

LA-6333-MS

Informal Report

UC-32 and UC-79 (Base Technology)
Reporting Date: April 1976
Issued: May 1976

C. 2

THRETRAN :
**A Program to Solve the Multigroup Discrete
Ordinates Transport Equation in (x, y, z) Geometry**

by

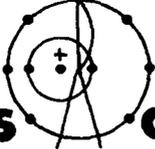
K. D. Lathrop

LOS ALAMOS NATL. LAB. LIBS.
3 9338 00394 2256

DO NOT CIRCULATE

PERMANENT RETENTION

REQUIRED BY CONTRACT


los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87545

An Affirmative Action/Equal Opportunity Employer

UNITED STATES
ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION
CONTRACT W-7405-ENG. 36

This work was supported by the US Energy Research and Development Administration, Division of Reactor Development and Demonstration. Program No. D538.

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161
Price: Printed Copy \$4.50 Microfiche \$2.25

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research and Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or 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.

CONTENTS

ABSTRACT - - - - - 1

I. COMPUTER PROGRAM OUTLINE - - - - - 1

II. INTRODUCTION - - - - - 3

III. EQUATIONS AND SOLUTION ALGORITHMS - - - - - 5

 A. Multigroup Transport Equation - - - - - 5

 B. Central Difference Approximation - - - - - 6

 C. Angular Partitioning - - - - - 7

 D. Solution Algorithm - - - - - 8

 1. Boundary Conditions - - - - - 8

 2. Progression Through the Space-Angle Mesh - - - - - 9

 3. Convergence Acceleration - - - - - 9

 4. Convergence Tests - - - - - 11

IV. PROGRAM OPTIONS AND INPUT DESCRIPTION - - - - - 12

 A. Program Options - - - - - 12

 1. Types of Problems - - - - - 12

 2. Source and Flux Inputs - - - - - 12

 3. Geometry Specification - - - - - 12

 4. Cross Sections - - - - - 12

 5. Dumps - - - - - 13

 6. Edits - - - - - 13

 B. Description of Problem Input - - - - - 13

 1. Input Formats - - - - - 13

 2. Input Data - - - - - 15

V. PROGRAMMING INFORMATION - - - - - 18

 A. Role and Function of Subprograms - - - - - 18

 B. Data Storage - - - - - 21

 C. Data Management Algorithms - - - - - 22



CONTENTS (cont)

1.	LCM and Disk Data Transmission - -	22
2.	Data Storage and Data Blocking - -	36
3.	Disk Assignment - - - - -	40
4.	Staged Data Transmission - - - - -	40
5.	Calculation of the Source-to-a- Group - - - - -	41
6.	Control of Inner Iteration - - - -	43
7.	Calculation of the New Fission Source - - - - -	44
8.	Dumps - - - - -	44
D.	Problem Printed Output - - - - -	46
VI.	PROGRAM TESTING AND SUGGESTED FUTURE MODIFICATIONS - - - - -	46
A.	Program Testing - - - - -	46
B.	Suggested Future Modifications of THREETRAN - - - - -	47
	REFERENCES - - - - -	48

LIST OF FIGURES

Figure 1.	Names of system boundaries. - - - - -	8
Figure 2.	Overall program flow in THREETRAN. - -	19
Figure 3.	Structure of A array. - - - - -	21
Figure 4.	Examples of blocking parameters for lines, planes, and groups. - - f - - -	37

LIST OF TABLES

Table I	Options for Special Read Formats - - -	14
Table II	Function of THREETRAN Service Routines - - - - -	20
Table III	Contents of Common Block IA within A Array - - - - -	23

LIST OF TABLES (cont)

Table IV	THREETRAN Storage Modes - - - - -	39
Table V	Disk Assignment in THREETRAN - - - - -	41
TEST PROBLEM LISTING - - - - -		49

THREETRAN:

A PROGRAM TO SOLVE THE MULTIGROUP DISCRETE ORDINATES
TRANSPORT EQUATION IN (x,y,z) GEOMETRY

by

K. D. Lathrop

ABSTRACT

Numerical formulations and programming algorithms are given for the THREETRAN computer program which solves the discrete ordinates, multigroup transport equation in (x,y,z) geometry. An efficient, flexible, and general data-handling strategy is derived to make use of three hierarchies of storage: small core memory, large core memory, and disk file. Data management, input instructions, and sample problem output are described. A six-group, S₄, 18 502 mesh point, 2 800 zone, k_{eff} calculation of the ZPPR-4 critical assembly required 144 min of CDC-7600 time to execute to a convergence tolerance of 5×10^{-4} and gave results in good qualitative agreement with experiment and other calculations.

I. COMPUTER PROGRAM OUTLINE

1. Program Identification: THREETRAN
2. Computer for which Program is Designed: CDC-7600
3. Description of Function: THREETRAN solves the time-independent three-dimensional discrete ordinates approximation to the multigroup transport equation in rectangular coordinates. Inhomogeneous source, k_{eff} , and source plus fission problems are solved.
4. Method of Solution: The central difference approximation is used in conjunction with standard finite difference

approximations to the spatial derivatives of the discrete ordinates streaming operator. The resulting system of equations is solved by within-group (inner) iterations on the scattering source and over all-group iterations on the fission source. The inner iterations are accelerated by updating the scalar flux after each quadrant of angular fluxes is computed and by a special line rebalance. The outer iterations are unaccelerated.

5. **Restrictions:** The program is presently restricted to isotropic scattering although linearly anisotropic scattering could be treated without increase of storage by dropping the scalar flux update. There is no negative flux fixup and no provision for upscattering cross sections.
6. **Running Time:** A six-group, S_4 , 18 502 mesh point k-calculation of the ZPPR-4 Phase 1 experiment required 144 min on a CDC-7600 for a 5×10^{-4} eigenvalue convergence.
7. **Unusual Features of the Program:** The program is organized to calculate a line (x-dimension) at a time. Storage is automatically apportioned between CDC small core memory (SCM), large core memory (LCM), and random access disk units to permit as large a problem as possible to be run with minimum use of disk files. Data transmission and computing are overlapped when disk files are used.
8. **Machine Requirements:** Ten disk units may be required depending on the problem type and output options. Two files are required for system input and output, one for a problem dump, one for a standard interface flux output tape, and one for microfiche recording of fluxes. If the problem data do not fit within LCM, a flux file and an inhomogeneous source file for source problems are required. For larger problems, two fission source files and a source-to-the-group file are also required.
9. **Related Programs:** None.
10. **Material Available:** Source deck, test problems, results of executed test problems, and this report are available from the author.

II. INTRODUCTION

The THREETRAN program was developed to examine the problems associated with writing and running a full three-dimensional (x,y,z) multigroup transport code. Because large running times were expected, the discrete ordinates angular representation was coupled with the central (diamond) difference approximation without a negative flux correction to minimize arithmetic operations. Consequently, the running times obtained with THREETRAN are to be regarded as minimums with respect to the numerical formulation; that is, use of a more sophisticated difference approximation or a finite-element representation of the transport equation would almost certainly require more computational time per mesh cell. On the other hand, the convergence acceleration algorithms used in THREETRAN are rudimentary, and consequently there is reasonable expectation that running times can be improved by using better convergence acceleration schemes. The convergence acceleration schemes in THREETRAN were deliberately kept simple for ease of development and because a potentially very powerful scheme due to Alcouffe and Lewis^{1,2} was being created concurrently.

Data storage and transfer were also expected to be a problem with a three-dimensional transport code, and the major research effort in writing THREETRAN was the creation of a flexible and relatively efficient data management strategy. The relatively small six-group, S_4 , 18 502 mesh point, k_{eff} test problem (a one-eighth core representation of the ZPPR-4 Phase 1 critical assembly) run with THREETRAN in 144 min (CDC-7600) to an eigenvalue convergence of 5×10^{-4} involved an unknown vector of 2.66×10^6 entries. This corresponds to a coefficient matrix of 7.10×10^{12} elements, and there is clearly no possibility of storing such a matrix on existing computers. Fortunately, the matrix of the discrete ordinates approximation is sparse and lower triangular so that it can be regenerated repetitively during the iterative sequence, and the angular flux vector itself usually need not be stored. Even so, there were 4.44×10^5 fluxes stored in the test problem.

THREETRAN was written assuming the CDC-7600 hierarchy of storage: 1) a fast small core memory (SCM) of about 40 000 words for problem data storage, 2) a fast large core memory (LCM) of about 370 000 words, and 3) random access disk files of about 30 million words each. Using system-dependent routines it was possible to overlap disk data transmission and computation. All disk transfers in THREETRAN are made to and from LCM with the exception of problem dumps which are written sequentially from SCM.

Within the three-level hierarchy, THREETRAN was structured to fit as much of the computation as possible into the fastest memory available. The line, one row of x -dimensional cells from an (x,y) plane of the (x,y,z) problem, was selected as the basic unit for SCM storage. If the data requirements for one line do not fit in SCM, THREETRAN will not execute the problem. Data for as many lines as possible, up to one full (x,y) plane, are stored in SCM. Similarly, the plane was chosen as the basic unit of LCM storage. If data for one plane will not fit in LCM, the problem cannot be executed. Data for as many planes as possible, up to the whole (x,y,z) problem, are stored in LCM. If more storage is available in LCM, data for as many groups as possible are stored in LCM. Surprisingly large problems can be contained entirely within SCM and LCM. If data for all planes fit in LCM, but all group data do not, the group structure is divided into blocks, each block of which is small enough to fit into LCM. In this storage mode only flux data and inhomogeneous source (if one is present) data are stored on disk files, and data for several groups may be transferred at once. When the problem is so large that all plane data do not fit in LCM, then additional disk files are required for the source-to-the-group data and fission source data.

In the following sections are described the details of basic difference schemes, numerical methods, program options, storage algorithms, and input preparation. The THREETRAN program was written for the purpose of solving the transport equation, and the reader will note that many service functions performed by lower dimensional transport codes are not performed by THREETRAN. For example, there is no cross-section processing in the code, either

in input or output, and in particular, the code accepts only macroscopic cross sections.

III. EQUATIONS AND SOLUTION ALGORITHMS

A. Multigroup Transport Equation

In the discrete ordinates approximation, we write the transport equation for each group as

$$\mu_m \frac{\partial \psi_m}{\partial x} + \eta_m \frac{\partial \psi_m}{\partial y} + \xi_m \frac{\partial \psi_m}{\partial z} + \sigma_t \psi_m = S_m \quad (1)$$

where μ_m , η_m , and ξ_m are the direction cosines of the discrete direction $\vec{\Omega}_m$ with respect to the x, y, and z axes. The angular flux in the discrete direction $\vec{\Omega}_m$ is denoted by $\psi_m(x, y, z)$ and σ_t is the macroscopic total cross section. The source term S_m contains contributions from scattering, both for the group being considered and from other groups, and fission, all evaluated at the discrete direction $\vec{\Omega}_m$. The details of the generation of S_m in a multigroup transport code are given in the TWOTRAN-II manual³ and are not repeated here.

To difference Eq. (1) we partition the spatial domain into boxes such that

$$\begin{aligned} X_{i-\frac{1}{2}} < X_i < X_{i+\frac{1}{2}} & \quad i = 1, 2, \dots, IT \\ Y_{j-\frac{1}{2}} < Y_j < Y_{j+\frac{1}{2}} & \quad j = 1, 2, \dots, JT \\ Z_{k-\frac{1}{2}} < Z_k < Z_{k+\frac{1}{2}} & \quad k = 1, 2, \dots, KT \end{aligned} \quad (2)$$

In this partitioning, subscripts with half-integer values denote box edges, and integer subscripts, consistent with our use of the central difference approximation, denote box center values.

Using a notation in which centered subscripts, including m, are not written, we difference Eq. (1) simply as

$$\frac{\mu(\psi_{i+\frac{1}{2}} - \psi_{i-\frac{1}{2}})}{\Delta X} + \frac{\eta(\psi_{j+\frac{1}{2}} - \psi_{j-\frac{1}{2}})}{\Delta Y} + \frac{\xi(\psi_{k+\frac{1}{2}} - \psi_{k-\frac{1}{2}})}{\Delta Z} + \sigma_t \psi = S. \quad (3)$$

In this equation $\psi_{i+\frac{1}{2}}$ is actually $\psi_{m,i+\frac{1}{2},j,k}$, the angular flux in direction m at the center of the box face at the $X_{i+\frac{1}{2}}$ edge. The mesh spacings are

$$\begin{aligned} \Delta X &= X_{i+\frac{1}{2}} - X_{i-\frac{1}{2}} \\ \Delta Y &= Y_{j+\frac{1}{2}} - Y_{j-\frac{1}{2}} \\ \Delta Z &= Z_{k+\frac{1}{2}} - Z_{k-\frac{1}{2}}. \end{aligned} \quad (4)$$

B. Central Difference Approximation

Throughout THREETRAN we always assume

$$\begin{aligned} \psi_{ijk} &= (\psi_{i+\frac{1}{2}} + \psi_{i-\frac{1}{2}})/2 \\ &= (\psi_{j+\frac{1}{2}} + \psi_{j-\frac{1}{2}})/2 \\ &= (\psi_{k+\frac{1}{2}} + \psi_{k-\frac{1}{2}})/2. \end{aligned} \quad (5)$$

With this assumption we can reduce the solving of Eq. (3) to evaluating the system

$$\psi = \frac{a \psi_{i-\frac{1}{2}} + b \psi_{j-\frac{1}{2}} + c \psi_{k-\frac{1}{2}} + S}{a + b + c + \sigma_t} \quad (6)$$

$$\psi_{i+\frac{1}{2}} = 2\psi - \psi_{i-\frac{1}{2}}$$

$$\psi_{j+\frac{1}{2}} = 2\psi - \psi_{j-\frac{1}{2}}$$

$$\psi_{k+\frac{1}{2}} = 2\psi - \psi_{k-\frac{1}{2}}$$

where

$$a = 2|\mu|/\Delta X$$

$$b = 2|\eta|/\Delta Y$$

$$c = 2|\xi|/\Delta Z . \quad (7)$$

In writing Eq. (6) we have assumed that $\psi_{i-\frac{1}{2}}$, $\psi_{j-\frac{1}{2}}$, and $\psi_{k-\frac{1}{2}}$ are known, either from calculations in adjoining cells or boundary conditions, implying Eq. (6) is to be used to step through the mesh with μ , η , and ξ positive. For negative μ , say, we interchange $\psi_{i+\frac{1}{2}}$ and $\psi_{i-\frac{1}{2}}$ in Eq. (6), and similarly for negative η and ξ .

While arithmetically simple and globally second-order accurate, Eq. (6) does not guarantee that $\psi_{i+\frac{1}{2}}$, $\psi_{j+\frac{1}{2}}$, or $\psi_{k+\frac{1}{2}}$ are positive, as is well known. Thus THRETRAN may yield negative fluxes.

C. Angular Partitioning

The discrete direction $\vec{\Omega}_m$ is assumed to lie within an unspecified angular range $\Delta\Omega_m$, and we define a quadrature weight, w_m , so that

$$w_m = \int_{\Delta\Omega_m} d\Omega/4\pi \quad m = 1, 2, \dots, MT . \quad (8)$$

Associated with each interval $\Delta\Omega_m$ is a cosine set (μ_m, η_m, ξ_m) . Angular integrals are performed by sums. For example, the scalar flux is given by

$$\phi = \sum_{m=1}^{MT} w_m \psi_m . \quad (9)$$

In THRETRAN it is assumed that w_m and the sets (μ_m, η_m, ξ_m) are the same for each octant of the unit sphere, and only values for one octant are required as code input. Otherwise, there are no restrictions on the quadrature set except that

$$\sum_{m=1}^{MT} w_m = 1 \quad (10)$$

or that the input weights sum to one-eighth for the octant.

D. Solution Algorithm

1. Boundary Conditions. Only vacuum and reflective boundary conditions are allowed in THREETRAN. At a vacuum boundary ψ_m is set to zero for incoming directions, and at a reflective boundary ψ_m for incoming directions is set to the value of ψ_m in that outgoing direction corresponding to specular reflection. As a convention in THREETRAN we assume the solution domain is located in the first quadrant and label the system boundaries as shown in Fig. 1. Reflective boundary conditions are allowed on left, bottom, back, and front boundaries only. Vacuum conditions are allowed on all boundaries.

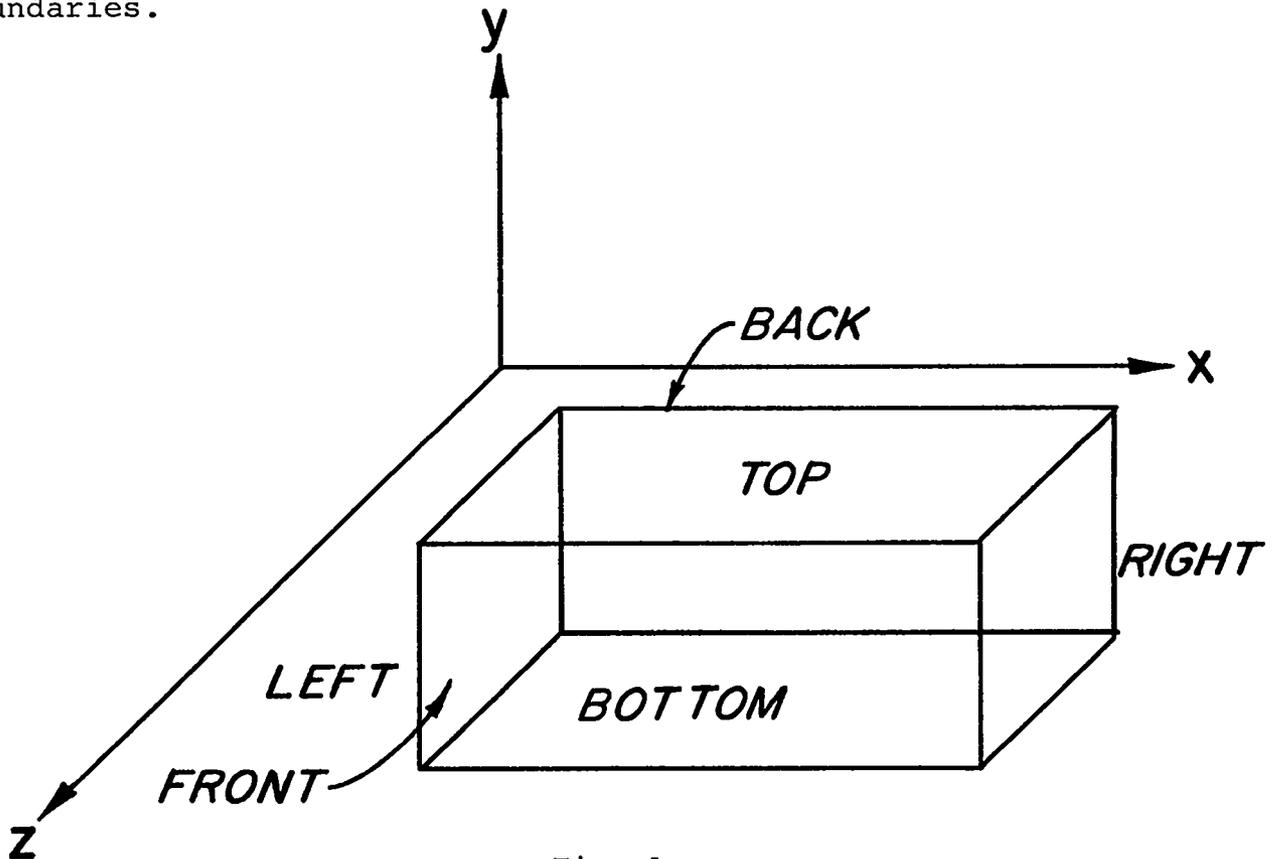


Fig. 1.
Names of system boundaries.

2. Progression Through the Space-Angle Mesh. THREETRAN uses the standard source iteration common to most discrete ordinates codes. It assumes S_m is known, makes one sweep through the entire space-angle mesh, and then recomputes those portions of S_m that depend on the newly calculated angular fluxes in the group being considered. The code is organized so that the geometric mesh is swept first for an octant of negative μ -directions (from right to left on an x-dimensional line) and then an octant of positive μ -directions (from left to right). This traverse is controlled by subroutine ISWEEP. A similar subroutine JSWEEP controls the traverse through each plane, first sweeping downward for negative η -directions and then upward for positive η -directions and calling ISWEEP for the traverse on each line. Finally, subroutine KSWEEP sweeps through the planes, first backward for all negative ξ -directions and then forward for positive ξ -directions. Subroutine KSWEEP calls JSWEEP for the traverse in each plane.

3. Convergence Acceleration. There are two convergence acceleration methods used in THREETRAN. The first of these is the updating of the source-term scalar flux after each quadrant of angular directions (each forward-backward sweep in ISWEEP) is calculated. This update is easy to execute, but requires storing the four separate components of the scalar flux. Note that because there are three currents required for linearly anisotropic scattering in (x,y,z) geometry, no additional storage would be required to treat P_1 scattering if the scalar flux and three currents were stored instead of four components of the scalar flux. At present THREETRAN is limited to an isotropic scattering source.

In addition to updating the source term, a spatially constant line rebalance factor is derived and used to scale all the fluxes associated with the forward-backward sweep for each quadrant of directions in ISWEEP. This rebalance differs from standard treatments such as that described in Ref. 3. To derive the rebalance factor we write Eq. (3) as

$$\begin{aligned} & \frac{\mu}{\Delta X} (\psi_{i+\frac{1}{2}} - \psi_{i-\frac{1}{2}}) + \frac{\eta}{\Delta Y} (\psi_{j+\frac{1}{2}} - \psi_{j-\frac{1}{2}}) + \frac{\xi}{\Delta Z} (\psi_{k+\frac{1}{2}} - \psi_{k-\frac{1}{2}}) + \sigma_t \psi \\ & = \sigma_s (\phi_1^p + \phi_2^p + \phi_3^p + \phi_4^p) + Q \end{aligned} \quad (11)$$

where σ_s is the macroscopic isotropic scattering cross section, Q is the isotropic source to the group, and ϕ_i , $i = 1, 2, 3, 4$, are the four components of the scalar flux. The superscript p indicates that these fluxes are obtained from a previous iteration. We now assume we are sweeping a line to generate a new value of ϕ_4 and that η and ξ are positive so that $\psi_{j-\frac{1}{2}}$ and $\psi_{k-\frac{1}{2}}$ can be assumed known from calculations on adjoining lines and planes. If we perform the weighted quadrant sum of Eq. (11) for the directions corresponding to ϕ_4 and also assume that the sum of the quadrant quadrature weights is one-fourth, then we can write the resulting equation as a pseudo-one-dimensional balance equation

$$\frac{I_{i+\frac{1}{2}} - I_{i-\frac{1}{2}}}{\Delta x} + \sigma_r \phi_4 = S \quad (12)$$

where the effective removal cross section is

$$\sigma_r = \sigma_t + (J_{j+\frac{1}{2}}/\Delta Y + K_{k+\frac{1}{2}}/\Delta Z)/\phi_4 \quad (13)$$

and the total source is

$$S_i = \sigma_s (\phi_1^p + \phi_2^p + \phi_3^p + \phi_4^p)/4 + Q/4 + J_{j-\frac{1}{2}}/\Delta Y + K_{k-\frac{1}{2}}/\Delta Z . \quad (14)$$

In these equations, I , J , and K are quadrant sums of μ , η , and ξ weighted angular fluxes. From Eq. (12) it is not difficult to show that the line rebalance factor is

$$f = (TS - \sum_i \Delta x_i \sigma_s \phi_4^p) / (TS - \sum_i \Delta x_i \sigma_s \phi_4) \quad (15)$$

with an integrated total source

$$TS = 4 \sum_i \Delta x_i S_i . \quad (16)$$

The above rebalance factor is simple to compute at the line-sweep level because TS can be accumulated before the sweep and the remaining terms are simple integrals of the previous and new scattering source. The calculation of the line rebalance factor is controlled by subroutine JSWEEP, where the angular integrals of $J_{j-\frac{1}{2}}$ and $K_{k-\frac{1}{2}}$ are performed by subroutine INFLOW. The rebalance factor, if positive, is applied to the new quadrant scalar flux and the angular fluxes $\psi_{k+\frac{1}{2}}$ and $\psi_{j+\frac{1}{2}}$ by subroutine SCALE. If the iteration converges, it is clear from Eq. (15) that f approaches unity.

There is no convergence acceleration of the outer iteration, which consists of a simple updating of the fission source.

4. Convergence Tests. A measure of the pointwise error that is readily calculated from available quantities is the ratio, for quadrant q ,

$$\epsilon_i = (S_i - \sigma_s \phi_{qi}^k) / (S_i - \sigma_s \phi_{qi}) \quad (17)$$

and the maximum of the absolute value of ϵ_i for the entire traverse through the spatial mesh is required to be less than the input value EPS before inner iteration is terminated. Even if the iteration does not converge, no more than an input value, LIMIT, inner iterations are made in one group.

For outer iterations, the ratio

$$\lambda = (TF + TQ) / (TFP + TQ) \quad (18)$$

is required to differ less than EPS from unity before stopping iteration. In Eq. (18) TQ is the volume integral of the isotropic inhomogeneous source, if there is one, TF is the volume integral of the fission source, and TFP is the same integral for the previous outer iteration. If there is no fission source, this criterion will

stop iteration after the first group traverse even if there is up-scattering. Thus if an upscattering capability were added to the code (a minor modification), this convergence test would have to be changed.

IV. PROGRAM OPTIONS AND INPUT DESCRIPTION

A. Program Options

1. Types of Problems. Only three kinds of problems are run by THRETRAN. Inhomogeneous source problems, with or without fission, and k_{eff} problems can be run, but no eigenvalue searches are performed.

2. Source and Flux Inputs. No flux input is required. If a k_{eff} problem is run, a constant, normalized fission source is used to begin the problem.

The only source input option presently allowed forms the source as the product of four shapes, one for the energy, and one for each of the spatial directions. Addition of the reading of a complete source, say from a standard interface format, would not be difficult.

3. Geometry Specification. The (x,y,z) system is assumed to be partitioned into IMxJMxKM coarse-mesh zones. The user enters coarse-mesh boundary dimensions and the number of equally spaced fine-mesh intervals he desires for each coarse-mesh interval in each dimension. That is, he specifies IM numbers to define the x-dimension fine-mesh spacing, JM numbers to specify the y-dimension fine-mesh spacing, and KM numbers to specify the z-dimension spacing. He also enters KM sets of IMxJM cross-section identification numbers. These numbers identify which cross section is in each coarse-mesh zone. For each z-dimension coarse-mesh zone, THRETRAN prints a schematic diagram showing boundary conditions, boundary dimensions, fine-mesh interval numbers, and cross-section identification numbers.

4. Cross Sections. Only macroscopic cross sections, in Los Alamos standard format (Ref. 3, p. 16) are allowed as input, although other formats could be added. There is no pre- or postprocessing of cross sections performed by the code. To run an adjoint

calculation, the adjoint reversals of the cross section would have to be performed separately.

5. Dumps. The program takes periodic, time limit, and restart dumps, and a problem restart may be performed from any of these dumps. The user may change any control variable upon restart, but some variables must not be changed or the restart will not function properly. To restart a problem, the user simply resubmits his input deck (after ensuring that the restart dump tape from unit seven has been mounted) with the control parameter ISTART set equal to one (time-limit or periodic dump) or two (final dump). The only other control parameters that it is meaningful to change are:

1. any boundary condition
2. IOUT, the output option
3. LIMIT
4. EPS
5. DTIME, time after which dump is taken and problem halted.

6. Edits. No edits are performed by THREEETRAN. Selected problem scalar fluxes are printed or microfiched, and a standard⁴ interface flux file is written. The standard had to be rewritten to allow blocking of three-dimensional arrays. Previously the standard seemed to imply that the whole array being written had to be core (SCM) contained.

B. Description of Problem Input

1. Input Formats. Except for the control parameters, cross sections, and edit parameters, all floating-point numbers and integers are read into THREEETRAN 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 end with a 3 in the I1 field after the last word of the block. The available options are given in Table I.

2. Input Data. On the following pages are listed the problem data required for input to THREEETRAN. The formats for each entry are given, and the special formats described in the preceding Section are denoted by S(E) for floating-point numbers and S(I) for integers.

<u>Number of Word on Card</u>	<u>Name of Variable</u>	<u>Comments</u>
CONTROL INTEGERS (6I12) - - - - -		Card 1
1	ITCARD	Specifies number of title cards to be read. ITCARD may be zero.
TITLE CARDS (18A4) - - - - -		
all words	TITLE(18)	Each card is read and printed immediately.
CONTROL INTEGERS (6I12) - - - - -		Card 2
1	M1	Number of discrete ordinates directions per octant.
2	IGM	Number of energy groups.
3	IM	Number of x-direction coarse-mesh zones.
4	JM	Number of y-direction coarse-mesh zones.
5	KM	Number of z-direction coarse-mesh zones.
6	IBL	Left boundary condition 0/1 vacuum/reflective.
CONTROL INTEGERS (6I12) - - - - -		Card 3
1	IBR	Right boundary condition vacuum (0) only.
2	IBB	Bottom boundary condition 0/1 vacuum/reflective.
3	IBT	Top boundary condition vacuum (0) only.
4	IBF	Front boundary condition 0/1 vacuum/reflective.
5	IBA	Back boundary condition 0/1 vacuum/reflective.
6	IEVT	Problem type 0/1/2 Inhomogeneous source/k-effective/source plus fission
CONTROL INTEGERS (6I12) - - - - -		Card 4
1	ISTART	0/1/2 Initial problem/restart from periodic or time limit dump/restart from final dump.

<u>Number of Word on Card</u>	<u>Name of Variable</u>	<u>Comments</u>
2	IQOPT	0/1 Source input option. No source/ energy and spatial shapes.
3	MT	Total number of macroscopic cross sections.
4	IHT	Row of total cross section in cross- section table.
5	IHM	Last row of cross-section table.
6	IOUT	Output option 0/1 print or/microfiche selected scalar fluxes.
CONTROL INTEGERS (6I12) - - - - - Card 5		
1	LIMIT	Maximum number of inner iterations al- lowed in each group.
CONTROL FLOATING-POINT NUMBERS (6E12.6) - - - - - Card 6		
1	EPS	Convergence precision.
2	NORM	Normalization factor. Volume integral of inhomogeneous source is made equal to NORM when IEVT = 0 or 2. If IEVT = 1, volume integral of fission source is made equal to NORM. If NORM = 0, it is not used.
3	DTIME	After DTIME seconds a problem dump is taken and execution is halted.

REMAINING DATA IN ORDER OF INPUT

<u>Block Name and Dimension</u>	<u>Format</u>	<u>Number of Entries</u>	<u>Comments</u>
IX(IM)	S(I)	IM	Number of equally spaced fine- mesh intervals for each x- dimension coarse-mesh zone.
JY(JM)	S(I)	JM	Number of equally spaced fine- mesh intervals for each y- dimension coarse-mesh zone.
KZ(KM)	S(I)	KM	Number of equally spaced fine- mesh intervals for each z- dimension coarse-mesh zone.
XB(IM+1)	S(E)	IM+1	Boundaries of x-dimension coarse-mesh zones.
YB(JM+1)	S(E)	IM+1	Boundaries of y-dimension coarse-mesh zones.
ZB(KM+1)	S(F)	IM+1	Boundaries of z-dimension coarse-mesh zones.

REMAINING DATA IN ORDER OF INPUT

Block Name and Dimension	Format	Number of Entries	Comments
WGT (M1)	S (E)	M1	Quadrature weights for one octant.
UMU (M1)	S (E)	M1	x-dimension direction cosine μ
ETA (M1)	S (E)	M1	y-dimension direction cosine η
ZI (M1)	S (E)	M1	z-dimension direction cosine ξ
NDSCAT (IGM, IGM)	S (I)	IGM*IGM	Matrix indicating for each group whether (entry = 1) or not (entry = 0) downscatter is possible from other groups. Used to avoid calculation of scattering source when scattering not possible or user knows scattering cross section is zero for all nuclides. Enter IGM numbers sequentially for first group, then for second group, etc. Self-scatter is a zero entry. For a two-group problem the array entries would be 0, 0, 1, 0. The one for the second group indicates scattering from group one to group two.
CHI (IGM)	S (E)	IGM	Fission spectrum.
IDCS (IM, JM)	S (I)	IM*JM*KM	Integers indicating which cross section is in each coarse-mesh zone. Enter IM*JM numbers in KM separate batches, each batch followed by a terminator. Cross sections are identified by the order in which they are input with ID numbers ranging from 1 to MT.
Cross Sections C (IHM, IGM)	6E12.5	IHM*IGM*MT	Each cross section is preceded by a header card read with an 18A4 format. Then the block of IHM*IGM cross sections is read continuously ((C(I,IG),I=1,IHM)IG=1,IGM). Cross sections are identified by the order in which they are input with ID numbers ranging from 1 to MT.

<u>Block Name and Dimension</u>	<u>REMAINING DATA IN ORDER OF INPUT</u>		
	<u>Format</u>	<u>Number of Entries</u>	<u>Comments</u>
Temporary	S(E)	IGM	Inhomogeneous source energy spectrum. Do not enter if IEVT = 1.
SX(IT)	S(E)	IT	Inhomogeneous source x-dimension shape. IT = number fine-mesh intervals in the x-dimension. Do not enter if IEVT = 1.
SY(JT)	S(E)	JT	Inhomogeneous source y-dimension shape. JT = number fine-mesh intervals in the y-dimension. Do not enter if IEVT = 1.
SZ(KT)	S(E)	KT	Inhomogeneous source z-dimension shape. KT = number fine-mesh intervals in the z-dimension. Do not enter if IEVT = 1.

V. PROGRAMMING INFORMATION

In this section we describe the overall organization of THREETRAN and define the function of its subroutines. We then concentrate on descriptions of the problem data storage and transfer algorithms which are the most complicated aspect of the code.

A. Role and Function of Subprograms

Figure 2 shows the flow of a problem through the major subroutines of THREETRAN. The main program sets the disk unit numbers, the assumed size of SCM and LCM, and the time in seconds between each periodic restart dump. Subroutine INITAL reads the problem title cards, the control data, and the number of fine-mesh intervals. It then computes the partitioning of storage between SCM, LCM, and disk file and assigns storage pointers for SCM and LCM. It also initializes the random access disk file request tables and zeroes LCM and the flux disk file if one is needed.

Subroutine INANDI reads the remaining problem input; calls GEOM to compute needed geometric functions; calls READQ to read

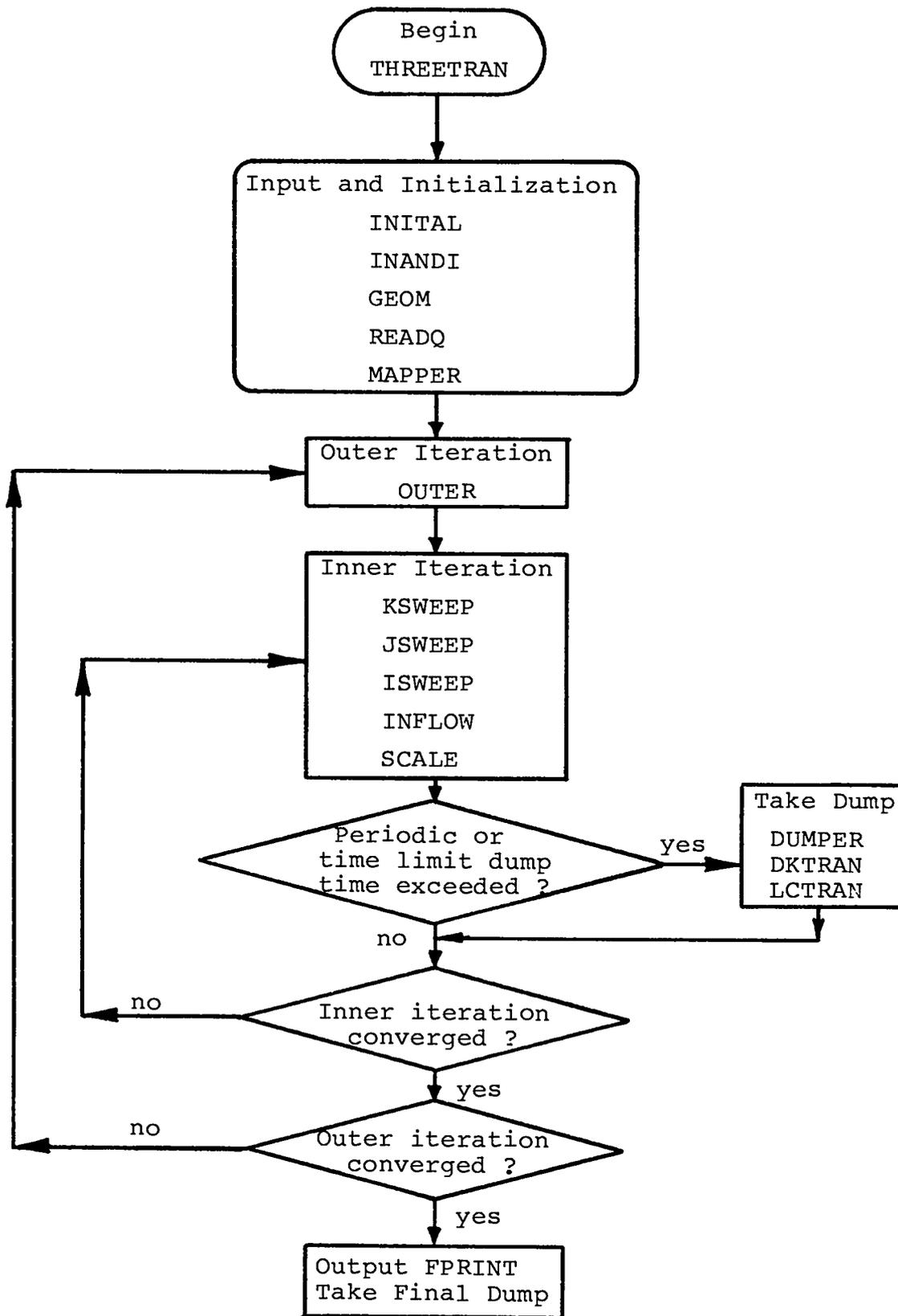


Fig. 2.
Overall program flow in THREETRAN.

inhomogeneous source data, initialize it, and store it; and uses MAPPER to draw a schematic representation of the system.

Subroutine OUTER initializes the fission source if there is one, calculates the source to a group, controls the inner iteration within a group, calculates a new fission source, and controls the iterations through the groups.

Within the sweeps through the planes made during inner iteration, the clock time is checked to see if a periodic or time limit restart dump should be taken. If so, DUMPER writes a dump on unit NDUMP using LCTRAN to write data from LCM through SCM to NDUMP and DKTRAN to transfer data from random access disk through LCM to SCM to NDUMP.

After convergence of the outer iteration, subroutine FPRINT is used to print or microfiche selected fluxes and to create an interface flux file on unit NTAPE.

During the problem execution, several service subroutines are used and their functions are listed in Table II.

TABLE II
FUNCTION OF THREETRAN SERVICE ROUTINES

<u>Name of Subroutine</u>	<u>Function</u>
REED	Transfers data from LCM to SCM or from disk to LCM
RITE	Transfers data from SCM to LCM or from LCM to disk
SECTOR	Calculates the number of 512 word sectors required on disk units
CLEAR	Stores a number in an array
ERROR	Writes error messages
LOAD	Reads problem input data
KADD	Used to switch LCM addresses for overlapped data transfer
DONE	Tests to see if a disk unit has finished data transmission

B. Data Storage

THREETRAN uses a single common container array, labeled A, to store all SCM data compactly. The length of this array is set in the main calling program, presently to 40 000, and the first action of the code is to zero the 40 000 words of SCM storage.

The general structure of the container array is shown in Fig. 3. The container array IA, used for control pointers and general problem data, is equivalenced to A(1) and extends to IAEND (currently equal to 220). From IAEND to LIX are stored disk request table data which must remain undisturbed during problem dump taking, and from LIX to LAST are stored general problem data.

A similar arrangement is used for storage in LCM with addresses running from 1 to KLAST with KLAST less than MAXLCM which is currently 370 000 words.

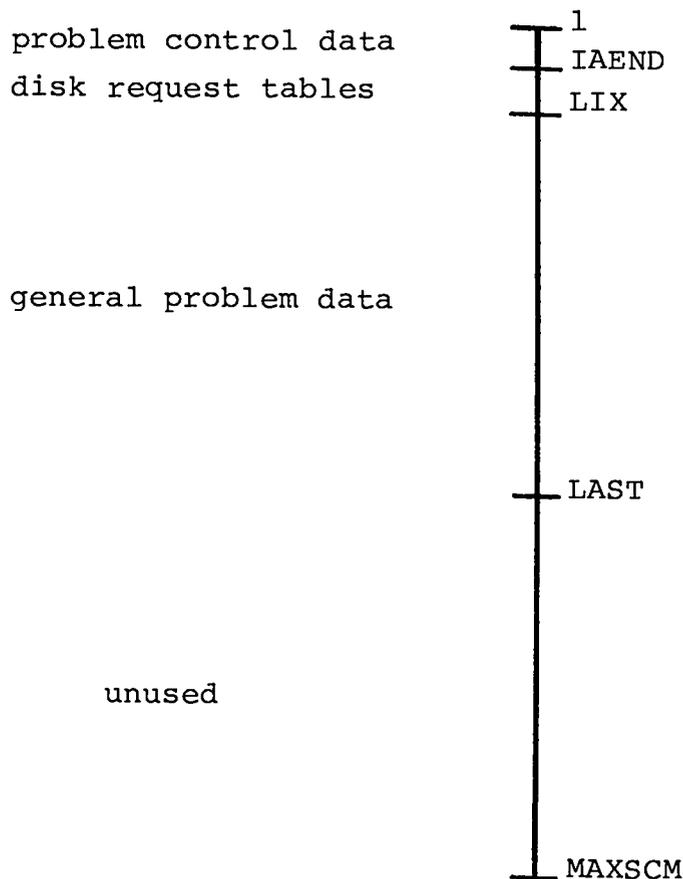


Fig. 3.
Structure of A array.

The contents of the IA position of the A array are defined in Table III. If there is no entry for some word in the table, the word is not used. A study of Table III conveys much general information about the THREETRAN program.

C. Data Management Algorithms

The numerical schemes used in THREETRAN are elementary; the data management techniques are not. That is not to say that data transmission is highly optimized in the code. Indeed, because of the complexity of the problem, simplicity was given equal weight with efficiency when decisions were made.

THREETRAN is organized to run as large problems as possible by fitting data within the three hierarchies of storage: SCM, LCM, and disk. For bulk transfers LCM is essentially as fast a memory as SCM so that the strategy chosen attempts to keep as much of the time-consuming parts of the calculation as possible within SCM and LCM and to use disk storage as a last resort. Similarly, if disk storage is used, an attempt is made to overlap data transmission with problem execution. Nevertheless, in large problems THREETRAN is data-transmission limited, particularly in the calculation of the source to the group where masses of data must be moved to perform trivial arithmetic. In this area we have tried to simplify the calculation by skipping the computation of the fission source contribution whenever the fission spectrum value for a group is zero and by skipping the computation of the scattering source whenever scattering from other groups to the group at hand is not possible. The latter decision is facilitated by having the user provide an array, NDSCAT, to define the possibilities.

In the following sections we define the data transmission algorithms in THREETRAN and describe their operation.

1. LCM and Disk Data Transmission. Two subroutines, REED and RITE, are used to transmit data in THREETRAN. Subroutine REED moves data from LCM to SCM or from random access disk file to LCM. Subroutine RITE transfers data in the opposite direction. Both use system-dependent routines. The LCM-SCM transfers are made from a starting address in LCM to a starting address in SCM in a block of

TABLE III
 CONTENTS OF COMMON BLOCK IA WITHIN A ARRAY

<u>Position</u>	<u>Name</u>	<u>Pointer for Array^a</u>	<u>Remarks</u>
1	M1		Number of directions
2	IGM		Number of groups
3	IM		Number of coarse-mesh intervals in x-direction
4	JM		Number of coarse-mesh intervals in y-direction
5	KM		Number of coarse-mesh intervals in z-direction
6	IBL		Left boundary condition indicator
7	IBR		Right boundary condition indicator
8	IBB		Bottom boundary condition indicator
9	IBT		Top boundary condition indicator
10	IBF		Front boundary condition indicator
11	IBA		Back boundary condition indicator
12	IEVT		Problem type indicator
13	ISTART		Starting option indicator
14	IQOPT		Inhomogeneous source option indicator
15	MT		Total number of macroscopic materials
16	IHT		Position (row) of total cross section in cross-section table
17	IHM		Length of cross-section table
18	IOUT		Output indicator
19	LIMIT		Maximum number of inner iterations per group
20			

^aPointers for SCM arrays begin with L; pointers for LCM arrays begin with K.

TABLE III (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for Array^a</u>	<u>Remarks</u>
21			
22			
23			
24			
25	EPS		Problem convergence precision
26	NORM		Problem normalization factor (floating point)
27	DTIME		Time after which problem dump is taken and problem halted
28			
29			
30			
31	M2		M2 = 2 x M1
32	IHS		Position (row) of scattering cross section in cross-section table
33	IHF		Position (row) of fission cross section in cross-section table
34	ITP		ITP = IT + 1
35	JTP		JTP = JT + 1
36	KTP		KTP = KT + 1
37	IT		Number of x-direction fine-mesh intervals = $\sum_{i=1}^{IM} IX_i$

^aPointers for SCM arrays begin with L; pointers for LCM arrays begin with K.

TABLE III (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for Array^a</u>	<u>Remarks</u>
38	JT		Number of y-direction fine-mesh intervals = $\sum_{j=1}^{JM} JY_j$
39	KT		Number of z-direction fine-mesh intervals = $\sum_{k=1}^{KM} KZ_k$
40	IG		Group index in OUTER
41	LIX	IX(IM)	Number of fine-mesh intervals in each x-direction coarse-mesh zone
42	LJY	JY(JM)	Number of fine-mesh intervals in each y-direction coarse-mesh zone
43	LKZ	KZ(KM)	Number of fine-mesh intervals in each z-direction coarse-mesh zone
44	LXB	XB(IM+1)	x-direction coarse-mesh boundaries
45	LYB	YB(JM+1)	y-direction coarse-mesh boundaries
46	LZB	ZB(KM+1)	z-direction coarse-mesh boundaries
47	LDX	DX(IT)	ΔX_i
48	LDY	DY(JT)	ΔY_j
49	LDZ	DZ(KT)	ΔZ_k
50	LRDX	RDX(IT)	$2/\Delta X_i$
51	LRDY	RDY(JT)	$2/\Delta Y_j$
52	LRDZ	RDZ(KT)	$2/\Delta Z_k$

25 ^aPointers for SCM arrays begin with L; pointers for LCM arrays begin with K.

TABLE III (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for Array^a</u>	<u>Remarks</u>
53	LIDX	IDX(IT)	Integers indicating which coarse-mesh zone each fine-mesh x-dimension interval belongs to. Array is preceded and followed by zeroes in memory
54	LIDY	IDY(JT)	Integers indicating which coarse-mesh zone each fine-mesh y-dimension interval belongs to. Array is preceded and followed by zeroes in memory
55	LIDZ	IDZ(KT)	Integers indicating which coarse-mesh zone each fine-mesh z-dimension interval belongs to. Array is preceded and followed by zeroes in memory
56	LE	E(IT)	Error at each space point during ISWEEP
57			
58			
59			
60			
61	LND	NDSCAT(IGM,IGM)	Array indicating whether or not downscatter to a group is possible
62	LC	C(IHM,MT)xIGM	Macroscopic cross sections rearranged to this configuration as they are loaded
63	LCT	CT(IT)	Total cross section for use in ISWEEP
64	LSI	SI(IT)	Total source for use in ISWEEP
65	LIDCS	IDCS(IM,JM)xKM	Cross-section zone identification numbers
66	LCHI	CHI(IGM)	Fission spectrum
67			

^aPointers for SCM arrays begin with L; pointers for LCM arrays begin with K.

TABLE III (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for array^a</u>	<u>Remarks</u>
68			
69			
70			
71	LWGT	WGT (M1)	Quadrature weight
72	LMU	UMU (M1)	x-direction cosine
73	LETA	ETA (M1)	y-direction cosine
74	LZI	ZI (M1)	z-direction cosine
75	LUY	UY (M1)	$2\eta_m/\Delta Y_j$ calculated in JSWEEP for use in ISWEEP
76	LUZ	UZ (M1)	$2\xi_m/\Delta Z_k$ calculated in KSWEEP for use in JSWEEP and ISWEEP
77			
78			
79			
80			
81	LJIN	JINDEX (NJBLOC)	Number of lines in each j (y-dimension) block
82	LKIN	KINDEX (NKBLOC)	Number of planes in each k (z-dimension) block
83	LGIN	IGNDEX (NGBLOC)	Number of groups in each g (energy-group) block
84			
85			
86	LBX	BX (M2)	Boundary angular flux array (x-dimension) used in ISWEEP
87	LBY	BY (M2, IT)	Boundary angular flux array (y-dimension) used in JSWEEP

27 ^aPointers for SCM arrays begin with L; pointers for LCM arrays begin with K.

TABLE III (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for array^a</u>	<u>Remarks</u>
88	LBZ1	BZ1 (M2,IT,JBLOC)	Boundary angular flux array (z-dimension, $\eta < 0$) used in KSWEEP
89	LBZ2	BZ2 (M2,IT,JBLOC)	Boundary angular flux array (z-dimension, $\eta > 0$) used in KSWEEP
90			
91	LF	F (IT,JBLOC)	Fission source array
92	LS	S (IT,JBLOC)	Source-to-the-group array
93	LPH1	PH1 (IT,JBLOC) xIGM	Integral of angular flux for directions $\eta < 0$, $\xi < 0$
94	LPH2	PH2 (IT,JBLOC) xIGM	Integral of angular flux for directions $\eta > 0$, $\xi < 0$
95	LPH3	PH3 (IT,JBLOC) xIGM	Integral of angular flux for directions $\eta < 0$, $\xi > 0$
96	LPH4	PH4 (IT,JBLOC) xIGM	Integral of angular flux for directions $\eta > 0$, $\xi > 0$
97	LQ	Q (IT,JBLOC) xIGM	Inhomogeneous source array (not required if IEVT = 1)
98	LSX	SX (IT)	Inhomogeneous source x-dimension shape (not required if IEVT = 1)
99	LSY	SY (JT)	Inhomogeneous source y-dimension shape (not required if IEVT = 1)
100	LSZ	SZ (KT)	Inhomogeneous source z-dimension shape (not required if IEVT = 1)
101	JBLOC		Size (number of lines) of j blocks
102	NJBLOC		Number of j blocks

^aPointers for SCM arrays begin with L; pointers for LCM arrays begin with K.

TABLE III (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for array^a</u>	<u>Remarks</u>
103	KBLOC		Size (number of planes) of k blocks
104	NKBLOC		Number of k blocks
105	KC		LCM address of cross-section storage
106	KIDCS		LCM address of cross-section identification numbers
107	KBZ1		LCM address of boundary angular fluxes BZ1
108	KBZ2		LCM address of boundary angular fluxes BZ2
109	KF		Indicator for LCM address of fission array KF = KFA or KFB
110	KS		Indicator for LCM address of source array KS = KSA or KSB
111	KPH1		Indicator for LCM address of flux array KPH1 = KPH1A or KPH1B
112	KPH2		Indicator for LCM address of flux array KPH2 = KPH2A or KPH2B
113	KPH3		Indicator for LCM address of flux array KPH3 = KPH3A or KPH3B
114	KPH4		Indicator for LCM address of flux array KPH4 = KPH4A or KPH4B
115	KQ		LCM address for inhomogeneous source array
116	KS1		LCM flux pointer generated in KSWEEP and used in JSWEEP
117	KS2		LCM flux pointer generated in KSWEEP and used in JSWEEP
118	KS3		LCM flux pointer generated in KSWEEP and used in JSWEEP

^aPointers for SCM arrays begin with L; pointers for LCM arrays begin with K.

TABLE III (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for array^a</u>	<u>Remarks</u>
119	IGBLOC		Size (number of groups) of g blocks
120	NGBLOC		Number of g blocks
121	KBZWDS		Number of LCM words in BZ1 array
122	KBZ2WD		KBZ2WD = 2X KBZWDS
123	KSWDS		Number of LCM words in S array
124	KPHWDS		Number of LCM words in PH1, PH2, PH3, or PH4 array
125	KPH2WD		KPH2WD = 2X KPHWDS
126	KPH4WD		KPH4WD = 4X KPHWDS
127	KIDWDS		Number of LCM words in IDCS array
128	KCWDS		Number of LCM words in cross-section array
129	KFWDS		Number of LCM words in fission array
130	KQWDS		Number of LCM words in inhomogeneous source array
131	KS4		LCM flux pointer generated in KSWEEP and used in JSWEEP
132	KFN		Not used
133			
134			
135			
136	NPHWDS		Number of disk words for one flux component
137	NPH2WD		NPH2WD = 2X NPHWDS
138	NPH4WD		NPH4WD = 4X NPHWDS

^aPointers for SCM arrays begin with L; pointers for LCM arrays begin with K.

TABLE III (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for array^a</u>	<u>Remarks</u>
139	NSWDS		Number of disk words for source-to-group
140	NQWDS		Number of disk words for inhomogeneous source
141	NFWDS		Number of disk words for fission source
142	NSECQ		Number of 512 word sectors on inhomogeneous source disk
143	NSECP		Number of 512 word sectors on flux disk
144	NSECS		Number of 512 word sectors on source-to-group disk
145	NSECF		Number of 512 word sectors on fission source disk
146	IJ		$IJ = IT*JBLOC$
147	IJN		$IJN = IJ*NJBLOC$
148	IJNK		$IJNK = IJN*KBLOC$
149	KSS		LCM indicator used in OUTER for inhomogeneous source
150	IG2		Index used in loop within each group block
151	MAXSCM		Size of SCM
152	MAXLCM		Size of LCM
153	NINP		Unit number of input unit
154	NOUT		Unit number of output unit
155	NSOUR		Unit number of source-to-group unit
156	NQ		Unit number of inhomogeneous source unit
157	NF		Unit number of first fission source unit

^aPointers for SCM arrays begin with L; pointers for LCM arrays begin with K.

TABLE III (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for array^a</u>	<u>Remarks</u>
158	NFN		Unit number of second fission source unit
159	NPHI		Unit number of flux unit
160	NC		Not used (to be used for cross-section file)
161	NREAD		Indicator for storage mode 0/1/2 all LCM/flux and inhomogeneous source on disk/flux, inhomogeneous source, source-to-group and fission source on disk
162	NTAPE		Unit number of final flux output
163	NFICHE		Unit number for microfiche output
164	NDUMP		Unit number for problem dumps
165			
166			
167			
168			
169	DUMPT		Time since last periodic dump or since problem began (seconds)
170	OINTO		Outer iteration number
171	EV		Current value of eigenvalue
172	TF		Volume integral of fission source
173	TQ		Volume integral of inhomogeneous source
174	AIROR		Error in inner iteration
175	ALA		Ratio of TF + TQ to TFP + TQ
176	TIME		Cumulative elapsed problem time in seconds

^aPointers for SCM arrays begin with L; pointers for LCM arrays begin with K.

TABLE III (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for array^a</u>	<u>Remarks</u>
177	TFP		Volume integral of fission source from previous outer iteration
178	REV		1/EV
179	EVA		Value of EV from previous outer iteration
180	LRTQ	RTQ(20)	Disk request table for inhomogeneous source disk
181	LRTS	RTS(20)	Disk request table for source-to-group disk
182	LRTF	RTF(20)	Disk request table for first fission source disk
183	L RTP	RPT(20)	Disk request table for flux disk
184	LRTNF	RTNF(20)	Disk request table for second (or new) fission source disk
185	KFA		LCM address for first block of fission source to be read
186	KFB		LCM address for second block of fission source to be read
187	KSA		LCM address for first block of source-to-group to be read
188	KSB		LCM address for second block of source-to-group to be read
189	KP1A		LCM address for first block of flux ϕ_1 to be read
190	KP2A		LCM address for first block of flux ϕ_2 to be read

33 ^aPointers for SCM arrays begin with L; pointers for LCM arrays begin with K.

TABLE III (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for array^a</u>	<u>Remarks</u>
191	KP3A		LCM address for first block of flux ϕ_3 to be read
192	KP4A		LCM address for first block of flux ϕ_4 to be read
193	KP1B		LCM address for second block of flux ϕ_1 to be read
194	KP2B		LCM address for second block of flux ϕ_2 to be read
195	KP3B		LCM address for second block of flux ϕ_3 to be read
196	KP4B		LCM address for second block of flux ϕ_4 to be read
197	PTIME		Time between periodic dumps
198	IGMAX		Number of groups in a g block
199	IG1		Index for loop over group blocks
200	IFB		After a restart from a periodic or time limit dump, IFB switches problem execution to a forward or backward sweep in KSWEEP
201	IITNO		Inner iteration number
202	K		Plane index
203	KMAX		Maximum number of planes in a k block
204	KMAXP		KMAXP = KMAX + 1
205	K1		Index used in loop over k blocks
206	K2		Index used in loop within k block
207	K3		Index used in loop within k block

^aPointers for SCM arrays begin with L; pointers for LCM arrays begin with K.

TABLE III (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for array^a</u>	<u>Remarks</u>
208	NSTART		Flux sector indicator used in KSWEEP
209	L		Index used in loop over k blocks
210	LAST		Address of last word in SCM
211	KLAST		Address of last word in LCM
212	LSTART		Address of first word in SCM to be used in dump data transfers
213	KSTART		Address of first word in LCM to be used in dump data transfers
214	NSECT		Flux unit sector address used in KSWEEP
215	IAEND		End of IA block within A array
216			
217			
218			
219			
220			

^aPointers for SCM arrays begin with L; pointers for LCM arrays begin with K.

arbitrary length. The disk-LCM transfers are made from the beginning of an arbitrary 512-word sector on a specified unit to a beginning address in LCM in a block of arbitrary length. Two disk options are allowed in both REED and RITE. One option insists that the disk unit be finished with prior transmission before beginning the desired transmission, but does not wait for the transmission to be finished before returning to problem execution. The other option does not check to ensure transmission completion before beginning, but does check for completion before returning to problem execution.

All of the random disk transmissions in THREETRAN are made using REED and RITE. However, sequential disk transmissions are made in FPRINT to write a microfiche file and a standard interface file and in DUMPER and INITAL in connection with reading and writing a dump file. A detailed definition of the argument lists in REED and RITE is given in the program listing.

2. Data Storage and Data Blocking. The general strategy of THREETRAN is to divide the range of the variables JT, KT, and IGM into blocks small enough to fit the problem into available storage. That is, JT, which defines the number of lines in a plane, is divided into NJBLOC blocks of JBLOC lines each. Then, the dimension of arrays associated with lines and stored in SCM is limited to JBLOC words rather than JT words. Because JT may not be evenly divisible into JBLOC words, we use an array, JINDEX(NJBLOC), to indicate the number of lines in each block. The first NJBLOC-1 of these blocks have JBLOC lines and the last block has \leq JBLOC lines. For example, suppose JT = 47 and JBLOC = 8. Then NJBLOC = 6, and JINDEX consists of 5 entries of 8 and a last entry of 7.

Similar blocking is performed for planes and groups. Figure 4 summarizes the variables involved.

The algorithm for determining block size proceeds as follows. All the arrays listed in Table III are stored in SCM except for two boundary angular flux arrays (BZ1 and BZ2), four flux arrays (PH1, PH2, PH3, and PH4), a fission source array (F), a source-to-the-group array (S), and if IEVT \neq 1, an inhomogeneous source array (Q). Allowing for the IA array space in the container, this storage requires LNOW words. Then JBLOC is computed from

Lines
JT = 13

Blocks
NJBLOC = 3
JBLOC = 5

Value of
JINDEX

Planes
KT = 9

Blocks
NRBLOC = 5
KBLOC = 2

Value of
KINDEX



5

5

3



2

2

2

2

1

Groups
IGM = 11

Blocks
NGBLOC = 4
IGBLOC = 3

Value of
IGNDEX



3

3

3

2

Fig. 4.
Examples of blocking parameters for lines, planes, and groups.

$$JBLOC = (MAXSCM - LNOW) / ((L + 2*M2) * IT) \quad (19)$$

where $L = 6$ unless a Q source is present and then $L = 7$. If this number is larger than or equal JT , then all storage will fit in LCM and no blocking is necessary, i.e., $NJBLOC = 1$ and $JINDEX(1) = JT$. If $JBLOC$ is negative or zero, the problem will not fit in LCM and is aborted. Otherwise, the appropriate number of blocks is determined, values of $JINDEX$ are calculated, and the storage for the blocked arrays is allocated in SCM as though each array had $JBLOC$ lines. A total value of LCM storage is calculated and printed as output.

A similar algorithm is used to determine the number of planes and groups that fit in LCM. All of the cross sections (C), zone identification numbers ($IDCS$), and boundary angular fluxes ($BZ1$ and $BZ2$) are placed in LCM. This requires $KNOW$ words of storage. Then, for $L = 4$ or 5 depending on whether or not there is an inhomogeneous source, $KBLOC$ is determined from

$$KBLOC = \frac{MAXLCM - KNOW}{IJN * (L+3)} \quad (20)$$

$$IJN = IT * NJBLOC * JBLOC .$$

At this point, if $KBLOC \geq KT$, then all planes and at least one group will fit in LCM and more groups may also fit. The next step is to calculate how many more groups will fit from

$$IGBLOC = 1 + \frac{MAXLCM - KNOW - IJNK * (L+3)}{IJNK * L} \quad (21)$$

$$IJNK = IJN * KT$$

and to assign values for $NGBLOC$ and $IGNDEX$. In this situation $NKBLOC = 1$ and $KINDEX(1) = KT$. However, if $KBLOC$ from Eq. (20) is less than KT then $KBLOC$ is recomputed from

$$KBLOC = \frac{MAXLCM - KNOW}{IJN*12} . \quad (22)$$

The extra LCM storage space required is to allow two areas in LCM for each array for staged data transmission. For example, fluxes are read from disk into one area while fluxes in the other area are being used for computation. Such staged transmission is performed only if the planes are blocked.

After KBLOC is calculated, values of NKBLOC and KINDEX are computed and the rest of LCM storage is allocated. In this situation there are IGM group blocks and IGBLOC = 1. If KBLOC from Eq. (22) is less than or equal zero, the problem will not fit in LCM and is aborted. In practice the controlling criterion is most likely to be available computing times rather than storage space. For example, a five-group, S_4 , 100 x 100 x 100 problem will fit within 40 000 words of SCM and 400 000 words of LCM.

The above algorithms define three distinct storage modes. These are described in Table IV. In the problem output a storage description is given and the value of NREAD is printed.

After storage allocation is complete, the required LCM storage is zeroed, and the number of disk sectors for each of the random disk units is computed. After initialization of random disk request tables, and if NREAD > 0, the flux disk file is initialized to zero.

With data blocking, normal do loops over an index are replaced by two loops. For example, the loop

TABLE IV
THREETRAN STORAGE MODES

Value of Mode Indicator NREAD	Description of Mode
0	Problem contained entirely within LCM. No disk files used.
1	All planes fit within LCM. Groups are blocked. Disks used for group-dependent data but no staged data transmission.
2	Planes and groups are blocked. Disk data transmission is staged.

```

DO      100      K = 1, KT
.
.
.
100 Continue

```

is replaced by

```

K = 0
DO      200      K1 = 1, NKBLOC
KMAX = KINDEX(K1)
.
.
DO      100      K2 = 1, KMAX
.
.
K = K + 1
100 Continue
200 Continue

```

with the separate K index being calculated to address arrays that are not blocked. If disk transfers are required, they are made outside the interior loop, either before or after. Similar loops are made for lines or groups as needed, but the only data transmission required in loops over lines is to or from LCM from or to SCM.

3. Disk Assignment. The disk units assigned for THREETRAN and their use are given in Table V. Units 1 through 5 are random access files for which system disk request tables are established at the start of a problem. No unit is opened unless the problem actually needs the unit.

4. Staged Data Transmission. If NREAD = 2 and planes are blocked, then simultaneous data transfer and computation are allowed. A similar algorithm is used in all cases, but an example from KSWEET is illustrative. There, inside the K1 loop, flux and source-to-the-group data are read before the K2 loop (refer to the sample do loops given in Sec. 2 above), and new fluxes are written after

TABLE V
DISK ASSIGNMENT IN THREETRAN

<u>File Name</u>	<u>File Number</u>	<u>Use</u>
NINP	10	System Input
NOUT	9	System Output
NSOUR	1	Source to a group
NQ	2	Inhomogeneous Source
NF	3	Fission Source
NFN	4	New Fission Source
NPFI	5	Flux
NTAPE	6	Flux interface file
NDUMP	7	Dump
NFICHE	14	Microfiche output

the K2 loop is complete. When $K1 = 1$, a read is initiated to fill the first LCM block for each array, and a second read is started for the second LCM block. Computation proceeds while the second read is being processed. At the end of the K2 loop new fluxes are written on the flux disk file from the first LCM block. Then LCM addresses for the first and second blocks are interchanged and the fluxes and sources are read into the second block (which now has first block addresses) while computation proceeds from the first block (but with second block addresses). This alternation continues until $K1 = NKBLOC$ when no more data need be read.

When $NREAD = 2$, staged data transfers are made for the inhomogeneous source, fission source, and fluxes in calculating the source to the group in OUTER; for generating the new fission source in OUTER; and, as just described, for reading the source-to-the-group and reading and writing fluxes in KSWEET.

5. Calculation of the Source-to-a-Group. Subroutine OUTER consists of three major functions, all performed within the same two-group do loops on the integers IG1 and IG2. These functions are the calculation of the source-to-the-group IG, the control of the inner iteration for this group, and, if $IEVT > 0$, the

calculation of a new fission source. Each of these operations involves its own sweep over the planes with do loops on the integers K1 and K2, with the inner iteration planar sweep being controlled in subroutine KSWEEP.

The first step in the OUTER group loop is to read all the cross sections for the group from LCM to SCM.

To begin the calculation of the source-to-the-group (STG), if IEVT \neq 1 and NREAD = 1, the inhomogeneous source is read from disk to LCM for all groups in a group block. If NREAD = 2 this read is done for all planes in each plane block inside the K1 loop and data transmission is staged. When NREAD = 0, this read is a transfer within LCM from the inhomogeneous source locations (KQ) to the STG locations (KS) within a K2 loop.

If IEVT > 0, the fission fraction CHI(IG) \neq 0, and NREAD = 2, the fission source is read in a staged data transfer from disk to LCM. A check is made at this point to ensure that the STG unit NSOUR is finished transmission (subroutine DONE) because the next calculation involves the LCM address of the STG. Within SCM, the STG block (S) is first cleared and then, if there is an inhomogeneous source present, it is read from LCM to SCM. If there is a fission source for the group in question, it is read from LCM to SCM, added to S and the result is stored in LCM. If there is no fission source for the group and no inhomogeneous source, zero is stored in LCM.

All the above manipulations create an STG in LCM for all planes in a block. The STG may be zero, may be an inhomogeneous source only, or may be an inhomogeneous source plus a fission source. To the STG is added the scattering source. Computation of the scattering source requires a loop over the groups. Because this loop is already within the main loop over groups, different indices, IH1 and IH2, are used. The index for the group from which scattering occurs is IH. Three special parameters are also used, ILAP, IREAD, and IHN. The first of these is used to indicate if staged data transfer is being performed so that LCM address interchange can be performed. Even if NREAD = 2 there may be no need to overlap data transmission and calculation because scattering from the

group of next highest energy may not be possible. The parameter IREAD is used when NREAD = 1 to indicate whether or not scattering is possible from any group in the group block, and IHN then indicates the current or "now" group from which scattering occurs. If, for NREAD = 1 and after scattering source calculations are complete, IHN = IGL, then the reading of the flux block for the within-group calculations can be omitted.

The calculation of the source-to-the-group begins by initializing IH, IHN, and ILAP to zero. Then, inside the IH1 loop, if NREAD = 1, IREAD is calculated as the sum of NDSCAT for all groups that can scatter to group IG, and, if IREAD \neq 0, IHN is set equal to IH1 and the appropriate block of group fluxes is read. Then the IH2 loop over the groups in the block is begun. Inside this loop, if no scattering is possible, the remainder of the calculation is skipped. Otherwise, there are two paths.

First, if NREAD = 2, the fluxes for group IH and plane block K1 are read from disk file if IH = 1 or NDSCAT (IH - 1, IG) = 0 and the fluxes for group IH + 1 are read if IH \neq IGM and NDSCAT(IH + 1, IG) \neq 0. The first time group IH + 1 fluxes are read, ILAP is set to one and LCM alternation begins. [Note that this flux reading and all the rest of the calculations in this paragraph are skipped if NDSCAT (IH, IG) = 0.] Next a sweep is performed through the planes in a plane block and the scattering source contribution from group IH to group IG is computed, added to the STG, and stored in LCM.

After the K2 loop is complete, i.e., after calculation of the scattering source for all planes in the block, LCM flux addresses are interchanged and the next scattering group is considered. When the scattering source group loop is complete, the entire STG is transferred to disk file if NREAD = 2 and the LCM addresses of the STG and fission source are interchanged.

All of the above calculations are repeated for each block of planes.

6. Control of Inner Iteration. If NREAD = 1 and IHN \neq IGL, then all the fluxes for the group block are read from disk. If NREAD = 0 the fluxes are read from LCM in KSWEEP and if NREAD = 2

they are read from disk to LCM in KSWEEP and then read from LCM to SCM.

After inner iteration and if NREAD = 1, the newly calculated fluxes from group IG are stored on disk. This must be done separately for each group in the group block rather than once for all groups so that the STG calculation can use the latest flux values.

7. Calculation of the New Fission Source. If IEVT = 0, calculation of the fission source is skipped. Otherwise, unless NREAD = 2, there is no disk storage required. When NREAD = 2, a staged read and write of the disk unit NFN is performed for blocks of planes. The fission source being used in the outer iteration is stored on unit NF, and after the outer iteration is finished the units are interchanged. The reading of unit NFN is skipped if IG = 1 because the read is to permit addition of the contribution being calculated to those previously calculated.

8. Dumps. Both the writing of the dump file and the reading of a restart dump file are performed by subroutine DUMPER. The dump is always written in the same way on the same unit. Dumps are taken from subroutine KSWEEP if the time since the last periodic dump exceeds the parameter PTIME seconds where PTIME is set in the main THREETRAN program or if the execution time of the problem, either restart or initial problem, exceeds the input parameter DTIME seconds. A dump is also taken at the end of OUTER before final printing. The use of the dumps is described in the section on program options above. Here, the data transfer associated with a dump is described.

Two auxiliary subroutines are used in transferring data during reading or writing of dumps. Subroutine LCTRAN transfers data between LCM, SCM, and the dump unit. Either the contents of LCM are passed through SCM in units of the available SCM memory to the dump unit, or LCM is filled, via SCM, by reading from the dump unit. Subroutine DKTRAN performs a similar function by transferring random disk data through LCM, in units of the available LCM storage, through SCM (using LCTRAN) to the dump unit or by performing the inverse operation.

During the writing of a dump, if NREAD = 2, the first step is to check to see if all other applicable disk transfers are complete.

Then certain basic parameters are computed and written on unit NDUMP along with the contents of SCM. Next the contents of LCM are written on NDUMP. If NREAD = 0, the contents of SCM, destroyed by the LCM to NDUMP transfers, are restored by reading from NDUMP to SCM and the taking of the dump is complete. If NREAD > 0, the contents of the appropriate random disk files are saved as well and in this case the original contents of LCM as well as of SCM must be restored. Here, in the restoring of SCM, it is essential not to disturb the disk request tables which have one status before the dump is taken and another afterwards. Thus, when reading from NDUMP to restore SCM, the request table area of SCM is skipped.

In reading a restart dump, the problem begins just as an initial problem until the control parameter input is read in INITAL. Then, if ISTART > 0, the dump parameters and the contents of SCM are read from NDUMP to initialize all storage parameters. At this point control is returned to the main program where IANDI is skipped and OUTER is entered. In OUTER, if ISTART = 2 for restart from a final dump, the dump is read immediately and a new outer iteration is begun. If ISTART = 1, control passes to the call of KSWEET in which the dump is read and control is switched further to the forward or backward sweep through the planes, depending on where the dump was written.

In reading the dump upon restart, two times initialized in the main program during restart are saved during the reading of the dump and restored after the reading is complete. Next, dump control parameters and the contents of LCM are read from NDUMP. If NREAD > 0, then the disk request tables are initialized and the random disk contents are read from NDUMP and placed on the appropriate disk unit. After this, or also if NREAD = 0, the contents of LCM and then SCM are restored by reading from NDUMP.

For a 10-group $50 \times 50 \times 50$ problem there would be 5×10^6 fluxes alone so that the disk file NDUMP would require several tapes to hold its contents.

D. Problem Printed Output

A copy of an output listing for a small three-dimensional test problem is displayed on the following pages. In the listing the printing of the input control parameters and the number of fine-mesh intervals is followed by a recapitulation of problem storage. In the sample problem, MAXSCM and MAXLCM have been artificially restricted to 4 000 and 30 000, respectively, to force NREAD to be 2.

After the storage summary, remaining problem input is printed in the order of input. Note that only the cross-section header cards are printed.

After problem input is printed, a mesh-spacing summary is given and a description of the system is drawn. At this point outer iteration begins. The problem shown was run taking periodic dumps every 30 s and restarted several times. The details of the outer iterations are omitted, but the final listing shows the problem required five outer iterations. The scalar flux is printed for the outside planes in both groups.

VI. PROGRAM TESTING AND SUGGESTED FUTURE MODIFICATIONS

A. Program Testing

THREETRAN was checked on 11 test problems before running a full-scale problem. These problems included an example of each eigenvalue type, i.e., an inhomogeneous source problem, a k_{eff} problem, and a source plus fission problem. In each case a two-dimensional version of these problems was first checked by comparison with results from a modified TWOTRAN-II.³ The TWOTRAN program was altered to prevent negative flux fixup and omit the coarse-mesh rebalance. THREETRAN was rendered two-dimensional by setting either UZ, UY, or UX to zero. Thus, the same two-dimensional problem could be run three different ways to check functioning of different parts of THREETRAN, and this was done.

After agreement with a two-dimensional calculation was obtained, a three-dimensional problem was run for each problem type in the fastest storage mode (NREAD = 0). Then, storage allowed was artificially reduced to permit checking of each of the problem

types in each of the other two storage modes. Finally each problem type was checked for each type of dump in each storage mode.

Two-dimensional testing was also performed before calculation of the full-scale test problem which was a one-eighth assembly representation of ZPPR-4 Phase 1 experiment. The details of this calculation are described in Ref. 5. An S_4 , $29 \times 29 \times 22$ (x,y,z), six-group, k_{eff} calculation was performed with 4 axial zones, 25 y-dimension zones, and 28 x-dimension zones. THRETRAN results at the system midplane for relative ^{235}U and ^{239}Pu fission rates as a function of x compared well with experiment and agreed qualitatively with Argonne National Laboratory (ANL) two-dimensional calculations.

B. Suggested Future Modifications of THRETRAN

THRETRAN is presently limited to downscatter and isotropic scattering problems. The first of these limitations is probably not serious for fast reactor calculations and in any event is trivial to remove. Addition of one input parameter, IHS, to the input would make possible the inclusion of upscattering. For IEVT = 0, however, no outer iteration would be performed unless the program was modified. Adding anisotropic scattering would be more difficult, although if the four-component flux treatment were eliminated, storage algorithms would, if anything, be simpler. For example, if PH1 were the scalar flux, then PH2, PH3, and PH4 could be used for the μ , η , and ξ components of the flux. Present data transmission to update the scalar flux after each quadrant of the computation could be eliminated.

Perhaps the most important improvement to THRETRAN, and the area in which I think work should concentrate, is the upgrading of convergence acceleration algorithms. With state-of-the-art methodology, perhaps the simplest improvement would be the use of a Chebyshev outer iteration acceleration. This would mean storing another fission source if the ANL algorithm as modified by LASL⁶ were used, but the fission source is not group dependent.

The potentially very powerful modified diffusion theory acceleration algorithm of Alcouffe and Lewis,^{1,2} might be too expensive

in both storage and computation time unless it can be implemented in one or a combination of the following variants:

1. Inner iteration acceleration only. Then perhaps available storage could be used for both the transport and diffusion theory calculation.
2. Diffusion theory calculation on a coarse mesh instead of the fine mesh.
3. Diffusion theory flux synthesis.

Experimentation presently under way in two dimensions should indicate the best options for three-dimensional application.

REFERENCES

1. Raymond E. Alcouffe and E. E. Lewis, "Synthetic Acceleration Methods for the Neutron Transport Equation," presented at Conf. on Methods of Neutron Transport Theory in Reactor Calculations, Bologna, Italy (November 1975).
2. W. F. Miller, Jr., "Transport and Reactor Theory, October 1—December 31, 1975," Los Alamos Scientific Laboratory report LA-6221-PR (February 1976), p. 1.
3. K. D. Lathrop and F. W. Brinkley, "TWOTRAN-II: An Interfaced, Exportable Version of the TWOTRAN Code for Two-Dimensional Transport," Los Alamos Scientific Laboratory report LA-4848-MS (July 1973).
4. B. M. Carmichael, "Standard Interface Files and Procedures for Reactor Physics Codes, Version III," Los Alamos Scientific Laboratory report LA-5486-MS (February 1974).
5. K. D. Lathrop, "Transport Theory, Reactor Theory, and Reactor Safety, July 1-September 30, 1975," Los Alamos Scientific Laboratory report LA-6137-PR (November 1975), p. 7.
6. Ibid., p. 4.

CASE PROCESSED BY THRELEMAN COUE OF R/15/75 ON 03/31/76
 TMPTAN DERIG TEST PROBLEM SEVEN UNSYMMETRIC (IN Y) WITH FISSION
 INTERCHANGE Y AND Z

3 M1 NUMBR OF DIRECTIONS PER OCTANT
 2 IGM NUMBER OF ENERGY GROUPS
 3 IM NUMBER OF X-DIRECTION COARSE MESH ZONES
 2 JM NUMBER OF Y-DIRECTION COARSE MESH ZONES
 4 KM NUMBER OF Z-DIRECTION COARSE MESH ZONES
 0 IRL LEFT BOUNDARY CONDITION 0/1 VACUUM/REFLECTING
 0 IRR RIGHT BOUNDARY CONDITION 0 ONLY FOR NOW
 0 IBB BOTTOM BOUNDARY CONDITION 0/1 VACUUM/REFLECTING
 0 IRT TOP BOUNDARY CONDITION 0 ONLY FOR NOW
 0 IRF FRONT BOUNDARY CONDITION 0 ONLY FOR NOW
 0 IBA BACK BOUNDARY CONDITION 0/1 VACUUM/REFLECTING
 2 IEVT PROBLEM TYPE 0/1/2 SOURCE/K EFFECTIVE/SOURCE PLUS FISSION
 0 ISTART 0/1/2 INITIAL PROBLEM/RESTART FROM PERIODIC DUMP/RESTART FROM FINAL DUMP
 1 IQOPT SOURCE INPUT OPTION 0/1 NO SOURCE/ENERGY AND SPATIAL SHAPES
 3 MT TOTAL NUMBER OF MACROSCOPIC CROSS SECTIONS
 3 IHT ROW OF TOTAL CROSS SECTION
 5 IHM LAST ROW OF CROSS SECTION TABLE
 -0 IOUT OUTPUT OPTION 0/1 PRINT/FICHE SELECTED FLUXES
 15 LIMIT MAXIMUM NUMBER OF INNER ITERATIONS PER GROUP
 1.000E-03 EPS CONVERGENCE PRECISION
 1.000E+00 NORM NORMALIZATION FACTOR (NOT USED IF ZERO)
 5.000E+02 DTIME RESTART DUMP TAKEN AFTER DTIME SECONDS AND PROBLEM HALTED

INPUT INT. PER X MESH 3
 2 4 8
 INPUT INT. PER Y MESH 2
 12 6
 INPUT INT. PER Z MESH 4
 ALL ENTRIES = 4

LINES IN CORE PLANES IN LCM GROUPS IN LCM SCM REQUIRED LCM REQUIRED
 12 6 1 3944 28279

STORAGE OPTIONS
 0/1/2 ALL LCM GROUPS GROUPS AND PLANES ON DISK
 2

INPUT X BOUNDARIES 4
 0. 5.0000E-01 1.5000E+00 3.5000E+00
 INPUT Y BOUNDARIES 3
 0. 3.0000E+00 4.5000E+00
 INPUT Z BOUNDARIES 5
 0. 1.0000E+00 2.0000E+00 3.0000E+00 4.0000E+00
 INPUT WEIGHTS 3
 ALL ENTRIES = 4.1667E-02
 INPUT MU COSTNES 3
 3.0164E-01 9.0445E-01 3.0164E-01
 INPUT ETA COSINES 3

9.0445E-01 3.0164E-01 3.0164E-01
 INPUT ZI COSINES 3
 3.0164E-01 3.0164E-01 9.0445E-01
 INPUT NO. OF PS. DOWN SCAT 4
 0 0 1 0
 INPUT FISSION SPECTRUM 2
 1.0000E+00 0.
 INPUT X-SECTION ID NOS 6
 ALL ENTRIES = 1
 INPUT X-SECTION ID NOS 6
 ALL ENTRIES = 2
 INPUT X-SECTION ID NOS 6
 ALL ENTRIES = 3
 INPUT X-SECTION ID NOS 6
 ALL ENTRIES = 1
 1 LOADED FROM CARDS XS 1
 2 LOADED FROM CARDS XS 2
 3 LOADED FROM CARDS XS 3

	X ZONE	DELTA X	Y ZONE	DELTA Y	Z ZONE	DELTA Z
1	1	.2500E+00	1	.2500E+00	1	.2500F+00
2	1	.2500E+00	1	.2500E+00	1	.2500F+00
3	2	.2500E+00	1	.2500E+00	1	.2500F+00
4	2	.2500E+00	1	.2500E+00	1	.2500F+00
5	2	.2500E+00	1	.2500E+00	2	.2500F+00
6	2	.2500E+00	1	.2500E+00	2	.2500F+00
7	3	.2500E+00	1	.2500E+00	3	.2500F+00
8	3	.2500E+00	1	.2500E+00	2	.2500F+00
9	3	.2500E+00	1	.2500E+00	3	.2500F+00
10	3	.2500E+00	1	.2500E+00	3	.2500F+00
11	3	.2500E+00	1	.2500E+00	3	.2500F+00
12	3	.2500E+00	1	.2500E+00	3	.2500F+00
13	3	.2500E+00	2	.2500E+00	4	.2500F+00
14	3	.2500E+00	2	.2500E+00	4	.2500F+00
15			2	.2500E+00	4	.2500F+00
16			2	.2500E+00	4	.2500F+00
17			2	.2500E+00		
18			2	.2500E+00		

INPUT Q ENERGY SHAPE 2
 1.0000E+00 0.
 INPUT Q X SHAPE 14
 0. 0. 1.0000E+00 1.0000E+00 1.0000E+00 1.0000F+00 1.0000E+00 1.0000E+00 1.0000E+00 1.0000E+00
 1.0000E+00 1.0000E+00 0. 0.
 INPUT Q Y SHAPE 18
 ALL ENTRIES = 1.0000E+00
 INPUT Q Z SHAPE 16
 0. 0. 0. 0. 1.0000E+00 1.0000E+00 1.0000E+00 1.0000E+00 1.0000E+00 1.0000E+00
 1.0000E+00 1.0000E+00 0. 0. 0. 0.

THE SYSTEM BACK BOUNDARY CONDITION IS 0
 BETWEEN Z BOUNDARY 0. AND Z BOUNDARY .10000E+01 THERE ARE 4 EQUAL INTERVALS

.....MATERIAL MAP.....

MATERIALS BY BROAD ZONE; ORIGIN AT LOWER LEFT.
 M IS NUMBER OF FINE INTERVALS/BROAD ROW(COLUMN).

Y	ROW	M				
4.5000	0	0	0	0	0	0
2	6	0	1	1	1	0
3.0000	0	0	0	0	0	0
1	12	0	1	1	1	0
0.0000	0	0	0	0	0	0
X			0	1	3	
			0	5000	5000	5000
M		2	4	8		
COLUMN		1	2	3		

BETWEEN Z BOUNDARY .10000E+01 AND Z BOUNDARY .20000E+01 THERE ARE 4 EQUAL INTERVALS

.....MATERIAL MAP.....

MATERIALS BY BROAD ZONE; ORIGIN AT LOWER LEFT.
 M IS NUMBER OF FINE INTERVALS/BROAD ROW(COLUMN).

Y	ROW	M				
4.5000	0	0	0	0	0	0
2	6	0	2	2	2	0
3.0000	0	0	0	0	0	0
1	12	0	2	2	2	0
0.0000	0	0	0	0	0	0
X			0	1	3	
			0	5000	5000	5000
M		2	4	8		
COLUMN		1	2	3		

BETWEEN Z BOUNDARY .20000E+01 AND Z BOUNDARY .30000E+01 THERE ARE 4 EQUAL INTERVALS

.....MATERIAL MAP.....

MATERIALS BY BROAD ZONE; ORIGIN AT LOWER LEFT.
 M IS NUMBER OF FINE INTERVALS/BROAD ROW(COLUMN).

Y	ROW	M				
4.5000	0	0	0	0	0	0
2	6	0	3	3	3	0
3.0000	0	0	0	0	0	0
1	12	0	3	3	3	0

```

0.0000 0 * 0
X 000000 000000 000000 000000
0. 0. 1. 3.
0 5000 5000 5000

M      2  4  8
COLUMN 1  2  3

```

BETWEEN Z BOUNDARY .30000E+01 AND Z BOUNDARY .40000E+01 THERE ARE 4 EQUAL INTERVALS

*****MATERIAL MAP*****

```

Y
ROW M
4.5000 00000000000000000000
0 0
2 6 0 1 * 1 * 1 0
0 0
3.0000 0*****0*****0
0 0
1 12 0 1 * 1 * 1 0
0 0
0.0000 000000 000000 000000
X 0. 0. 1. 3.
0 5000 5000 5000

M      2  4  8
COLUMN 1  2  3

```

MATERIALS BY BROAD ZONE: ORIGIN AT LOWER LEFT.
M IS NUMBER OF FINE INTERVALS/BROAD ROW (COLUMN).

THE SYSTEM FRONT BOUNDARY CONDITION IS 0

OUTER ITERATION 0 EIGENVALUE 0.

LAMBDA 0.

GROUP	ITERATION NUMBER	ERROR
1	1	1.84197E+03
1	2	4.47019E+02
1	3	1.81598E+02
1	4	3.09053E-01
1	5	6.05539E-02
1	6	1.41852E-02
1	7	3.46197E-03
1	8	8.54724E-04
2	1	2.16073E+02
2	2	4.86925E+00
2	3	8.75894E-01
2	4	3.64781E-01
2	5	1.86903E-01
2	6	1.04252E-01
2	7	6.06544E-02

RESTART DUMP TAKEN

CASE PROCESSED BY THRETRAN CODE OF 8/15/75 ON 24/1/76
 THRETRAN DEBUG TEST PROBLEM SEVEN UNSYMMETRIC (IN Y) WITH FISSION
 INTERCHANGE Y AND Z

3 N1 NUMBER OF DIRECTIONS PER OCTANT
 2 IGM NUMBER OF ENERGY GROUPS
 2 IM NUMBER OF X-DIRECTION COARSE MESH ZONES
 2 JM NUMBER OF Y-DIRECTION COARSE MESH ZONES
 4 KM NUMBER OF Z-DIRECTION COARSE MESH ZONES
 0 IRL LEFT BOUNDARY CONDITION 0/1 VACUUM/REFLECTING
 0 IRD RIGHT BOUNDARY CONDITION 0 ONLY FOR NOW
 0 IRR HGTICM BOUNDARY CONDITION 0/1 VACUUM/REFLECTING
 0 IRT TOP BOUNDARY CONDITION 0 ONLY FOR NOW
 0 IRF FRONT BOUNDARY CONDITION 0 ONLY FOR NOW
 0 IRA BACK BOUNDARY CONDITION 0/1 VACUUM/REFLECTING
 2 IEVT PROBLEM TYPE -1/2 SOURCE/K EFFECTIVE/SOURCE PLUS FISSION
 1 ISTART 0/1/2 INITIAL PROBLEM/RESTART FROM PERIODIC DUMP/RESTART FROM FINAL DUMP
 1 IQOPT SOURCE INPUT OPTION 0/1 NO SOURCE/ENERGY AND SPATIAL SHAPES
 3 MT TOTAL NUMBER OF MACROSCOPIC CROSS SECTIONS
 3 IMT ROW OF TOTAL CROSS SECTION
 5 IHM LAST ROW OF CROSS SECTION TABLE
 -0 IOHT OUTPLT OPTION 0/1 PRINT/FICHE SELECTED FLUXES
 15 LIMIT MAXIMUM NUMBER OF INNER ITERATIONS PER GROUP
 1.000E-03 EPS CONVERGENCE PRECISION
 1.000E+00 NORM NORMALIZATION FACTOR (NOT USED IF ZERO)
 5.000E+02 OTIME RESTART DUMP TAKEN AFTER OTIME SECONDS AND PROBLEM HALTED

RESTART DUMP READ

1	4	9.52367E-04
2	1	3.33178E-03
2	2	2.52811E-03
2	3	1.55627E-03
2	4	8.53272E-04

OUTER ITERATION 5 EIGENVALUE .10000E+01 LAMDA .10010E+01

EXECUTION TIME (MINUTES) = 5.34942E-01

RESTART DUMP TAKEN

SCALAR FLUX FOR GROUP 1: PLANE 1

	X MESH 1	X MESH 2	X MESH 3	X MESH 4	X MESH 5	X MESH 6	X MESH 7	X MESH 8
18	.232187E-02	.795892E-02	.339342E-02	.997480E-02	.466548E-02	.527922E-02	.558867E-02	.550867E-02
17	.289534E-02	.381601E-02	.442809E-02	.522709E-02	.617435E-02	.696131E-02	.717342E-02	.717342E-02
16	.351497E-02	.476129E-02	.562702E-02	.667810E-02	.793039E-02	.891181E-02	.913646E-02	.913646E-02
15	.411427E-02	.554252E-02	.655346E-02	.773836E-02	.916435E-02	.103289E-01	.106625E-01	.106625E-01
14	.447181E-02	.591753E-02	.692907E-02	.812191E-02	.953193E-02	.107481E-01	.112336E-01	.112336E-01
13	.457077E-02	.602962E-02	.710700E-02	.831922E-02	.967086E-02	.108765E-01	.114034E-01	.114034E-01
	X MESH 9	X MESH 10	X MESH 11	X MESH 12	X MESH 13	X MESH 14	X MESH	

18	.527922E-12	.466544E-12	.397480E-02	.379342E-02	.295892E-02	.232183E-02
17	.696131E-12	.617435E-12	.522769E-02	.442809E-02	.381601E-02	.289534E-02
16	.891181E-12	.793639E-12	.667810E-02	.562702E-02	.476129E-02	.351493E-02
15	.103289E-11	.916635E-12	.773836E-02	.655346E-02	.554252E-02	.411427E-02
14	.107481E-11	.953193E-12	.812191E-02	.692907E-02	.591753E-02	.447181E-02
13	.108765E-11	.967086E-12	.831922E-02	.710700E-02	.602962E-02	.457070E-02

SCALAR FLUX FOR GROUP 1, PLANE 1

	X MESH 1	X MESH 2	X MESH 3	X MESH 4	X MESH 5	X MESH 6	X MESH 7	X MESH 8
12	.456691E-12	.606797E-12	.719890E-02	.845908E-02	.983564E-02	.110238E-01	.115165E-01	.115165E-01
11	.458192E-12	.611795E-12	.723247E-02	.851440E-02	.991200E-02	.110864E-01	.115963E-01	.115963E-01
10	.464293E-12	.616716E-12	.733821E-02	.864252E-02	.100171E-01	.112050E-01	.117214E-01	.117214E-01
9	.464293E-12	.616716E-12	.733821E-02	.864252E-02	.100171E-01	.112050E-01	.117214E-01	.117214E-01
8	.458192E-12	.611795E-12	.723247E-02	.851440E-02	.991200E-02	.110864E-01	.115963E-01	.115963E-01
7	.456691E-12	.606797E-12	.719890E-02	.845908E-02	.983564E-02	.110238E-01	.115165E-01	.115165E-01
6	.457117E-12	.602962E-12	.710699E-02	.819222E-02	.962185E-02	.108165E-01	.114034E-01	.114034E-01
5	.447181E-12	.591753E-12	.692907E-02	.812191E-02	.953192E-02	.107481E-01	.112336E-01	.112336E-01
4	.411427E-12	.554251E-12	.655345E-02	.773835E-02	.916634E-02	.103289E-01	.106625E-01	.106625E-01
3	.351493E-12	.476129E-12	.562702E-02	.667808E-02	.793037E-02	.891179E-02	.913645E-02	.913645E-02
2	.289534E-12	.351493E-12	.442808E-02	.522707E-02	.617433E-02	.696129E-02	.717340E-02	.717340E-02
1	.232183E-12	.255890E-12	.339339E-02	.397477E-02	.466544E-02	.527919E-02	.550863E-02	.550863E-02
	X MESH 9	X MESH 10	X MESH 11	X MESH 12	X MESH 13	X MESH 14	X MESH	X MESH
12	.110238E-01	.115165E-01						
11	.110864E-01	.115963E-01						
10	.112050E-01	.117214E-01						
9	.112050E-01	.117214E-01						
8	.110864E-01	.115963E-01						
7	.110238E-01	.115165E-01						
6	.108165E-01	.114034E-01						
5	.107481E-01	.112336E-01						
4	.103289E-01	.106625E-01						
3	.891179E-02	.913645E-02						
2	.696129E-02	.717340E-02						
1	.527919E-02	.550863E-02						

SCALAR FLUX FOR GROUP 1, PLANE 1A

	X MESH 1	X MESH 2	X MESH 3	X MESH 4	X MESH 5	X MESH 6	X MESH 7	X MESH 8
18	.180113E-12	.227115E-12	.255194E-02	.297323E-02	.353509E-02	.406202E-02	.423575E-02	.423575E-02
17	.223181E-12	.251169E-12	.330916E-02	.393707E-02	.460614E-02	.533360E-02	.547365E-02	.547365E-02
16	.269147E-12	.361465E-12	.418976E-02	.495802E-02	.597491E-02	.679734E-02	.693165E-02	.693165E-02
15	.315825E-12	.420157E-12	.486516E-02	.578961E-02	.685905E-02	.783630E-02	.805700E-02	.805700E-02
14	.346611E-12	.451175E-12	.516222E-02	.601153E-02	.713443E-02	.816590E-02	.852453E-02	.852453E-02
13	.357730E-12	.463634E-12	.534281E-02	.620502E-02	.722973E-02	.833376E-02	.873506E-02	.873506E-02
	X MESH 9	X MESH 10	X MESH 11	X MESH 12	X MESH 13	X MESH 14	X MESH	X MESH
18	.406202E-02	.423575E-02						
17	.533360E-02	.547365E-02						
16	.679734E-02	.693165E-02						
15	.783630E-02	.805700E-02						
14	.816590E-02	.852453E-02						
13	.833376E-02	.873506E-02						

SCALAR FLUX FOR GROUP 1, PLANE 1A

	X MESH 1	X MESH 2	X MESH 3	X MESH 4	X MESH 5	X MESH 6	X MESH 7	X MESH 8
12	.356655E-02	.465920F-02	.540476E-02	.670540E-02	.742385E-02	.844324E-02	.841110E-02	.811166F-02
11	.354314E-02	.464803E-02	.537703E-02	.628323E-02	.740844E-02	.840750E-02	.878690E-02	.878686F-02
10	.361377E-02	.471554E-02	.548154E-02	.640780E-02	.751920E-02	.833535E-02	.892007E-02	.892007F-02
9	.361377E-02	.471554E-02	.548154E-02	.640780E-02	.751920E-02	.833535E-02	.892007E-02	.892007F-02
8	.354314E-02	.464803E-02	.537703E-02	.628323E-02	.740844E-02	.840750E-02	.878690E-02	.878686F-02
7	.356655E-02	.465920E-02	.540476E-02	.670540E-02	.742385E-02	.844324E-02	.841110E-02	.811166F-02
6	.357734E-02	.463634E-02	.534281E-02	.620502E-02	.729972E-02	.833375E-02	.873506E-02	.873506F-02
5	.346611E-02	.451174E-02	.516222E-02	.600152E-02	.713443E-02	.816590E-02	.852453E-02	.852453F-02
4	.315875E-02	.420156E-02	.486515E-02	.570961E-02	.685905E-02	.783630E-02	.805700E-02	.805700F-02
3	.269144E-02	.361465E-02	.418975E-02	.485801E-02	.597490E-02	.679733E-02	.693164E-02	.693164F-02
2	.223187E-02	.291168E-02	.330915E-02	.389369E-02	.466613E-02	.533359E-02	.547364E-02	.547364F-02
1	.180112E-02	.227114E-02	.255193E-02	.297322E-02	.353508E-02	.406200E-02	.423573E-02	.423573F-02

	X MESH 9	X MESH 10	X MESH 11	X MESH 12	X MESH 13	X MESH 14	X MESH
12	.844324E-02	.742385E-02	.630540E-02	.540476E-02	.465920E-02	.356655E-02	
11	.840750E-02	.740844E-02	.628323E-02	.537703E-02	.464803E-02	.354314E-02	
10	.853535E-02	.751920E-02	.640780E-02	.548154E-02	.471554E-02	.361377E-02	
9	.853535E-02	.751920E-02	.640780E-02	.548154E-02	.471554E-02	.361377E-02	
8	.840750E-02	.740844E-02	.628323E-02	.537703E-02	.464803E-02	.354314E-02	
7	.844324E-02	.742385E-02	.630540E-02	.540476E-02	.465920E-02	.356655E-02	
6	.833375E-02	.729972E-02	.620502E-02	.524281E-02	.463034E-02	.357732E-02	
5	.816590E-02	.713443E-02	.600152E-02	.516222E-02	.451174E-02	.346611E-02	
4	.783630E-02	.685905E-02	.570961E-02	.486515E-02	.420156E-02	.315875E-02	
3	.679733E-02	.597490E-02	.495801E-02	.418975E-02	.361465E-02	.269144E-02	
2	.533359E-02	.466613E-02	.389369E-02	.330915E-02	.291168E-02	.223187E-02	
1	.406200E-02	.353508E-02	.297322E-02	.255193E-02	.227114E-02	.180112E-02	

SCALAR FLUX FOR GROUP 20 PLANE 1

	X MESH 1	X MESH 2	X MESH 3	X MESH 4	X MESH 5	X MESH 6	X MESH 7	X MESH 8
18	.180999E-02	.245624E-02	.305285E-02	.370129E-02	.402047E-02	.432593E-02	.447994E-02	.446794E-02
17	.242000E-02	.338666E-02	.425401E-02	.518488E-02	.562346E-02	.605115E-02	.624667E-02	.624567E-02
16	.300241E-02	.424722E-02	.537372E-02	.634781E-02	.711613E-02	.765406E-02	.789903E-02	.789903F-02
15	.351952E-02	.497106E-02	.627970E-02	.740533E-02	.820921E-02	.890668E-02	.919697E-02	.919697F-02
14	.388657E-02	.545977E-02	.687373E-02	.815154E-02	.901799E-02	.967244E-02	.999207E-02	.999209F-02
13	.411935E-02	.577929E-02	.727575E-02	.854142E-02	.951622E-02	.101947E-01	.105267E-01	.105267F-01
	X MESH 9	X MESH 10	X MESH 11	X MESH 12	X MESH 13	X MESH 14	X MESH	
18	.437594E-02	.402047E-02	.359129E-02	.305285E-02	.245624E-02	.180898E-02		
17	.605115E-02	.562346E-02	.518488E-02	.475401E-02	.432593E-02	.405115E-02		
16	.765406E-02	.711613E-02	.634781E-02	.537372E-02	.424722E-02	.300241E-02		
15	.890668E-02	.820921E-02	.740533E-02	.627970E-02	.497106E-02	.351952E-02		
14	.919697E-02	.901799E-02	.815154E-02	.687373E-02	.545977E-02	.388657E-02		
13	.101987E-01	.951622E-02	.854142E-02	.727575E-02	.577929E-02	.411935E-02		

SCALAR FLUX FOR GROUP 20 PLANE 1

	X MESH 1	X MESH 2	X MESH 3	X MESH 4	X MESH 5	X MESH 6	X MESH 7	X MESH 8
12	.425822E-02	.558571E-02	.755478E-02	.887455E-02	.987561E-02	.105011E-01	.109261E-01	.109261F-01
11	.434123E-02	.610591E-02	.771120E-02	.906811E-02	.100881E-01	.108017E-01	.111551E-01	.111551F-01
10	.439757E-02	.518080E-02	.781996E-02	.919483E-02	.102209E-01	.109421E-01	.112992E-01	.112992F-01
9	.439740E-02	.518070E-02	.781995E-02	.919482E-02	.102209E-01	.109421E-01	.112992E-01	.112992F-01
8	.434123E-02	.610580E-02	.771117E-02	.906808E-02	.100880E-01	.108017E-01	.111550E-01	.111550F-01
7	.425819E-02	.558567E-02	.755473E-02	.887449E-02	.987555E-02	.105011E-01	.109260E-01	.109260F-01

6	.41193F-2	.577924E-2	.727568E-02	.854135E-02	.951054E-02	.101986E-01	.105266E-01	.105266F-01
5	.398652F-2	.545970E-2	.687365E-02	.828144E-02	.901788E-02	.967413E-02	.999197E-02	.999197F-02
4	.351944F-2	.457998E-2	.627960E-02	.740621E-02	.82098E-02	.890654E-02	.919683E-02	.919683F-02
3	.300234F-2	.424711E-2	.537359E-02	.634766E-02	.711597E-02	.765388E-02	.789885E-02	.789885F-02
2	.241998F-2	.328653E-2	.425385E-02	.511829E-02	.562326E-02	.605094E-02	.624645E-02	.624645F-02
1	.186884F-2	.245609E-2	.305266E-02	.359107E-02	.402022E-02	.432567E-02	.446768E-02	.446768F-02

X MESH 9 X MESH 10 X MESH 11 X MESH 12 X MESH 13 X MESH 14 X MESH

12	.105811E-1	.987561E-2	.887455E-02	.755478E-02	.598571E-02	.425822E-02		
11	.108017E-1	.104881E-1	.966811E-02	.771120E-02	.610591E-02	.434123E-02		
10	.109421E-1	.102209E-1	.919483E-02	.781996E-02	.618608E-02	.439750E-02		
9	.109421E-1	.102209E-1	.919482E-02	.781995E-02	.618607E-02	.439749E-02		
8	.108517E-1	.108880E-1	.966808E-02	.771117E-02	.610588E-02	.434122E-02		
7	.105811E-1	.987555E-2	.887449E-02	.755473E-02	.598567E-02	.425819E-02		
6	.101986E-1	.951054E-2	.854135E-02	.727568E-02	.577424E-02	.411930E-02		
5	.967917E-2	.901788E-2	.808144E-02	.687365E-02	.545970E-02	.388652E-02		
4	.890654E-2	.828008E-2	.740621E-02	.627960E-02	.497098E-02	.351945E-02		
3	.765388E-2	.711597E-2	.634766E-02	.577359E-02	.424711E-02	.300233E-02		
2	.605094E-2	.562326E-2	.511829E-02	.425385E-02	.338653E-02	.241998E-02		
1	.432567E-2	.402022E-2	.359107E-02	.305266E-02	.245609E-02	.186886E-02		

SCALAR FLUX FOR GROUP 2, PLANE 1A

	X MESH 1	X MESH 2	X MESH 3	X MESH 4	X MESH 5	X MESH 6	X MESH 7	X MESH 8
18	.194637F-2	.263459E-2	.328121E-02	.379714E-02	.434785E-02	.467102E-02	.482646E-02	.482646F-02
17	.254994F-2	.361193E-2	.455141E-02	.540047E-02	.605819E-02	.650995E-02	.672217E-02	.672217F-02
16	.322121F-2	.455098E-2	.578643F-02	.688219E-02	.772521E-02	.829438E-02	.856373E-02	.856373F-02
15	.378874E-2	.525384E-2	.680525E-02	.808608E-02	.900196E-02	.971714E-02	.100376E-01	.100376F-01
14	.418288F-2	.587752E-2	.744421E-02	.881688E-02	.985045E-02	.105581E-01	.108957E-01	.108957F-01
13	.442507E-2	.628892E-2	.786121E-02	.929489E-02	.103614E-01	.110863E-01	.114474E-01	.114474F-01

X MESH 9 X MESH 10 X MESH 11 X MESH 12 X MESH 13 X MESH 14 X MESH

18	.447102E-2	.434785E-2	.387914E-02	.328121E-02	.263459E-02	.194630E-02		
17	.650995E-2	.605819E-02	.540047E-02	.455141E-02	.361193E-02	.258942E-02		
16	.829438E-2	.772521E-02	.688219E-02	.578643E-02	.455098E-02	.322121E-02		
15	.971714E-2	.900196E-02	.808608E-02	.680525E-02	.535384E-02	.378874E-02		
14	.105581E-01	.985045E-02	.881688E-02	.744421E-02	.587752E-02	.418283E-02		
13	.110863E-01	.103614E-01	.929489E-02	.786121E-02	.628892E-02	.442507E-02		

SCALAR FLUX FOR GROUP 2, PLANE 1A

	X MESH 1	X MESH 2	X MESH 3	X MESH 4	X MESH 5	X MESH 6	X MESH 7	X MESH 8
12	.457163E-2	.642776E-2	.815823E-02	.965101E-02	.107512E-01	.114935E-01	.118715E-01	.118716F-01
11	.466797E-2	.654602E-02	.833768E-02	.987310E-02	.109956E-01	.117474E-01	.121353E-01	.121353F-01
10	.472311E-2	.664619E-02	.844768E-02	.100151E-01	.111297E-01	.118884E-01	.122796E-01	.122796F-01
9	.472314E-2	.664618E-02	.844767E-02	.100151E-01	.111297E-01	.118884E-01	.122795E-01	.122796F-01
8	.466788E-2	.654600E-02	.833765E-02	.987307E-02	.109955E-01	.117474E-01	.121352E-01	.121352F-01
7	.457167E-2	.642772E-02	.815818E-02	.965096E-02	.107511E-01	.114935E-01	.118715E-01	.118715F-01
6	.442500E-2	.628887E-02	.786115E-02	.929482E-02	.103613E-01	.110862E-01	.114473E-01	.114473F-01
5	.418277E-2	.587745E-02	.744413E-02	.881678E-02	.985035E-02	.105580E-01	.108956E-01	.108956F-01
4	.378868E-2	.525376E-02	.680515E-02	.808597E-02	.900184E-02	.971701E-02	.100375E-01	.100375F-01
3	.322112E-2	.455088E-02	.578631E-02	.688205E-02	.772506E-02	.829422E-02	.856357E-02	.856357F-02
2	.258933E-2	.361181E-02	.455126E-02	.540029E-02	.605800E-02	.650874E-02	.672196E-02	.672196F-02
1	.194619E-2	.263446E-02	.328104E-02	.387894E-02	.434764E-02	.467079E-02	.482623E-02	.482623F-02

X MESH 9 X MESH 10 X MESH 11 X MESH 12 X MESH 13 X MESH 14 X MESH

72	.114935E-01	.107512E-01	.965101E-02	.A15823E-02	.642776E-02	.457163E-02
71	.117474E-01	.109956E-01	.9A7310E-02	.A73708E-02	.A566A2E-02	.466790E-02
70	.118A84E-01	.111247E-01	.100015E-01	.A44768E-02	.664619E-02	.472315E-02
9	.118A84E-01	.111247E-01	.100015E-01	.A44767E-02	.664618E-02	.472314E-02
8	.117474E-01	.109955E-01	.9A7307E-02	.A73765E-02	.A566A0E-02	.466788E-02
7	.114935E-01	.107511E-01	.965096E-02	.A15818E-02	.642772E-02	.457160E-02
6	.118A62E-01	.107613E-01	.929482E-02	.7A6115E-02	.A708A7E-02	.442502E-02
5	.105500E-01	.985035E-02	.881678E-02	.744413E-02	.587745E-02	.418277E-02
4	.971701E-02	.906184E-02	.8A0597E-02	.A00515E-02	.535376E-02	.378868E-02
3	.829422E-02	.772506E-02	.680205E-02	.57A631E-02	.455088E-02	.322112E-02
2	.650874E-02	.605800E-02	.540029E-02	.455126E-02	.361181E-02	.258933E-02
1	.467470E-02	.434764E-02	.387894E-02	.328104E-02	.263446E-02	.194619E-02

RECEIVED

SEP -8 '76

IASL LIBRARY



SEP 8 '76