANGULAR FLUX
  4 IMC     NO. OF HORIZON. MAT.MESH INTERVALS (IF.NE.0, REBAL.MESH.NE.MAT.MESH WHICH IS EDIT MESH)
  4 JMC     NO. OF VERTICAL MAT.MESH INTERVALS
 -0 IFO     0/1 NO/YES STANDARD INTERFACE FILE OUTPUT


 0.          EV      EIGENVALUE GUESS
-1.000E-01   EVM     EIGENVALUE MODIFIER
 0.          PV      PARAMETRIC EIGENVALUE
 1.000E-02   XLAL    SEARCH LAMBDA LOWER LIMIT
 5.000E-01   XLAH    SEARCH LAMBDA UPPER LIMIT
 1.000E-03   XLAX    FINE MESH SEARCH PRECISION

 1.000E-04   EPS     CONVERGENCE PRECISION
 1.000E+00   NORM    NORMALIZATION AMPLITUDE
 1.000E+00   POD     PARAMETER OSCILLATION DAMPER
 0.          BHGT    TOTAL BUCKLING HEIGHT IN CM FOR (X,Y) AND (R,T) ONLY


INPUT FINE R MESH        2
          6              4
```

```
INPUT FINE Z MESH       2
          6             4


STORAGE    REQUIRED ALLOWED
SMALL CORE    1819   40000
LARGE CORE    6542  375000

*******************************************INPUT CROSS SECTIONS****************************************
     1                     LOADED FROM CARDS      IX-F1  COM. MIX FOR F.E.1            P0 IHT= 5  IHS= 6 IHM= 7
     2                     LOADED FROM CARDS      IX-F1  COM. MIX FOR F.E.1            P1 IHT= 5  IHS= 6 IHM= 7
     3                     LOADED FROM CARDS       ABSORBER      IHT=5
     4                     LOADED FROM CARDS      SPECIAL FOR EDIT   IHT=5

 INPUT CROSS SECT       1

        GROUP    1    GROUP    2

    1   .207048E-03  0.
    2   .104011E+00   .339009E+00
    3   .208784E-01   .633738E-01
    4   .701805E-01   .116041E+00
    5   .201152E+00   .564013E+00
    6   .106444E+00   .500640E+00
    7  0.             .650869E-01
 INPUT CROSS SECT       2

        GROUP    1    GROUP    2

    1  0.            0.
    2  0.            0.
    3  0.            0.
    4  0.            0.
    5  0.            0.
    6   .902235E-01   .166424E+00
    7  0.             .163226E-01
 INPUT CROSS SECT       3

        GROUP    1    GROUP    2

    1  0.            0.
    2  0.            0.
    3   .100470E-01   .135048E-01
    4  0.            0.
    5   .100470E-01   .135048E-01
    6  0.            0.
    7  0.           -0.

 INPUT CROSS SECT       4

        GROUP    1    GROUP    2

    1  0.            0.
    2  0.            0.
    3   .100000E+00   .100000E+00
    4  0.            0.
    5   .100000E+01   .100000E+01
    6   .500000E+00   .900000E+00
    7  0.             .400000E+00                                                                                         (3)

IN MATERIAL  1.GROUP  1 INPUT ABSORPTION (  .29878E-01).NE.EFFECTIVE ABSORPTION (  .29621E-01)

* * * * *  INPUT FOR FLUX   * * * * *

 ISOTROPIC COMPONENT

INPUT FLUX  ESHAPE     2
ALL ENTRIES =   1.0000E+00



S 6 CONSTANTS

                  MU             ETA             WEIGHT


      1       .23009194E+00   .94557676E+00   .42361640E-01
      2       .68813432E+00   .68813432E+00   .40971693E-01
      3       .23009194E+00   .68813432E+00   .40971693E-01
      4       .94557676E+00   .23009194E+00   .42361640E-01
      5       .68813432E+00   .23009194E+00   .40971693E-01
      6       .23009194E+00   .23009194E+00   .42361640E-01

INPUT COARSE RMESH     3
  0.          6.5400E+00  1.0900E+01

INPUT COARSE ZMESH     3
  0.          6.5400E+00  1.0900E+01

INPUT CROSS SEC ID    16
          -5          -5          -5           7           7           7          -5          -5          -5          -5
          -5          -5          -5          -5           7           7

INPUT FISSION SPEC     2
  7.5382E-01  2.4618E-01

INPUT VELOCITIES       2
ALL ENTRIES =   1.0000E+00

INPUT MIX NUMBERS     10
           5           5           5           6           6           7           7           7           7           7

INPUT MIX COMMANDS    10
           0           1           3           0           2           0           4           0           1           3

INPUT MIX DENSITY     10
  0.          9.9000E-01  9.9000E-01  0.          9.9000E-01  0.          1.0000E+00  0.          9.9000E-01  9.9000E-01

INPUT R MESH MODS      4
ALL ENTRIES =   1.0000E+00

INPUT Z MESH MODS      4
  0.          5.0000E-01  0.          5.0000E-01

INPUT MATERL RMESH     4
           3           3           2           2

INPUT MATERL ZMESH     4
```

```
          3             3             2             2                                          (4)
INPUT MATBND RMESH      5
  0.          3.2700E+00  6.5400E+00  8.7200E+00  1.0900E+01

INPUT MATBND ZMESH      5
  0.          3.2700E+00  6.5400E+00  8.7200E+00  1.0900E+01
```

```
*************************************************REBALANCE MAP*************************************************************
      Z                                              MATERIALS BY BROAD ZONE, ORIGIN AT LOWER LEFT.
   ROW    M                                          M IS NUMBER OF FINE INTERVALS/BROAD ROW(COLUMN).
   10.9000   1000000000000
             1     *     0
     2    4  1   3 *   4 0
             1     *     0
*    6.5400  1***********0
             1     *     0
     1    6  1   1 *   2 0
             1     *     0
    0.0000   1111111111111
       R     0.    6.    10.
              0   5399   8999

       M          6     4
   COLUMN         1     2
```

```
*********************************************************MATERIAL MAP**********************************************************

     Z                                               MATERIALS BY BROAD ZONE, ORIGIN AT LOWER LEFT.                        (5)
   ROW   M                                           M IS NUMBER OF FINE INTERVALS/BROAD ROW(COLUMN).
  10.9000   1000000000000000000000000
            1     *     *     *     0
    4   2   1  -5 *  -5 *   7 *   7 0
            1     *     *     *     0
   8.7200   1***********************0
            1     *     *     *     0
    3   2   1  -5 *  -5 *  -5 *  -5 0
            1     *     *     *     0
   6.5400   1***********************0
            1     *     *     *     0
    2   3   1   7 *   7 *  -5 *  -5 0
            1     *     *     *     0
   3.2700   1***********************0
            1     *     *     *     0
    1   3   1  -5 *  -5 *  -5 *   7 0
            1     *     *     *     0
   0.0000   1111111111111111111111111
       R    0.    3.    6.    8.    10.
             0  2699  5399  7200  8999

       M       3     3     2     2
   COLUMN      1     2     3     4



   2 NEDS   NUMBER OF EDITS


   4 MN     NUMBER OF MICROSCOPIC ACTIVITIES TO BE COMPUTED

INPUT MICRO XS IDS      4
           1            2           3           4

   2 NZ     NUMBER OF EDIT ZONES
   1 NORMZ  ZONE TO WHICH POWER DENSITY IS NORMALIZED

INPUT EDIT ZONES       16
           1            1           1           2           2           2           1           1           1           1
           1            1           1           1           2           2

   3 NZ     NUMBER OF EDIT ZONES
   0 NORMZ  ZONE TO WHICH POWER DENSITY IS NORMALIZED

INPUT EDIT ZONES       16
           1            2           3           1           2           1           2           3           3           2
           1            2           1           3           2           1

MIXTURE NUMBER MIXTURE COMMAND  MATERIAL ATOMIC DENSITY
          5               0              0.                        1
          5               1              9.9000000E-01             2
          5               3              9.9000000E-01             3
          6               0              0.                        4
          6               2              9.9000000E-01             5
```

```
        /            0             0.                          6                               (6)
        /            4             1.0000000E+00               7
        /            0             0.                          8
        /            1             9.9000000E-01               9
        /            3             9.9000000E-01              10

GROUP NUMBER   1

MIXED X-SECT

           MATERL     5   MATERL     6   MATERL     7

     1   .254478E-03  0.              .254478E-03
     2   .172271E+00  0.              .172271E+00
     3   .474461E-01  0.              .474461E-01
     4   .754187E-01  0.              .754187E-01
     5   .296207E+00  0.              .296207E+00
     6   .184580E+00   .893213E-01    .184580E+00
     7  0.            0.             0.

GROUP NUMBER   2

MIXED X-SECT

           MATERL     5   MATERL     6   MATERL     7

     1  0.            0.             0.
     2   .335619E+00  0.              .335619E+00
     3   .761098E-01  0.              .761098E-01
     4   .114881E+00  0.              .114881E+00
     5   .571743E+00  0.              .571743E+00
     6   .495634E+00   .164760E+00    .495634E+00
     7   .644360E-01   .161594E-01    .644360E-01


GROUP FISSION SPECTRUM    VELOCITIES

  1      7.5382000E-01  1.0000000E+00
  2      2.4618300E-01  1.0000000E+00
  3      1.0000030E+00  0.


EPSO   1.00E-04      XLAX   1.00E-03
EPSI   1.00E-04      XLAH   5.00E-01
EPSR   5.00E-05      XLAL   1.00E-02
EPSX   1.00E-04      POD    1.00E+00
NORM   1.00E+00      BHGT   0.

IITL    9



TIME IN    OUTER       INNER        NEUTRON         EIGENVALUE        EIGENVALUE         LAMBDA         REBALANCE
MINUTES ITERATIONS  ITERATIONS      BALANCE                             SLOPE                          CONVERGENCE
                   TOTAL/BY GROUP

 4.38E-03     0         0           0.              0.                0.               0.              0.
   8.75E-03                      1    7
   1.23E-02                      2   10 1.28E-03
 2.86E-02     1        17         -8.4373053E-07 0.                0.               8.31439311E-01 3.53487771E-01
```

(7)

```
  8.61E-03                        1  7
  1.23E-02                        2  10 4.58E-04
5.36E-02      2       17               -2.55920924E-07 0.              0.              8.59568314E-01 1.48390330E-01
  7.40E-03                        1  6
  1.23E-02                        2  10 1.32E-04
7.75E-02      3       16               -2.57087706E-07 0.              0.              8.70325166E-01 1.39550827E-01
  6.18E-03                        1  5
  1.11E-02                        2  9
9.86E-02      4       14               -8.17839720E-07 0.              0.              8.74050990E-01 1.30979484E-01
  4.97E-03                        1  4
  8.64E-03                        2  7
1.16E-01      5       11               -4.97837334E-07 0.              0.              8.75727371E-01 1.26889389E-01
  4.96E-03                        1  4
  8.70E-03                        2  7
1.38E-01      6       11               -6.50675730E-07 0.              0.              8.76540594E-01 1.24810663E-01
  3.79E-03                        1  3
  6.24E-03                        2  5
1.65E-01      7        8               -3.51055292E-07 0.              0.              8.76942096E-01 1.23747867E-01
  3.77E-03                        1  3
  3.76E-03                        2  3
1.82E-01      8        6                3.67590296E-07 0.              0.              8.77138567E-01 1.23215061E-01
  3.77E-03                        1  3
  2.54E-03                        2  2
1.93E-01      9        5               -6.49568861E-08 0.              0.              8.77238135E-01 1.22943782E-01
  2.55E-03                        1  2
  2.54E-03                        2  2
2.04E-01     10        4               -7.44427155E-07 1.00000000E-01 0.              8.77289834E-01 1.22811157E-01
  6.18E-03                        1  5
  9.86E-03                        2  8
2.22E-01     11       13                3.04832092E-06 1.00000000E-01 0.              9.20094769E-01 8.18013200E-02
  4.98E-03                        1  4
  6.20E-03                        2  5
2.37E-01     12        9                1.20308755E-06 1.00000000E-01 0.              9.21449377E-01 7.93644573E-02
  3.76E-03                        1  3
  6.20E-03                        2  5
2.53E-01     13        8                9.20160275E-07 2.75012932E-01 2.24115851E+00 9.21909614E-01 7.84586618E-02
  6.19E-03                        1  5
  1.11E-02                        2  9
2.72E-01     14       14                3.19720752E-06 2.75012932E-01 2.24115851E+00 9.85725693E-01 1.75741772E-02
  4.97E-03                        1  4
  8.65E-03                        2  7
2.90E-01     15       11                1.00063659E-06 2.75012932E-01 2.24115851E+00 9.87886464E-01 1.36637125E-02
  4.97E-03                        1  4
  7.43E-03                        2  6
3.09E-01     16       10                2.46734729E-07 3.09162051E-01 2.93982388E+00 9.88658745E-01 1.21018475E-02
  6.18E-03                        1  5
  8.65E-03                        2  7
3.25E-01     17       12                1.75278399E-06 3.09162051E-01 2.93982388E+00 1.00011483E+00 1.15718105E-03
  3.76E-03                        1  3
  6.19E-03                        2  5
3.42E-01     18        8                1.18265880E-06 3.07375016E-01 2.93982388E+00 1.00060787E+00 1.14425674E-03
  3.77E-03                        1  3
  3.76E-03                        2  3
3.53E-01     19        6                1.20253638E-07 3.06504086E-01 2.93982388E+00 1.00029625E+00 3.81731518E-04
  2.53E-03                        1  2
  3.77E-03                        2  3
3.61E-01     20        5               -1.03613625E-06 3.06504086E-01 2.93982388E+00 1.00007697E+00 1.07849879E-04
```

```
COARSE MESH BALANCE TABLE FOR GROUP    1                                                                                   (8)


ZONE  SOURCE       LEFT        RIGHT       BOTTOM       TOP        ABSORPTION   LEFT        RIGHT       BOTTOM       TOP
                   INFLOW      INFLOW      INFLOW      INFLOW                  OUTFLOW     OUTFLOW     OUTFLOW     OUTFLOW

    1 3.0267E-01 0.          6.5315E-02 7.7891E-02 3.6033E-02 2.1905E-01 0.          1.1745E-01 7.7891E-02 6.7520E-02
    2 2.5964E-01 1.1745E-01 0.          6.8890E-02 3.3533E-02 1.9321E-01 6.5315E-02 9.1048E-02 6.8890E-02 6.1049E-02
    3 1.0108E-01 0.          2.3979E-02 6.7520E-02 0.          7.5712E-02 0.          4.1056E-02 3.6033E-02 3.9782E-02
    4 9.0420E-02 4.1056E-02 0.          6.1049E-02 0.          6.9365E-02 2.3979E-02 3.3118E-02 3.3533E-02 3.2530E-02


COARSE MESH BALANCE TABLE FOR GROUP    2


ZONE  SOURCE       LEFT        RIGHT       BOTTOM       TOP        ABSORPTION   LEFT        RIGHT       BOTTOM       TOP
                   INFLOW      INFLOW      INFLOW      INFLOW                  OUTFLOW     OUTFLOW     OUTFLOW     OUTFLOW

    1 2.2529E-01 0.          8.3229E-02 8.8128E-02 4.6182E-02 1.6799E-01 0.          1.1880E-01 8.8128E-02 6.7903E-02
    2 1.9632E-01 1.1880E-01 0.          7.5893E-02 4.1194E-02 1.4172E-01 8.3229E-02 7.2048E-02 7.5893E-02 5.9335E-02
    3 7.6717E-02 0.          2.9336E-02 6.7903E-02 0.          5.4960E-02 0.          3.9902E-02 4.6182E-02 3.2909E-02
    4 6.9570E-02 3.9902E-02 0.          5.9335E-02 0.          4.8422E-02 2.9336E-02 2.5371E-02 4.1194E-02 2.4485E-02


COARSE MESH BALANCE TABLE FOR ALL GROUPS


ZONE  SOURCE       LEFT        RIGHT       BOTTOM       TOP        ABSORPTION   LEFT        RIGHT       BOTTOM       TOP
                   INFLOW      INFLOW      INFLOW      INFLOW                  OUTFLOW     OUTFLOW     OUTFLOW     OUTFLOW

    1 4.0164E-01 0.          1.4853E-01 1.6600E-01 8.2208E-02 2.6058E-01 0.          2.3623E-01 1.6600E-01 1.3541E-01
    2 3.4442E-01 2.3623E-01 0.          1.4477E-01 7.4721E-02 2.2338E-01 1.4853E-01 1.6308E-01 1.4477E-01 1.2037E-01
    3 1.3410E-01 0.          5.3311E-02 1.3541E-01 0.          8.6962E-02 0.          8.0951E-02 8.2208E-02 7.2685E-02
    4 1.1994E-01 8.0951E-02 0.          1.2037E-01 0.          7.7739E-02 5.3311E-02 5.8484E-02 7.4721E-02 5.7011E-02
*******************************************************************************************************************************

*FINAL*DUMP TAKEN ON UNIT        7

*******************************************************************************************************************************



FINAL PRINTING
```

```
                                                                                                              (9)

                 SOURCE       FISSION SOURCE          IN SCATTER         SELF SCATTER          OUT SCATTER         NET LEAKAGE
  1    0.                       7.5381774E-01       7.1054274E-14       9.2157971E-01        3.2044820E-01       1.9647810E-01
  2    0.                       2.4618226E-01       3.2172001E-01       2.6901061E+00       -4.2906674E-06       1.5481336E-01
  3    0.                       1.0000000E+00       3.2172001E-01       3.6116858E+00        3.2171447E-01       3.5129140E-01


            ABSORPTION       NEUTRON BALANCE        RIGHT LEAKAGE    HORIZONTAL LEAKAGE       TOP LEAKAGE         VERTICAL LEAKAGE
  1    2.3689191E-01          -6.2313688E-07       1.2416558E-01        1.2416558E-01        7.2312520E-02       7.2312520E-02
  2    4.1309442E-01          -2.1249935E-06       9.7419234E-02        9.7419234E-02        5.7394131E-02       5.7394131E-02
  3    6.4998633E-01          -1.2684383E-06       2.2158481E-01        2.2158481E-01        1.2970665E-01       1.2970665E-01


TIME IN     OUTER        INNER        NEUTRON         EIGENVALUE       EIGENVALUE        LAMBDA         REBALANCE
MINUTES ITERATIONS  ITERATIONS      BALANCE                           SLOPE                          CONVERGENCE
                   TOTAL/BY GROUP

 3.78E-01     21         4           -1.26843830E-06 3.06504086E-01 2.93982388E+00 1.00008499E+00 1.07849879E-04



 DELTA CALCULATION FINAL RESULTS

      I          MODIFIED R       ORIGINAL R


      1       0.               0.
      2        .42722684E+01    .32700000E+01
      3        .85445367E+01    .65400000E+01
      4        .11392716E+02    .87200000E+01
      5        .14240895E+02    .10900000E+02

      J          MODIFIED Z       ORIGINAL Z


      1       0.               0.
      2        .32700000E+01    .32700000E+01
      3        .70411342E+01    .65400000E+01
      4        .92211342E+01    .87200000E+01
      5        .11735224E+02    .10900000E+02



FLUX COMPONENTS FOR GROUP     1

SECOND COMPONENT IS R DIRECTION CURRENT, THIRD IS Z DIRECTION CURRENT


COMPONENT NUMBER  1
```

```
                                                                                                              (10)
          R MESH      1   R MESH      2   R MESH      3   R MESH      4   R MESH      5   R MESH      6   R MESH      7   R MESH      8

    10    .480101E-03     .468084E-03     .448607E-03     .420580E-03     .387254E-03     .347658E-03     .295371E-03     .237444E-03
     9    .701594E-03     .684644E-03     .656183E-03     .616136E-03     .566170E-03     .513295E-03     .450297E-03     .369848E-03
     8    .877149E-03     .855807E-03     .820070E-03     .769203E-03     .705295E-03     .636419E-03     .556962E-03     .461444E-03
     7    .102753E-02     .100219E-02     .958695E-03     .896443E-03     .817381E-03     .728525E-03     .632822E-03     .525764E-03
     6    .120255E-02     .117165E-02     .111945E-02     .104360E-02     .946482E-03     .831021E-03     .712875E-03     .591809E-03
     5    .138341E-02     .134866E-02     .128848E-02     .120063E-02     .108642E-02     .943884E-03     .799745E-03     .659856E-03
     4    .151889E-02     .148182E-02     .141534E-02     .131963E-02     .119590E-02     .104110E-02     .882135E-03     .726900E-03
     3    .159906E-02     .155861E-02     .148802E-02     .138836E-02     .126087E-02     .110797E-02     .944999E-03     .782975E-03
     2    .163296E-02     .159134E-02     .152021E-02     .141963E-02     .129206E-02     .114417E-02     .984992E-03     .822096E-03
     1    .164755E-02     .160622E-02     .153531E-02     .143626E-02     .131031E-02     .116313E-02     .100415E-02     .840173E-03

          R MESH      9   R MESH     10

    10    .179760E-03     .112963E-03
     9    .280832E-03     .176485E-03
     8    .352822E-03     .226108E-03
     7    .404083E-03     .261829E-03
     6    .455326E-03     .295358E-03
     5    .506065E-03     .327319E-03
     4    .553875E-03     .350577E-03
     3    .596831E-03     .366151E-03
     2    .627265E-03     .379067E-03
     1    .641118E-03     .387441E-03

COMPONENT NUMBER  2


          R MESH      1   R MESH      2   R MESH      3   R MESH      4   R MESH      5   R MESH      6   R MESH      7   R MESH      8

    10    .465584E-05     .137028E-04     .223001E-04     .300791E-04     .362957E-04     .410399E-04     .430999E-04     .437270E-04
     9    .677893E-05     .200542E-04     .324792E-04     .434500E-04     .521668E-04     .574819E-04     .620765E-04     .671708E-04
     8    .863379E-05     .257174E-04     .416771E-04     .559089E-04     .669827E-04     .752145E-04     .846827E-04     .939407E-04
     7    .969855E-05     .291502E-04     .475919E-04     .642931E-04     .785513E-04     .897775E-04     .100991E-03     .112169E-03
     6    .104270E-04     .309914E-04     .506367E-04     .692485E-04     .859134E-04     .100703E-03     .115455E-03     .127247E-03
     5    .112860E-04     .333301E-04     .545985E-04     .747684E-04     .941129E-04     .113009E-03     .130093E-03     .14335[illegible]E-03
     4    .132995E-04     .392211E-04     .640955E-04     .874337E-04     .109007E-03     .129167E-03     .144719E-03     .15452[illegible]E-03
     3    .156819E-04     .466571E-04     .763153E-04     .103127E-03     .126513E-03     .144673E-03     .155146E-03     .159256E-03
     2    .169069E-04     .501055E-04     .817218E-04     .110467E-03     .134234E-03     .150643E-03     .158965E-03     .1610[illegible]E-03
     1    .166256E-04     .500129E-04     .816772E-04     .109673E-03     .132991E-03     .149933E-03     .159817E-03     .163[illegible]E-03

          R MESH      9   R MESH     10

    10    .449116E-04     .448307E-04
     9    .701569E-04     .706976E-04
     8    .985207E-04     .972015E-04
     7    .117819E-03     .115448E-03
     6    .132348E-03     .129995E-03
     5    .149527E-03     .145729E-03
     4    .159277E-03     .154128E-03
     3    .158217E-03     .152452E-03
     2    .160211E-03     .155037E-03
     1    .164411E-03     .159221E-03

COMPONENT NUMBER  3
```

```
       R MESH     1   R MESH     2   R MESH     3   R MESH     4   R MESH     5   R MESH     6   R MESH     7   R MESH     8

 10   .209523E-03    .203889E-03    .195242E-03    .182015E-03    .167572E-03    .147700E-03    .118616E-03    .949916E-04
  9   .204499E-03    .198704E-03    .190125E-03    .176601E-03    .161349E-03    .140446E-03    .112493E-03    .915840E-04
  8   .194656E-03    .189040E-03    .180329E-03    .166956E-03    .150504E-03    .129194E-03    .106257E-03    .871436E-04
  7   .181459E-03    .176561E-03    .167711E-03    .155201E-03    .138015E-03    .118046E-03    .100776E-03    .837507E-04
  6   .160710E-03    .156658E-03    .148407E-03    .137954E-03    .121841E-03    .105211E-03    .941877E-04    .789519E-04
  5   .130512E-03    .126971E-03    .120578E-03    .113039E-03    .100966E-03    .906411E-04    .857674E-04    .724160E-04
  4   .956602E-04    .924164E-04    .885311E-04    .838448E-04    .775046E-04    .743417E-04    .730213E-04    .633781E-04
  3   .645203E-04    .617987E-04    .598691E-04    .571110E-04    .549211E-04    .555775E-04    .551649E-04    .489595E-04
  2   .387053E-04    .365860E-04    .359832E-04    .343208E-04    .337260E-04    .341328E-04    .338025E-04    .299806E-04
  1   .131850E-04    .122198E-04    .122343E-04    .116784E-04    .114398E-04    .113696E-04    .111735E-04    .981249E-05

       R MESH     9   R MESH    10

 10   .724464E-04    .457861E-04
  9   .708286E-04    .463380E-04
  8   .675234E-04    .452543E-04
  7   .636794E-04    .421893E-04
  6   .585962E-04    .381326E-04
  5   .522448E-04    .324599E-04
  4   .446539E-04    .242995E-04
  3   .335596E-04    .167944E-04
  2   .199705E-04    .105953E-04
  1   .604478E-05    .359030E-05


FLUX COMPONENTS FOR GROUP    2

SECOND COMPONENT IS R DIRECTION CURRENT, THIRD IS Z DIRECTION CURRENT


COMPONENT NUMBER  1


       R MESH     1   R MESH     2   R MESH     3   R MESH     4   R MESH     5   R MESH     6   R MESH     7   R MESH     8

 10   .466562E-03    .455886E-03    .437471E-03    .410412E-03    .377897E-03    .338360E-03    .284128E-03    .223994E-03
  9   .744618E-03    .728012E-03    .697758E-03    .655101E-03    .601499E-03    .545475E-03    .478553E-03    .388178E-03
  8   .950829E-03    .929203E-03    .890110E-03    .834126E-03    .763811E-03    .688631E-03    .601928E-03    .492941E-03
  7   .112909E-02    .110308E-02    .105524E-02    .986350E-03    .898247E-03    .797960E-03    .690207E-03    .565912E-03
  6   .133845E-02    .130665E-02    .124875E-02    .116374E-02    .105346E-02    .919685E-03    .783564E-03    .641650E-03
  5   .155773E-02    .152077E-02    .145254E-02    .135209E-02    .122023E-02    .105243E-02    .884008E-03    .719223E-03
  4   .171739E-02    .167718E-02    .160167E-02    .149127E-02    .134785E-02    .116595E-02    .979952E-03    .797043E-03
  3   .180651E-02    .176367E-02    .168437E-02    .156986E-02    .142233E-02    .124423E-02    .105402E-02    .864048E-03
  2   .184344E-02    .179970E-02    .171967E-02    .160417E-02    .145691E-02    .128561E-02    .110079E-02    .910710E-03
  1   .186154E-02    .181779E-02    .173748E-02    .162268E-02    .147681E-02    .130668E-02    .112272E-02    .931682E-03

       R MESH     9   R MESH    10

 10   .165319E-03    .927713E-04
  9   .287649E-03    .164118E-03
  8   .368125E-03    .216120E-03
  7   .424741E-03    .252874E-03
  6   .481918E-03    .287158E-03
  5   .538222E-03    .320129E-03
  4   .592544E-03    .343209E-03
```

(12)

```
  3   .644111E-03   .357964E-03
  2   .690472E-03   .371252E-03
  1   .696164E-03   .379909E-03

COMPONENT NUMBER  2


      R MESH     1   R MESH     2   R MESH     3   R MESH     4   R MESH     5   R MESH     6   R MESH     7   R MESH     8

 10   .274806E-05   .801060E-05   .128370E-04   .170860E-04   .204710E-04   .233047E-04   .237978E-04   .235585E-04
  9   .445343E-05   .131167E-04   .210768E-04   .278805E-04   .328889E-04   .348929E-04   .375944E-04   .425447E-04
  8   .586105E-05   .173239E-04   .277959E-04   .371815E-04   .437699E-04   .480438E-04   .554281E-04   .648036E-04
  7   .638072E-05   .190469E-04   .309179E-04   .416073E-04   .506500E-04   .580541E-04   .673210E-04   .789833E-04
  6   .648488E-05   .191953E-04   .312321E-04   .428670E-04   .536919E-04   .645000E-04   .776889E-04   .901029E-04
  5   .693690E-05   .204022E-04   .334140E-04   .459249E-04   .588549E-04   .734510E-04   .887599E-04   .102825E-03
  4   .852030E-05   .250380E-04   .408906E-04   .560692E-04   .707839E-04   .863927E-04   .100071E-03   .110363E-03
  3   .107212E-04   .317759E-04   .518779E-04   .702631E-04   .871361E-04   .100721E-03   .108323E-03   .111555E-03
  2   .119223E-04   .350799E-04   .569288E-04   .770590E-04   .939555E-04   .105378E-03   .110584E-03   .111353E-03
  1   .118270E-04   .351986E-04   .570578E-04   .764880E-04   .925699E-04   .104079E-03   .110522E-03   .113396E-03

      R MESH     9   R MESH    10

 10   .253767E-04   .276154E-04
  9   .465639E-04   .505979E-04
  8   .713468E-04   .745604E-04
  7   .872862E-04   .901578E-04
  6   .977694E-04   .101339E-03
  5   .112231E-03   .115038E-03
  4   .118525E-03   .120513E-03
  3   .112003E-03   .113573E-03
  2   .112166E-03   .114597E-03
  1   .115826E-03   .118299E-03

COMPONENT NUMBER  3


      R MESH     1   R MESH     2   R MESH     3   R MESH     4   R MESH     5   R MESH     6   R MESH     7   R MESH     8

 10   .166800E-03   .162682E-03   .156259E-03   .145741E-03   .134844E-03   .117904E-03   .894569E-04   .696644E-04
  9   .154298E-03   .150306E-03   .144101E-03   .133768E-03   .122783E-03   .105491E-03   .792073E-04   .628774E-04
  8   .142052E-03   .138291E-03   .131980E-03   .122005E-03   .109592E-03   .921377E-04   .727234E-04   .582702E-04
  7   .128886E-03   .125626E-03   .119236E-03   .110097E-03   .966656E-04   .808561E-04   .682199E-04   .562389E-04
  6   .110605E-03   .108058E-03   .102194E-03   .948107E-04   .823960E-04   .699781E-04   .641461E-04   .538740E-04
  5   .805260E-04   .845389E-04   .800832E-04   .750345E-04   .662335E-04   .596421E-04   .599037E-04   .509651E-04
  4   .609356E-04   .592670E-04   .566581E-04   .536938E-04   .496939E-04   .496324E-04   .525596E-04   .466933E-04
  3   .396042E-04   .383507E-04   .371735E-04   .356247E-04   .350531E-04   .379411E-04   .404605E-04   .370096E-04
  2   .232698E-04   .223637E-04   .221164E-04   .213394E-04   .219847E-04   .235081E-04   .248478E-04   .224594E-04
  1   .706794E-05   .743286E-05   .751479E-05   .732622E-05   .757518E-05   .784120E-05   .819432E-05   .719228E-05

      R MESH     9   R MESH    10

 10   .524438E-04   .293526E-04
  9   .483164E-04   .285949E-04
  8   .448988E-04   .277787E-04
  7   .421400E-04   .255508E-04
  6   .308701E-04   .230103E-04
  5   .352239E-04   .195221E-04
  4   .314556E-04   .137165E-04
  3   .239169E-04   .908377E-05
```

```
     2   .137046E-04   .609102E-05
     1   .452284E-05   .208391E-05


FISSION EDIT


COMPONENT NUMBER  1

         R MESH     1   R MESH     2   R MESH     3   R MESH     4   R MESH     5   R MESH     6   H MESH     7   R MESH     8

    10   .898075E-04   .876747E-04   .840903E-04   .788680E-04   .726192E-04   .650909E-04   .549173E-04   .436403E-04
     9   .138455E-03   .135269E-03   .129647E-03   .121727E-03   .111800E-03   .101377E-03   .889372E-04   .724875E-04
     8   .175385E-03   .171291E-03   .164105E-03   .153837E-03   .140940E-03   .127108E-03   .111155E-03   .914308E-04
     7   .207206E-03   .202306E-03   .193530E-03   .180921E-03   .164837E-03   .146615E-03   .127018E-03   .104665E-03
     6   .244457E-03   .238473E-03   .227885E-03   .212398E-03   .192405E-03   .168328E-03   .143780E-03   .118347E-03
     5   .283288E-03   .276421E-03   .264044E-03   .245879E-03   .222117E-03   .192090E-03   .161871E-03   .132390E-03
     4   .311848E-03   .304433E-03   .290744E-03   .270842E-03   .245035E-03   .212464E-03   .179107E-03   .146387E-03
     3   .328132E-03   .320159E-03   .3-5726E-03   .285054E-03   .258492E-03   .226499E-03   .192357E-03   .158313E-03
     2   .334932E-03   .326768E-03   .312209E-03   .291356E-03   .264817E-03   .233983E-03   .200746E-03   .166624E-03
     1   .338110E-03   .329967E-03   .315394E-03   .294735E-03   .268478E-03   .237834E-03   .204710E-03   .170397E-03
         R MESH     9   R MESH    10

    10   .325492E-04   .191772E-04
     9   .542253E-04   .321642E-04
     8   .688998E-04   .418808E-04
     7   .792699E-04   .487971E-04
     6   .897031E-04   .552643E-04
     5   .999981E-04   .614626E-04
     4   .109844E-03   .658681E-04
     3   .119008E-03   .687378E-04
     2   .125481E-03   .712384E-04
     1   .128328E-03   .728644E-04
```

```
        Z                                        EDIT NUMBER    1 EDIT SPECIFICATIONS BY BROAD ZONE. ORIGIN AT LOWER LEFT.
  ROW    M                                       M IS NUMBER OF FINE INTERVALS/BROAD ROW(COLUMN).
  11.7352   1000000000000000000000000
            1     *     *     *     0
     4   2  1   1 *   1 *   2 *   2 0
            1     *     *     *     0
   9.2211   1************************0
            1     *     *     *     0
     3   2  1   1 *   1 *   1 *   1 0
            1     *     *     *     0
   7.0411   1************************0
            1     *     *     *     0
     2   3  1   2 *   2 *   1 *   1 0
            1     *     *     *     0
   3.2700   1************************0
            1     *     *     *     0
     1   3  1   1 *   1 *   1 *   2 0
            1     *     *     *     0
   0.0000   1111111111111111111111111
       R    0.    4.    8.    11.   14.
             0  2722  5445  3927  2408

      M          3     3     2     2
  COLUMN         1     2     3     4

EDIT NUMBER    1 GROUP NUMBER    1

EDIT ZONE  RIGHT LEAKAGE    LEFT LEAKAGE     TOP LEAKAGE    BOTTOM LEAKAGE     NET LEAKAGE        SOURCE              FLUX

    1        1.1184246E-01  -2.4904060E-02   8.6077031E-02  -3.8241308E-02   1.3477412E-01   0.                  3.4026746E+00
    2        8.2039085E-02  -4.4811903E-02   7.0771535E-02  -4.6294738E-02   6.1703978E-02   0.                  1.5897601E+00


MACROSCOPIC ACTIVITY EDIT

ACT.FOR XS 1--IHT        1

          ZONE       1    ZONE       2

    1   .805904E-03    .404558E-03
    2   .506182E+00    .273869E+00
    3   .101444E+00    .754280E-01
    4   .206625E+00    .119898E+00
    5   .100790E+01    .470898E+00


MICROSCOPIC ACTIVITY EDIT FOR MATERIAL      1 GROUP    1

ACT.FOR XS 1--IHT        1

          ZONE       1    ZONE       2

    1   .805904E-03    .404558E-03
    2   .506182E+00    .273869E+00
   3   .100650E+00   .470245E-01
   4   .256625E+00   .119898E+00
   5   .947102E+00   .442495E+00


MICROSCOPIC ACTIVITY EDIT FOR MATERIAL     2 GROUP    1

ACT.FOR XS 1--IHT

ALL ENTRIES OF THIS ARRAY EQUAL  0.


MICROSCOPIC ACTIVITY EDIT FOR MATERIAL     3 GROUP    1

ACT.FOR XS 1--IHT       1

          ZONE      1     ZONE      2

   1  0.              0.
   2  0.              0.
   3   .607940E-01     .284035E-01
   4  0.              0.
   5   .607940E-01     .284035E-01


MICROSCOPIC ACTIVITY EDIT FOR MATERIAL     4 GROUP    1

ACT.FOR XS 1--IHT       1

          ZONE      1     ZONE      2

   1  0.              0.
   2  0.              0.
   3  0.               .158976E+00
   4  0.              0.
   5  0.               .158976E+01


EDIT NUMBER    1 GROUP NUMBER   2

EDIT ZONE  RIGHT LEAKAGE   LEFT LEAKAGE     TOP LEAKAGE   BOTTOM LEAKAGE    NET LEAKAGE        SOURCE           FLUX

   1        8.4411574E-02  -1.6606927E-02  6.3380690E-02  -2.6141895E-02  1.0504344E-01   0.                3.7[illegible]8793E+00
   2        6.0000776E-02  -3.0386189E-02  5.0627262E-02  -3.0471926E-02  4.9769923E-02   0.                1.[illegible]192698E+00
```

(16)

```
      Z                                            EDIT NUMBER   2 EDIT SPECIFICATIONS BY BROAD ZONE. ORIGIN AT LOWER LEFT.
  ROW   M                                          A IS NUMBER OF FINE INTERVALS/BROAD ROW(COLUMN).
 11.7352  1000000000000000000000000
          1     *     *     *     0
    4   2 1   1 *   3 *   2 *   1 0
          1     *     *     *     0
  9.2211  1************************0
          1     *     *     *     0
    3   2 1   3 *   2 *   1 *   2 0
          1     *     *     *     0
  7.0411  1************************0
          1     *     *     *     0
    2   3 1   2 *   1 *   2 *   3 0
          1     *     *     *     0
  3.2700  1************************0
          1     *     *     *     0
    1   3 1   1 *   2 *   3 *   1 0
          1     *     *     *     0
  0.0000  1111111111111111111111111
     R    0.    4.    8.    11.   14.
          0   2722  5445  3927  2408

     M      3     3     2     2
  COLUMN    1     2     3     4

EDIT NUMBER   2 GROUP NUMBER   1


EDIT ZONE  RIGHT LEAKAGE   LEFT LEAKAGE     TOP LEAKAGE   BOTTOM LEAKAGE    NET LEAKAGE       SOURCE          FLUX

    1      1.0930403E-01  -6.5119921E-02  7.5372930E-02  -5.1414391E-02  6.8142652E-02  0.             1.7394286E+00
    2      1.1414072E-01  -5.9966461E-02  9.5225050E-02  -6.6227026E-02  8.3172285E-02  0.             2.0625890E+00
    3      9.4704625E-02  -6.8897423E-02  6.1641005E-02  -4.2285048E-02  4.5163158E-02  0.             1.1904171E+00



MACROSCOPIC ACTIVITY EDIT

ACT.FOR XS 1--IHT       1

         ZONE      1     ZONE      2     ZONE      3

    1   .442645E-03   .524883E-03   .302934E-03
    2   .299653E+00   .355324E+00   .205074E+00
    3   .845292E-01   .978619E-01   .564807E-01
    4   .131185E+00   .155558E+00   .897797E-01
    5   .515231E+00   .610953E+00   .352610E+00



MICROSCOPIC ACTIVITY EDIT FOR MATERIAL      1 GROUP   1

ACT.FOR XS 1--IHT       1

         ZONE      1     ZONE      2     ZONE      3

    1   .442645E-03   .524883E-03   .302934E-03
```

IV. PROGRAMMING INFORMATION

In this section we give some of the details of the TWOTRAN-II program. The material contained in this section is designed to help in the future local modification of the program. Much supplementary information is provided by the program comment cards.

A. Program Structure

1. Role and Function of Subprograms

We describe in Table V (p. 53) the function of all the subprograms in TWOTRAN-II. In Table V, we follow the overlay structure defined in Table IV (p. 52). The overlay is of the simple control, input, execute and output kind. Note that on IBM machines with large central memories it is more efficient to remove the substructure of OVERLAY(2,0).

2. Program Flow

In Fig. 11 we show an amplified version of Fig. 8, depicting the overall flow of the TWOTRAN-II program and showing subroutine names.

3. Relation of Problem Variables and Program Mnemonics

In much of the material in this manual we have used variables actually appearing in the FORTRAN of the program. To supplement this information we define the geometric functions generated by the program in Table VI (p. 52) and give a list of the relation between problem variable symbols and program variable names in Table VII (p. 55).

4. Definition of Variables in Common Blocks

Tables VIII through XIII (pp. 56-65) define the variables stored in blank common block IA and the named common blocks of TWOTRAN-II. The container array, A, for problem data is also in blank common. Block IA contains problem input parameters, first word addresses of data stored in the A array, and data generated by the program. Block FWBGN1 contains information necessary to a problem restart.

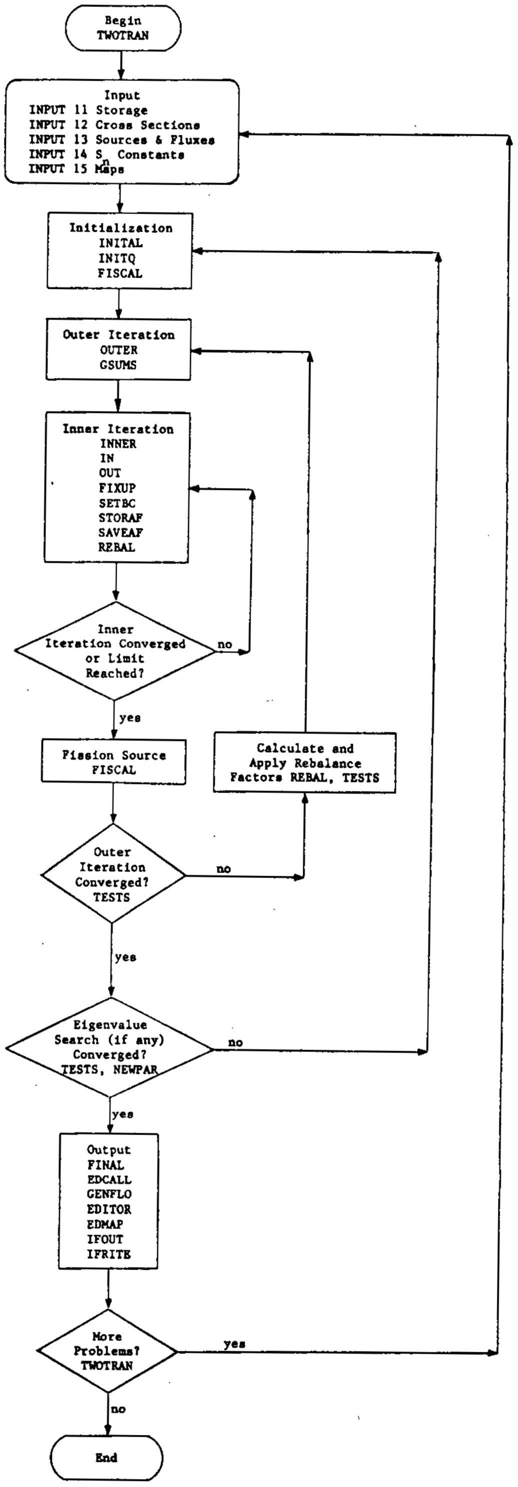


Fig. 11. Amplified form of flow diagram in Fig. 8 showing subroutine names in overall program logic.

TABLE IV

STRUCTURE OF THE TWOTRAN-II PROGRAM[a]

Overlay (0,0)	Overlay (1,0)	Overlay (2,0)	Overlay (3,0)
TWOTRAN	INPUT1	GRIND2	OUTPUT3
1. MONITR	1. LOAD	1. REBAL	1. OUTPT31
2. ERROR	2. INPUT11	2. GRIND21	a. FINAL
3. CLEAR	a. DUMPRD	a. INITAL	2. OUTPT32
4. MPLY	3. INPUT12	b. INITQ	a. EDCALL
5. WRITE	a. CSPREP	c. FISCAL	b. GENFLO
6. ECHECK	b. IFINXS	3. GRIND22	c. EDITOR
7. DUMPER	4. INPUT13	a. OUTER	d. EDMAP
8. PCMBAL	a. READQF	b. INNER	3. IFOUT
9. REED	b. IFINQF	c. IN	a. IFRITE
10. RITE	5. INPUT14	d. OUT	
	a. SNCON	e. FIXUP	
	b. IFINSN	f. SETBC	
	c. PNGEN	g. STORAF	
	6. INPUT15	h. SAVEAF	
	a. CSMESH	i. GSUMS	
	b. MAPPER	4. GRIND23	
		a. TESTS	
		b. NEWPAR	

[a] On IBM 360 machines with large bulk core storage, it is computationally more efficient to remove the substructure of OVERLAY(2,0), locating GRIND21, GRIND22, and GRIND23 in GRIND2. For the IBM version, variable names of seven characters are shortened to six.

TABLE VI

GEOMETRIC FUNCTIONS

Array Name		Geometry	
and Dimension	(x,y)	(r,θ)	(r,z)
XH(IM)	Δx_i	Δr_i	Δr_i
YH(JT)	Δy_j	$\Delta \theta_j$	Δz_j
A3(IT)	Δx_i	Δr_i	$\pi\left(r^2_{i+\frac{1}{2}} - r^2_{i-\frac{1}{2}}\right)$
A4(IT+1)	1.0	$2\pi r_{i-\frac{1}{2}}$	$2\pi r_{i-\frac{1}{2}}$
A5(IT)	Δx_i	$\pi\left(r^2_{i+\frac{1}{2}} - r^2_{i-\frac{1}{2}}\right)$	$\pi\left(r^2_{i+\frac{1}{2}} - r^2_{i-\frac{1}{2}}\right)$
A1(IT)	2.0	$2\pi(r_{i+\frac{1}{2}} + r_{i-\frac{1}{2}})$	$2\pi(r_{i+\frac{1}{2}} + r_{i-\frac{1}{2}})$
A2(IT)	0.0	$2\pi(r_{i+\frac{1}{2}} - r_{i-\frac{1}{2}})$	$2\pi(r_{i+\frac{1}{2}} - r_{i-\frac{1}{2}})$
B1(JT)	$1/\Delta y_j$	$1/\Delta\theta_j$	$1/\Delta z_j$

Volume, V_{ij} = A5(I)*YH(J)

Area, $A_{i+\frac{1}{2},j}$ = A4(I+1)*YH(J)

Area, B_i = A3(I)

TABLE V

FUNCTION OF TWOTRAN-II SUBROUTINES

(Overlay) Subroutine	Function
(0,0) TWOTRAN	Main driver of program. Initializes program parameters; calls input, calculation and output overlays; and attempts to execute sequential problems.
1. MONITR	Prints resume of convergence parameters, monitor line headings, and outer iteration monitor data.
2. ERROR	Prints messages, including error notices.
3. CLEAR	Stores a constant in an array.
4. MPLY	Multiplies an array by a constant.
5. WRITE	Generalized output routine for printing 1D, 2D or 3D arrays, either integer or floating point.
6. ECHECK	Checks equality of elements in a vector. Used to reduce printing in WRITE.
7. DUMPER	Writes a restart dump.
8. PCMBAL	Prints the coarse-mesh balance tables.
9. REED	Handles all binary reading operations including rewind and bulk memory transfers (ECS or LCM).
10. RITE	Handles all binary writing operations including end of file and rewind and bulk memory transfers.
(1,0) INPUT1	Calls five sections of this overlay.
1. LOAD	Los Alamos data loader.
(1,1) 2. INPUT11	Reads basic parameter data, checks it, calculates storage locations in the container array and in bulk storage, prints basic parameter data. Calls DUMPRD.
a. DUMPRD	Reads restart dump.
(1,2) 3. INPUT12	Clears cross-section, flux, and flow bulk storage. Calls CSPREP.
a. CSPREP	Reads cross sections in standard LASL format, FIDO format or from interface file by calling IFINXS. Prints cross sections, performs adjoint transpositions and reversals, checks cross sections and calculates effective absorptions.
b. IFINXS	Interface Input of cross sections from standard file ISOTXS.
(1,3) 4. INPUT13	Calls READQF to read source or flux guess.
a. READQF	Reads source or flux guess from cards or calls IFINQF to read source or flux guess from interface files. Prints either source or flux if required.
b. IFINQF	Reads source or flux from standard interface file.
(1,4) 5. INPUT14	Reads S_n constants and generates spherical harmonic functions.
a. SNCON	Provides builtin library of S_n constants, S_2 through S_{16}.
b. IFINSN	Interface Input of S_n constants from standard file SNCONS.
c. PNGEN	Generates polynomials $R_n^k\ (\mu,\phi)$.
(1,5) 6. INPUT15	Reads data specified by input parameters, including edit information. Calls MAPPER and, if required, CSMESH.
a. CSMESH	In the event the material mesh is different from the rebalance mesh, CSMESH checks input, generates mesh and zone label indices.
b. MAPPER	Draws system picture.
(2,0) GRIND2	Calls overlay (2,1), (2,2) and (2,3). Controls outer iteration process.
1. REBAL	Calculates rebalance factors, either for inner or outer iterations for either coarse mesh or whole system. Called by INNER and TESTS.

(2,1) 2. GRIND21 Calls INITAL and FISCAL.

a. INITAL Sets initialization trigger for FISCAL, generates special angular arrays, performs adjoint reversals for fission spectrum and velocities, mixes cross sections, modifies boundary locations for delta searches, generates rebalance mesh indices and geometric functions, initializes fission spectrum and source if any, generates total cross section array and fission rate array.

b. INITQ Generates source rebalance information; normalises sources and related quantities if required. Prints normalized sources if desired.

c. FISCAL Calculates volume integral of fission source and related rebalance information, and normalizes it if required. When called after INITAL does not calculate λ; when called after OUTER, calculates λ.

(2,2) 3. GRIND22 Calls OUTER.

a. OUTER Calculates source to group, calls INNER, generates coarse-mesh balance (over-all group) data and group sum information, calls GSUMS, calculates new fission rate.

b. INNER Calculates total source for group, performs space-angle mesh sweeps by calling SETBC, IN and OUT, saves and stores angular fluxes, if required, by calling SAVEAF and STORAF, applies rebalance factors calculated by REBAL and calculates inner iteration error and controls convergence.

c. IN Makes inward space-angle traverse for all I and M on level J. Calculates partial flows and calls FIXUP.

d. OUT Makes outward space-angle traverse for all I and M on level J. Calculates partial flows and calls FIXUP.

e. FIXUP Prevents negative fluxes unless sources are negative.

f. SETBC Sets boundary conditions and computes partial flows on boundaries.

g. STORAF Removes boundary source, if any, from angular flux and stores angular flux in bulk memory.

h. SAVEAF Removes boundary source, if any, from angular flux and stores flux by I space point for all M directions on each J level.

i. GSUMS Computes outscatter, leakages, and neutron balance for group balance tables.

(2,3) 4. GRIND23 Calls TESTS.

a. TESTS Calculates rebalance factors for outer iteration acceleration, checks for outer convergence, applies rebalance factors, and, if a search problem, calls NEWPAR.

b. NEWPAR Computes new parameters for implicit eigenvalue search.

(3,0) OUTPUT3 Controls prints, interface file output, and edits. Takes final dump.

(3,1) 1. OUTPT31 Calls FINAL.

a. FINAL Prints final output.

(3,2) 2. OUTPT32 Calls EDCALL.

a. EDCALL Rearranges storage for edit, calls GENFLO (if needed) and EDITOR.

b. GENFLO If the material mesh and rebalance mesh do not coincide, GENFLO computes partial flows from angular fluxes.

c. EDITOR Calls EDMAP and performs edit.

d. EDMAP Draws picture of edit zones.

(3,3) 3. IFOUT Creates interface output files.

a. IFRITE Writes or prints interface file data.

TABLE VII

RELATION OF PROBLEM VARIABLES TO PROGRAM MNEMONICS

Program Mnemonic	Problem Variable	Equation in Text
1. C(ROW,NUCLIDE)	σ_a, $\nu\sigma_f$, σ_t, $\sigma_{snh\rightarrow g}$	Section III.B.1
2. Q(N,I,J)	Q^k_{ngij} (See Note a)	(22)
3. FLUX(N,I,J)	N^k_{nhij} (See Note a)	(17)
4. FISSA(I,J)	FS_{gijm}/χ_g	(21)
5. WGT(M)	w_m	(15)
6. COSMU(M)	μ_m	Table I
7. COSETA(M)	η_m	Table I
8. WMU(M)	$w_m\mu_m$	
9. WETA(M)	$w_m\eta_m$	
10. AL1(M)	$\alpha_{m+\frac{1}{2}}/w_m$	(16)
11. AL2(M)	$\alpha_{m-\frac{1}{2}}/w_m$	(16)
12. P1(N,M)	R^j_{nm} $\mu < 0$, $\eta < 0$	(19)
13. P2(N,M)	R^j_{nm} $\mu > 0$, $\eta < 0$	(19)
14. P3(N,M)	R^j_{nm} $\mu < 0$, $\eta > 0$	(19)
15. P4(N,M)	R^j_{nm} $\mu > 0$, $\eta > 0$	(19)
16. F(K,L)	$f_{k\ell}$	(47)
17. BR1(J,M)	$N_{i+\frac{1}{2},j,m}$ at column i, $\eta < 0$	
18. BR2(J,M)	$N_{i+\frac{1}{2},j,m}$ at column i, $\eta > 0$	
19. BT1(I,M)	$N_{i,j+\frac{1}{2},m}$ at row j, $\mu < 0$	
20. BT2(I,M)	$N_{i,j+\frac{1}{2},m}$ at row j, $\mu > 0$	
21. ALFL(N,I)	$N_{i,j,m+\frac{1}{2}}$ for η level N, row j	
22. CTOT(I,J)	$A5(I)*\sigma_{tij}$	

[a] The subscripts n and k are both contained in N. The subscripts g and h are not used because data are processed for only one group at a time.

TABLE VIII

CONTENTS OF BLANK COMMON BLOCK IA

Position	Name	Pointer for Array	Remarks
1	ITH		Theory
2	ISCT		Scattering order
3	ISN		Order of S_n
4	IGM		Number of groups
5	IM		Number of radial coarse-mesh intervals
6	JM		Number of axial coarse-mesh intervals
7	IBL		Left boundary specification
8	IBR		Right boundary specification
9	IBB		Bottom boundary specification
10	IBT		Top boundary specification
11	IEVT		Eigenvalue type specification
12	ISTART		Flux input option
13	MT		Total number of materials
14	MIN		Total number of input nuclides from both library and cards
15	MS		Number of mixture instructions
16	IHT		Position in table of total cross section
17	IHS		Position in table of self-scatter cross section
18	IHM		Cross-section table length
19	IQOPT		Source input options
20	IQAN		Distributed source anisotropy order
21	IQB		Bottom boundary source indicator
22			Not used
23	IPVT		Parametric eigenvalue or k_{eff} indicator
24	IANG		Prepare angular flux indicator
25	IMC		Number of <u>material</u> coarse-mesh intervals in the radial direction
26	IITL		Maximum number of inner iterations
27	JMC		Number of <u>material</u> coarse-mesh intervals in the axial direction
28	IRBM		Maximum number of rebalance iterations which is set in INPUT11
29	IXM		Radial-mesh modification indicator
30	IYM		Axial-mesh modification indicator
31	IEDOPT		Edit options
32	IGEOM		Geometry type
33	IQR		Right boundary source indicator
34	IQT		Top boundary source indicator
35	ISDF		Density factor input indicator
36			Not used
37	EV		Eigenvalue guess
38	EVM		Eigenvalue modifier
39	PV		Parametric value of k_{eff} or alpha
40	XLAL		Search lambda lower limit
41	XLAH		Search lambda upper limit
42	XLAX		Fine-mesh search precision
43	EPS(EPSO)		Convergence precision and outer convergence precision
44	EPSI		Inner convergence precision = EPSO

45	EPSR		Within-group rebalance convergence precision = EPSO/2
46	EPSX		Whole-system rebalance convergence precision = EPSO
47			Not used
48	NORM		Normalization amplitude
49	POD		New parameter modifier
50	BHGT		Buckling height
51	IUP		IHS-IHT-1 (upscatter indicator)
52	IHF		IHT-1 (position of $\nu\sigma_f$ in cross-section table)
53	IHA		IHT-2 (position of σ_a in cross-section table)
54	IHTR		IHT-3 (position of σ_{tr} transport cross section in cross-section table, if present)
55	IHNN		IHT-4 (position of $\sigma_{n,2n}$ cross section in cross-section table, if present)
56	IMJM		Product IM*JM
57	MM		(ISN*(ISN+2))/8, number of directions per octant
58	NM		((ISCT+1)*(ISCT+2))/2, number of anisotropic components of flux
59	NMQ		((IQAN+1)*(IQAN+2))/2, number of anisotropic source components
60	IP		Sum IM+1
61	JP		Sum JM+1
62	IGP		Sum IGM+1
63	IJMM		Product IM*JM*MM
64	IT		Total number of radial fine-mesh intervals
65	JT		Total number of axial fine-mesh intervals
66	ITJT		Product IT*JT
67	ITMM		Product IT*MM
68	JTMM		Product JT*MM
69	NMIJ		Product NM*IT*JT
70	NMM		Product NM*MM
71	ISPANC		Last word address of cross-section block
72	IHMT		Product IHM*MT
73	ISPANF		Last word address of flux block
74	ISCP		Sum ISCT+1
75	IMJP		Product IM*JP
76	IPJM		Product IP*JM
77	ITP		Sum IT+1
78	JTP		Sum JT+1
79	ICLIM		Length of the C and AAJ blocks
80	LIHX	IHX(IM)	Number of radial fine-mesh intervals per coarse-mesh interval for rebalance mesh (the block is also the number of radial fine-mesh intervals per coarse-mesh interval for the material mesh when the rebalance mesh and material mesh are the same)
81	LIHY	IHY(JM)	Number of axial fine-mesh intervals per coarse-mesh interval for rebalance mesh (the block is also the number of axial fine-mesh intervals per coarse-mesh for the material mesh when the rebalance mesh and the material mesh are the same)
82	LC	C(IHM,MT)	Cross-sections for a group
83	LA1	A1(ITP)	A4(I+1)+A4(I)
84	LA2	A2(ITP)	A4(I+1)-A4(I)

85	LA3	A3(ITP)	Axial direction surface area
86	LA4	A4(ITP)	Radial direction surface area = A4*YH
87	LA5	A5(ITP)	Volume = A5*YH
88	LQ	Q(NM,IT,JT)	Distributed source (overstorage assigns maximum of NMIJ and IPJM as length)
89	LQR1	QR1(JT,MM)	Input right boundary source for in-down directions (conditional on IQR.EQ.1)
90	LQR2	QR2(JT,MM)	Input right boundary source for in-up directions (conditional on IQR.EQ.1)
91	LQT1	QT1(IT,MM)	Input top boundary source for in-down directions (conditional on IQT.EQ.1)
92	LQT2	QT2(IT,MM)	Input top boundary source for out-down directions (conditional on IQT.EQ.1)
93	LBR1	BR1(JT,MM)	Right boundary flux (in-down and out-down)
94	LBR2	BR2(JT,MM)	Right boundary flux (in-up and out-up)
95	LBT1	BT1(IT,MM)	Top boundary flux (in-down and in-up)
96	LBT2	BT2(IT,MM)	Top boundary flux (out-down and out-up)
97	LXDF	XDF(IT)	Radial fine-mesh density factors
98	LYDF	YDF(JT)	Axial fine-mesh density factors
99	LIHXC	IHXC(IMC)	Number of radial fine-mesh intervals per coarse-mesh interval for the material mesh. When the rebalance mesh and material mesh are the same, array coincides with IHX
100	LIHYC	IHYC(JMC)	Number of axial fine-mesh intervals per coarse-mesh interval for the material mesh. When the rebalance mesh and material mesh are the same, array coincides with IHY
101	LDXC	IDXC(IT)	Material radial direction indicators showing which radial coarse-mesh interval a fine radial mesh interval belongs to for the material mesh. When the rebalance mesh and material mesh are the same, array coincides with IDX
102	LDYC	IDYC(JT)	Material axial direction zone index multiples for the material mesh. When the rebalance mesh and material mesh are the same, array coincides with IDY
103	LDYAC	IDYAC(JT)	Material axial direction indicators showing which axial coarse-mesh interval a fine axial mesh interval belongs to for the material mesh. When the rebalance mesh and material mesh are the same, array coincides with IDYA
104			Not used
105	LFL	FLUX(NM,IT,JT)	Flux components
106	LFLA	FLUXA(IT,JT)	Scalar flux from previous inner iteration (over-storage assigns maximum of ITJT and IPJM as length)
107			Not used
108	LFIS	FISS(IT,JT)	Same origin as FISSA
109	LFISA	FISSA(IT,JT)	Fission source
110			Not used
111	LDC	IDCS(IC)	Cross-section zone identification numbers (IC is IM*JM when the rebalance mesh and material mesh are the same and IMC*JMC when the rebalance mesh and material mesh are not the same)

112	LXR	XRAD(I)	Input radial coarse-mesh boundaries (I is IM+1 when the rebalance mesh and material mesh are the same and IMC+1 when the rebalance mesh and material mesh are not the same)
113	LYR	YRAD(J)	Input axial coarse-mesh boundaries (J is JM+1 when the rebalance mesh and material mesh are the same and JMC+1 when the rebalance mesh and material mesh are not the same)
114	LDX	IDX(IT)	Rebalance radial direction indicators showing which radial coarse-mesh interval a fine-mesh interval belongs to (when the rebalance mesh and material mesh are the same, the block IDXC coincides with IDX)
115	LDY	IDY(JT)	Rebalance axial direction zone index multiples (when the rebalance mesh and material mesh are the same, the block IDYC coincides with IDY)
116	LDYA	IDYA(JT)	Rebalance axial direction indicators showing which axial coarse-mesh interval a fine axial mesh interval belongs to (when the rebalance mesh and material mesh are the same, the block IDYAC coincides with IDYA)
117	LXH	XH(I)	Material mesh radial fine-mesh spacing (I is IM when the rebalance mesh and material mesh are the same and IMC when the rebalance mesh and material mesh are not the same)
118	LYH	YH(JT)	Material mesh axial fine-mesh spacing
119	LW	WGT(MM)	Direction weights
120	LCM	COSMU(MM)	Radial direction cosines
121	LCE	COSETA(MM)	Axial direction cosines
122	LWM	WMU(MM)	Product WGT*COSMU
123	LWE	WETA(MM)	Product WGT*COSETA
124	LP1	P1(NM,MM)	Spherical harmonic functions for in-down sweep
125	LP2	P2(NM,MM)	Spherical harmonic functions for out-down sweep
126	LP3	P3(NM,MM)	Spherical harmonic functions for in-up sweep
127	LP4	P4(NM,MM)	Spherical harmonic functions for out-up sweep
128	LMN	MIXNUM(MS)	Input mixture numbers (conditional on MS.GT.0)
129	LMC	MIXCOM(MS)	Input mixture instructions (conditional on MS.GT.0)
130	LMD	MIXDEN(MS)	Input mixture densities (conditional on MS.GT.0)
131	LF	F(IM,JM)	Coarse-mesh rebalance factors (overstorage assigns IMJP as length)
132	LFU	FU(IM,JP)	Coarse-mesh upward partial current
133	LFD	FD(IM,JP)	Coarse-mesh downward partial current
134	LFR	FR(IP,JM)	Coarse-mesh rightward partial current
135	LFLL	FL(IP,JM)	Coarse-mesh leftward partial current
136	LAB	AB(IM,JM)	Coarse-mesh absorption removal rate
137	LQQ	QQ(IM,JM)	Coarse-mesh source (overstorage assigns IMJP as length)
138	LQQG	QQG(IM,JM)	Coarse-mesh source over all groups (conditional on IEVT.EQ.0)
139	LFUT	FUT(IM,JP)	Sum of FU over all groups (same origin as FU)
140	LFDT	FDT(IM,JP)	Sum of FD over all groups (same origin as FD)
141	LFRT	FRT(IP,JM)	Sum of FR over all groups (same origin as FR)
142	LFLT	FLT(IP,JM)	Sum of FL over all groups (same origin as FL)
143	LABT	ABT(IM,JM)	Sum of absorption rate over all groups
144	LHA	HA(IM)	Used in subroutine REBAL for inversion
145	LGA	GA(IM)	Used in subroutine REBAL for inversion
146	LQG	QG(IGP)	Space integral of Q
147	LFG	FG(IGP)	Space integral of FISSA
148	LSIN	SIN(IGP)	Space integral of group inscatter source

149	LSS	SS(IGP)	Space integral of group selfscatter source
150	LSOUT	SOUT(IGP)	Space integral of group outscatter source
151	LHL	HL(IGP)	Group horizontal leakage
152	LVL	VL(IGP)	Group vertical leakage
153	LTL	TL(IGP)	Group total leakage
154	LNL	NL(IGP)	Group net leakage
155	LRL	RL(IGP)	Group right leakage
156	LABG	ABG(IGP)	Space integral of group absorption rate
157	LBAL	BAL(IGP)	Group neutron balance
158	LCHI	CHI(IGP)	Input fission spectrum
159	LCHIA	CHIA(IGP)	Fission spectrum used in the calculation
160	LVEL	VEL(IGP)	Group velocities
161	LYM	YM(J)	Material axial-mesh modification factors (J is JM when the rebalance mesh and material mesh are the same and JMC when the rebalance mesh and material mesh are not the same)
162	LXM	XM(I)	Material radial-mesh modification factors (I is IM when the rebalance mesh and material mesh are the same and IMC when the rebalance mesh and material mesh are not the same)
163	LXRA	XRADA(I)	Modified material coarse-mesh radial boundaries (I is IM+1 when the rebalance mesh and material mesh are the same and IMC+1 when the rebalance mesh and material mesh are not the same)
164	LYRA	YRADA(J)	Modified material coarse-mesh axial boundaries (J is JM+1 when the rebalance mesh and material mesh are the same and JMC+1 when the rebalance mesh and material mesh are not the same)
165	LSOU	SOURCE(NM,IT,JT)	Total source in a group (same origin as Q)
166	LANF	ANF(ITP,MM,2)	Horizontal angular flux temporary storage (conditional on IANG.NE.0)
167	LAA	AAJ(MT)	Effective absorption cross-section
168	LB1	B1(JT)	1.0/axial delta - used to multiply all areas and volumes in major recursion formulas
169	LAL1	AL1(MM)	α coefficient (MM+1/2)/WGT
170	LAL2	AL2(MM)	α coefficient (MM-1/2)/WGT
171	LALF	ALFL(NN,IT)	α flux due to curvature streaming (NN=ISN/2)
172	LQB1	QB1(IT,MM)	Input bottom boundary source for in-up directions (conditional on IQB.EQ.1)
173	LQB2	QB2(IT,MM)	Input bottom boundary source for out-up directions (conditional on IQB.EQ.1)
174	LCTOT	CTOT(IT,JT)	Effective total cross section
175	JCONV		Final convergence indicator
176	TN2N		N,2N reaction term used in balance equations
177	XLAPP		Value of lambda from sequence of outer iterations previous to that of XLAP
178	XLAP		Value of lambda from previous sequence of outer iterations
179	ICNT		Iteration trigger used in NEWPAR
180	E2		Temporary storage
181	E1		Temporary storage
182	EVPP		Eigenvalue from cycle of outer iteration previous to that of EVP
183	EVP		Eigenvalue from previous sequence of outer iterations
184	E4		Temporary storage

185	NGO		Return indicator set in NEWPAR
186	ALAR		Value of lambda from previous outer iteration
187	MESH		Rebalance mesh and material mesh the same indicator (0/1 = yes/no)
188	IITNO		Inner iteration number
189	TS		Total integrated source to a group
190	G		Number of current group (Integer)
191	ICONV		Secondary convergence indicator
192	NGOTO		Return indicator set in TESTS
193	E3		Temporary storage
194	EVS		Slope used in eigenvalue search
195	IITOT		Total number of inner iterations
196	ALA		Parameter lambda
197	TIN		Time
198	FTP		Previous fission total
199	IFN		Fission calculation indicator set in INITAL
200	OITNO		Outer iteration number
201	ZZ		Radial geometric function used in FIXUP
202	BB		Axial geometric function used in FIXUP
203	CC		Angular function used in FIXUP
204	DD		Angular function used in FIXUP
205	T		Cell-centered flux used in FIXUP
206	S		Source used in FIXUP
207	CT		Total cross section used in FIXUP
208	SUMMU		$\sum_{m=1}^{MM}$ COSMU(M)*WGT(M)
209	SUMETA		$\sum_{m=1}^{MM}$ COSETA(M)*WGT(M)
210	NN		ISN/2
211	AA		Radial geometric function used in FIXUP
212	TI		Temporary i-flux used in FIXUP
213	TJ		Temporary j-flux used in FIXUP
214	TM		Temporary m-flux used in FIXUP
215			Not used
216	ERR		Scalar flux error from comparison with previous flux
217			Not used
218			Not used
219			Not used
220	LBT3	BT3(IT,MM)	Top boundary flux (conditional on IBT.EQ.3)
221			Not used
222			Not used
223	LBT4	BT4(IT,MM)	Top boundary flux (conditional on IBT.EQ.3)
224			Not used
225	NLIMIT		Rebalance constant associated with maximum number of inner iterations (IITL) used to determine type of rebalance in INNER
226	IFLAG		Whole-system rebalance indicator
227 thru 245			Not used

246	IANGPR	Signed angular flux storage indicator
247	TIMACC	Accumulated problem running time
248	MCRRDS	Signed number of input nuclides requested from the code dependent input file
249	NOSGUP	Sigma up included in cross-sections indicator
250	IOLYCS	Overlay indicator (first digit gives primary overlay and second digit gives secondary overlay when read in octal)

TABLE IX

CONTENTS OF NAMED COMMON BLOCK FWBGN1

The named common block FWBGN1 contains the information required to continue the processing of the current problem if it is restarted.

Position	Name	Contents and Remarks
1	IDUSE	A vector used for the title of the problem in A4 format (length of 18 words)
2	LAST	Length of common block A used by the current problem
3	LASTEC	Length of LCM used by the current problem
4	IGCDMP	Group number of restart dump
5	IPSO	LCM pointer of the source-to-the-group block which is calculated by subroutine OUTER and used by subroutine INNER
6	LTSO	Length of the source-to-the-group block
7	IPFL	LCM pointer for the first group of the flow block (each group block contains the coarse-mesh partial currents matrices for the upward, downward, rightward and leftward directions)
8	LTFL	Length of the flow block for a group
9	IPFX	LCM pointer for the first group of the flux block (each group block contains the three-dimensional flux array as well as boundary arrays stored consecutively)
10	LTFX	Length of the flux block for a group
11	LXFX	Length of the three-dimensional flux array
12	IPXS	LCM pointer for the first group of the cross-section block (each group block contains the cross-section array, the effective absorption vector and the spatial total cross-section array)
13	IPXSCT	LCM pointer for the first group of the spatial total cross-section array within the cross-section block
14	LTXS	Length of the cross-section block for a group
15	LTOXS	Length of the cross-section array for a group
16	LTAXS	Length of the cross-section array and the effective absorption vector for a group
17	IPQS	LCM pointer for the first group of the Q-source block (each group block contains the Q-source array and associated boundary source arrays (right, top, and bottom)
18	LTQS	Length of the Q-source block for a group (zero when not a Q-calculation)
19	IEREC	Not used
20	I2	Final flux print indicator (0/1/2 = all/isotropic/none)
21	I4	Final fission print indicator (0/1 = yes/no)
22	I6	Coarse-mesh balance table print indicator (0/1 - yes/no - caution - an extra outer iteration is required to print these tabels when angular fluxes are not requested)
23	ISPANQ	Last word address of the Q-source block in core
24	IPHAF	LCM pointer for the first axial level horizontal angular flux block (total number of levels is 2*JT)
25	IPVAF	LCM pointer for the first axial level vertical angular flux block (total number of levels is 2*JTP)
26	LTHAF	Length of the horizontal angular flux block for an axial level (zero if IANG.EQ.0)
27	LTVAF	Length of the vertical angular flux block for an axial level (zero if IANG.EQ.0)
28	IFO	Interface output indicator (0/1 = no/yes)

TABLE X

CONTENTS OF NAMED COMMON BLOCK FWBGN2

The named common block FWBGN2 consists primarily of those indicators used by the program but not vital to restart. Parameters which define options are set in program TWOTRAN.

Position	Name	Contents and Remarks
1	TIMBDP	Minimum time between periodic dumps (0/F = no periodic dumps/seconds)
2	TIMSLD	Elapsed time since last dump
3	TIMOFF	Floating-point form of the input fixed-point parameter ITLIM
4	MAXLEN	Maximum length of the main common data block A
5	MAXECS	Maximum length of LCM storage available to the problem
6	LENMCB	Length of the named common block FWBGN1 which must be saved for restarts
7	LENCIA	Length of the common parameter block IA which must be saved for restarts
8	IFNOVY	Overlay file name for CDC machine usage (given in left justified, zero fill Hollerith form)
9	IRCOVY	Recall overlay indicator for CDC machine usage (0/6HRECALL = no reloading of overlay when in core/reloading overlay when in core)
10	I1	Full flux guess input print indicator (indicator effective when ISTART is 2 or -2, 0/1 = yes/no)
11	I3	Cross-section print indicator (0/1/2 = all/mixed/none)
12	I5	Input source print indicator (0/1/2/3 = all/input/normalized/none)

TABLE XI

CONTENTS OF NAMED COMMON BLOCK LOCAL

The named common block LOCAL contains information that is passed from overlay to overlay for a problem but is not needed in restart.

Position	Name	Contents and Remarks
1	NERROR	Parameter input-error indicator
2	ITLIM	Fixed-point time problem removal value (0/N = no/seconds)
3	ISNT	S_n library request indicator
4	MCR	Number of nuclides requested from the code dependent input file (minus if FIDO format)
5	MTP	Number of library nuclides requested from ISOTXS interface file (MTPS*(ISCT+1))
6	MTPS	Number of nuclide sets requested from ISOTXS interface file (each set yields ISCT+1 blocks)
7	NISOXS	Number of isotopes in the set as read from the ISOTXS interface file
8	LMTP	Core pointer for the ISOTXS interface file position number data block
9	IEDOPS	Signed value of IEDOPT
10	NEXTER	Fetch next case indicator (if any error was detected after all input was successfully read, the next problem is fetched)
11	JFISC	Return indicator used in the computational overlay (2,0)
12	IEDIT	A vector of length two which contains the edit values NZ and NORMZ
13	LIMIT	Maximum storage length required by input cross-sections and calculation of anisotropic scattering coefficients
14	LENCLR	Length of the partial block to be cleared during input

TABLE XII

CONTENTS OF NAMED COMMON BLOCK SWEEP

The named common block SWEEP is used to pass the current value of variables from subroutine INNER to subroutines IN and OUT without using a calling sequence entry.

Position	Name	Contents and Remarks
1	BA	Two times one over the axial delta for the current J level
2	BC	YH(J) which is the current axial fine-mesh spacing
3	J	Current axial J level index
4	J1	IDYA(J) which is the current axial direction indicator for the axial fine-mesh
5	J2	IDYA(J-1) for downward (J decreasing) sweeps and IDYA(J+1) for upward (J increasing) sweeps

TABLE XIII

CONTENTS OF NAMED COMMON BLOCK UNITS

The named common block UNITS contains the symbolic names of all input, output, and scratch devices required by TWOTRAN-II which are set in the main program TWOTRAN.

Position	Name	Contents and Remarks
1	NINP	Problem code dependent decimal input
2	NOUT	Problem decimal output
3	NAFLUX	Angular flux by group
4	NDUMP1	First restart dump unit
5	NDUMP2	Second restart dump unit
6	NEXTRA	Scratch unit
7	NEDIT	Edit input storage
8	IAFLUX	Interface form of angular flux
9	ITFLUX	Interface form of total flux
10	ISNCON	Interface form of S_n constants
11	IFIXSR	Interface form of inhomogeneous Source (Q source)
12	ISOTXS	Interface form of multigroup cross section file ISOTXS

5. Machine Dependent Subprograms

a. LCM System Routines

LCM (large core memory) is a large bulk memory from which blocks of words may be quickly transferred to or from SCM (small core memory). This random bulk memory is accessed through two system routines - ECRD (transfers LCM to SCM) and ECWR (transfers SCM to LCM) - which process consecutive words of SCM and consecutive words of LCM given an SCM address and a pointer value for LCM. The pointer value given may be thought of as the index of a container array. To read from or write into a block of core, it is necessary to provide the read/write routines with the core origin, the LCM pointer value and the number of consecutive words to be transferred. For example, if we consider reading the entire FLUX block for group IG from LCM to SCM, we would have the Fortran IV statements

```
CALL REED(0,IPFX+(IG-1)*LTFX,FLUX,LTFX,1) and

CALL ECRD(FLUX,IPFX+(IG-1)*LTFX,LTFX,IER)
```

In these statements FLUX is the SCM container array, IPFX+(IG-1)*LTFX is the location of the first word of the IGth group flux array in LCM, and LTFX words are transferred. IER is an error indicator. On the CDC 6600 Extended Core Storage (ECS) plays the role of LCM. On the IBM 360/195 ECRD and ECWR are replaced with core-to-core transfers.

b. General System Routines

Additional system routines required by the code are SECOND (obtains current time), DATE1 (obtains current date), ATAN (arctangent), SQRT (floating point square root), and EXIT (returns control to system for next job).

Use of an end of file test is made in INPUT11 to detect the last case of a sequence of cases. The test must be replaced by an equivalent statement to obtain a normal exit.

B. External and Internal Data Files

All files used for input, output and scratch data are referred to by symbolic name throughout the code. The user may easily change the physical unit assigned a file by modification of the symbolic name which is initialized in the main program of TWOTRAN-II. Table XIV indicates the files required by TWOTRAN-II.

TABLE XIV

TWOTRAN-II FILE REQUIREMENTS

Name	Logical Unit (CDC Machines Only)	Contents	Remarks
NINP	10	Problem code dependent decimal input	The user may wish to equate this file to the system input file.
NOUT	9	Problem decimal output	The user should equate this file to the system decimal printed output file.
NAFLUX	6	Binary angular flux by group generated only on a special last outer iteration when requested	The contents for each group consists of 2*JTP records of length LTVAF plus 2*JT records of length LTHAF.
NDUMP1	7	Restart dump	This unit is used to receive the first restart dump when the problem is not restarted from a previous dump. The unit must contain the restart dump information when the problem is restarted and will then be used to receive the second restart dump (NDUMP2 receives the first dump).
NDUMP2	5	Restart dump	Second restart dump unit
NEXTRA	18	Scratch file	The file is used in both the decimal and binary mode. The records at any one time are not mixed mode. The decimal mode is used for Hollerith conversions rather than the core to core conversions given by the FORTRAN statements of ENCODE and DECODE of CDC machines. The binary mode is used to reorder data required by interface output file routines and editing routines.
NEDIT	17	Edit input storage	To save core, edit input is stored on this file until time of edit.
IAFLUX	31	Interface form of angular flux (either adjoint or regular)	Output of the angular fluxes in interface form is placed on this file. The file is rewound prior to processing the fluxes and an end of file is placed on the file after the last write. Data for one problem only is kept on this file.

ITFLUX	30	Interface form of total flux (either adjoint or regular)	The code requires that this unit be used when a flux guess is requested from the total flux interface file. The unit is rewound and the records of the first file are used as the input guess. The interface form of the total flux is prepared on this file as problem output by rewinding the file and writing the file in standard format. An end of file is placed on the file after the last write instruction.
ISNCON	32	Interface form of S_n constants	When the file is used as input, only the first one-fourth of the values of weights and direction cosines are read as input. When used as input, the file is rewound and read. When used as output the file is rewound and written, including an end of file.
IFIXSR	33	Interface form of source	This file is used only as input for the cell centered inhomogenous source. Boundary sources (if any) are obtained from the code dependent input file.
ISOTXS	34	Interface form of the cross-section multigroup file ISOTXS	This file is only used as input when cross sections are requested from an interface file library.
N5	40	Scratch storage for FIDO subroutine which appears only in the IBM code version	Necessary for reading FIDO format cross sections on IBM 360 machines only

C. Hardware Requirements

The TWOTRAN-II code does not require any special hardware. The LASL CDC 7600 provides 65K (decimal) SCM and 512K LCM 60-bit words. Only 370K LCM are available to the user with the operating system and buffers using the remainder. Type 7638 disk units provide 84 million decimal words of peripheral store per unit.

D. Software Requirements

1. CDC Machines

The code was designed to operate on the CDC 7600 under the CROS operating system[12] which was developed at Los Alamos. The system uses the CDC RUN compiler with a CDC optimizer attached.

The code operates on the CDC 6600 under SCOPE 3.1 which has been heavily modified[13] at Los Alamos, and the CDC RUN compiler is again used.

On both machines, the disk units provide storage for input, output, scratch, and resident files.

2. TWOTRAN-II for the IBM-360

TWOTRAN-II was prepared for the IBM-360/195 from the CDC-7600 version of the code. The IBM-360 version operates in the four byte floating point mode. The CDC-7600 version of the code was prepared so that conversion to the IBM-360 would involve as few changes as possible and so that an eight byte floating point version of the code could easily be made, if needed. Entries of named common blocks were ordered to make this possible, and formats for reading input title data were changed to A4 formats.

The major change made in the conversion of the code was in the treatment of peripheral storage. The vast amount of fast core available on the IBM-360 is one of the cheaper resources of that machine. It was thus decided to place the data normally kept in LCM (large core memory) in an expanded container array. This was accomplished by using part of the container array (indexed from 1 through LAST) as small core (SCM) and the remainder as LCM (indexed starting at LAST+1 and ending at MAXLEN). The system routines ECRD and ECWR of the CDC-7600 were replaced by simple routines which move data to and from sections of the container array. It was thus possible to keep the LCM pointer structure of the code with no change in logic and with little cost in time for data movement. It should be noted that the amount of core needed (MAXLEN) is computed first and the amount of LCM needed is computed as MAXLEN minus LAST. The size of the container array is changed simply by changing the dimension of common block A and assigning a new value to MAXLEN. Both changes need be made only in program TWOTRAN.

In the edit and interface output sections of the code, the amount of SCM required is recomputed and tested against LAST. In rare cases these tests may fail, but enough core may exist in the container array for both SCM and LCM. Then, the value LAST may be reset and that portion of the container array corresponding to LCM moved to accomodate the increased requirement for SCM.

Because of the amount of core storage available on the IBM machines, it was found to be computationally advantageous to combine the computational overlay GRIND2 with its secondary overlays.

In addition to storage reorganization, the following changes were made to effect the IBM conversion of TWOTRAN-II:

1. Dummy subroutines were substituted for system subroutines DATE1 (returns date as an A8 word) and SECOND (returns floating point value of the current time). The dummy routine for SECOND must be replaced by a local system routine or the periodic and time limit dump options will not work.
2. A subroutine FIDO was added to process the FIDO cross-section format. The CDC-7600 algorithm to read this format uses a rewind command. On the IBM system a prohibitively large amount of wait time was required because of the rewind command.
3. Hollerith 6H constants throughout the code were typed as real eight byte data, including subroutine call list variables. Subroutine WRITE uses one such variable as either an integer (four bytes) or an 6H constant (eight bytes), and this situation was treated by equivalencing statements.
4. The IF(EOF) CDC job termination test was replaced with a read with the IBM END parameter.
5. CDC overlay cards were removed, overlay programs were changed to subroutines, calls to overlay segments were replaced with calls to the subroutines, and the setting of overlay parameters IFNOVY and IRCOVY was eliminated.
6. Several options were added to subroutine REED to treat interface file data. All such data is assumed to be four-byte words, but the specification record of such files is three eight-byte words plus one four-byte integer. Two arrays of the ISOTXS file contain mixed 6H data (8 bytes), and floating point and integer (4 bytes). Finally, reading the SNCONS file required skipping a portion of the record before reading additional data. All of these problems were treated by adding options to REED.
7. Shortened names (six character maximum) were required for the following routines:

CDC	IBM-360
TWOTRAN	(Main program)
INPUT11	INPT11
INPUT12	INPT12
INPUT13	INPT13
INPUT14	INPT14
INPUT15	INPT15
GRIND21	GRID21
GRIND22	GRID22
GRIND23	GRID23
OUTPUT3	OUTPT3
OUTPT31	OUTT31
OUTPT32	OUTT32

E. Programming Considerations

1. Storage Management

a. Variable Dimensioning

A single container array, A, in common is used for the blocks of data required in executing a problem. The storage of all data is consecutive and compact in the A array so that the size of a problem is limited by the total storage required rather than by the size of individual parameters. A pointer word is associated with each data block and is used to index A to locate the block. For example LFL is the first word address of the flux block in A and A(LFL) is the first word of the flux array. When subroutine calls are written, the address of a data block, say A(LFL), is passed through the argument call. In the subroutine the data block is variably dimensioned so that it may be easily indexed by its subscripts, e.g. FLUX(N,I,J).

With the exception of subroutines IN and OUT, consistent use is made of data mnemonics. In subroutines IN and OUT, variable names are shortened and dimension sizes are transmitted strictly to variably dimension arrays. Index limits IT, MM, NM, and ITP, which coincide with dimension sizes, are passed through equivalences. This arrangement saves computing time when the RUN compiler is used.

b. Allocation of Large Core Memory (LCM)

The allocation of storage in large core memory (LCM) is handled in the same manner as core storage. Most of the group-dependent arrays are stored in LCM so the dimensionality is IGM times the core requirement of the array. For example, there are IGM*NM*IT*JT LCM locations required for FLUX(NM,IT,JT).

Certain blocks of data are stored contiguously in core so that they may be read in and out of LCM in a single stream. For example, the flux block includes FLUX(NM,IT,JT), BR1(JT,MM), BR2(JT,MM), BT1(IT,MM), and BT2,(IT,MM) when the boundary conditions are implicit; and, if a periodic boundary condition is used on the top and bottom boundaries, BT3(IT,MM) and BT4(IT,MM) are also included. The first word of this block is LFL, and the last word is ISPANF. The cross-section block included the cross sections C(IHM,MT), the effective absorption AAJ(MT) and the total cross section CTOT(IT,JT). The first word of this block is LC, and the last word ISPANC. The Q source block is conditional on a source type calculation. When present it includes Q(NM,IT,JT), QR1(JT,MM) and QR2(JT,MM) when IQR is one, QT1(IT,MM) and QT2(IT,MM) when IQT is one, and QB1(IT,MM) and QB2(IT,MM) when IQB is one. The first word of this block LQ and the last word is ISPANQ.

Angular flux data is written in LCM not by group but by an axial level index. A complete list of LCM storage is given in Table XV.

c. Computation of Required Storage

The easiest way to compute the storage required by a problem is to load the problem for a short run and let the code compute LAST, the amount of SCM and LASTEC, the amount of LCM. The computation is made very early in problem execution and the result is printed before most of the data is read. An approximate formula for LAST is

TABLE XV

LCM STORAGE PARAMETERS

LCM First Word Address	Length Per Block	No. of Blocks	Contents
IPFX	LTFX (ISPANF+1-LFL)	IGM	Flux blocks by group
IPSO	LTSO (NM*IT*JT)	1	Source to group not including within group scatter
IPXS	LTXS (IT*JT+MT*(IHM+1))	IGM	Cross section blocks by group
IPQS	LTQS (ISPANQ+1-LQ)	IGM	Q-source blocks by group, present only if IEVT.EQ.0
IPFL	LTFL (IPJM+IMJP)*2)	IGM	Flow blocks by group
IPHAF	LTHAF (ITP*MM*2)	2*JT	Horizontal angular flux by axial level, present only if IANG.NE.0
IPVAF	LTVAF (IT*MM*2)	2*JTP	Vertical angular flux by axial level, present only if IANG.NE.0

$$LAST = IT*JT*(2*NM+3)+2*MM*(IT+JT)+8*(IM*JM)$$
$$+(IMC*JMC)+(IHM*MT)$$

where the variables in the formula are defined in Table VIII. The formula does not include several conditional data blocks. The largest of these is the angular flux which can be very expensive, especially for LCM storage. The sizes of the conditional blocks are:

2*MM*ITP	IANG.NE.0
2*MM*JT	IQR.NE.0
2*MM*IT	IQT.NE.0
2*MM*IT	IQB.NE.0
IM*JM	IEVT.EQ.0
IMC	IXM.NE.0
JMC	IYM.NE.0
2*MM*IT	IBT.EQ.3
2*JT+IT+IMC+JMC	MESH.NE.0

The amount of LCM is given by

$$LASTEC = IGM*\{IT*JT*(NM+1)+MT*(IHM+1)$$
$$+2*[JM*(IM+1)+IM*(JM+1)]\}+NM*IT*JT$$

with conditional block sizes:

IGM*MAXO[NM*IT*JT,(IM+1)*JM]	IEVT.EQ.0
2*IGM*IT*MM	IBT.EQ.3
2*IGM*MM*(IT+JT)	IBR.GT.0 or IBT.GT.0
4*MM(ITP*JT+IT*JTP)	IANG.NE.0
2*IGM*MM*JT	IQR.NE.0
2*IGM*MM*IT	IQT.NE.0
2*IGM*MM*IT	IQB.NE.0

d. Temporary Storage Requirements

The amount of storage actually calculated for LAST is the maximum of three quantities. The first of these is the total amount of SCM required for problem execution and the other two are temporary storage requirements, one for input of cross sections and the other for calculation of anisotropic scattering coefficients. Usually, the problem data requirement is much larger than the temporary storage requirement, but occasionally the input cross section requirement (IGM*IHM*MT words) is largest.

A test is made for each of the temporary storages and the user is informed if more SCM is required than is available. Additional tests are made at the time of edit information input (The edit data is stored peripherally during problem execution, but it is conceivable that reading the data on input in INPUT15 would require too much storage.), when storage is redefined for editing in EDCALL and when storage is redefined for interface output in IFOUT.

e. Overstorage of Data In Core

In TWOTRAN-II, a certain amount of overstorage is used to reduce the total amount of small core memory (SCM) required; i.e., more than one array may reside in the same SCM locations as the problem progresses. The most important example of this is that the large arrays Q(NM,IT,JT) and SOURCE(NM,IT,JT) are stored in the same spaces. The Q array is used in OUTER to generate the total source to a group, including the inhomogeneous source, if any. This source is then written in LCM. In INNER this source is read into SCM after each inner iteration to initialize the SOURCE array. Then the rest of the group source is added to SOURCE.

The coarse-mesh total flows (sum over all groups of the group flows) are stored in the same space as the group flows. That is, FUT, FDT, FRT, FLT are stored over FU, FD, FR, and FL. The group values of FU, FD, FR, and FL are stored in LCM and, after all groups are finished, the total values are accumulated. To perform this accumulation, we make temporary use of the arrays Q(NM,IT,JT), FLUXA(IT,JT), F(IM,JM), and QQ(IM,JM). For this reason when we allocate storage for, say, FLUXA, we must assign the larger of IT*JT and (IM+1)*JM storage words because both sizes of arrays are stored in the same SCM area. Similar precautions are necessary for the other arrays.

When input cross sections are read, they are read into location LQ and when anisotropic scattering spherical harmonic coefficients are calculated, storage beginning at location LC is used. In each of these cases, the amount of storage required is checked (previous section) and the core is reinitialized after the operation is complete. Similar overstorage is performed when maps are written (subroutine MAPPER) and when the microscopic material edit block is read into locations beginning at LFU.

2. Restart Tape Composition

The restart dump is composed of the following records: restart parameter information, named common block FWBGN1, common block IA, data common block A (only the portion used by the problem), group flux blocks from LCM, blocks of the remainder of LCM (starting after the last flux block and with a maximum length of the data portion of common block A), and angular flux records by group (2*JTP records of length LTVAF plus 2*JT records of length LTHAF per group). The angular flux information is only generated when requested and in the last pass of the calculation for groups one through the current group. The number of data entries is the first word of each record on the dump tape. The final dump contains the current group (IGCDMP) value of zero.

The restart parameter information is a vector which contains five words in the following order:

1.	IGM	Number of groups.
2.	NORDM	Number of LCM records excluding the flux records.
3.	NORDAF	Number of angular flux records.
4.	ITJT	Product of fine radial mesh total and fine axial mesh total.
5.	NM	Number of flux components.

The flux by group was separated from the rest of LCM to permit easy access to that block.

3. Standard Interface Files

The standard interface files read and written by TWOTRAN-II are version II files.[3] As far as possible the codes which process these files are subroutines. The user should note that in reading, the very first entry on a file is used as the input, and in writing, the file is rewound prior to the output of data. No physical unit distinction is made between regular and adjoint input or output files. If a standard interface file is used to provide a flux guess, and a standard interface file flux output is requested, the input file information is destroyed.

ACKNOWLEDGMENT

The authors are very grateful to Neva Noyes for the outstanding job she did in preparing and typing this report.

REFERENCES

1. K. D. Lathrop and F. W. Brinkley, "Theory and Use of the General-Geometry TWOTRAN Program," Los Alamos Scientific Laboratory report LA-4432 (May 1970).

2. Wm. H. Reed, T. R. Hill, F. W. Brinkley and K. D. Lathrop, "TRIPLET, a Two-Dimensional, Multigroup Code for Triangular Mesh, Planar Geometry, Explicit Transport," to be published as a Los Alamos Scientific Laboratory report.

3. B. M. Carmichael, D. A. Meneley, and D. R. Vondy, "Report of the Subcommittee on Standard Interface Files," prepared for the Committee on Computer Code Coordination and issued as Los Alamos Scientific Laboratory report LA-5324-MS.

4. W. W. Engle, Jr., "A Users Manual for ANISN, a One Dimensional Discrete Ordinates Transport Code With Anisotropic Scattering," Union Carbide Corporation report K-1693 (March, 1967).

5. Draft #8 of the American Nuclear Society Standard 10.3, "Guidelines for the Documentation of Digital Computer Programs," to be published.

6. G. I. Bell, G. E. Hansen, and H. A. Sandmeier, "Multitable Treatments of Anisotropic Scattering in S_n Multigroup Transport Calculations," Nucl. Sci. Eng., 28, 376 (1967).

7. K. D. Lathrop, "Spatial Differencing of the Transport Equation: Positivity vs. Accuracy," J. Comput. Phys., 4, 475 (1969).

8. Shoichiro Nakamura, "A Variational Rebalancing Method for Linear Iterative Convergence Schemes of Neutron Diffusion and Transport Equations," Nucl. Sci. Eng., 39, 278 (1970).

9. B. G. Carlson and G. I. Bell,"Solution of the Transport Equation by the S_n Method," Proceedings Second U. N. Intern. Conf. Peaceful Uses At. Energy 16, 535 (1958).

10. K. D. Lathrop, "DTF-IV, A Fortran Program for Solving the Multigroup Transport Equation with Anisotropic Scattering," Los Alamos Scientific Laboratory report LA-3373 (November 1965), page 42.

11. B. G. Carlson and K. D. Lathrop, "Transport Theory - Method of Discrete Ordinates," Chapter III of Computing Methods in Reactor Physics, Gordon and Breach, New York (1968), page 211.

12. "Elementary Guide to the Control Data 7600," Ray Davenport, Ed., Programmer's Information Manual, Vol. 5A, Los Alamos Scientific Laboratory (1972).

13. Programmer's Information Manual, Vol. 4, CDC 6600, Anne Browing, Ed., Los Alamos Scientific Laboratory (1973).