LA-5990-MS
Informal Report

UC-32
Reporting Date: May 1975
Issued: June 1975

ONETRAN: A Discrete Ordinates Finite Element Code for the Solution of the One-Dimensional Multigroup Transport Equation

by

T. R. Hill

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

An Affirmative Action/Equal Opportunity Employer

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

In the interest of prompt distribution, this report was not edited by the Technical Information staff.

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22151
Price: Printed Copy $5.45 Microfiche $2.25

TABLE OF CONTENTS

		Page
ABSTRACT		1
I.	INTRODUCTION	2
II.	THEORY	3
	A. The Analytic Transport Equation	3
	1. Particular Forms of the Divergence Operator	3
	2. Spherical Harmonics Expansion of the Source Term	4
	B. Energy and Angular Approximations	6
	1. Multigroup Equations	6
	2. Discrete Ordinate Equations	7
	a. Plane Geometry	7
	b. Cylindrical Geometry	7
	c. Spherical Geometry	8
	d. Diamond Difference Assumption and Starting Directions	9
	C. Discretization of the Spatial Variable	9
	D. Solution Algorithms	13
	1. Boundary Conditions	13
	2. Sweep of the Space-Angle Mesh	14
	3. Negative Flux Fixup	14
	4. Adjoint Problems	14
	5. Iterative Processes	15
	6. Convergence Acceleration Methods	16
	a. Rebalance Inner Iteration Acceleration	16
	b. Chebyshev Inner Iteration Acceleration	17
	c. Rebalance Outer Iteration Acceleration	18
	7. Convergence Tests	18
III.	A Guide to User Application	21
	A. Overall Program Flow	21
	B. Details of Program Options	21
	1. Cross Sections	21
	a. Input Formats	21
	b. Cross Section Mixing	22
	c. Anisotropic Cross Sections	23
	d. Adjoint Cross Sections	23
	e. Cross Section Checking	23
	f. Fission Fractions	23
	g. (n,2n) Reactions	24
	h. Fine-Mesh Density Factors	24

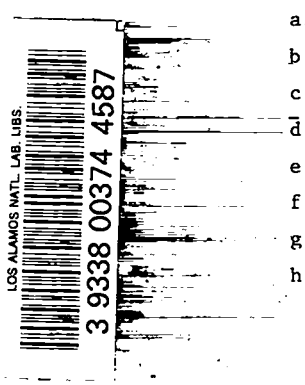


TABLE OF CONTENTS (continued)

		Page
	2. Geometry and Boundary Condition Specifications	24
	a. Spatial Mesh	24
	b. Boundary Conditions	24
	c. Buckling Absorption	25
	3. Angular Quadrature Coefficient Specifications	25
	4. Source Options	26
	5. Source Input Options	26
	6. Flux Input Options	27
	7. Flux Dumps and Restart Procedures	27
	8. Iteration Acceleration Options	28
	9. Eigenvalue Searches	28
	10. Adjoint Computations	30
	11. Edit Options	30
	a. Zone Edit	30
	b. Point Edit	31
C.	Data Input Rules	31
D.	Description of Input Data	32
	1. Job Title Card	32
	2. Input of Control Numbers	32
	3. Problem-Dependent Data	34
	4. Edit Input	36
E.	Output Description for a Test Problem	37
IV.	PROGRAMMING INFORMATION	55
A.	Program Structure	55
	1. Role and Function of Subprograms	55
	2. Relation of Problem Variables and Program Mnemonics	55
	3. Definition of Variables in Common Blocks	55
	4. Machine-Dependent Subprograms	65
	a. LCM System Routines	65
	b. General System Routines	65
B.	External and Internal Data Files	66
C.	Hardware Requirements	66
D.	Software Requirements	66
	1. CDC Machines	66
	2. ONETRAN for the IBM-360	66
E.	Programming Considerations	66
	1. Storage Management	66
	a. Variable Dimensioning	66
	b. Allocation of Large Core Memory	68
	c. Computation of Required Storage	68
	d. Temporary Storage Requirements	68
	e. Overstorage of Data in Core	69

TABLE OF CONTENTS (continued)

		Page
2.	Restart Tape Composition	69
3.	Standard Interface Files	69
ACKNOWLEDGEMENTS		69
REFERENCES		69

LIST OF FIGURES

Fig. 1.	Coordinates in Plane Geometry	4
Fig. 2.	Coordinates in Cylindrical Geometry	4
Fig. 3.	Coordinates in Spherical Geometry	4
Fig. 4.	Ordering of S_6 Directions in Plane and Spherical Geometries	8
Fig. 5.	Ordering of S_6 Directions in Cylindrical Geometry	8
Fig. 6.	Ordering of S_4 Directions in Two-Angle Plane Geometry	8
Fig. 7.	Linear Discontinuous Representation of the Angular Flux in the i^{th} Mesh Cell	10
Fig. 8.	Angular Flux Nodal Values for the i,m (Space,Angle) Mesh Cell	10
Fig. 9.	Simplified Logical Flow Diagram for ONETRAN	21
Fig. 10.	Variation of λ During a Hypothetical Eigenvalue Search	29

LIST OF TABLES

Table I.	Forms of $\nabla \cdot \underline{\Omega}\, \psi$	3
Table II.	Tabular Array of Spherical Harmonics: P_n^{ℓ}	6
Table III.	Spherical Harmonics Storage: Cylindrical Geometry	6
Table IV.	Spherical Harmonics Storage: Two-angle Plane Geometry	6
Table V.	Table of Geometric Functions	12
Table VI.	Options for Special Read Formats in LOAD	31
Table VII.	Function of ONETRAN Subroutines	55
Table VIII.	Relation of Problem Variables to Program Mnemonics	57
Table IX.	Contents of Blank Common Block IA	58
Table X.	Contents of Named Common Block /UNITS/	65
Table XI.	ONETRAN File Requirements	67
Table XII.	LCM Storage Parameters	68

ONETRAN:

A DISCRETE ORDINATES FINITE ELEMENT CODE

FOR THE

SOLUTION OF THE ONE-DIMENSIONAL MULTIGROUP TRANSPORT EQUATION

by

T. R. Hill

ABSTRACT

1. Program Identification: ONETRAN

2. Computer for which program is designed: CDC-7600, IBM-360

3. Description of Function: ONETRAN solves the one-dimensional multigroup transport equation in plane, cylindrical, spherical, and two-angle plane geometries. Both regular and adjoint, inhomogeneous and homogeneous (k_{eff} and eigenvalue searches) problems subject to vacuum, reflective, periodic, white, albedo or inhomogeneous boundary flux conditions are solved. General anisotropic scattering is allowed and anisotropic inhomogeneous sources are permitted.

4. Method of solution: The discrete ordinates approximation for the angular variable is used with the diamond (central) difference approximation for the angular extrapolation in curved geometries. A linear discontinuous finite element representation for the angular flux in each spatial mesh cell is used. Negative fluxes are eliminated by a local set-to-zero and correct algorithm. Standard inner (within-group) iteration cycles are accelerated by system rebalance, coarse-mesh rebalance, or Chebyshev acceleration. Outer iteration cycles are accelerated by coarse-mesh rebalance.

5. Restrictions: Variable dimensioning is used so that any combination of problem parameters leading to a container array less than MAXCOR can be accommodated. On CDC machines MAXCOR can be about 25 000 words and peripheral storage is used for most group-dependent data.

6. Running Time: ONETRAN is approximately twice as slow as DTF-IV per inner iteration for the same space-angle mesh. However, ONETRAN has twice as many unknowns per spatial mesh cell and a coarser spatial mesh than DTF-IV will normally give equivalent accuracy. Furthermore, ONETRAN will usually converge to DTF-IV equivalent accuracy in fewer total iterations.

   A 6-group, 106-interval mesh, S_2 k_{eff} calculation with four outer iterations of an EBR-II core requires 7.5 s on the CDC 7600. A 20-group, 134-interval mesh, S_4 k_{eff} cell calculation with 12 outer iterations requires 5.5 min on the CDC 7600.

7. Unusual Features of the Program: Provision is made for creation of standard interface output files for S_N constants, inhomogeneous sources, angle-integrated fluxes, and angular fluxes. Standard interface input files for S_N constants, inhomogeneous sources, cross sections, and total or angular fluxes may be read. All binary operations are localized in subroutines REED and RITE. Flexible edit options, including restart capability are provided.

8. Machine Requirements: Five interface units (use of interface units is optional), five output units, and two system input/output units are required. A large bulk memory is desirable, but may be replaced by disk, drum, or tape storage.

9. Related Programs: ONETRAN may be used to provide initial conditions to the TIMEX code, a time-dependent kinetics version of ONETRAN.

10. Material Available: Source deck, test problems, results of executed test problems, and this report are available from the Argonne Code Center.

I. INTRODUCTION

ONETRAN is a program designed to solve the one-dimensional multigroup transport equation in plane, cylindrical, spherical, and two-angle plane geometry. The program solves both regular or adjoint, homogeneous or inhomogeneous, time-independent problems subject to a variety of boundary conditions.

ONETRAN was created primarily to provide initial conditions compatible with the TIMEX[1] kinetics code. In addition, ONETRAN provides a significant advance over presently available 1-D transport codes by implementing the most current techniques available; namely:

- coarse-mesh rebalance[2] of both the inner and outer iterations, and the
- discontinuous linear finite element scheme resulting in a very accurate and stable discretization of the spatial variable.

ONETRAN is very similar to TWOTRAN-II[3] and TRIPLET[4] in nomenclature, coding, and input specifications.

The major features of ONETRAN include:

(1) direct or adjoint capability,

(2) plane, cylindrical, spherical, or two-angle plane geometry options,

(3) arbitrary anisotropic scattering order,

(4) two different sets of built-in S_N constants,

(5) vacuum, reflective, periodic, white, albedo, or inhomogeneous source boundary conditions,

(6) inhomogeneous source, k_{eff}, alpha or time-absorption, concentration, and delta or critical size calculation options.

(7) user choice of none, whole system rebalance, coarse-mesh rebalance, or Chebyshev acceleration of the inner iterations; and coarse-mesh rebalance acceleration of the outer iterations,

(8) optional print suppress of large input and output arrays,

(9) user choice of a single fission spectrum, zone-dependent fission spectra, a single fission matrix, or zone-dependent fission matrices,

(10) flexible edit and restart options,

(11) optional input of flux guess, inhomogeneous distributed and boundary sources, S_N constants, and cross sections from standard interface files,[5]

(12) optional output of scalar and angular fluxes, inhomogeneous distributed and boundary sources, and S_N constants to standard interface files,

(13) optional FIDO format[6] input of cross sections,

(14) optional specification of a pointwise density for cross-section spatial dependence, and

(15) a group-at-a-time storage organization to permit execution of exceptionally large problems.

The next section of this report contains the theoretical development of all the methods and approximatons used in ONETRAN. Section III is a user's guide for preparation of ONETRAN input and Sec. IV contains detailed programming information to facilitate local modification of the code. The contents of this report follow the guidelines[7] for documentation of digital computer programs accepted as an American Nuclear Society standard.

II. THEORY

In this section the energy, angular, and spatial variables of the transport equation are discretized to obtain a set of linear algebraic equations. The exact transport equation is discussed and the spherical harmonics expansion of the scattering sources is performed in Sec. II.A. The multigroup treatment of the energy variable and the discrete ordinates approximation of the angular variable are treated in Sec. II.B. Section II.C. is devoted to a discussion of the discontinuous linear finite element scheme used to discretize the spatial variable. The solution algorithms used to solve the set of algebraic equations are presented in Sec. II.D.

A. The Analytic Transport Equation

The time-independent inhomogeneous transport equation is

$$\nabla \cdot (\underline{\Omega}\psi) + \sigma(r,E)\,\psi(r,E,\underline{\Omega}) = \iint dE'\, d\underline{\Omega}'\, \sigma_s(r,E'\to E,\underline{\Omega}\cdot\underline{\Omega}')\,\psi(r,E',\underline{\Omega}') + \frac{1}{4\pi}\iint dE'\, d\underline{\Omega}'\, \chi(r,E'\to E)\,\nu\sigma_f(r,E')\,\psi(r,E',\Omega') + Q(r,E,\underline{\Omega}), \quad (1)$$

where ψ is the particle flux (particle number density times their speed) defined such that $\psi\, dE\, dV d\underline{\Omega}$ is the flux of particles in the volume element dV about r, in the element of solid angle $d\underline{\Omega}$ about $\underline{\Omega}$, in the energy range dE about E. Similarly, Q dV dE $d\underline{\Omega}$ is the number of particles in the same element of phase space emitted by sources independent of ψ. The macroscopic total interaction cross section is denoted by σ, the macroscopic scattering transfer probability (from energy E' to E through a scattering angle with cosine $\underline{\Omega}\cdot\underline{\Omega}'$) by σ_s, and the macroscopic fission cross section by σ_f. All of these quantities may be spatially dependent. The number of particles omitted isotropically ($1/4\pi$) per fission is ν, and the fraction of these liberated in the range dE about E from fissions in dE' about E' is $\chi(r,E'\to E)$.

The homogeneous transport equation is written in the same manner as Eq. (1) except that Q is zero and the term representing a source of neutrons due to fission is divided by the eigenvalue k_{eff}. In this report the inhomogeneous problem will be referred to as a source problem and the homogeneous or eigenvalue problem will be referred to as a k_{eff} problem. The ONETRAN code will solve both types of problems.

1. Particular Forms of the Divergence Operator

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

In the standard plane geometry, the angular flux is assumed independent of the azimuthal angle ϕ so that the angular dependence can be reduced to the μ interval of (-1, +1). ONETRAN also permits the two-angle option in plane geometry where no assumptions of symmetry are imposed. In this case the complete unit sphere of angular directions must be considered. In cylindrical geometry, the angular flux is assumed symmetric in the ξ angular cosine and symmetric about the $\phi = 0^\circ$-180° plane. Thus, only one-quarter of the unit sphere must be considered in the angular dependence. In spherical geometry, the angular flux

TABLE I

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

Geometry	Dependence of	Definition of Variables	$\nabla \cdot \underline{\Omega}\psi$
Plane	$\psi(x,\mu)$ or $\psi(x,\mu,\phi)$	$\mu=\hat{e}_x\cdot\underline{\Omega}$ $\xi=(1-\mu^2)^{1/2}\cos\phi$ $\eta=(1-\mu^2)^{1/2}\sin\phi$	$\mu\frac{\partial\psi}{\partial x}$
Cylindrical	$\psi(r,\mu,\eta)$	$\mu=\hat{e}_r\cdot\underline{\Omega}$ $\xi=\hat{e}_z\cdot\underline{\Omega}$ $\eta=(1-\xi^2)^{1/2}\sin\phi$ $\mu=(1-\xi^2)^{1/2}\cos\phi$	$\frac{\mu}{r}\frac{\partial(r\psi)}{\partial r} - \frac{1}{r}\frac{\partial(\eta\psi)}{\partial\phi}$
Spherical	$\psi(r,\mu)$	$\mu=\hat{e}_r\cdot\underline{\Omega}$	$\frac{\mu}{r^2}\frac{\partial(r^2\psi)}{\partial r} + \frac{1}{r}\frac{\partial[(1-\mu^2)\psi]}{\partial\mu}$

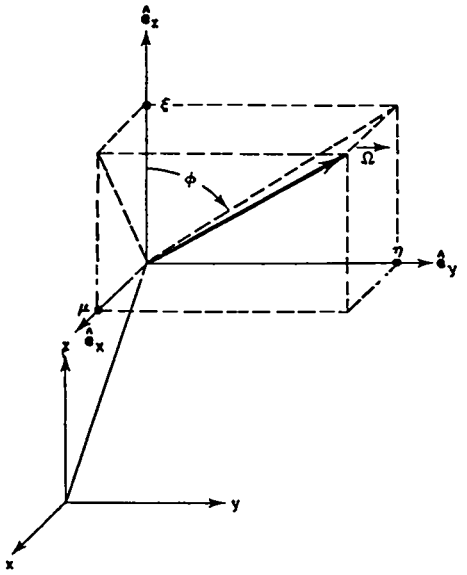


Fig. 1. Coordinates in plane geometry.

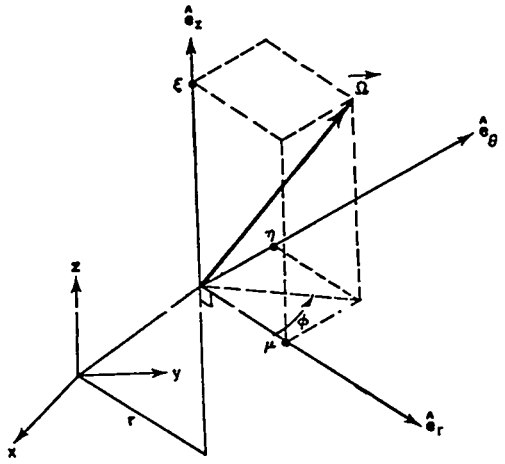


Fig. 2. Coordinates in cylindrical geometry.

is also assumed independent of the azimuthal angle ϕ so that the angular dependence is reduced to the μ interval of (-1, +1).

2. Spherical Harmonics Expansion of the Source Term

In the ONETRAN program, the scattering transfer probability is assumed to be represented by a finite Legendre polynomial expansion of order ISCT

$$\sigma_s(r,E'\to E,\underline{\Omega}\cdot\underline{\Omega}') = \sum_{n=0}^{ISCT} \frac{2n+1}{4\pi}\,\sigma_s^n(r,E'\to E)\,P_n(\underline{\Omega}\cdot\underline{\Omega}'). \quad (2)$$

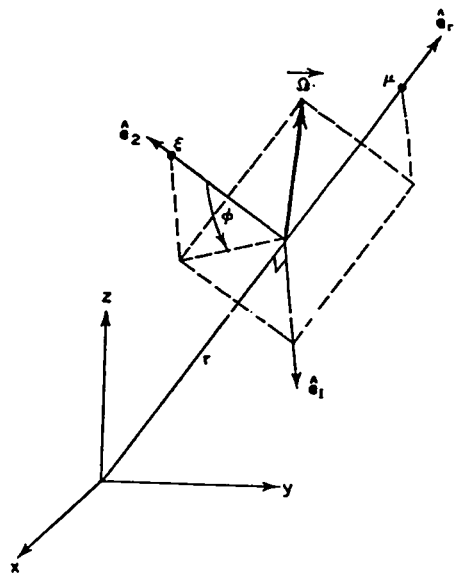


Fig. 3. Coordinates in spherical geometry.

If this expansion is inserted into Eq. (1), and the addition theorem for spherical harmonics used to expand $P_n(\underline{\Omega}\cdot\underline{\Omega}')$, the scattering term may be written after integration over the azimuthal angle for plane and spherical geometries as

$$\iint dE'\; d\underline{\Omega}'\; \sigma_s(r,E'\to E,\underline{\Omega}\cdot\underline{\Omega}')\;\psi(r,E',\underline{\Omega}')$$

$$= \int dE' \sum_{n=0}^{ISCT} (2n+1)\sigma_s^n(r,E'\to E)\;P_n(\mu)\;\phi_n(r,E'), \quad (3)$$

where the moments of the angular flux are defined

$$\phi_n(r,E) = \int_{-1}^{1} \frac{d\mu}{2} P_n(\mu)\;\psi(r,E,\mu). \quad (4)$$

For cylindrical and two-angle plane geometries, the scattering term becomes more complicated since the associated Legendre polynomial terms from the addition theorem cannot be integrated out. Dropping the spatial and energy variables, this term is

$$\iint dE'\ d\underline{\Omega}'\ \sigma_s(\underline{\Omega}\cdot\underline{\Omega}')\ \psi(\underline{\Omega}') = \sum_{n=0}^{ISCT} \frac{2n+1}{4\pi}\ \sigma_s^n \left[P_n(\xi) \int_{-1}^{1} d\xi' \int_0^{2\pi} d\phi'\ P_n(\xi')\ \psi(\xi',\phi') \right.$$

$$\left. + 2 \sum_{\ell=1}^{n} \frac{(n-\ell)!}{(n+\ell)!}\ P_n^{\ell}(\xi) \int_{-1}^{1} d\xi' \int_0^{2\pi} d\phi'\ P_n^{\ell}(\xi')\ \cos \ell(\phi-\phi')\ \psi(\xi',\phi') \right], \qquad (5)$$

where for two-angle plane geometry the variable ξ is replaced by μ. Using the trigonometric relation

$$\cos \ell(\phi-\phi') = \cos \ell\,\phi \cos \ell\,\phi' + \sin \ell\phi \sin \ell\phi', \qquad (6)$$

Eq. (5) may be written

$$= \sum_{n=0}^{ISCT} (2n+1)\sigma_s^n \left\{ P_n(\xi)\ \phi_n + \sum_{\ell=1}^{n} \sqrt{2\frac{(n-\ell)!}{(n+\ell)!}}\ [P_n^{\ell}(\xi) \cos \ell\phi\ \phi_n^{\ell} + P_n^{\ell}(\xi) \sin \ell\phi\ \phi_n^{\ell}] \right\}, \qquad (7)$$

where the moments of the angular flux are defined

$$\phi_n = \int_{-1}^{1} d\xi' \int_0^{2\pi} d\phi'\ P_n(\xi')\ \psi(\xi',\phi')/4\pi \qquad (8)$$

and

$$\phi_n^{\ell} = \sqrt{2\frac{(n-\ell)!}{(n+\ell)!}} \int_{-1}^{1} d\xi' \int_0^{2\pi} d\phi'\ P_n^{\ell}(\xi') \begin{Bmatrix} \cos \ell\,\phi' \\ \sin \ell\,\phi' \end{Bmatrix} \psi(\xi',\phi')/4\pi, \qquad (9)$$

and where either the sin or cos of Eq. (9) is chosen depending on its coefficient (sin or cos) in Eq. (7).

A tabular array of the $P_n(\xi)$ Legendre functions is shown in Table II. In cylindrical geometry, the angular flux is assumed symmetric in ξ so that the odd flux moments ($n - \ell$ odd) vanish. Likewise, the angular flux is assumed symmetric in ϕ so that the sine moments of the flux vanish. Thus, in cylindrical geometry, only the moments indicated in Table III are required. In the two-angle plane geometry, no assumptions of symmetry are made on the angular flux. Thus the complete array of spherical harmonics, as indicated in Table IV, is required.

In all cases, the scattering term may be written in the general form

$$S.T. = \int_0^{\infty} dE' \sum_{n=1}^{NM} (2n-1)\ \sigma_s^n(r,E'\to E)\ R_n(\underline{\Omega})\ \phi_n(r,E'), \qquad (10)$$

where NM is the number of flux moments given by

$$NM = \begin{cases} ISCT+1 & \text{for plane and spherical geometry,} \\ (ISCT+2)^2/4 & \text{for cylindrical geometry, or} \\ (ISCT+1)^2 & \text{for two-angle plane geometry,} \end{cases}$$

and $R_n(\underline{\Omega})$ is a spherical harmonic appropriate to that geometry and ϕ_n the corresponding angular flux moment. For cylindrical and two-angle plane geometry, the two-dimensional arrays of spherical harmonic moments are stored as one-dimensional arrays, indexed in the order shown in Tables III and IV.

In a similar fashion, the inhomogeneous source term of Eq. (1) is expanded in spherical harmonics

$$\text{Inhomogeneous Source} = \sum_{n=1}^{NMQ} (2n-1)\ R_n(\underline{\Omega})\ Q_n(r,E), \qquad (11)$$

where the number of source moments NMQ is related to the order of source anisotropy IQAN by

$$NMQ = \begin{cases} IQAN+1 & \text{for plane and spherical geometry} \\ (IQAN+2)^2/4 & \text{for cylindrical geometry, or} \\ (IQAN+1)^2 & \text{for two-angle plane geometry,} \end{cases}$$

and $Q_n(r,E)$ is the corresponding source moment.

TABLE II

TABULAR ARRAY OF SPHERICAL HARMONICS: P_n^ℓ

n \ ℓ →	0	1	2	3	4
0	P_0	-	-	-	-
1	P_1	P_1^1	-	-	-
2	P_2	P_2^1	P_2^2	-	-
3	P_3	P_3^1	P_3^2	P_3^3	-
4	P_4	P_4^1	P_4^2	P_4^3	P_4^4

TABLE III

SPHERICAL HARMONICS STORAGE: CYLINDRICAL GEOMETRY

1. P_0		-		-		-		-
x	2. $P_1^1 \cos\phi$		-		-		-	
3. P_2		x	4. $P_2^2 \cos 2\phi$		-		-	
x	5. $P_3^1 \cos\phi$		x	6. $P_3^3 \cos 3\phi$		-		
7. P_4		x	8. $P_4^2 \cos 2\phi$		x		9. $P_4^4 \cos 4\phi$	

x - storage not required.

TABLE IV

SPHERICAL HARMONICS STORAGE: TWO-ANGLE PLANE GEOMETRY

1. P_0		-		-		-		-	
2. P_1	3. $P_1^1 \cos\phi$ 4. $P_1^1 \sin\phi$		-		-		-		
5. P_2	6. $P_2^1 \cos\phi$ 7. $P_2^1 \sin\phi$		8. $P_2^2 \cos 2\phi$ 9. $P_2^2 \sin 2\phi$		-		-		
10. P_3	11. $P_3^1 \cos\phi$ 12. $P_3^1 \sin\phi$		13. $P_3^2 \cos 2\phi$ 14. $P_3^2 \sin 2\phi$		15. $P_3^3 \cos 3\phi$ 16. $P_3^3 \sin 3\phi$		-		
17. P_4	18. $P_4^1 \cos\phi$ 19. $P_4^1 \sin\phi$		20. $P_4^2 \cos 2\phi$ 21. $P_4^2 \sin 2\phi$		22. $P_4^3 \cos 3\phi$ 23. $P_4^3 \sin 3\phi$		24. $P_4^4 \cos 4\phi$ 25. $P_4^4 \sin 4\phi$		

B. Energy and Angular Approximations

In this section the multigroup approximation of the energy variable and the discrete ordinates approximation of the angular variable are given.

1. Multigroup Equations

The energy domain of interest in assumed to be partitioned into IGM intervals of width ΔE_g, g = 1, 2, ..., IGM. By convention, increasing g represents decreasing energy. If we integrate Eq. (1) over ΔE_g after making the spherical harmonic expansion of Eqs. (10) and (11), we can write

$$\nabla \cdot (\underline{\Omega}\, \psi_g) + \sigma_g\, \psi_g(r,\underline{\Omega})$$

$$= \sum_{h=1}^{IGM} \sum_{n=1}^{NM} (2n-1)\, \sigma_{s,h\to g}^n\, R_n(\underline{\Omega})\, \phi_{nh}(r)$$

$$+ \sum_{h=1}^{IGM} \nu\sigma_{fh}\, \chi_{h\to g}\, \phi_{1h}(r)$$

$$+ \sum_{n=1}^{NMQ} (2n-1)\, R_n(\underline{\Omega})\, Q_{ng}(r), \qquad (12)$$

$$g = 1,2,\ldots,IGM.$$

Here, the flux for group g,

$$\psi_g = \int_{\Delta E_g} \psi\, dE, \qquad (13)$$

is no longer a distribution in energy, but is the total number of particles in the energy interval. For this reason, when group structures are changed, the effect on the angular flux or its moments must be evaluated by comparing $\psi_g/\Delta E_g$. Because of Eq. (13), energy integrals in ONETRAN are evaluated by simple sums.

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

$$\sigma_g = \int_{\Delta E_g} \sigma\, \psi\, dE \Big/ \int_{\Delta E_g} \psi\, dE, \qquad (14)$$

but, of course, ψ is not known and must be approximated by some means. If in Eq. (14) the angular dependence of ψ is nonseparable, then σ_g will depend on angle. No provision for such dependence is made in ONETRAN. Recipes for taking this dependence into account, as well as for improving the averages $\sigma_{sh\to g}^n$ when scattering is severely anisotropic, are given by Bell, Hansen, and Sandmeier.[8]

2. Discrete Ordinates Equations

The three terms on the right-hand side of Eq. (12) represent sources of neutrons due to scattering, fission, and inhomogeneous sources, respectively. All coupling between the IGM multigroup equations is contained in these terms. To simplify notation for the following analysis, we write all three sources simply as S_g, which we will treat as a known quantity. Of course, S_g depends on the unknown flux ψ_g, but we will generate S_g iteratively using the latest values of ψ_g. For details of the iterative processes in ONETRAN, see Sec. II.D.5. Omitting the group subscript, Eq. (12) is written

$$\nabla \cdot (\underline{\Omega}\, \psi) + \sigma\, \psi(r,\underline{\Omega}) = S(r,\underline{\Omega}). \tag{15}$$

The discrete ordinates equation may be derived by directly differencing the angular variable in Eq. (15). The resulting equation in cylindrical and spherical geometries will conserve neutrons only in the limit of small angular intervals and may result in complex and unrealistic coupling of the angular variable.[9] The customary procedure is to difference both the angular and spatial variables simultaneously, but due to the finite element approach used on the spatial variable this will not be done. Instead a heuristic derivation of the discrete ordinates equation for each of the three geometries will be given.

a. Plane Geometry

From Table I, it is observed that no angular derivative appears in the divergence operator so that no angular coupling is present. For the standard plane geometry, the angular interval of $\mu\varepsilon(-1, -1)$ is discretized into a set of MM quadrature points μ_m and associated quadrature weights w_m ordered, as shown in Fig. 4. The quadrature weights are normalized so that $\sum_{m=1}^{MM} w_m = 1$, analogous to $d\mu/2$ in Eq. (4). The (angular) cell-centered angular flux is then assumed to be given by

$$\psi_m(r) \approx \psi(r, \mu_m) \tag{16}$$

and the angular flux moments of Eq. (4) are approximated by

$$\phi_n(r) \approx \sum_{m=1}^{MM} w_m\, P_n(\mu_m)\, \psi_m(r). \tag{17}$$

For the two-angle plane geometry option, the angular domain of the complete unit sphere is again discretized into a set of MM quadrature points (μ_m, ϕ_m) and associated quadrature weights w_m. The normalization is again $\sum_{m=1}^{MM} w_m = 1$, analogous to $d\underline{\Omega}/4\pi$ in Eqs. (8) and (9). The ordering of these quadrature points is illustrated in Fig. 6 for an S_4 quadrature. The built-in quadrature set of ONETRAN actually chooses μ_m and ξ_m as either the Gauss-Legendre or the double Gauss-Legendre quadrature points. The cell-centered angular flux is again assumed to be given by

$$\psi_m(r) \approx \psi(r, \mu_m, \phi_m) \tag{18}$$

and the angular flux moments of Eqs. (8) and (9) are approximated by

$$\phi_n(r) \approx \sum_{m=1}^{MM} w_m\, P_n(\mu_m)\, \psi_m(r) \tag{19}$$

and

$$\phi_n^{\ell}(r) \approx \sqrt{2\frac{(n-\ell)!}{(n+\ell)!}} \sum_{m=1}^{MM} w_m\, P_n^{\ell}(\mu_m) \begin{Bmatrix} \cos \ell\, \phi_m \\ \sin \ell\, \phi_m \end{Bmatrix} \psi_m(r). \tag{20}$$

For both of the plane geometries, the discrete ordinates approximation of the multigroup transport Eq. (15) is

$$\mu_m \frac{\partial \psi_m}{\partial x} + \sigma\, \psi_m(x) = S_m(x), \tag{21}$$

where $S_m(x)$ is the (known) source evaluated at the m^{th} angular quadrature point.

b. Cylindrical Geometry

From Table I, the multigroup transport Eq. (15) may be written in cylindrical geometry as

$$\mu \frac{\partial(r\, \psi)}{\partial r} - \frac{\partial(\eta\, \psi)}{\partial \phi} + r\, \sigma\, \psi(r, \underline{\Omega}) = r\, S(r, \underline{\Omega}). \tag{22}$$

The angular domain of one quadrant of the unit sphere is discretized into a set of MM quadrature points (μ_m, η_m) and associated quadrature weights w_m, normalized so that $\sum_{M=1}^{MM} w_m = 1$. The ordering of these quadrature points is illustrated in Fig. 5 for

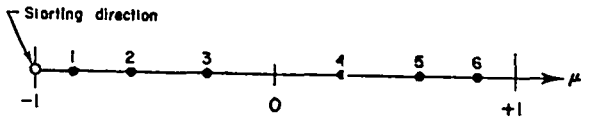


Fig. 4. Ordering of S_6 directions in plane and spherical geometries. The starting direction only applies to spherical geometry.

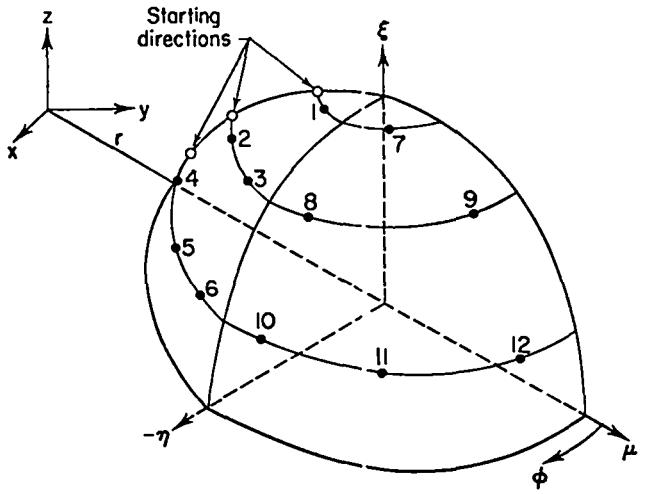


Fig. 5. Ordering of S_6 directions in cylindrical geometry.

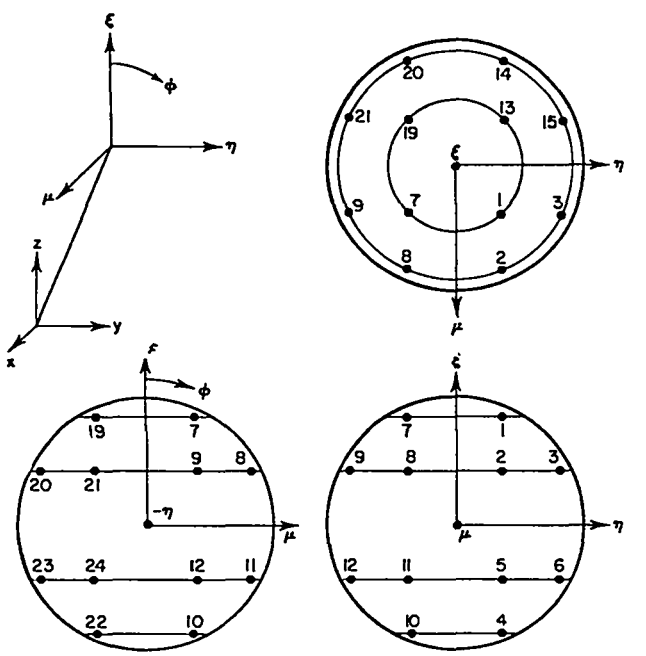


Fig. 6. Ordering of S_4 directions in two-angle plane geometry. The ordinates in the 8th octant are not shown.

an S_6 quadrature. The (angular) cell-centered angular flux is assumed to be given by

$$\psi_m(r) \approx \psi(r, \mu_m, \eta_m) \tag{23}$$

and the angular flux moments are approximated by Eqs. (19) and (20). In addition we have the (angular) cell edge fluxes on the same ξ level denoted by $\psi_{m-\frac{1}{2}}(r)$ and $\psi_{m+\frac{1}{2}}(r)$. We then write the discrete ordinates approximation to Eq. (22) as

$$\mu_m \frac{\partial(r\,\psi_m)}{\partial r} + \frac{\alpha_{m+\frac{1}{2}}}{w_m}\psi_{m+\frac{1}{2}}(r) - \frac{\alpha_{m-\frac{1}{2}}}{w_m}\psi_{m-\frac{1}{2}}(r) + r\,\sigma\,\psi_m(r) = r\,S_m(r). \tag{24}$$

Consider now the case of divergenceless flow in which $\psi = S/\sigma = \text{constant}$. Since $\eta = \sqrt{1-\xi^2}\sin\phi$ and $\mu = \sqrt{1-\xi^2}\cos\phi$, then $\partial\eta/\partial\phi = \mu$ and it is easily shown that the angular coupling coefficients α must satisfy the recursion relation

$$\alpha_{m+\frac{1}{2}} - \alpha_{m-\frac{1}{2}} = -\,w_m\,\mu_m, \tag{25}$$

with the requirement from neutron conservation that the first ($\alpha_{\frac{1}{2}}$) and last ($\alpha_{M+\frac{1}{2}}$) coefficients on a ξ level must vanish. It can be shown[10] that Eq. (24) becomes identical to Eq. (22) in the limit of vanishingly small angular intervals.

c. Spherical Geometry

From Table I, the multigroup transport Eq. (15) is written in spherical geometry as

$$\mu\frac{\partial(r^2\psi)}{\partial r} + r\frac{\partial[(1-\mu^2)\psi]}{\partial\mu} + r^2\,\sigma\,\psi(r,\mu) = r^2\,S(r,\mu). \tag{26}$$

The angular domain of $\mu\varepsilon(-1,+1)$ is discretized into a set of MM quadrature points μ_m and associated quadrature weights w_m, normalized so that $\sum_{m=1}^{MM} w_m = 1$. The ordering of these quadrature points is illustrated in Fig. 4 for an S_6 quadrature. The (angular) cell-centered angular flux is assumed to be given by

$$\psi_m(r) \approx \psi(r, \mu_m) \tag{27}$$

and the angular flux moments approximated by Eq. (17). The (angular) cell edge fluxes are denoted by $\psi_{m-\frac{1}{2}}(r)$

and $\psi_{m+\frac{1}{2}}(r)$. We write the discrete ordinates approximation to Eq. (26) as

$$\mu_m \frac{\partial(r^2 \psi_m)}{\partial r} + \left[\frac{\beta_{m+\frac{1}{2}}}{w_m} \psi_{m+\frac{1}{2}}(r) - \frac{\beta_{m-\frac{1}{2}}}{w_m} \psi_{m-\frac{1}{2}}(r)\right] r$$

$$+ r^2 \sigma \psi_m(r) = r^2 S_m(r). \qquad (28)$$

Considering the case of divergenceless flow, then the angular coefficients β must satisfy the recursion relation

$$\beta_{m+\frac{1}{2}} - \beta_{m-\frac{1}{2}} = -2 w_m \mu_m, \quad m = 1, \ldots, MM, \qquad (29)$$

with the requirement from neutron conservation that the first ($\beta_{\frac{1}{2}}$) and last ($\beta_{MM+\frac{1}{2}}$) coefficients must vanish. Again it can be shown[10] that Eq. (28) becomes identical to Eq. (26) in the limit of vanishingly small angular intervals.

d. Diamond Difference Assumption and Starting Directions

For the curved geometries discrete ordinates transport equation, Eqs. (24) and (28), there are three unknown functions: the (angular) mesh cell edge fluxes, $\psi_{m+\frac{1}{2}}(r)$ and $\psi_{m-\frac{1}{2}}(r)$, and the cell-centered angular flux, $\psi_m(r)$. The $\psi_{m-\frac{1}{2}}(r)$ edge flux will be assumed known from the previous angular mesh cell computation and imposing continuity at the (angular) mesh cell boundaries. The standard diamond difference[11] assumption (in angle only) is made to relate the edge and cell-centered fluxes, viz.,

$$\psi_m(r) = \frac{1}{2}\left[\psi_{m-\frac{1}{2}}(r) + \psi_{m+\frac{1}{2}}(r)\right]. \qquad (30)$$

Thus in each (angular) mesh cell, we need only solve the transport equation for one function, $\psi_m(r)$.

To initiate the computation in the first angular mesh cell, ONETRAN uses special zero-weighted directions to calculate $\psi_{\frac{1}{2}}(r)$. For spherical geometry, this special direction is the straight-inward direction $\mu = -1$, as illustrated in Fig. 4. For cylindrical geometry, these special directions correspond to ordinates directed towards the cylindrical axis, $\eta = 0$, $\phi = 180^\circ$, as illustrated in Fig. 5.

The starting direction calculations are treated separately from the calculations for the other directions and are discussed further in the following section.

C. Discretization of the Spatial Variable

The approximations that have been made thus far are independent of the treatment of the spatial variable and are identical to those used in other one-dimensional discrete ordinate transport codes.[6,12] In this section, we will depart from the traditional usage of the diamond difference scheme and develop a linear discontinuous finite element scheme for the spatial discretization. Use of the discontinuous scheme is based on the favorable experience of such a method in the two-dimensional, triangular mesh transport code TRIPLET.[4] These discontinuous methods are found to result in a very accurate and stable difference scheme (especially for optically thick mesh cells) that interacts well with the rebalance acceleration (see Sec. II.D.5.) algorithm.

Difference schemes for the transport equation fall into two broad categories: implicit and explicit methods. In an implicit method no attempt is made to solve in the direction in which neutrons are streaming. Instead, variational or weighted residual methods are used to determine a set of linear algebraic equations for all the unknowns. This set of equations is then solved, often by direct methods, to obtain the final solution. An implicit method couples all or some adjacent mesh cell with no regard for the direction of neutron travel. An explicit method, on the other hand, sweeps once through the mesh, solving for the unknowns in the direction in which neutrons are streaming. This is also equivalent to solving a set of linear algebraic equations, but here the matrix to be inverted is triangular. An explicit method couples only the mesh cells visible when looking backward along the direction in which neutrons are traveling. The finite element method developed below, like the diamond difference scheme, is explicit in nature.

Finite element methods for solving differential equations like Eqs. (21), (24), and (28) usually involve the assumption that the unknown function $\psi_m(r)$ can be approximated by some member of a finite-dimensional set of functions. This set of functions is often referred to as the trial space. A particular member of this function space is selected by some procedure like minimizing a functional or requiring the residual to be orthogonal to a set of weighting functions. The selected member is the desired approximate solution to the differential equation.

The finite element method used in ONETRAN is derived using a weight and integrate technique. The trial space consists of functions that are piecewise linear and discontinuous across mesh cell boundaries. More precisely, if $\psi_i(r)$ is our approximation of the exact solution to the discrete ordinates equation in mesh cell i (dropping the discrete ordinate index m), then we assume a linear Lagrangian representation of the form

$$\psi_i(r) \approx \frac{1}{\Delta r_i} [(r_{i+\frac{1}{2}} - r)\ \psi_{i-\frac{1}{2}} + (r - r_{i-\frac{1}{2}})\ \psi_{i+\frac{1}{2}}], \quad (31)$$

where $\Delta r_i = r_{i+\frac{1}{2}} - r_{i-\frac{1}{2}}$ and $\psi_{i-\frac{1}{2}}$, $\psi_{i+\frac{1}{2}}$ are the unknown discrete ordinates angular flux on the mesh cell left and right boundaries, respectively. To complete the specification of the trial space, we must assign a unique value to the approximate flux on the mesh cell boundaries. It is essential to the following analysis that the flux on the mesh cell boundary is the limit of the flux as one approaches the boundary in the direction in which neutrons are streaming. This is illustrated in Fig. 7 for the two possible cases, $\mu > 0$ and $\mu < 0$.

For the angular mesh cell, we impose continuity on the mesh cell edges and assume the diamond difference relation, so that the (angle) extrapolated angular flux is

$$\psi_{m+\frac{1}{2}} = 2\ [\tfrac{1}{2}(\psi_{i+\frac{1}{2}} + \psi_{i-\frac{1}{2}})] - \psi_{m-\frac{1}{2}}. \quad (32)$$

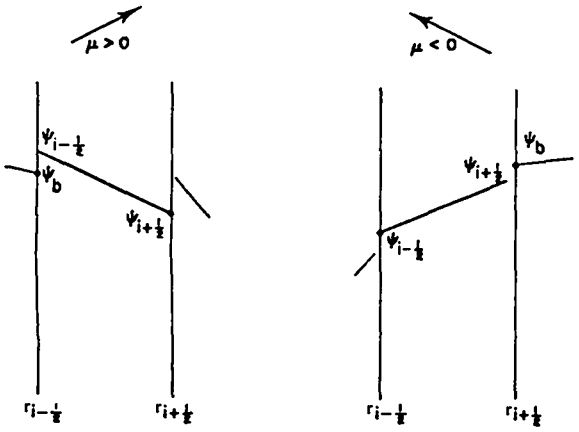


Fig. 7. Linear discontinuous representation of the angular flux in the i^{th} mesh cell. The • indicates the actual value of the angular flux on the mesh cell boundary. The angular flux from the previous mesh-cell boundary is denoted ψ_b.

The arrangement of the angular flux node points in a single space-angle mesh cell is illustrated in Fig. 8.

With the above assumptions inserted therein, the discrete ordinates equations Eqs. (21), (24), and (28) in the $(i,m)^{th}$ mesh cell become, respectively,

$$\frac{\mu_m}{\Delta r_i} \frac{d}{dr} [(r_{i+\frac{1}{2}} - r)\ \psi_{i-\frac{1}{2}} + (r - r_{i-\frac{1}{2}})\ \psi_{i+\frac{1}{2}}]$$

$$+ \frac{\sigma}{\Delta r_i} [(r_{i+\frac{1}{2}} - r)\ \psi_{i-\frac{1}{2}} + (r - r_{i-\frac{1}{2}})\ \psi_{i+\frac{1}{2}}]$$

$$\approx \frac{1}{\Delta r_i} [(r_{i+\frac{1}{2}} - r)\ S_{i-\frac{1}{2}} + (r - r_{i-\frac{1}{2}})\ S_{i+\frac{1}{2}}], \quad (33a)$$

$$\frac{\mu_m}{\Delta r_i} \frac{d}{dr} [r(r_{i+\frac{1}{2}} - r)\ \psi_{i-\frac{1}{2}} + r(r - r_{i-\frac{1}{2}})\ \psi_{i+\frac{1}{2}}]$$

$$+ \frac{\alpha_{m+\frac{1}{2}}}{w_m} [\psi_{i+\frac{1}{2}} + \psi_{i-\frac{1}{2}} - \psi_{m-\frac{1}{2}}] - \frac{\alpha_{m-\frac{1}{2}}}{w_m} \psi_{m-\frac{1}{2}}$$

$$+ \frac{\sigma}{\Delta r_i} [r(r_{i+\frac{1}{2}} - r)\ \psi_{i-\frac{1}{2}} + r(r - r_{i-\frac{1}{2}})\ \psi_{i+\frac{1}{2}}]$$

$$\approx \frac{1}{\Delta r_i} [r(r_{i+\frac{1}{2}} - r)\ S_{i-\frac{1}{2}} + r(r - r_{i-\frac{1}{2}})\ S_{i+\frac{1}{2}}] \quad (33b)$$

and

$$\frac{\mu_m}{\Delta r_i} \frac{d}{dr} [r^2(r_{i+\frac{1}{2}} - r)\ \psi_{i-\frac{1}{2}} + r^2(r - r_{i-\frac{1}{2}})\ \psi_{i+\frac{1}{2}}]$$

$$+ \frac{\alpha_{m+\frac{1}{2}}}{w_m} [\psi_{i+\frac{1}{2}} + \psi_{i-\frac{1}{2}} - \psi_{m-\frac{1}{2}}]2r - \frac{\alpha_{m-\frac{1}{2}}}{w_m} \psi_{m-\frac{1}{2}}\ 2r$$

$$+ \frac{\sigma}{\Delta r_i} [r^2(r_{i+\frac{1}{2}} - r)\ \psi_{i-\frac{1}{2}} + r^2(r - r_{i-\frac{1}{2}})\ \psi_{i+\frac{1}{2}}]$$

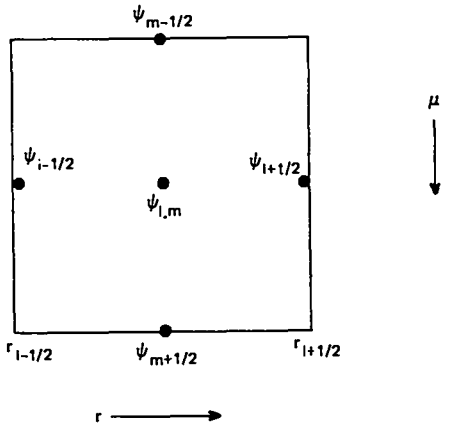


Fig. 8. Angular flux nodal values for the i,m (space, angle) mesh cell.

$$\approx \frac{1}{\Delta r_i}\left[r^2(r_{i+\frac{1}{2}} - r)\, S_{i-\frac{1}{2}} + r^2(r - r_{i-\frac{1}{2}})\, S_{i+\frac{1}{2}}\right],$$

$$\text{for} \qquad r\varepsilon(r_{i-\frac{1}{2}},\ r_{i+\frac{1}{2}}), \tag{33c}$$

where the sources on the right-hand side have also been approximated by a linear Lagrangian representation analogous to Eq. (31). In the spherical geometry Eq. (33c), the relation

$$\beta_{m\pm\frac{1}{2}} = 2\ \alpha_{m\pm\frac{1}{2}} \tag{34}$$

has been used, where the new curvature coefficients α satisfy the recursion relation of Eq. (25).

Since Eq. (33) cannot be satisfied identically for all $r\varepsilon(r_{i\frac{1}{2}},\ r_{i+\frac{1}{2}})$ we require the residual to be orthogonal to certain weight functions. Specifically, we operate on Eq. (33) with $\int_{r_{i-\frac{1}{2}}}^{r_{i+\frac{1}{2}}} dr$ and $\int_{r_{i-\frac{1}{2}}}^{r_{i+\frac{1}{2}}} (r - r_{i-\frac{1}{2}})\, dr$ for the rightward-directed sweeps ($\mu > 0$), and $\int_{r_{i-\frac{1}{2}}}^{r_{i+\frac{1}{2}}} dr$ and $\int_{r_{i-\frac{1}{2}}}^{r_{i+\frac{1}{2}}} (r_{i+\frac{1}{2}} - r)dr$ for the leftward-directed sweeps ($\mu < 0$). This results in the following system of equations for the mesh cell edge fluxes:

$$\begin{bmatrix} \Delta A_i \frac{\alpha_{m+\frac{1}{2}}}{w_m} + \sigma V_{i-\frac{1}{2}} & \mu A_{i+\frac{1}{2}} + \Delta A_i \frac{\alpha_{m+\frac{1}{2}}}{w_m} + \sigma V_{i+\frac{1}{2}} \\ \mu z_3 + z_5 \frac{\alpha_{m+\frac{1}{2}}}{w_m} + \sigma z_1 & \mu z_4 + z_5 \frac{\alpha_{m+\frac{1}{2}}}{w_m} + \sigma z_2 \end{bmatrix} \begin{Bmatrix} \psi_{i-\frac{1}{2}} \\ \psi_{i+\frac{1}{2}} \end{Bmatrix}$$

$$= \begin{Bmatrix} S_{i-\frac{1}{2}} V_{i-\frac{1}{2}} + S_{i+\frac{1}{2}} V_{i+\frac{1}{2}} + \Delta A_i \frac{\alpha}{w_m}\ \psi_{m-\frac{1}{2}} + \mu A_{i-\frac{1}{2}} \psi_b \\ S_{i-\frac{1}{2}} z_1 + S_{i+\frac{1}{2}} z_2 + z_5 \frac{\alpha}{w_m}\ \psi_{m-\frac{1}{2}} \end{Bmatrix}, \qquad \mu > 0, \tag{35a}$$

and

$$\begin{bmatrix} -\mu A_{i-\frac{1}{2}} + \Delta A_i \frac{\alpha_{m+\frac{1}{2}}}{w_m} + \sigma V_{i-\frac{1}{2}} & \Delta A_i \frac{\alpha_{m+\frac{1}{2}}}{w_m} + \sigma V_{i+\frac{1}{2}} \\ \mu z_8 + z_{10} \frac{\alpha_{m+\frac{1}{2}}}{w_m} + \sigma z_6 & \mu z_9 + z_{10} \frac{\alpha_{m+\frac{1}{2}}}{w_m} + \sigma z_7 \end{bmatrix} \begin{Bmatrix} \psi_{i-\frac{1}{2}} \\ \psi_{i+\frac{1}{2}} \end{Bmatrix}$$

$$= \begin{Bmatrix} S_{i-\frac{1}{2}} V_{i-\frac{1}{2}} + S_{i+\frac{1}{2}} V_{i+\frac{1}{2}} + \Delta A_i \frac{\alpha}{w_m} \psi_{m-\frac{1}{2}} - \mu A_{i+\frac{1}{2}} \psi_b \\ S_{i-\frac{1}{2}} z_6 + S_{i+\frac{1}{2}} z_7 + z_{10} \frac{\alpha}{w_m} \psi_{m-\frac{1}{2}} \end{Bmatrix}, \qquad \mu < 0. \tag{35b}$$

Here, $\alpha = \alpha_{m-\frac{1}{2}} + \alpha_{m+\frac{1}{2}}$, ψ_b is the angular flux on the boundary of the previous mesh cell as indicated in Fig. 7, and the remaining symbols are defined in Table V.

For the starting direction sweeps in cylindrical and spherical geometry there is no angular redistribution so that

$$\psi_{m+\frac{1}{2}} = \psi_{m-\frac{1}{2}} = \frac{1}{2}(\psi_{i-\frac{1}{2}} + \psi_{i+\frac{1}{2}}), \tag{36}$$

for m = starting direction. Since $\alpha_{m-\frac{1}{2}} = 0$, so that

$$\frac{\alpha_{m+\frac{1}{2}}}{w_m} - 0 = -\mu_m \tag{37}$$

for m = starting direction from Eq. (25), then

$$\frac{\alpha_{m+\frac{1}{2}}}{w_m} \psi_{m+\frac{1}{2}} = -\frac{1}{2}\, \mu_m(\psi_{i-\frac{1}{2}} + \psi_{i+\frac{1}{2}}) \tag{38}$$

for m = starting direction.

The relationship of Eq. (38) replaces the curvature terms in Eq. (33). We again operate with $\int_{r_{i-\frac{1}{2}}}^{r_{i+\frac{1}{2}}} dr$ and $\int_{r_{i-\frac{1}{2}}}^{r_{i+\frac{1}{2}}} (r_{i+\frac{1}{2}} - r)dr$. This results in

TABLE V

TABLE OF GEOMETRIC FUNCTIONS

Notation: $r_+ = r_{i+\frac{1}{2}}$, $r_- = r_{i-\frac{1}{2}}$

The i subscript is omitted from all quantities

Quantity	Plane Geometry	Cylindrical Geometry	Spherical Geometry
Δr	$r_+ - r_-$	$r_+ - r_-$	$r_+ - r_-$
A_-	1	$2\pi r_-$	$4\pi r_-^2$
A_+	1	$2\pi r_+$	$4\pi r_+^2$
ΔA	0	$A_+ - A_-$	$A_+ - A_-$
V_-	$\frac{1}{2}\Delta r$	$\frac{\pi}{3}\Delta r(r_+ + 2r_-)$	$\frac{\pi}{3}\Delta r(r_+^2 + 2r_+r_- + 3r_-^2)$
V_+	$\frac{1}{2}\Delta r$	$\frac{\pi}{3}\Delta r(2r_+ + r_-)$	$\frac{\pi}{3}\Delta r(3r_+^2 + 2r_+r_- + r_-^2)$
z_1	$10\Delta r$	$5(r_+ + r_-)\Delta r$	$(3r_+^2 + 4r_+r_- + 3r_-^2)\Delta r$
z_2	$20\Delta r$	$5(3r_+ + r_-)\Delta r$	$(12r_+^2 + 6r_+r_- + 2r_-^2)\Delta r$
z_3	-30	$-10(r_+ + 2r_-)$	$-5(r_+^2 + 2r_+r_- + 3r_-^2)$
z_4	$+30$	$10(4r_+ - r_-)$	$5(9r_+^2 - 2r_+r_- - r_-^2)$
z_5	0	$30\Delta r$	$20(2r_+ + r_-)\Delta r$
z_6	$20\Delta r$	$5(r_+ + 3r_-)\Delta r$	$(2r_+^2 + 6r_+r_- + 12r_-^2)\Delta r$
z_7	$10\Delta r$	$5(r_+ + r_-)\Delta r$	$(3r_+^2 + 4r_+r_- + 3r_-^2)\Delta r$
z_8	-30	$10(r_+ - 4r_-)$	$5(r_+^2 + 2r_+r_- - 9r_-^2)$
z_9	$+30$	$10(2r_+ + r_-)$	$5(3r_+^2 + 2r_+r_- + r_-^2)$
z_{10}	0	$30\Delta r$	$20(r_+ + 2r_-)\Delta r$

---

the following system of equations for the mesh cell edge fluxes:

$$\begin{bmatrix} -\frac{1}{2}\mu(A_{i-\frac{1}{2}} + A_{i+\frac{1}{2}}) + \sigma V_{i-\frac{1}{2}} & -\frac{1}{2}\mu\,\Delta A_i + \sigma V_{i+\frac{1}{2}} \\ \mu(z_8 - \frac{1}{2}z_{10}) + \sigma z_6 & \mu(z_9 - \frac{1}{2}z_{10}) + \sigma z_7 \end{bmatrix} \begin{Bmatrix} \psi_{i-\frac{1}{2}} \\ \psi_{i+\frac{1}{2}} \end{Bmatrix}$$

$$= \begin{Bmatrix} S_{i-\frac{1}{2}} V_{i-\frac{1}{2}} + S_{i+\frac{1}{2}} V_{i+\frac{1}{2}} - \mu A_{i+\frac{1}{2}} \psi_b \\ S_{i-\frac{1}{2}} z_6 + S_{i+\frac{1}{2}} z_7 \end{Bmatrix}, \text{ for} \qquad (35c)$$

μ = starting directions. It should be noted that by imposing continuity on the mesh cell boundary for the second equation in Eq. (35) ($\psi_{i-\frac{1}{2}} = \psi_b$ for $\mu > 0$ and $\psi_{i+\frac{1}{2}} = \psi_b$ for $\mu < 0$, the diamond difference equations are obtained. For curved geometries, these are a weighted diamond difference slightly different from that of Reed and Lathrop.[13]

For the diamond difference case in a source-free plane geometry mesh cell (S = 0), the solution of Eq. (35a) is easily shown to be $\psi_{i+\frac{1}{2}} = -\psi_b$ as the optical thickness of a mesh cell becomes infinitely large ($\sigma\Delta r/\mu \to \infty$). Thus, negative fluxes are a problem for such mesh cells. For the discontinuous case, it is easily shown the $\psi_{i-\frac{1}{2}} = \psi_{i+\frac{1}{2}} \to 0$ for mesh cells with an infinite optical thickness. Negative fluxes may still appear for the discontinuous case, but the worst possible situation occurs for $\sigma\Delta r/\mu = 8.196$ at which $\psi_{i-\frac{1}{2}} = +0.366\ \psi_b$ and $\psi_{i+\frac{1}{2}} = -0.0981\ \psi_b$. Thus, negative fluxes are much less in magnitude with the discontinuous scheme than for the diamond difference solution.

There are two important disadvantages to the linear discontinuous finite element scheme: computational times and core storage requirements. Although an explicit solution of Eq. (35) may be expressed, it is much more complicated than the corresponding diamond difference solution, requiring approximately twice as many operations (other than divides). Thus the computation costs will be greater than for other transport codes based on the diamond difference. Since the mesh cell edge values for the fluxes and sources are required (as compared to only the cell-centered quantities for a diamond difference code), the core storage for these quantities are doubled. In addition, all the finite element arrays for quantities in Table V must be stored on the fine mesh. It is possible that a choice of different weight functions in the derivation of Eq. (35) could result in a system both simpler to solve and requiring less core storage, but this has not been investigated.

On the other hand, it is found that a coarser spatial mesh is usually sufficient to give an accuracy equivalent to that obtained by a diamond difference solution.

D. Solution Algorithms

The basic algebraic equation that is actually solved by ONETRAN is Eq. (35) for each space-angle mesh cell. In this section we detail the algorithms used to implement the solution of Eq. (35) in the ONETRAN code.

1. Boundary Conditions

Information about the right and left boundary flux values may be specified by the ONETRAN user designating one of the following boundary conditions:

- Vacuum boundary condition -- the angular flux on the boundary is set to zero for all incoming directions: $\psi_{incoming}(\mu_m) = 0$.

- Reflective boundary condition -- the incoming angular flux on the boundary is set equal to the outgoing angular flux on that boundary in the direction corresponding to specular reflection: $\psi_{incoming}(\mu_m) = \psi_{outgoing}(-\mu_m)$.

- Periodic boundary condition -- the incoming angular flux on the boundary is set equal to the outgoing angular flux in the same direction on the opposite boundary:
  $\psi_{incoming}(r = r_{left}, \mu_m) = \psi_{outgoing}(r = r_{right}, \mu_m)$
  and
  $\psi_{incoming}(r = r_{right}, \mu_m) = \psi_{outgoing}(r = r_{left}, \mu_m)$.

- White boundary condition -- the incoming angular flux on the boundary is set equal to a single value such that the net flow through the boundary is zero, viz.,

$$\psi_{incoming}(\mu_m) = \frac{\sum_m w_m \mu_m \psi_{outgoing}(\mu_m)}{\sum_m w_m \mu_m} .$$

  where the sums range over all outgoing directions. This condition is used primarily for cell calculations in cylindrical and spherical geometry where it is applied to the outer radial boundary.

- Albedo boundary condition -- the incoming angular flux on the boundary is set equal a user-supplied albedo times the outgoing angular flux on that boundary in the direction corresponding to specular reflection:

$$\psi_{incoming}(\mu_m) = \alpha\ \psi_{outgoing}(-\mu_m).$$

  The albedo factor α is an energy group-dependent quantity. The reflective boundary condition corresponds to $\alpha = 1$.

- Inhomogeneous source boundary condition -- the incoming angular flux on the boundary is set

equal to a user-supplied source:

$$\psi_{incoming}(\mu_m) = Q_m.$$

The inhomogeneous boundary source is both group- and angle-dependent.

Use of the reflective or albedo boundary condition requires the S_N quadrature ordinates to be symmetric about $\mu = 0$.

At the start of each sweep of the spatial mesh (for a given discrete ordinate direction), the subroutine SETBC is called which returns the value of the boundary angular flux for that direction, namely ψ_b. This is the boundary flux, ψ_b, of the adjacent mesh cell as used in Eq. (35). Thus the equalities indicated above for the boundary conditions will not actually be true due to the discontinuity of the angular flux at the mesh cell boundary. Furthermore, for the reflecting boundary condition at the origin in cylindrical or spherical geometry, this reflecting boundary makes no contribution to the source term in Eq. (35) since $A_{i-\frac{1}{2}} = 0$.

2. Sweep of the Space-Angle Mesh

The unknown angular fluxes are ordered so that the difference scheme is stable and so that the coefficient matrix is lower triangular. Physically, this corresponds to proceeding in the direction of particle flow.

The angular mesh is swept in the same sequence in which the quadrature directions are ordered as indicated in Figs. 4, 5, and 6. For a particular quadrature direction, the spatial mesh is then swept either from left to right ($\mu > 0$) or from right to left ($\mu < 0$). For curved geometries, there are NLEV starting directions (ISN/2 for cylindrical, 1 for spherical). The angular flux, $\psi_{m-\frac{1}{2}}$, is generated on the fine mesh for all NLEV starting directions (stored in the array AFE). The angular mesh is then swept for all inward directions ($\mu < 0$). For each direction, the spatial mesh is swept from right to left, generating the mesh cell edge fluxes $\psi_{i-\frac{1}{2}}$, $\psi_{i+\frac{1}{2}}$ (stored in the array AFC). In each spatial mesh cell, the angular extrapolation is made by Eq. (32) (overstored in array AFE). The angular mesh is then swept for all outward directions ($\mu > 0$), with the spatial mesh now swept from left to right. This sweeping of the space-angle mesh is performed in subroutine INNER.

3. Negative Flux Fixup

As briefly mentioned in Sec. II.C., it is possible for the mesh cell edge angular fluxes in Eq. (35) to be negative, primarily in optically-thick mesh cells with no sources present. In such cases the offending angular fluxes are usually small in magnitude, are rapidly damped in space, and have little effect on integral quantities. Nonetheless, many transport code users become alarmed by the presence of these negative fluxes. Consequently, a negative flux fixup has been incorporated into ONETRAN, which results in an increase in computation time of (at least) approximately 3%. This is the cost of the test for negative fluxes only, and does not include the expense of computing the fixed-up flux.

The flux fixup algorithm proceeds as follows: (for rightward sweeps):

(a) After the mesh cell edge fluxes are computed, the far or extrapolated angular flux, $\psi_{i+\frac{1}{2}}$, is checked for positivity. (The near angular flux, $\psi_{i-\frac{1}{2}}$, will always be positive for positive sources.) If it is positive, the computation proceeds normally. If the total source in the mesh cell is negative, the flux fixup is bypassed.

(b) If $\psi_{i+\frac{1}{2}}$ is negative, the second equation of Eq. (35) is replaced with the requirement that this flux vanish, $\psi_{i+\frac{1}{2}} = 0$, and the first (conservation) equation of Eq. (35) is solved for the near angular flux, $\psi_{i-\frac{1}{2}}$.

If the computation time for a problem is significant and negative fluxes are not a serious problem, the negative flux fixup algorithm (in subroutine INNER) may easily be deleted.

It is also possible to generate negative fluxes with the angular diamond difference extrapolation of Eq. (32). This is not normally encountered and a fixup test and correction is not made.

4. Adjoint Problems

The ONETRAN program solves the adjoint transport equation by transposing the scattering and fission matrices and inverting the group order of the problem. Tranposition of the scattering matrix converts the normal, predominately downscattering problem to an upscattering problem while the group order inversion restores this downscattering dominance and eliminates unnecessary upscattering iteration. The solution of the resulting problem in the direction $\underline{\Omega}$ is then

identified with the adjoint solution in the direction $-\underline{\Omega}$.[14]

5. Iterative Processes

We assumed in writing Eq. (15) that the source $S(r,\underline{\Omega})$ is a known function. It is clear that if scattering or fission is present, this source function is not known but depends upon the moments of the angular fluxes being computed. In ONETRAN this source is generated in an iterative fashion using the latest flux information available. For the initial iteration a flux guess must be supplied as input that permits the generation of the source function. Sources in ONETRAN are approximted, like the angular flux, by the linear discontinuous Lagrangian representation of the form of Eq. (31). This requires the sources, $S_{i-\frac{1}{2}}$ and $S_{i+\frac{1}{2}}$, to be computed at the mesh cell edges.

In the following analysis we develop the iterative strategies used in ONETRAN for solving the discrete transport equation by writing these strategies for the analytic multigroup equations. We believe that details of the iteration process are clearer when presented in this manner, but the reader must keep in mind that all implied operations are actually performed in the discrete domain by methods described earlier in this report.

The multigroup transport equations can be written in operator notation as

$$L\,\vec{\psi} + \Sigma\,\vec{\psi} = (S_s + S_d + S_u)\vec{\psi} + F\,\vec{\psi} + \vec{Q}, \qquad (39)$$

where the matrix operators, L, Σ, S_s, S_d, S_u, and F represent streaming, collision, in-group or self-scattering, downscattering, upscattering, and fission processes, respectively in the g^{th} energy group. The form of the operators appearing in Eq. (39) can be inferred by comparing that equation with Eq. (12).

ONETRAN uses the standard dual iteration strategy for solving the discrete analog of Eq. (39). The two nested iterations are referred to as outer and inner iterations. The outer iteration represents a sweep through all the groups, while the inner iteration is performed within each energy group. Let us assume that the angular flux $\vec{\psi}^{k}$ is available from a previous outer iteration or from the input flux guess, if k = 0. The outer iteration then takes the form

$$L\,\vec{\psi}^{k+1} + \Sigma\,\vec{\psi}^{k+1} = (S_s + S_d)\vec{\psi}^{k+1} + (S_u + F)\vec{\psi}^{k} + \vec{Q}. \qquad (40)$$

Note that upscatter and fission sources are computed from the old flux $\vec{\psi}^{k}$ (in subroutine SOURCE) but that self-scatter and downscatter sources are computed using the new flux $\vec{\psi}^{k+1}$.

We can solve Eq. (40) for this new flux in the following manner. We first note that the matrix S_d is lower triangular, so that if the groups are solved in order beginning with the first group this term causes us no difficulty. That is, the downscatter source into group g involves only the new flux in groups g' such that g' < g. An effective source $\tilde{Q}_g^k$ to the g^{th} group can then be computed as

$$\tilde{Q}_g^k = (S_d\,\vec{\psi}^{k+1})_g + (S_u\,\vec{\psi}^{k})_g + (F\,\vec{\psi}^{k})_g + (\vec{Q})_g, \qquad (41)$$

where the notation $(\ )_g$ signifies the g^{th} component of the vector in parentheses. Having calculated $\tilde{Q}_g^k$, we must solve the following equation for the new flux ψ_g^{k+1} in the g^{th} group

$$L_g\,\psi_g^{k+1} + \Sigma_g\,\psi_g^{k+1} - S_{sg}\,\psi_g^{k+1} = \tilde{Q}_g^k. \qquad (42)$$

The operators L_g, Σ_g, and S_{sg} represent the g^{th} component of the diagonal matrix operators, L, Σ, and S_s. The above equation cannot be solved easily because of the presence of the self-scatter term, which couples all directions. The methods developed in Sec. II.B. and Sec. II.C. are capable of solving the discrete form of Eq. (42) if scattering sources are assumed known. Thus a second iteration, the inner iteration, is suggested. In ONETRAN, this iteration takes the form

$$(L_g + \Sigma_g)\psi_g^{k+1,\ell+1} = S_{sg}\,\psi_g^{k+1,\ell} + \tilde{Q}_g^k,$$

where ℓ is the inner iteration index. The discrete form of the operator $L_g + \Sigma_g$ can be inverted easily by a sweep through the space-angle mesh as described in Sec. II.D.2.

The inner and outer iterations have been described above for an inhomogeneous source problem. The inner iteration remains unchanged for a k_{eff} problem, but the outer iteration is altered slightly.

In place of Eq. (40) we have

$$L\,\vec{\psi}^{k+1} + \Sigma\,\vec{\psi}^{k+1} = (S_s + S_d)\vec{\psi}^{k+1} + (S_u + \frac{F}{\kappa_k})\vec{\psi}^{k}.$$

In the above equation we have divided the fission source by the parameter κ_k. This parameter is computed as

$$\kappa_k = \frac{\langle F\,\vec{\psi}^{k}\rangle}{\langle F\,\vec{\psi}^{k-1}\rangle}\,\kappa_{k-1},$$

with $\kappa_o = 1$ and $\langle\cdot\rangle$ representing an integration over group, angle, and space variables. The parameters κ_k approach k_{eff} for the system:

$$\kappa_k \to k_{eff} \text{ as } k \to \infty.$$

6. Convergence Acceleration Methods

In most problems the inner and outer iterations described above converge rapidly. There exist problems (primarily optically-thick regions with a scattering ratio near unity), however, for which these algorithms require excessive iterations for convergence to a satisfactory precision. The ONETRAN user is provided a choice of three methods for acceleration of the inner iterations:

- whole-system rebalance,
- coarse-mesh rebalance, or
- Chebyshev acceleration.

Each outer iteration is accelerated by coarse-mesh rebalance. The details of these techniques are described below. The recommended option for acceleration of the inner iterations is coarse-mesh rebalance. There are some problems in which coarse-mesh rebalance has a destabilizing effect on the inner iterations.[2] Experience with the TRIPLET code indicates this to be much less likely with a discontinuous spatial difference scheme than with the continuous diamond difference scheme. Should the coarse-mesh rebalance cause the inner iterations to diverge, or have minimal effect on the convergence of the inner iterations, then the Chebyshev acceleration is the recommended alternative. In the unlikely event that the inner iterations diverge with either Chebyshev or whole-system rebalance, the acceleration of the inners may be bypassed completely by appropriate choice of the input parameter IACC.

a. Rebalance Inner Iteration Acceleration

If the conservation equation, the first equation of Eq. (35), is multiplied by w_m and summed over all MM directions for the i^{th} mesh cell, the resulting equation (suppressing the m subscript on the angular fluxes)

$$\sum_{\mu_m<0} w_m|\mu_m|\,(A_{i-\frac{1}{2}}\,\psi_{i-\frac{1}{2}} - A_{i+\frac{1}{2}}\,\psi_{br})$$

$$+ \sum_{\mu_m>0} w_m\mu_m\,(A_{i+\frac{1}{2}}\,\psi_{i+\frac{1}{2}} - A_{i-\frac{1}{2}}\,\psi_{b\ell})$$

$$+ \sigma(V_{i-\frac{1}{2}}\,\phi_{i-\frac{1}{2}} + V_{i+\frac{1}{2}}\,\phi_{i+\frac{1}{2}})$$

$$= (S_{i-\frac{1}{2}}\,V_{i-\frac{1}{2}} + S_{i+\frac{1}{2}}\,V_{i+\frac{1}{2}})$$

$$+ \sigma^{o}_{s,g\to g}\,(V_{i-\frac{1}{2}}\,\phi^{p}_{i-\frac{1}{2}} + V_{i+\frac{1}{2}}\,\phi^{p}_{i+\frac{1}{2}}) \qquad (43)$$

is a statement of particle balance. Here we denote $\psi_{b\ell}$ and ψ_{br} as the m^{th} angular flux on the boundary of the left and right mesh cells, respectively, and $\phi_{i\pm\frac{1}{2}}$ as the mesh cell edge scalar fluxes, viz.,

$$\phi_{i\pm\frac{1}{2}} = \sum_{m=1}^{MM} w_m\,\psi_{i\pm\frac{1}{2}}. \qquad (44)$$

In Eq. (43) the self-scatter term has been separated out of the sources $S_{i\pm\frac{1}{2}}$ to indicate that this scattering source depends upon the flux from the previous iteration as denoted by the p superscript. All anisotropic sources (scattering or otherwise) vanish under the m summation due to orthogonality of the spherical harmonic functions, provided the quadrature set correctly integrates these functions. If the quadrature set does not correctly integrate these higher spherical harmonic polynomials, the numerical quadrature error term will appear as a nonphysical contribution to the source term in the balance equation, Eq. (43). If this quadrature error is greater than the input precision specified for problem convergence, then convergence to this precision will be unattainable.

The particle balance in Eq. (43) is satisfied only when $\phi = \phi^p$ (i.e., the converged solution). It has long been realized[15] that enforcing particle balance accelerates the iterative convergence. The object of the rebalance technique is to find a factor

by which all fluxes in a zone may be multiplied to ensure that leakage plus absorption equals sources in that zone. Usually, application of the factor quickly brings all flux amplitudes within the zone close to their final amplitude, and subsequent iterations merely refine the flux shapes in the zone.

To describe the terms entering the rebalance equation, we assume a coarse mesh superposed upon the fine mesh. In ONETRAN, the coarse mesh is taken to be the material mesh, and no special coarse mesh is used for the rebalance. For the k^{th} coarse-mesh zone we compute the leftward and rightward flows,

$$FL_{k-\frac{1}{2}} = \sum_{\mu_m<0} w_m |\mu_m| A_{k-\frac{1}{2}} \psi_{k-\frac{1}{2}}, \tag{45}$$

$$FR_{k+\frac{1}{2}} = \sum_{\mu_m>0} w_m \mu_m A_{i+\frac{1}{2}} \psi_{k+\frac{1}{2}}, \tag{46}$$

the zone effective absorption,

$$AB_k = \sum_{i\epsilon k} (\sigma_t - \sigma^o_{s,g\to g}) \sum_{m=1}^{MM} (V_{i-\frac{1}{2}} \psi_{i-\frac{1}{2}} + V_{i+\frac{1}{2}} \psi_{i+\frac{1}{2}}) w_m, \tag{47}$$

and the isotropic component of the zone source,

$$QQ_k = \sum_{i\epsilon k} \sum_{m=1}^{MM} (V_{i-\frac{1}{2}} S_{i-\frac{1}{2}} + V_{i+\frac{1}{2}} S_{i+\frac{1}{2}}) w_m \tag{48}$$

where $S_{i\pm\frac{1}{2}}$ is the nonself-scatter portion of S used in Eq. (43), i.e., the portion that does not change during the inner iteration. If boundary flux sources occur, they are placed in QQ_k in the zones adjacent to the boundary.

If all fluxes are now multiplied by the appropriate rebalance factor, f_k, we obtain the rebalance equation,

$$f_k (FL_{k-\frac{1}{2}} + FR_{k+\frac{1}{2}} + AB_k) = QQ_k + f_{k-1} FR_{k-\frac{1}{2}} + f_{k+1} FL_{k+\frac{1}{2}}, \tag{49}$$

by equating losses in the coarse-mesh zone (outflows plus absorption) to the sources (true source plus inflows from adjoining zones). This equation represents a tridiagonal system of equations for the rebalance factors, f_k, which may be solved directly by forward elimination-backward substitution. If the outer boundary condition is a vacuum condition, then the corresponding incoming flow is zero. For reflecting and white boundary conditions, we set the factor outside the boundary equal the factor just inside the boundary. For example, suppose the right boundary is reflecting. Then, at the boundary we set $f_{k+1} = f_k$ so that

$$f_k (FL_{k-\frac{1}{2}} + FR_{k+\frac{1}{2}} - FL_{k+\frac{1}{2}} + AB_k) = QQ_k + f_{k-1} FR_{k-\frac{1}{2}}. \tag{50}$$

The term $FR_{k+\frac{1}{2}} - FL_{k+\frac{1}{2}}$ is the net flow through the boundary and should vanish when the reflecting condition is satisfied. Such conditions are identically satisfied at the nonimplicit left boundary. With a periodic boundary condition, the outgoing flux on the left, say, is used as the incoming flux on the right. Thus we set f_o of Eq. (49) to f_{IM}. This results in a nontridiagonal system and an iterative solution for the rebalance factors is now required.

The above discussion described coarse-mesh rebalance acceleration. The ONETRAN user also has the option of using whole-system rebalance in which the entire system is assumed to be a single coarse-mesh zone. This single rebalance factor is easily seen to be the ratio of the total source to the net leakage plus the absorption.

b. Chebyshev Inner Iteration Acceleration

The ONETRAN user is also provided the alternative of using a modified form of Chebyshev acceleration[16] on the inner iterations. We can write the inner iteration of Eq. (43) in the form

$$\vec{\psi}^{\ell+1} = B \vec{\psi}^{\ell} + Q, \tag{51}$$

where ℓ is the inner iteration index and $B = (L + \Sigma)^{-1} S_s$ is the iteration matrix. The spectral radius of the iteration matrix, $\rho(B)$, is estimated by

$$\rho(B) \approx \frac{|| \epsilon^{\ell+1} ||}{|| \epsilon^{\ell} ||} \tag{52}$$

where the Euclidean norm of the error vector is calculated from the scalar flux as

$$||\varepsilon^{\ell}|| = \sqrt{\sum_i (\phi_i^{\ell} - \phi_i^{\ell-1})^2} \qquad (53)$$

and where the summation ranges over all spatial points. The relaxation factors, ω_ℓ, are then calculated recursively by

$$\omega_{\ell+1} = \frac{1}{1 - \frac{\rho^2(B)\ \omega_\ell}{4}} . \qquad (54)$$

It is known[16] that in the limit as $\ell \to \infty$, this Chebyshev relaxation factor becomes identical to the optimum relaxation factor of successive over-relaxation. The Chebyshev-accelerated scalar flux, $\tilde{\phi}^{\ell+1}$, is then given by

$$\tilde{\phi}^{\ell+1} = \omega_{\ell+1}\ \phi^{\ell+1} + (1 - \omega_{\ell+1})\ \phi^{\ell-1}. \qquad (55)$$

The Chebyshev acceleration is applied only to the scalar flux, but not the higher moments. The Chebyshev acceleration factor, which is group-dependent, is actually

$$AF = \frac{1}{\rho(B)} , \qquad (56)$$

and may be input at the option of the user.

In the Chebyshev acceleration in ONETRAN, the whole-system rebalance factor is calculated and applied to the flux moments prior to the calculation and application of the Chebyshev acceleration factors.

c. Rebalance Outer Iteration Acceleration

To accelerate the outer iteration, ONETRAN calculates a different set of coarse-mesh rebalance factors for each group. This outer iteration rebalance process is advantageous because it accelerates all types of problems, e.g., inhomogeneous source problems with upscatter and/or fission, or eigenvalue problems with or without upscatter. These outer rebalance factors are group-dependent and calculated by the flows and absorption defined in Sec. II.D.6.a.

The source for the outer rebalance consists of the inhomogeneous source (if any) plus the scattering source (if any) plus the fission source (if any). If there is a fission source present, a source iteration is performed to determine the outer rebalance factors. If there is no inhomogeneous source present, this source iteration can also be used to estimate the eigenvalue, say k_{eff}. In this case, we replace the inhomogeneous source, QQ_k, of the rebalance equation, Eq. (49), with the fission source plus scattering source, viz.,

$$f_{k-1}^{m}\ FR_{k-\frac{1}{2}} + f_k^m(\ FL_{k-\frac{1}{2}} + \ FR_{k+\frac{1}{2}} + AB_k)$$

$$+ f_{k+1}^{m}\ FL_{k+\frac{1}{2}} = (REV * FS_k + SS_k) f_k^{m-1}. \qquad (57)$$

Here FS and SS are the fission and scattering source in the k^{th} coarse mesh zone and REV is the outer rebalance eigenvalue. The m superscript is the index for this power iteration to determine the rebalance factors, which are now the eigensolutions of Eq. (57).

7. Convergence Tests

There are three levels of iterative processes in the ONETRAN program:

(1) the inner iteration in which the within-group scattering source and/or the boundary flux at an implicit boundary changes,

(2) the outer iteration in which the fission or upscattering source changes or which is necessitated by incompletely converged inner iterations (usually in slowly convergent inhomogeneous source problems), and

(3) the parametric eigenvalue search iteration in which, after a converged outer iteration, the value of a material concentration, a coarse-mesh boundary, or a time absorption (see Sec. III.B.9.) is changed.

Two additional iterations are also required: iteration for the coarse-mesh rebalance factors when the periodic boundary condition is present (in subroutine REBAL) and the power iteration on the fission source for the outer iteration rebalance factors (in subroutine GREBAL).

All of the iterative processes are compared to various convergence precisions to terminate the iterations. These convergence precisions are:

EPSI	Inner iteration convergence precision. This convergence precision is an input parameter and is set to 10^{-4} if a zero (blank) is entered.
EPSO=EPSI	Outer iteration convergence precision.
EPSX=(1+IGM*$e^{-100\ EPSI}$)*EPSI	Outer iteration rebalance factor convergence precision.
EPST=10^{-2}	Chebyshev norm convergence precision.
XLAX	Search lambda convergence precision for parametric eigenvalue searches, an input parameter. Default value if not specified on the input: XLAX=10*EPSI.
EPSR=10^{-1}*EPSI	Rebalance factor iteration convergence precision for periodic boundary condition.

For the inner iteration process, the iterations are terminated when

$$\max_{i} \left| 1 - \frac{\phi^{\ell-1}_{i+\frac{1}{2}}}{\phi^{\ell}_{i+\frac{1}{2}}} \right| < \text{EPSI},$$

where $\phi^{\ell}_{i+\frac{1}{2}}$ is the i^{th} mesh cell edge scalar flux for the ℓ^{th} inner iteration after application of the rebalance factors. If the number of inner iterations exceeds the value of IITL, an input variable, the inner iterations are terminated. When $|1 - \lambda| <$ 10*EPSO (see below for definition of λ), then IITL is switched to IITM, another input variable.

For the Chebyshev acceleration of the inner iterations, the first few estimates of the spectral radius, $\rho(B)$ in Eq. (52), may be inaccurate and lead to unstable accelerations. Consequently, the Chebyshev acceleration is not applied to the scalar flux until the change in this spectral radius has stabilized and is less than EPST,

$$|\rho^{\ell+1}(B) - \rho^{\ell}(B)| < \text{EPST}.$$

Both of these convergence tests on the inner iterations are made in subroutine INNER.

In determination of convergence of the outer iterations, ONETRAN calculates the parameter

$$\lambda^k = \frac{\text{Fission source}^k + \text{Inhomogeneous Source}}{\text{Fission source}^{k-1} + \text{Inhomogeneous Source}} \quad (58)$$

for the k^{th} outer iteration. Thus $\lambda < 1$ for a subcritical system, $\lambda = 1$ for a critical system, and $\lambda > 1$ for a supercritical system. The outer iterations are terminated when

$$|1 - \lambda| < \text{EPSO} \quad \underline{\text{and}} \quad \max_{i,g} |1 - f_{ig}| < \text{EPSX}, \quad (59)$$

where f_{ig} are the outer iteration coarse-mesh rebalance factors for group g and coarse-mesh zone i.

For the power iteration on the fission source for the outer rebalance factors as described in Sec. II.D.6.c., we terminate the iteration when

$$\max_{i,g} \left| 1 - \frac{f^{n}_{ig}}{f^{n-1}_{ig}} \right| < \text{EPSX}$$

for inhomogeneous source problem with upscatter but no fission, or

$$\left| 1 - \frac{\lambda^{n}_{x}}{\lambda^{n-1}_{x}} \right| < \text{EPSX}$$

for inhomogeneous source problems with fission or eigenvalue searches (IEVT>1), or

$$\left| 1 - \lambda^{n}_{x} \right| < \text{EPSX}$$

for k_{eff} calculations (IEVT=1). Here we denote n as the index for this fission source power iteration and λ_x the same ratio of Eq. (58) for each of these power iterations. These above outer iteration convergence tests are all performed in subroutine GREBAL.

For parametric eigenvalue searches, the outer iterations are continued for the initial system (i.e., the system described by the initial eigenvalue guess) until

$$\left| \lambda^k - \lambda^{k-1} \right| < \text{EPSO},$$

at which time the initial eigenvalue is adjusted as described in Sec. III.B.9. For all subsequent systems, the outer iterations are continued until

$$\left| \lambda^k - \lambda^{k-1} \right| < \text{XLAX}$$

before the eigenvalue is again modified. The

eigenvalue modifications will continue until the outer iteration convergence criteria of Eq. (59) are finally satisfied. These parametric eigenvalue convergence tests are performed in subroutine NEWPAR.

Finally, for problems with periodic boundary conditions, the rebalance factor iteration is terminated when

$$\max_i \left| 1 - \frac{f_i^j}{f_i^{j-1}} \right| < \text{EPSR},$$

where f_i^j is the i^{th} coarse-mesh reblance factor for this j^{th} iteration. This convergence test is performed in subroutine REBAL.

In difficult problems with a large amount of upscattering and fission, it is frequently found that convergence of the outer rebalance factor, the second criteria of Eq. (59), is the most difficult condition to satisfy. Many times the eigenvalue will be accurately converged, yet the fluxes will still not be in very good balance. In such cases, it may become necessary to limit the number of outer iterations (with the input parameter OITM) or modify the code so that EPSX is a larger multiple of EPSI.

III. A GUIDE TO USER APPLICATION

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

A. Overall Program Flow

A schematic flow chart for ONETRAN is given in Fig. 9. The subroutine names in which that computation is performed is indicated beside each block.

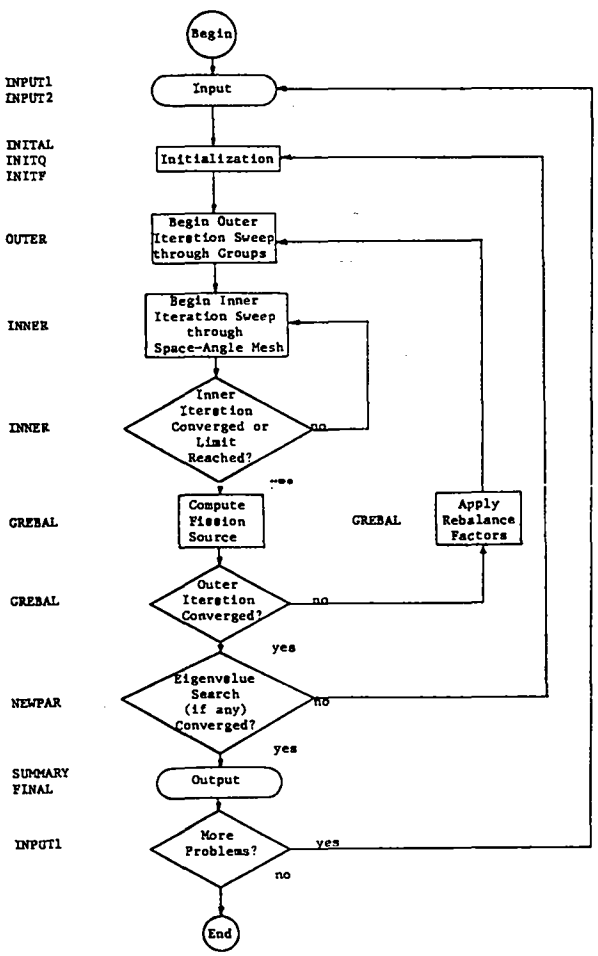


Fig. 9. Simplified logical flow diagram for ONETRAN.

B. Details of Program Options

1. Cross Sections

a. Input Formats

The ONETRAN program accepts cross sections either from the standard file ISOTXS,[5] in FIDO format,[6] or in the standard Los Alamos format as described in this section. In upscattering problems, the program does not need the special σ^{up} cross section which is required in earlier Los Alamos programs.[12] In ONETRAN, it is assumed that σ^{up} is NOT present, but σ^{up} is automatically removed from the card input cross section sets if the user tags the input number IHS with a minus sign. Note that this is the opposite procedure for removal of σ^{up} from that used in other LASL transport codes.[3,4] Cross sections read with the FIDO format may not contain σ^{up}.

The Los Alamos cross section format assumes that each nuclide is described by a block of cross sections of IHM rows for IGM group columns. The row position of cross sections is specified relative to the total cross section, σ_t (row IHT), and the within-group scattering cross section, $\sigma_{s,g\to g}$, (row IHS). It is assumed that the row order of the cross sections is as follows:

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

In this format, group g+1 corresponds to a group of lower energy than group g. The symbol $\sigma_{s,g-2\to g}$ denotes the scattering transfer probability from group g-2 to group g. The format allows N groups of upscatter and M groups of downscatter; i.e., the scattering matrix need not be square. However, all cross section blocks must have the same values for IHM, IHS, and IHT. The fission cross section, σ_f, times the mean number of neutrons per fission, ν, must be located in row IHT-1, and the absorption cross section, σ_a, must be entered in row IHT-2. The transport cross section, σ_{tr}, must be entered in position IHT-3 if the transverse buckling correction is to be made using σ_{tr} rather than σ_t as detailed in Sec. III.B.2.c. The (n,2n) scattering cross section, $\sigma_{n,2n}$, must be entered in position IHT-4 if the scattering matrix is used to represent (n,2n) reactions, as detailed in Sec. III.B.1.g. The user is free to enter additional cross sections at the top of the format. These extra cross sections are not used in the calculation, but are used for reaction-rate computations in the flux edits.

b. Cross Section Mixing

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

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

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

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

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

For this example we have MS = 17 instructions. In the first three instructions, block 46 is cleared (set to zero) and then made up of 0.0478 parts of block 1 and 0.0333 parts of block 20. If block 1 and 20 are microscopic cross sections in barns, then 0.0333 and 0.478 times 10^{24} are the atomic densities. In the second set of instructions, block 47 is cleared and then made up of 0.1179 times the result of adding three-fourths of block 2 to one-fourth of block 3. In the next set of instructions, block 48 is cleared and made up of portions of blocks 15 and 14. If IEVT (the input eigenvalue type option) is 3, then the resulting block 48 is multiplied by EV (the input eigenvalue guess). In this type of problem the program attempts to find a value of EV such that the resulting concentration of block 48 renders the system critical. If IEVT ≠ 3, the line of instructions 48, 48,

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

0.0 would not alter the composition of block 48. In the final sequence, block 49 is made up of 0.187 times one-half of block 33 and block 34; provision is made to search for the concentration of this portion of 49 to which is always added 0.1 of the previously mixed block 46. It should be clear that there are many possibilities not covered in this example, but by examining the FORTRAN instructions above, the user should be able to prepare his own set of mixture instructions.

c. Anisotropic Cross Sections

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

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

where ISCT is an input control integer. Thus if ISCT > 0, additional blocks of scattering transfer cross sections must be entered for those nuclides for which anisotropic scattering sources are to be computed. Note that the anisotropic scattering blocks do NOT contain the (2n+1) factor as in some transport codes.[6] Should the cross section blocks contain this factor, they may easily be removed via the mixing tables. In these blocks, the rows 1 through IHT are zero, and $\sigma^n_{s,g\to h}$ (the energy average of $\sigma_s^n(E' \to E)$ in groups g and h) is entered as for the isotropic component of the cross section. It is assumed in ONETRAN that blocks of anisotropic cross sections which are used in the calculation have block numbers in ascending sequence, starting with the isotropic cross section block. For example, suppose that block 50 is the isotropic cross section block for hydrogen and that ISCT = 3. Then, block 51 must be σ_s^1 for hydrogen, block 52 must be σ_s^2, and block 53 must be σ_s^3. If a material is made by mixing two anisotropic scatterers, then the anisotropic blocks must also be mixed with the same densities to form anisotropic blocks for the material. In each zone in which anisotropic scattering sources are computed the number of anisotropic scattering blocks must be the same, namely ISCT.

d. Adjoint Cross Sections

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

$$(\sigma_a)_{eff} = \sigma_t - \sum_{\text{all } h} \sigma^0_{s,g\to h}. \qquad (61a)$$

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

$$(\sigma_a)_{eff} = \sigma_t - \sum_{\text{all } h} \sigma^0_{s,h\to g}. \qquad (61b)$$

e. Cross Section Checking

As input cross sections are processed in subroutine CSPREP, the effective absorption of Eq. (61) is computed and compared to the input value of σ_a. If the relative difference between the input total cross section and the computed total cross section exceeds EPSI (inner convergence precision), an error message is printed. However, the computation will proceed normally using the input absorption cross section with no attempt being made to correct this inconsistency.

f. Fission Fractions

The ONETRAN user may specify the fission fractions as either a spectrum (χ_g: the probability of a fission in any group releasing a neutron in group g) or a matrix ($\chi_{h\to g}$: the probability of a fission in group h releasing a neutron in group g). These fission fractions may also be coarse-mesh-dependent. The fission fractions are conventionally normalized to $\sum_g \chi_g = 1$ or $\sum_g \chi_{h\to g} = 1$. This normalization is not checked by ONETRAN and any lack of normalization will be reflected proportionally in k_{eff}.

The fission fractions are specified by the input parameter IFISS:

IFISS	Option
1	A single fission spectrum for the entire system
2	A fission spectra for each of the IM coarse-mesh regions.
3	A single fission matrix for the entire system.
4	A fission matrix for each of the IM coarse-mesh regions.

For coarse-mesh-dependent fission spectra (IFISS=2), the fission fractions are ordered as:

$$[\chi_1 \cdots \chi_{IGM}]_{i=1} \cdots\cdots\cdots [\chi_1 \cdots \chi_{IGM}]_{i=IM},$$

and loaded as a single block. For coarse-mesh-dependent fission matrices (IFISS=4), the fission fractions are ordered as:

$$\begin{bmatrix} [\chi_{1\to1}\cdots\chi_{IGM\to1}]_{i=1}\cdots[\chi_{1\to1}\cdots\chi_{IGM\to1}]_{i=IM} \\ [\chi_{1\to IGM}\cdots\chi_{IGM\to IGM}]_{i=1}\cdots[\chi_{1\to IGM}\cdots\chi_{IGM\to IGM}]_{i=IM} \end{bmatrix},$$

and each row is loaded as a single block.

g. (n,2n) Reactions

The ONETRAN user may utilize the scattering matrices to represent (n,2n) reactions by flagging the input parameter IHT negative. If $\sigma_{(n,2n)h\to g}$ is the reaction cross section for a neutron in group h releasing two neutrons in group g, then $2 * \sigma_{(n,2n)h\to g}$ must be entered as the scattering transfer matrix, $\sigma_{s,h\to g}$, in order to obtain the proper neutron multiplication in the scattering computation by ONETRAN. The total (n,2n) reaction cross section,

$$\sigma_{n,2n} = \sum_g \sigma_{(n,2n)h\to g},$$

must then be entered in cross section position IHNN = IHT - 4. This cross section is then used to correct the group sum of the outscatter term in the system balance tables.

If IHT is not flagged negative, ONETRAN assumes no (n,2n) reactions are present and cross section position IHNN may be used for any other cross section to be used in the reaction-rate computations in the flux edits.

h. Fine-Mesh Density Factors

The ONETRAN user has the option of specifying fine-mesh density factors to describe a pointwise spatial variation of the macroscopic cross sections. Thus, the macroscopic cross section is multiplied by DEN(I) whenever the cross section is required in mesh cell I. These density factors are very useful in problems such as air transport calculations where a single material is present but with a continuously varying spatial density.

2. Geometry and Boundary Condition Specifications

a. Spatial Mesh

To specify the spatial domain on the problem, user supplies IM+1 coarse-mesh boundaries (defining IM intervals). Except for the first radius in cylindrical or spherical geometries, this set need not begin at 0.0, but must form a monotone increasing sequence. The user also supplies IM integers which indicate how many fine-mesh intervals are in each coarse-mesh interval. The fine-mesh spacing is uniform between the coarse-mesh boundaries. This results in a total of IT fine-mesh intervals, indexed from left to right. Finally, the user specifies the boundary condition on the left and right boundaries.

The coarse-mesh boundaries define IM zones. The user must supply a number for each of these zones (IDC array) to designate which cross section block belongs in the zone. That is, the material mesh is identical to the coarse mesh. If anisotropic scattering is desired in any zone, the block number for that zone is flagged negative, otherwise the scattering is computed as isotropic. This indicates that the next ISCT blocks in numerical sequence contain the anisotropic scattering cross sections for this zone. The scattering will be computed as isotropic if ISCT > 0 and IDC is not flagged negative or if ISCT = 0 (regardless of the sign on IDC).

All of the above information is converted by subroutine MAPPER into a pictorial description of the system.

b. Boundary Conditions

The ONETRAN user must select one of the following five boundary conditions for each of the system boundaries:

- Vacuum boundary condition -- the angular flux on the boundary is set to zero for all incoming directions.

- Reflective boundary condition -- the incoming angular flux on the boundary is set equal to the outgoing flux in the direction corresponding to specular reflection.

- Periodic boundary condition -- the incoming angular flux on the boundary is set equal to to outgoing flux on the same direction on the opposite boundary.

- White boundary condition -- the incoming angular flux in the boundary is set equal to the single value such that the net flow through the boundary is zero, namely:

$$\psi_{\text{incoming}}(\mu_{m'}) = \frac{\sum_{m} w_m \mu_m \psi(\mu_m)_{\text{outgoing}}}{\sum_{m} w_m \mu_m},$$

  where the sums range over all outgoing directions. This condition is used primarily for cell calculations in cylindrical and spherical geometry where it is applied to the outer radial boundary.

- Albedo boundary condition -- the incoming angular flux on the boundary is set equal to a user-supplied albedo times the outgoing flux in the direction corresponding to specular reflection.

Use of reflective or albedo boundary conditions requires the S_N quadrature set to be symmetric about $\mu = 0$.

c. Buckling Absorption

Leakage from the transverse dimensions of a multi-dimensional system may be simulated by a user-specified buckling height and width (for plane geometry only). These buckling dimensions must be in units consistent with the cross sections (in cm if cross sections are in cm^{-1}). If diffusion theory is assumed adequate then the flux shape in the transverse direction z is of the form $\cos \pi z/\tilde{h}$, so that the flux vanishes at the extrapolated half-heights $\pm \tilde{h}/2$. If this assumption is substituted into the multi-dimensional form of the transport equation, Eq. (1), then the transverse leakage appears as a buckling absorption cross section of the form

$$\sigma_{a,BHT} = \frac{\sigma}{3} \left(\frac{\pi}{\sigma * BHT + 1.4209}\right)^2,$$

where σ is the total cross section, BHT is the actual buckling transverse dimension (height or width), and $1.4209/\sigma$ is twice the Milne problem extrapolation distance. If the input buckling height (BHGT) is flagged negative, then the transport cross section, σ_{tr}, is assumed to be in cross section position IHTR = IHT - 3. The extrapolation distance of $1.4209/\sigma_{tr}$ is then used so that the buckling absorption is

$$\sigma_{a,BHT} = \frac{\sigma}{3} \left(\frac{\pi}{\sigma * BHT + 1.4209\ \sigma/\sigma_{tr}}\right)^2.$$

The buckling absorption is added to both the total cross section (CT) and absorption cross section (CA) arrays in subroutine INITAL. Consequently, the absorption in the output coarse mesh balance table also contains this buckling absorption. The activities computed in the final edits do not contain this buckling absorption.

If BHGT is not flagged negative, then σ_{tr} is assumed to not be present and cross section position IHTR may be used for any other cross section to be used in reaction-rate computations in the flux edits.

3. Angular Quadrature Coefficient Specifications

The ONETRAN user has the option of obtaining the angular quadrature coefficients from interface file ISNCON,[5] one of two built-in sets in subroutine SNCON, or from card input. The input parameter IQUAD specifies the source of these coefficients. The number of quadrature coefficients (MM) is determined from the input S_N order parameter ISN and the geometry type specification (IGEOM) as

$$MM = \begin{cases} \text{ISN for plane and spherical geometry} \\ \text{ISN*(ISN+2)/4 for cylindrical geometry (IGEOM=2), or} \\ \text{ISN*(ISN+2) for two-angle plane geometry (IGEOM=4).} \end{cases}$$

The built-in constants are either the P_N (Gaussian) quadrature constants for: S_2, S_4, S_6, S_8, S_{12}, S_{16}, S_{20}, S_{24}, S_{32}, or S_{48}; or the DP_N (double Gaussian) quadrature constants for: S_4, S_8, S_{12}, S_{16}, S_{24}, S_{32}, S_{40}, S_{48}, S_{64}, or S_{96}. For most problems, the P_N set is the recommended set. However, for thin-slab problems in which the angular representation of the leakage flux is important, use of the DP_N quadrature set is recommended.

For problems with anisotropic scattering, it is important that the S_N order be chosen sufficiently

large such that the spherical harmonic polynomials are correctly integrated. Otherwise, the numerical quadrature error may introduce a nonphysical contribution to the neutron balance, preventing convergence of the problem to the desired precision.

For user input S_N constants, it is necessary that they be correctly ordered as illustrated in Sec. II.B.2. In addition, if the sums $1 - \sum_m w_m$, and $\sum_m \mu_m$, and $\sum_m w_m \mu_m$ are greater than 10^{-5}, an error message is printed.

4. Source Options

The ONETRAN user may specify an anisotropic distributed source and the boundary flux at either boundary of the system. The inhomogeneous distributed source must be represented by a finite spherical harmonic expansion of the form

$$Q(r,\underline{\Omega}) = \sum_{n=1}^{NMQ} (2n-1)\, R_n(\underline{\Omega})\, Q_n(r), \qquad (62)$$

where the energy group index has been omitted. For standard plane or spherical geometry, the moments of the source are

$$Q_n(r) = \frac{1}{2}\int_{-1}^{1} d\mu\, P_n(\mu)\, Q(r,\mu), \qquad (63)$$

and for cylindrical and two-angle plane geometry

$$Q_n(r) = \frac{1}{4\pi}\int_{-1}^{1} d\xi \int_{0}^{2\pi} d\phi\, P_n(\xi)\, Q(r,\xi,\phi) \qquad (64)$$

and

$$Q_n^{\ell}(r) = \frac{1}{4\pi}\sqrt{2\,\frac{(n-\ell)!}{(n+\ell)!}}\int_{-1}^{1} d\xi \int_{0}^{2\pi} d\phi\, P_n^{\ell}(\xi) \begin{Bmatrix} \cos \ell\phi \\ \sin \ell\phi \end{Bmatrix} Q(r,\xi,\phi), \qquad (65)$$

as defined in a similar fashion for the flux moments in Sec. II.A.2. The anisotropic source components are entered in the order indicated in Tables III and IV.

When using the anisotropic distributed source option, the order of anisotropic scattering, ISCT, must be at least as large as IQAN so that the requisite number of spherical harmonics, $R_n(\underline{\Omega})$, are computed.

The ONETRAN user is also allowed to specify the incoming flux on either boundary. This boundary source is specified for the MM/2 incoming directions in the same order as the S_N quadrature ordinates as illustrated in Sec. II.B.2.

5. Source Input Options

If a distributed source of anisotropy IQAN is designated, then

$$NMQ = \begin{cases} IQAN+1 \text{ for plane and spherical geometry, or} \\ (IQAN+2)^2/4 \text{ for cylindrical geometry, or} \\ (IQAN+1)^2 \text{ for two-angle plane geometry,} \end{cases}$$

components (spherical harmonic moments) of the source must be entered for each group in the order listed in Tables III and IV. The complete dimensions of the inhomogeneous distributed source for a single group are Q(NMQ,2,IT). Appropriate choice of the source input parameter IQOPT will reduce the amount of input required as specified below.

Boundary sources may also be specified by setting the input boundary source triggers IQL=1 and/or IQR=1 for the left and/or right boundary sources, respectively. This requires the input of the boundary sources for all MM/2 incoming directions and for each group. For IQOPT positive or zero, the complete boundary sources at each direction for each group are required input. For IQOPT negative, the energy spectra of the boundary sources are required input, and the boundary sources are assumed isotropic in angle.

The ordering of the source input is:

1. Distributed sources (if any); for all groups of an anisotropic order; for all orders of source anisotropy, then
2. Left boundary sources (if any), right boundary sources (if any); for all groups.

One can imagine the sources to be read by the following FORTRAN statements:

```
      DO      10      N=1,NMQ
      DO      10      G=1,IGM
10    READ            ((Q_G(N,K,I),K=1,2),I=1,IT)
      DO      12      G=1,IGM
      READ            (QL_G(M),M=1,MM/2)

12    READ            (QR_G(M),M=1,MM/2)
```

The IQOPT parameters available to the user are:

IQOPT	Option
0	Zero distributed source (no input)
± 1	Energy spectrum for the distributed source: EQ(IGM). One spectrum for each NMQ component.
± 2	Flat distributed source on the fine mesh: Q(IT). One distribution for each group and each NMQ component.
± 3	Linear distributed source on the fine mesh: Q(2,IT). The first subscript is the left edge and the right edge sources, respectively. One distribution for each group and for each NMQ component.
± 4	A single energy spectrum for the distributed source: EQ(IGM), followed by a single linear distributed source on the fine mesh: Q(2,IT). The distributed source is formed by the product of the energy spectrum and the fine-mesh spatial distribution. One spectrum and one spatial distribution for each NMQ component.
5	Input of both distributed and boundary sources from standard interface file FIXSRC mounted on unit IFIXSR.

6. Flux Input Options

Options for reading the input flux guess are specified by the input integer ISTART. If ISCT is the order of anisotropic scattering, then

$$NM = \begin{cases} ISCT+1 \text{ for plane and spherical geometry,} \\ (ISCT+2)^2/4 \text{ for cylindrical geometry, or} \\ (ISCT+1)^2 \text{ for two-angle plane geometry,} \end{cases}$$

spherical harmonic components of the angular flux must be specified, ordered as in Tables III and IV.

The ISTART options available to the user are:

ISTART	Option
0	A unit fission guess is automatically supplied in every mesh cell. No input required.
± 1	Energy spectrum: EQ(IGM). NM spectra input for each component for ISTART=+1; one spectrum input for the scalar flux component only for ISTART=-1, the higher components being assumed zero.
± 2	Flat distributed flux distribution on the fine mesh: F(IT). NM distributions input for each component for ISTART=+2; one distribution input for the scalar flux component only for ISTART=-2, the higher components being assumed zero.
3	A problem restart dump is read from unit NDMP1. See Sec. III.B.7.
± 4	The scalar flux guess is read from standard interface file RTFLUX or ATFLUX mounted on unit ITFLUX for ISTART=+4. The complete angular flux is read from standard interface file RAFLUX or AAFLUX mounted on unit IAFLUX for ISTART=-4. If the interface file output is requested (IFO=1), these flux files will be overwritten with the computed fluxes at the end of the problem.

7. Flux Dumps and Restart Procedures

The three types of dumps that are taken have the same form, and each may be used to restart a problem. A periodic dump is taken every DUMPT minutes where DUMPT is a program variable which can be set to meet particular installation requirements in the main program segment. A final dump is always taken after the successful completion of a problem, and a time limit dump is taken after a user-specified period of time (ITLIM). Dumps are written alternately on units NDMP1 and NDMP2 depending on which is free; an output message is written to indicate which unit contains the latest dump.

When problem execution is continued using a restart dump, certain input parameters can be changed and edit specifications can be added or modified. It is possible to use the program to edit a dump.

To restart a problem, the first card (only) of input control integers is read, with ISTART=3. All other control integers on this card are ignored. ONETRAN will then read the dump file, restoring both small core and large core memories to their contents at the time the dump was taken.

The second card input contains the following changed values of the control integers (on a 6I6 format):

IACC - Acceleration option

OITM - outer iteration limit

IITL - inner iteration limit until $|1-\lambda|<10*EPSO$

IITM - inner iteration limit after $|1-\lambda|<10*EPSO$

IEDOPT - edit option trigger

IFO - interface file output trigger.

The acceleration option cannot be changed to

Chebyshev (IACC=3) if the problem was not originally run with that option.

The third card input contains the following changed value of the floating point parameter (on an E12.6 format):

EPSI - inner iteration convergence precision

All further input (excluding edit) and problem initiation is then bypassed and execution is resumed in the subroutine OUTER for the current group at the time the dump was taken.

If the edit option trigger is on (IEDOPT=1), all edit option input must follow the restart input on the standard card input. Any edit input included in the original problem run is not saved.

8. Iteration Acceleration Options

The user is provided a choice of four methods for acceleration of the inner (within group) iterations by the IACC input parameter. These options are:

1. No acceleration -- recommended only if all other options fail,
2. System rebalance -- particle balance is enforced over the entire system,
3. Coarse-mesh rebalance -- particle balance is enforced over each coarse-mesh zone. This is the recommended option for most rapidly obtaining the converged solution, or
4. Chebyshev acceleration -- the Chebyshev semi-iterative scheme is used to accelerate the scalar flux after application of the system rebalance factor. For some problems in which coarse-mesh rebalance is very slow to converge or even divergent, this option is the recommended alternative.

Coarse-mesh rebalance is always performed to accelerate the outer iterations.

9. Eigenvalue Searches

It is possible in ONETRAN to perform an eigenvalue search on nuclide concentration (concentration search), system dimensions (delta search), or the time absorption (alpha search) to achieve a desired value of k_{eff}, normal unit (a critical system). The type of eigenvalue search is chosen by the input parameter IEVT as follows*:

IEVT	Type of Eigenvalue Search
2	Time absorption (alpha)
3	Concentration
4	Critical size (delta)

For time absorption calculations, the time-dependent angular flux is assumed to be separable in time and space, viz.,

$$\psi(r, \underline{\Omega}, t) = \psi(r, \underline{\Omega})\, e^{\alpha t}.$$

If this assumption is inserted into the time-dependent transport equation, the exponentials cancel and a fictitious cross-section term of the form α/v_g appears as a correction to the total and absorption cross sections. Here v_g is the neutron speed associated with energy group g. The exponential factor α is then the eigenvalue sought in the time-absorption eigenvalue search. Obviously, $\alpha = 0$ for an exactly critical system, and $\alpha > 0$ for a supercritical system.

For concentration searches, the modification of the cross-section concentrations takes place as indicated in Sec. III.B.1.b.

For delta searches, the coarse-mesh boundaries can be modified selectively to obtain a critical system. The modified coarse-mesh boundaries, $\tilde{R}_k$, are calculated from the initial input boundaries, R_k, by

$$\tilde{R}_{k+1} = \tilde{R}_k + (R_{k+1} - R_k)*(1 + EV*RM_k),$$

$$k = 1, 2, \ldots, IM, \qquad (66)$$

where EV is the eigenvalue sought in the delta eigenvalue search. The factors RM_k are the coarse-mesh radii modifiers which are input by the ONETRAN user and control how the coarse-mesh boundaries are modified. Clearly, if RM_k is zero, the thickness of the k^{th} zone is not altered. If all RM_k are unity, the system dimensions are uniformly expanded (EV > 0) or contracted (EV < 0). Many sophisticated changes can be made, limited only by the ingenuity of the user. For example, an interface between two zones

*Not included here are the options IEVT=0 for inhomogeneous source problems and IEVT=1 for k_{eff} calculations.

may be moved while the remainder of the system is left unchanged.

In all three eigenvalue searches, the appropriate system parameter is adjusted to achieve a desired value of k_{eff}. This value is taken to be unity (criticality) unless the input parametric eigenvalue trigger (IPVT) is set to unity. In this case, the parametric value of k_{eff} is entered as an input number (PV).

For concentration and delta searches, it is also possible to adjust the appropriate system parameter to achieve a system changing exponentially in time at the rate $e^{\alpha t}$ by setting the parametric eigenvalue trigger equal to 2. In this case, the parametric eigenvalue (PV) entered by the ONETRAN user is the desired exponential factor α. Obviously, $\alpha = 0$ corresponds to the normal concentration or delta search.

Regardless of the parameter being adjusted, the search is executed by performing a sequence of k_{eff} calculations, each for a different value of the parameter being treated as the eigenvalue. Each of the successive k_{eff} calculations is accelerated by coarse-mesh rebalance, but the search for the desired value of k_{eff} is conducted by subroutine NEWPAR. Regardless of the nature of the problem, the search is for a value of the parameter which makes the value of λ defined in Eq. (58) unity.

In the following description of NEWPAR, it is helpful to refer to Fig. 10 in which the deviation of λ from unity is plotted against outer iteration number.

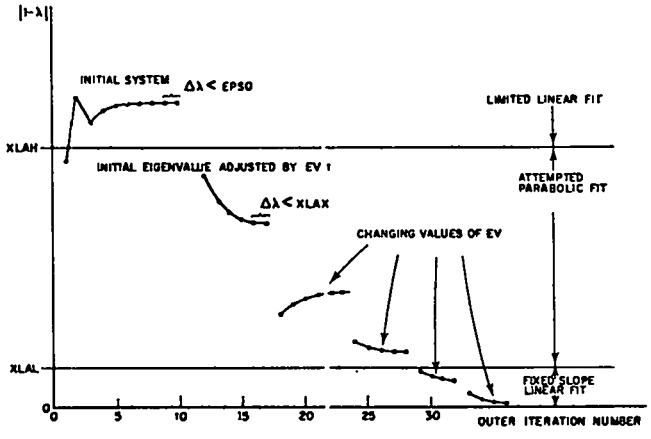


Fig. 10. Variation of λ during a hypothetical eigenvalue search.

For the initial system, NEWPAR continues the outer iteration until two successive values of λ differ by less than EPSO. For subsequent sequences of λ values, a different convergence precision, XLAX, is used. After the first converged λ sequence is obtained, the initial value of the eigenvalue (EV) is altered by EVM, an input value. If $\lambda > 1$ (multiplying system), the new eigenvalue is equal to EV + EVM; if $\lambda < 1$ (decaying system), the new value is EV - EVM. These alterations correspond to the addition or the subtraction of an absorption, e.g., as in a time-absorption search or a poison-concentration search. For delta calculations (IEVT=4), EVM must be negative to change EV in the right direction.

Basically, after two values of $k_{eff}(\lambda)$ are obtained for two different system configurations, subroutine NEWPAR attempts to fit a curve through the most recent values to extrapolate or interpolate to a value of unity. Depending on the amount of information available and the size of $|1 - \lambda|$, this fit proceeds in different ways. A parabolic fit cannot be made until three converged values of λ are available, and is not attempted unless $|1 - \lambda|$ is greater than an input search lower limit (XLAL) and less than an input search upper limit (XLAH). If a parabolic fit is tried and the roots are imaginary, a straight-line fit is used. If the roots are not imaginary, the closest root is used as the new value of EV. Once a bracket is obtained (change of sign of $\lambda - 1$), the fit procedure is not allowed to move outside the region of the bracket. Should a parabolic fit select an eigenvalue outside the bracket region, this value is rejected and the new value is taken to be one-half the sum of the previous value and the value previous to that.

Whenever the parabolic fit is not used, (i.e., $|1 - \lambda| <$ XLAL) a linear fit is used and the new eigenvalue is computed from

$$(EV)_{new} = (EV)_{old} + POD*EVS*(1 - \lambda), \qquad (67)$$

where POD is an input "parameter oscillation damper" which may be used to restrict the amount of change in the eigenvalue. In Eq. (67), EVS is a measure of the slope of the curve. When $|1 - \lambda| >$ XLAH, $(1 - \lambda)$ in Eq. (67) is replaced by XLAH (with the correct sign) to prevent too large a change in EV. After $|1 - \lambda| <$ XLAL, the value of EVS is fixed and

kept constant until convergence to prevent numerical difficulty in the approximation of the derivative when λ is close to unity.

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

Convergence in time-absorption calculations is typically one-sided. If EV (the eigenvalue α) is negative, then there is a possibility that the corrected total cross section will become negative. If this happens, the automatic search procedure may fail dramatically. For this reason POD = 0.5 or less is frequently used in such searches.

10. Adjoint Computations

The ONETRAN program solves the adjoint transport equation by transposing the matrices of scattering coefficients and inverting the group order of the problem. The solution of the resulting problem in direction $\underline{\Omega}$ is then identified with the solution of the adjoint equation in direction $-\underline{\Omega}$.[14]

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

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

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

(c) Remember that any output is in inverse group order, i.e., that groups labeled 1, 2, ..., are really groups IGM, IGM - 1, etc. Similarly, the output flux from an adjoint problem must be inverted before insertion into a direct problem. On the other hand, an output flux from one adjoint problem is in the proper group order for use in another adjoint problem.

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

11. Edit Options

The ONETRAN user is provided with two types of edit options, zone edits and point edits. As many different zone and point edits as desired may be performed.

a. Zone Edit

An edit zone is a collection of fine-mesh intervals which have the same zone number. The user defines a zone by entering a set of IT numbers (NEDZ array) which associate with each interval on the fine mesh a zone identification number (zone i.d.). The intervals of an edit zone need not be contiguous. For each group and zone, a table containing the macroscopic activities (for cross-section positions 1 through IHT) is given. The macroscopic activity $A_k(g,IPOS)$ in zone k and group g for cross-section position IPOS is defined by

$$A_k(g,IPOS) = \sum_i C(g,IPOS,m_i)\phi_i V_i \text{ for } i \in \text{zone } k, \tag{68}$$

where m_i is the material i.d. (cross-section block identification number) for mesh cell i, $C(g,IPOS,m_i)$ is the cross section for group g in position IPOS for material m_i, V_i is the mesh cell volume, and ϕ_i is the average flux in mesh cell i. Thus A_k is the activity computed with the macroscopic cross section actually used in the problem, summed over all mesh cells in zone k.

For each zone edit, the ONETRAN user is provided the option of calculating constituent activities and microscopic activities for any material desired. The constituent activity $A_k^j(g,IPOS)$ for material j in zone k is defined by

$$A_k^j(g,IPOS) = \sum_i C(g,IPOS,m_i)\phi_i V_i \delta_{jm_i} \text{ for } i \in \text{zone } k. \tag{69}$$

Here δ_{jm_i} equals unity if material j equals material m_i, the mixture table density (MIXDEN) if material j is a "constituent" of material m_i, and is zero otherwise. A "constituent" means that material j appears as an entry in the MIXNUM array with density MIXDEN (see Sec. III.B.1.3.) that is used to form material m_i. Thus, if material j is used to form a matieral j', which is used to form material m_i, then

material j is not a "constituent" of material m_i within this definition.

The microscopic activity for material j in zone k is defined by

$$A_k^j(g,IPOS) = \sum_i C(g,IPOS,j)\phi_i V_i \text{ for } i \in \text{zone } k. \quad (70)$$

Thus A_k^j would be the activity obtained in zone k if material j were uniformly distributed throughout the system, even though material j may not actually have appeared in the problem cross sections.

The edit input parameters NCA and NMA specify the number of constituent activities and number of microscopic activities to be calculated. The user must then enter NCA material i.d.'s for the constituent activities and NMA material i.d.'s for the microscopic activities.

To edit a material which is not actually a part of the problem, the ONETRAN user may add a mixture instruction to the mixture tables; or, if interested in only a few cross sections, he may add these cross sections to other blocks in rows IHT-5, IHT-6, etc.

Finally, following any constituent activities or microscopic activities, the zone edit provides the zone relative power density (group sum of the zone volume integral of $\nu \times$ fission rate divided by the zone volume), normalized to that of a user-designated zone. The zone relative power density (unnormalized) is defined by

$$PD_k = \frac{\sum_g \sum_i C(g,IHT-1,m_i)\phi_i V_i}{\sum_i V_i} \text{ for } i \in \text{zone } k. \quad (71)$$

If the user selects zone zero (NORMZ=0), the normalization is to the whole system power density.

b. Point Edit

The point edit feature of ONETRAN provides the user with the option of obtaining the pointwise variation of the activity across each fine-mesh interval. The user must enter the fine-mesh i.d.'s over which the point edit is desired (NEDPT array) and the cross-section material i.d.'s for the microscopic activities (IDMA array). The pointwise microscopic activity for material j in mesh cell i is

$$A^j_{i\pm\frac{1}{2}}(g,IPOS) = C(g,IPOS,m_i)\ \phi_{i\pm\frac{1}{2}}. \quad (72)$$

C. Data Input Rules

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

TABLE VI

OPTIONS FOR SPECIAL READ FORMATS IN LOAD

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

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

1 - Zero is repeated 47 times,

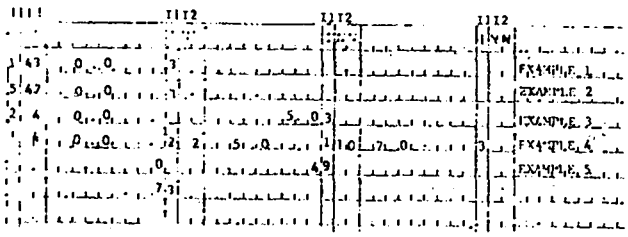
2 - Zero is repeated 470 times,

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

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

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

A special routine, WRITE, is used to print some of the two- and three-dimensional arrays that occur in the program. This routine can be used for one-,



two-, or three-dimensional arrays and has an option for printing a portion of an array, e.g., the mixed cross-section blocks, if any.

D. Description of Input Data

In the following pages the input data for ONETRAN are listed in exactly the order in which they are entered in the code. The data are divided into four categories: (1) job title cards, (2) control integers on cards 1 through 3 and control floating-point numbers on cards 4, 5, and 6, (3) problem-dependent data on subsequent cards through the LOAD routine, and (4) edit input.

1. Job Title Cards

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

2. Input of Control Numbers

On cards 1 through 3, the user enters the following control integers which are read in a 12I6 format and on cards, 4, 5, and 6 the following control floating-point numbers in a 6E12.4 format:

Number of Word on Card	Name of Variable	Comments
CONTROL INTEGERS (12I6)----------		----------CARD 1
1	ITH	0/1 (direct/adjoint) type of calculation performed.
2	ISCT	0/N (isotropic/N^{th} order anisotropic) order of scattering calculations. NM spherical harmonics flux components are computed.
3	ISN	S_N angular quadrature order. Must be an even number.
4	IGM	Number of energy groups.
5	IM	Number of coarse-mesh intervals.
6	IBL	0/1/2/3/4 (vacuum/reflective/periodic/white/albedo) left boundary condition.
7	IBR	0/1/2/3/4 right boundary condition.
8	IEVT	0/1/2/3/4 (inhomogeneous source/k_{eff}/alpha or time absorption/concentration search/delta or critical size search) eigenvalue type.
9	ISTART	0/±1/±2/3/±4 input flux guess and starting option. See Sec. III.B.6.
10	IQOPT	0/±1/±2/±3/±4/5 inhomogeneous source input option. See Sec. III.B.5.
11	IGEOM	1/2/3/4 (plane/cylindrical/spherical/two-angle plane) geometry option.
12	IQUAD	1/2/±3 (built-in P_n/built-in DP_n/+: card input, -: interface file) source of S_N quadrature constants w and μ.
CONTROL INTEGERS (12I6)----------		----------CARD 2
1	MT	Total number of materials (cross section blocks including anisotropic cross sections) in the problem.
2	MTP	Number of input material sets from interface file ISOTXS. CAUTION: Each material set from this file yields ISCT+1 materials. See LMTP below.
3	MCR	Number of input materials from the code-dependent input file. If MCR is negative, each MCR material is read as a single block on the FIDO format, terminated by the FIDO terminator: T.

4	MS	Number of mixture instructions. See Sec. III.B.1.b. and items MIXNUM, MIXCOM, MIXDEN below.
5	IHT	Row of total cross section in the cross-section format. If IHT is flagged negative, then n,2n scattering is present in the scattering matrices and cross-section position IHT-4.
6	IHS	Row of within-group scattering cross section in the cross-section format. For problems with upscattering (IHS>IHT+1), IHS must be flagged negative if σ^{up} is present in the cross-section input table and to be removed. Not applicable for FIDO format input.
7	IHM	Total number of rows in the cross-section format.
8	IDEN	0/1 (no/yes) space-dependent density factors.
9	IQAN	0/N (isotropic/N^{th} order anisotropic) order of source anisotropy. NMQ spherical harmonics source components are required input. CAUTION: ISCT≥IQAN is required.
10	IQL	0/1 (no/yes) left boundary source.
11	IQR	0/1 (no/yes) right boundary source.
12	IACC	0/1/2/±3 (none/system rebalance/coarse mesh rebalance/Chebyshev, -: read Chebyshev factors) inner iteration acceleration option.

CONTROL INTEGERS (7I6, TI1, 4I6)----------CARD 3

1	OITM	Maximum number of outer iterations.
2	IITL	Maximum number of inner iterations (per group) until $\|1 - \lambda\| < 10*EPSO$.
3	IITM	Maximum number of inner iterations (per group) after $\|1 - \lambda\| < 10*EPSO$. ONETRAN requires that IITM≥IITL.
4	IFISS	1/2/3/4 (fission spectrum/zone-dependent fission spectrum/fission matrix/zone-dependent fission matrix) type of fission fractions.
5	IPVT	0/1/2 (none/k_{eff}/alpha) parametric eigenvalue trigger. Valid only for IEVT>1 if IPVT=1. Valid for all IEVT if IPVT=2.
6	IEDOPT	0/1 (no/yes) edit option input.
7	IPLOT	0/1/2 (no/semi-log/linear) scalar flux plotting option.
8	I1	0/1 (no/yes) input flux print suppression trigger.
	I2	0/1/2/3/4 (all/isotropic/none/all cell-centered/isotropic cell-centered) final flux print trigger. The standard flux print contains both cell-centered and cell-edge fluxes. The cell-centered options print only the cell-centered fluxes, greatly reducing the volume of output.
	I3	0/1/2 (all/mixed/none) cross-section print trigger.
	I4	0/1/2 (none/all/cell-centered) final fission print trigger.
	I5	0/1/2/3 (all/unnormalized/normalized/none) source print trigger.
	I6	0/1 (no/yes) fine mesh geometry table print suppression trigger.
9	ITLIM	0/N (none/N second) time limit.
10	IFO	0/1 (no/yes) interface file output trigger.
11	IANG	0/±1 (no/yes) store of angular flux. IANG is negative for print of angular flux. If IANG≠0, the TIMEX angular flux file NTIMEX is written.

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

1	EV	Eigenvalue guess. It is satisfactory to enter 1.0 for concentration search (IEVT=3) and 0.0 for all other problems.
2	EVM	Eigenvalue modifier used only if IEVT>1. See Sec. III.B.9. above.

3	PV	Parametric value of k_{eff} for subcritical or supercritical systems or α for 1/v time absorption. Used only if IPVT≠0. See Sec. III.B.9. above.
4	XLAL	Lambda lower limit for eigenvalue searches. Default value is 0.01. See Sec. III.B.9. above.
5	XLAH	Search lambda upper limit. Default value is 0.5.
6	XLAX	Search lambda convergence precision for second and subsequent values of the eigenvalue. Default value is 10*EPSI.

CONTROL FLOATING POINT DATA (3E12.4)----------CARD 5

1	EPSI	Inner iteration convergence precision. Default value is 1.E-4.
2	NORM	Normalization factor. The total number of source (IEVT=0) or fission particles (IEVT≠0) is normalized to this number if it is nonzero. No normalization is performed if NORM=0.0.
3	POD	Parameter oscillation damper used in eigenvalue searches. Default value if not entered is 1.0. See Sec. III.B.9.

CONTROL FLOATING POINT DATA (2E12.4)----------CARD 6

1	BHGT	Buckling height (in cm if cross sections are in cm^{-1}). If BHGT is flagged negative, the transport cross section in position IHT-3 is used for calculation of the buckling absorption.
2	BWTH	Buckling width (plane geometry only).

3. Problem-Dependent Data

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

With the exception of the cross sections from the code-dependent card input file, all the following data are loaded by the LASL block loader using the special formats described in Sec. III.C. We denote these formats by S(I) for integers and S(E) for floating point numbers.

Block Name and Dimension	Format	Number of Entries	Comments
IHR(IM)	S(I)	IM	Number of fine mesh intervals in each coarse mesh.
WGT(MM)	S(E)	MM	S_N quadrature weights. Enter only if IQUAD=+3.
U(MM)	S(E)	MM	S_N quadrature μ cosines. Enter only if IQUAD=+3.
C(IHM,IGM,MIN)	-	-	Cross-section blocks. MIN=MCR+MTP*(ISCT+1). Three options are available for reading cross sections. The LASL input format may not be mixed with the FIDO format. 1. LASL Input. If MCR>0, MCR blocks of IHM*IGM numbers are read in a 6E12.5 format. Each block is preceeded by an identification card read in a 18A4 format.

2. FIDO Input. If MCR<0, MCR blocks of data are created from FIDO input. The 14* card must not preceed the FIDO input data.

3. Interface File ISOTXS. When MTP≠0, MTP material sets are read from standard interface file ISOTXS. On this file each material set consists of ISCT+1 cross-section blocks for the isotropic and ISCT anisotropic cross sections. The first (isotropic) component of the first material is stored in cross-section block MCR+1, the first component of the second material is stored in cross section block MCR+ISCT+2, etc. Should the ISOTXS file not contain ISCT anisotropic components, zeroes are supplied for the components not present. If the ISOTXS file contains more components than needed, only the first ISCT+1 components are read. The maximum number of upscatter groups and downscatter groups (MAXUP, MAXDN) in the ISOTXS file must be consistent with the choice of IHT, IHS, and IHM. If they are not consistent, this will be flagged as an error.

LMTP(MTP)	S(I)	MTP	Position numbers of material sets to be read from ISOTXS. Do not enter unless MTP>0. The material sets are loaded into the C block in the order they appear on the ISOTXS file, and not in the order they appear in the LMTP array.
Input flux guess FLUX(NM,2,IT)	S(E)	-	Number of entries depends on option. See Sec. III.B.6.

ISTART	Number of entries
-4	Angular flux from standard interface file on unit IAFLUX.
-2	IT
-1	IGM
0	None.
+1	NM sets of IGM.
+2	NM sets of IT.
3	Problem restart dump from unit NDMP1.
+4	Scalar flux from standard interface file on unit ITFLUX.

Input sources Q(NMQ,2,IT) QL(MM/2) QR(MM/2)	S(E)	-	Number of entries depends on option. See Sec. III.B.5. The sources are loaded as: (a) distributed source (if any) for each group; for each anisotropic component; (b) left boundary source (if any) and right boundary source (if any); for each group. For IQOPT flagged negative, an energy spectrum is input for each (assumed isotropic) boundary source. For IQOPT positive or zero, the complete angular distribution of the boundary sources is input, a distribution for each group.

			IQOPT — Number of entries for distributed source 0 — None. ± 1 — IGM; one for each NMQ components ± 2 — IT; one for each group; for each NMQ components ± 3 — 2*IT; one for each group; for each NMQ components ± 4 — IGM and 2*IT; both for each NMQ components.
RAD(IM+1)	S(E)	IM+1	Coarse-mesh radii.
IDC(IM)	S(I)	IM	Cross-section material identification numbers. These numbers assign a cross-section block to each coarse-mesh interval. These numbers must be flagged negative for an anisotropic scattering source to be calculated in that coarse-mesh interval.
CHI(IGM,IM)	S(E)	-	Fission fractions. Fraction of fission yield emerging in each group. May be either a spectrum (χ_g) or a matrix ($\chi_{g'\to g}$) and may be coarse-mesh zone-dependent. See Sec. III.B.1.f. IFISS — Number of entries 1 — IGM; single fission spectrum. 2 — IGM*IM; IM sets of spectra loaded as a single block. 3 — IGM sets of length IGM; single fission matrix, loaded by rows in blocks of IGM length. 4 — IGM sets of length IGM*IM: IM sets of matrices, loaded by rows in blocks of IGM*IM length.
VEL(IGM)	S(E)	IGM	Group speeds. Used only in time-absorption calculations.
MIXNUM(MS)	S(I)	MS	Numbers identifying cross-section block being mixed. See Sec. III.B.1.b. Do not enter if MS=0.
MIXCOM(MS)	S(I)	MS	Numbers controlling cross-section mixture process. See Sec. III.B.1.b. Do not enter if MS=0.
MIXDEN(MS)	S(E)	MS	Mixture densities. See Sec. III.B.1.b. Do not enter if MS=0.
RM(IM)	S(E)	IM	Coarse-mesh radii modifiers. See Sec. III.B.9. Enter only if IEVT=4.
DEN(IT)	S(E)	IT	Fine-mesh density factors. Enter only if IDEN=1.
LB(IGM)	S(E)	IGM	Left boundary group albedos. Enter only if IBL=4.
RB(IGM)	S(E)	IGM	Right boundary group albedos. Enter only if IBR=4.
AF(IGM)	S(E)	IGM	Chebyshev acceleration factors. Enter only if IACC=-3.

4. Edit Input

The edit input, entered only if IEDOPT=1, consists of control integers entered on cards indicated by EDIT 1, 2, or 3; and the remaining edit input entered in the special format through the LOAD subroutine discussed above in Sec. III.C. The zone-edit control integers and the zone-edit arrays are read first for all NZEDS edits, then the point edit control integers and point edit arrays are read for all NPEDS edits.

EDIT CONTROL INTEGERS (2I6)----------EDIT 1

1	NZEDS	Number of zone edits.
2	NPEDS	Number of point edits.

ZONE EDIT CONTROL INTEGERS, Enter only if NZED > 0 (4I6)----------EDIT 2

1	NZ	Total number of zones.
2	NCA	Number of constituent activities calculated.
3	NMA	Number of microscopic activities calculated.
4	NORMZ	Zone identification number for normalization of power density. If NORMZ=0, whole system normalization is performed.

ZONE EDIT ARRAYS THROUGH LOAD----------

IDCA(NCA)	S(I)	NCA	Cross-section material identification numbers for constituent activities. Enter only if NCA>0.
IDMA(NMA)	S(I)	NMA	Cross-section material identification numbers for microscopic activities. Enter only if NMA>0.
NEDZ(IT)	S(I)	IT	Zone identification numbers. These numbers assign a zone number to each fine-mesh interval.

POINT EDIT CONTROL INTEGERS, Enter only if NPEDS > 0 (2I6)----------EDIT 3

1	NIPE	Number of fine-mesh intervals to be included in the point edit.
2	NPMA	Number of microscopic activities calculated in the point edit.

POINT EDIT ARRAYS THROUGH LOAD----------

IDMA(NPMA)	S(I)	NPMA	Cross-section material identification numbers for microscopic activities.
NEDP(NIPE)	S(I)	NIPE	Fine-mesh identification numbers to be included in the point edit. Do not enter if NIPE=IT (point edit over all fine-mesh intervals).

E. Output Description for a Test Problem

The ONETRAN program comes with a set of twelve test problems plus an example problem designed to illustrate many of the ONETRAN options whose output is presented on the following pages. Each page of the output is numbered, and we refer to these numbers in the text below.

The problem is a ^{239}Pu cylindrical core containing a central absorbing rod and a weakly anisotropic scattering ^{238}U (with some ^{239}Pu) blanket. The object of the calculation is to obtain the critical thickness of the Pu core, maintaining the absorber and blanket radii constant. As seen from the first output page (1), the problem is run with S_4 angular quadrature, three energy groups, and coarse-mesh rebalance acceleration of the inner iterations. The fission fractions are zone-dependent fission matrices with the values:

$$\chi_{g'\to g}(^{239}Pu) = \begin{bmatrix} .6 & .3 & .1 \\ .4 & .5 & .1 \\ .4 & .4 & .2 \end{bmatrix}$$

$$\chi_{g'\to g}(^{238}U) = \begin{bmatrix} .7 & .25 & .05 \\ .6 & .35 & .05 \\ .5 & .45 & .05 \end{bmatrix}$$

All the integer and floating point input control data is printed on output page (1). The S_N angular quadrature coefficients are the built-in S_4 Gaussian quadrature set and are printed on output page (2). The level index, level weights, and level cosine columns refer to the μ levels of Fig. 5. The LI, XI, and PHI columns refer to the ξ level index, ξ angle cosine, and ϕ angle of Fig. 5. The remaining problem input is printed on output pages (2) through (4).

The coarse mesh and material map is printed on output page (4), indicating anisotropic scattering in the blanket and five fine-mesh intervals in each coarse mesh. The cross-section mixing instructions, the mixed cross sections, the coarse-mesh and fine-mesh geometry tables, and the fission fractions follow on output pages (4) through (6). Following the summary of convergence precisions on output page (7), a monitor of the calculation is printed. The "rebalance convergence" column contains the maximum deviation from unity of any rebalance factor for the coarse-mesh rebalance performed on each outer iteration. The lambda column is the λ factor of Eq. (58). The eigenvalue in this case is the EV of Eq. (66).

On output page (8), the system balance tables for each group and the group sum are printed. These group-dependent quantities are computed in subroutine SUMS and defined as follows:

(a) SOURCE = total inhomogeneous source = QG_g =

$$\sum_{i=1}^{IT} Q_i \, V_i + \sum_{\mu_m<0} w_m |\mu_m| A_{IT+\frac{1}{2}} \, QR_m + \sum_{\mu_m>0} w_m \, \mu_m \, A_{\frac{1}{2}} \, QL_m,$$

where Q_i is the inhomogeneous distributed source, QL_m is the left boundary source, and QR_m is the right boundary source;

(b) FISSION SOURCE = total fission source to group g = FG_g =

$$\sum_{h=1}^{IGM} \chi_{h\to g} \, (\nu\sigma_f)_h \sum_{i=1}^{IT} \phi_{i,h} \, V_i;$$

(c) SELF-SCATTER = self-scatter in group g = SS_g =

$$\sum_{i=1}^{IT} \sigma^0_{s,g\to g} \, \phi_i \, V_i;$$

(d) OUT-SCATTER = out-scatter from group g = $SOUT_g$ =

$$\sum_{i=1}^{IT} \sigma'_{t,g} \, \phi_i \, V_i,$$

where $\sigma'_{t,g}$ is the total cross section for group g plus any buckling absorption plus any time absorption (EV/v_g);

(e) ABSORPTION = absorption in group g = ABG_g =

$$\sum_{i=1}^{IT} \sigma'_{a,g} \, \phi_i \, V_i,$$

where $\sigma'_{a,g}$ is the absorption cross section for group g plus any buckling absorption plus any time absorption (EV/v_g);

(f) IN-SCATTER = in-scatter source to group g = SIN_g =

$$\sum_{\substack{h=1 \\ h\neq g}}^{IGM} \sigma^0_{s,h\to g} \sum_{i=1}^{IT} \phi_{i,h} \, V_i;$$

(g) RIGHT LEAKAGE = Net current out of system right boundary = RL_g =

$$FR_{IT+\frac{1}{2}} - FL_{IT+\frac{1}{2}}$$

$$= \sum_{\mu_m>0} w_m \, \mu_m \, A_{IT+\frac{1}{2}} \, \psi_{m,IT+\frac{1}{2}}$$

$$- \sum_{\mu_m<0} w_m |\mu_m| A_{IT+\frac{1}{2}} \, \psi_{m,IT+\frac{1}{2}};$$

(h) NET LEAKAGE = Net current from the whole system = $NL_g = RL_g - (FR_{\frac{1}{2}} - FL_{\frac{1}{2}})$

$$= RL_g - \Big(\sum_{\mu_m>0} w_m \, \mu_m \, A_{\frac{1}{2}} \, \psi_{m,\frac{1}{2}}$$

$$- \sum_{\mu_m<0} w_m |\mu_m| A_{\frac{1}{2}} \, \psi_{m,\frac{1}{2}}\Big);$$

and

(i) NEUTRON BALANCE = BAL_g

$$= 1 - \frac{NL_g + ABG_g + SOUT_g}{QG_g + FG_g + SIN_g}.$$

A repeat of the convergence parameters, the final iteration monitor and the final coarse-mesh radii follow the balance tables. Output pages (9) through

(11) contain the scalar fluxes and fission rates for each group. The cell-centered flux is simply the average of the the two cell-edge fluxes, $\phi_{i\pm\frac{1}{2}}$. The fission rate is printed on the cell-centered format, reducing the amount of printed output for this array by a factor of seven. The zone edit begins on output page (12) with the print of the zone edit input arrays. The edit zones are seen to be identical to the coarse-mesh material zones. The constituent activities are calculated for materials 2 (^{239}Pu) and 3 (^{238}U). These are followed by the zone relative power densities (normalized to the whole system).

The point edit begins on output page (14) with the print of the point edit input. This edit calculated the pointwise activities for material 1 (absorber) over the first five mesh intervals (coarse-mesh zone 1).

Following the point edit output, the messages indicating the successful completion of the plot and the writing of the interface file output. Output page (15) shows the semilog plot of the scalar flux.

```
THIS ONETRAN PROBLEM RUN ON  05/14/75 WITH VERSION     1/1/75
 ONETRAN EXAMPLE PROBLEM

         0  ITH      0/1  DIRECT/ADJOINT
         1  ISCT     0/N  ISOTROPIC/NTH ORDER ANISOTROPIC
         4  ISN      SN ORDER
         3  IGM      NUMBER OF GROUPS
         3  IM       NUMBER OF COARSE MESH INTERVALS
         1  I8L      LEFT/RIGHT BOUNDARY CONDITION=0/1/2/3/4
         0  I8R       VACUUM/REFLECTIVE/PERIODIC/WHITE/ALBEDO
         4  IEVT     0/1/2/3/4 Q/K/ALPHA/C/DELTA CALCULATION
        -1  ISTART   0 THRU 4 STARTING OPTIONS (SEE MANUAL)
         0  IQOPT    0 THRU 5 SOURCE INPUT OPTIONS (SEE MANUAL)
         2  IGEOM    1/2/3/4 PLANE/CYLINDER/SPHERE/2 ANGLE PLANE
         1  IQUAD    1-PN W AND MU,2-OPN W AND MU,+3-CARD INPUT W AND MU
                    -3-INTERFACE INPUT W AND MU

         6  MT       TOTAL NUMBER OF MATERIALS
         0  MTP      NUMBER OF MATERIALS FROM LIBRARY
         4  MCR      NUMBER OF MATERIALS FROM CARDS (= FOR FIDO FORMAT)
         9  MS       NUMBER OF MIXTURE INSTRUCTIONS
         3  IHT      ROW OF TOTAL CROSS SECTION (= FOR N,2N REACTION PRESENT)
         4  IHS      ROW OF SELF SCATTER CROSS SECTION  (= INDICATES SIGMA-UP PRESENT    )
         6  IHM      LAST ROW OF CROSS SECTION TABLE
         1  IDEN     0/1 NO/YES SPACE DEPENDENT MATERIAL DENSITY
         0  IQAN     0/N ISOTROPIC/NTH ORDER ANISOTROPIC SOURCE
         0  IQL      0/1 NO/YES LEFT BOUNDARY SOURCE
         0  IQR      0/1 NO/YES RIGHT BOUNDARY SOURCE
         2  IACC     0-NOTHING,1-SYSTEM REBALANCE,2-COARSE MESH REBALANCE
                    3-CHEBYSHEV (= TO READ ACC,FACTORS)

        50  OITM     MAXIMUM NUMBER OF OUTER ITERATIONS
         5  IITL     MAXIMUM NUMBER OF INNER ITERATIONS UNTIL (1.-LAMBDA),LT,10*EPSO
        20  IITM     MAXIMUM NUMBER OF INNER ITERATIONS AFTER (1.-LAMBDA),LT,10*EPSO
         4  IFISS    1/2/3/4 FISSION FRACTIONS/ZONE FISSION FRACTIONS/FISSION MATRIX/ZONE FISSION MATRIX
         0  IPVT     0/1/2 NONE/K/ALPHA PARAMETRIC EIGENVALUE TRIGGER
         1  IEDOPT   0/1 NO/YES  EDIT OPTIONS
         1  IPLOT    0/1/2 NO/SEMI-LOG/LINEAR  PLOTTING OPTIONS
         0  I1       0/1 NO/YES SUPPRESS INPUT FLUX PRINT
         1  I2       0/1/2/3/4 ALL/ISOTROPIC/NONE/ALL CELL CENTERED/ISOTROPIC CELL CENTERED  FINAL FLUX PRINT
         0  I3       0/1/2 ALL/MIXED/NONE CROSS SECTION PRINT
         2  I4       0/1/2 NONE/FULL/CELL-CENTERED  FISSION RATE PRINT
         0  I5       0/1/2/3 ALL/UNNORMALIZED/NORMALIZED/NONE SOURCE PRINT
         1  I6       0/1 YES/NO PRINT F,M,GEOMETRY TABLE
         0  ITLIM    0/N NO/N=SECOND TIME LIMIT
         1  IFO      0/1 NO/YES INTERFACE FILE OUTPUT
         0  IANG     0/1 NO/YES STORE ANGULAR FLUX(= FOR PRINT OF ANGULAR FLUX)

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

  1.000E-04   EPSI    INNER CONVERGENCE PRECISION
  1.000E+00   NORM    NORMALIZATION AMPLITUDE
  1.000E+00   POO     PARAMETER OSCILLATION DAMPER

  0.          BHGT    BUCKLING HEIGHT (= FOR TRANSPORT CROSS SECTION IN CORRECTION)
 -0.          BWTH    BUCKLING WIDTH
```

```
INPUT FINE R MESH      3
ALL ENTRIES =          5


STORAGE    REQUIRED ALLOWED
SMALL CORE      819   24500
LARGE CORE      889  375000


LEVEL  LEVEL WEIGHT  LEVEL COSINE  STARTING COSINE
   1   1,739274E-01 -8,611363E-01 -5,083741E-01
   2   3,260726E-01 -3,399810E-01 -9,404323E-01
   3   3,260726E-01  3,399810E-01  0,
   4   1,739274E-01  8,611363E-01  0,

  M   REFL M   LI   POINT WEIGHT   MU COSINE       WGT*MU      BETA PLUS     BETA MINUS          XI          PHI(DEG)
  1       4    1   1,630363E-01 -3,399810E-01 -5,542925E-02 3,399810E-01  0,              ,86113631     131,9715
  2       6    2   1,739274E-01 -8,611363E-01 -1,497752E-01 8,611363E-01  0,              ,33998104     156,3027
  3       5    2   1,630363E-01 -3,399810E-01 -5,542925E-02 1,258643E+00  9,186619E-01    ,33998104     111,1933
  4       1    1   1,630363E-01  3,399810E-01  5,542925E-02 0,            3,399810E-01    ,86113631      48,0285
  5       3    2   1,630363E-01  3,399810E-01  5,542925E-02 9,186619E-01  1,258643E+00    ,33998104      68,8067
  6       2    2   1,739274E-01  8,611363E-01  1,497752E-01 0,            8,611363E-01    ,33998104      23,6973


**************************************************INPUT CROSS SECTIONS*****************************************
     1                    LOADED FROM CARDS ABSORBER
     2                    LOADED FROM CARDS PU-239
     3                    LOADED FROM CARDS U-238
     4                    LOADED FROM CARDS U-238(P1)


INPUT CROSS SECTION   1

        GROUP     1   GROUP     2   GROUP     3

   1   ,100000E+01   ,500000E+01   ,200000E+02
   2  0.            0.            0.
   3   ,125000E+01   ,550000E+01   ,203000E+02
   4   ,200000E+00   ,400000E+00   ,300000E+00
   5  0.             ,500000E-01   ,100000E+00
   6  0.            0.            0.
```

```
          GROUP      1    GROUP      2    GROUP      3
    1    .100000E+01     .150000E+01     .300000E+01
    2    .290000E+01     .320000E+01     .420000E+01
    3    .240000E+01     .370000E+01     .420000E+01
    4    .110000E+01     .200000E+01     .120000E+01
    5   0.               .200000E+00     .200000E+00
    6   0.              0.               .100000E+00

 INPUT CROSS SECTION    3

          GROUP      1    GROUP      2    GROUP      3
    1    .200000E+01     .300000E+01     .200000E+01
    2    .280000E+01     .320000E+01    0.
    3    .750000E+01     .540000E+01     .300000E+01
    4    .300000E+01     .200000E+01     .100000E+01
    5   0.               .150000E+01     .400000E+00
    6   0.              0.               .100000E+01

 INPUT CROSS SECTION    4

          GROUP      1    GROUP      2    GROUP      3
    1   0.              0.              0.
    2   0.              0.              0.
    3   0.              0.              0.
    4    .500000E+00     .200000E+00     .500000E-01
    5   0.              0.              0.
    6   0.              0.              0.

 INPUT ENERGY SHAPE      3
 ALL ENTRIES =    1.0000E+00

 INPUT COARSE MESH       4
   0.          1.0000E-01  6.0000E+00  1.0000E+01

 INPUT CROSS SEC IO      3
             1           2          .5

 INPUT FISSN G SPEC      9
   0.          0.          0.          6.0000E-01  3.0000E-01  1.0000E-01  7.0000E-01  2.5000E-01  5.0000E-02

 INPUT FISSN G SPEC      9
   0.          0.          0.          4.0000E-01  5.0000E-01  1.0000E-01  6.0000E-01  3.5000E-01  5.0000E-02

 INPUT FISSN Q SPEC      9
   0.          0.          0.          4.0000E-01  6.0000E-01  2.0000E-01  5.0000E-01  4.5000E-01  5.0000E-02

 INPUT VELOCITIES        3
   1.0000E+07  5.0000E+05  1.0000E+05
```

```
INPUT MIX NUMBERS       9
          1             2           3           4           5           5           5           6           6

INPUT MIX COMMANDS      9
          0             0           0           0           0           2           3           0           4

INPUT MIX DENSITY       9
  6.0000E-02  4.8000E-02  4.8000E-02  4.8000E-02  0.          2.0000E-02  9.0000E-01  0.          9.0000E-01

INPUT MESH MODS         3
  0.          1.0000E+00  0.

INPUT R DENSITY        15
  1.0000E+00  1.0000E+00  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.          1.0000E+00  1.0000E+00  1.0000E+00  1.0000E+00
```

M IS NUMBER OF FINE INTERVALS IN EACH COARSE INTERVAL

```
          1111111111111111111
          1     *     *      0
          1   1 *   2 *  -5  0
          1     *     *      0
          1111111111111111111
    R     0.    0.    6.    10.
          0  1000     0      0

    M         5     5     5
 COLUMN       1     2     3
```

MIXTURE NUMBER	MIXTURE COMMAND	MATERIAL ATOMIC DENSITY	
1	0	6.0000000E-02	1
2	0	4.8000000E-02	2
3	0	4.8000000E-02	3
4	0	4.8000000E-02	4
5	0	0.	5
5	2	2.0000000E-02	6
5	3	9.0000000E-01	7
6	0	0.	8
6	4	9.0000000E-01	9

GROUP NUMBER 1

MIXED X-SECT

	MATERL 5	MATERL 6
1	.950400E-01	0.
2	.134496E+00	0.
3	.355104E+00	0.
4	.142176E+00	.235200E-01
5	0.	0.
6	0.	0.

GROUP NUMBER 2

MIXED X-SECT

	MATERL 5	MATERL 6
1	.142560E+00	0.
2	.153600E+00	0.
3	.257568E+00	0.
4	.960000E-01	.940800E-02
5	.787520E-01	0.
6	0.	0.

GROUP NUMBER 3

MIXED X-SECT

	MATERL 5	MATERL 6
1	.969600E-01	0.
2	.403200E-02	0.
3	.145152E+00	0.
4	.481920E-01	.235200E-02
5	.190080E-01	0.
6	.471360E-01	0.

COARSE MESH GEOMETRY

	NO. OF INTERVALS	WIDTH	FINE MESH SIZE	LEFT BOUNDARY
1	5	.10000000E+00	.20000000E-01	0.
2	5	.59000000E+01	.11800000E+01	.10100000E+00
3	5	.40000000E+01	.80000000E+00	.60100000E+01
4	0	0.	0.	.10100000E+02

UNALTERED FISSION FRACTIONS FOR GROUP 1

GROUPS BY ROWS 1

	ZONE 1	ZONE 2	ZONE 3
1	0.	.600000E+00	.700000E+00
2	0.	.300000E+00	.250000E+00
3	0.	.100000E+00	.500000E-01

UNALTERED FISSION FRACTIONS FOR GROUP 2

GROUPS BY ROWS 1

	ZONE 1	ZONE 2	ZONE 3
1	0.	.400000E+00	.600000E+00
2	0.	.500000E+00	.350000E+00
3	0.	.100000E+00	.500000E-01

UNALTERED FISSION FRACTIONS FOR GROUP 3

GROUPS BY ROWS 1

	ZONE 1	ZONE 2	ZONE 3
1	0.	.400000E+00	.500000E+00
2	0.	.400000E+00	.450000E+00
3	0.	.200000E+00	.500000E-01

(6)

```
-0.         PV
 1.0000E-02 XLAL
 5.0000E-01 XLAH
 1.0000E-03 XLAX
 1.0000E-04 EPS OUTER
 1.0000E-04 EPS INNER
 3.9701E-04 EPS OUTER REBALANCE
 1.0000E-02 EPS TN
 1.0000E+00 POO
 1.0000E+00 NORM
```

TIME IN MINUTES	OUTER ITERATIONS	INNER ITERATIONS TOTAL/PER OUTER	/GROUP/PER GROUP	ERROR	NEUTRON BALANCE	EIGENVALUE	EIGENVALUE SLOPE	LAMBDA	REBALANCE CONVERGENCE
2.18E-03	0	0	0		0.	0.	0.	0.	0.
			1	5 6.61E-05					
			2	5 3.13E-04					
			3	5 3.54E-05					
3.84E-03	1	15	15		4.87216241E-07	0.	0.	1.02885230E+00	2.24579490E-01
			1	5 1.49E-05					
			2	5 8.72E-05					
			3	4 9.14E-05					
5.34E-03	2	29	14		-4.38861534E-08	0.	0.	1.00587777E+00	3.03682564E-02
			1	4 2.53E-05					
			2	5 2.01E-05					
			3	4 1.54E-05					
6.73E-03	3	42	13		5.20622905E-08	0.	0.	1.02320038E+00	2.52276003E-02
			1	4 7.54E-06					
			2	4 3.23E-05					
			3	3 3.79E-05					
7.91E-03	4	53	11		-9.87937696E-08	0.	0.	1.02292738E+00	2.31608639E-02
			1	3 7.61E-05					
			2	3 5.91E-05					
			3	3 9.42E-06					
8.93E-03	5	62	9		-3.78985163E-07	0.	0.	1.02333561E+00	2.34265229E-02
			1	3 7.84E-05					
			2	3 3.47E-05					
			3	3 8.40E-06					
9.94E-03	6	71	9		-3.80335086E-07	-1.00000080E-03	0.	1.02341083E+00	2.34252459E-02
			1	3 7.78E-05					
			2	3 3.17E-05					
			3	3 7.58E-06					
1.10E-02	7	80	9		-3.67810941E-07	3.56209326E-02	1.52155815E+00	1.02275360E+00	2.28233890E-02
			1	3 6.20E-05					
			2	4 2.73E-05					
			3	3 6.93E-06					
1.21E-02	8	90	10		9.55630952E-08	3.56209326E-02	1.52155815E+00	9.98998222E-01	2.11168830E-03
			1	3 1.52E-05					
			2	3 4.84E-05					
			3	3 1.48E-05					
1.31E-02	9	99	9		1.73554540E-08	3.40105482E-02	1.52155815E+00	9.98941622E-01	1.09096430E-03
			1	2 3.96E-05					

```
                                        2       2  3,46E-05
                                        3       2  8,83E-06
 1,38E-02    10      105     6                                -1,09608412E-07-3,38908948E-02 1,52155815E+00 9,99921361E-01 1,39389527E-04
                                        1       1  3,11E-05
                                        2       1  2,86E-05
                                        3       1  3,08E-05
 1,43E-02    11      108     3                                -1,08241477E-07-3,38766748E-02 1,52155815E+00 9,99990654E-01 1,44552668E-05
                                        1       1  1,46E-05
                                        2       1  7,43E-06
                                        3       1  7,24E-06
```

SUMMARY PRINT

ONETRAN EXAMPLE PROBLEM

SYSTEM BALANCE TABLE

GP		SOURCE	FISSION SOURCE	IN SCATTER	SELF SCATTER	OUT SCATTER
1	0.		3,2291218E-01	6,9862769E-06	1,8160669E-01	1,0866373E-01
2	0.		3,3552658E-01	6,6590459E-02	2,4295178E-01	3,3701747E-02
3	0.		3,4156124E-01	7,5190985E-02	1,1968234E-01	-4,4408921E-16
SUM	0.		1,0000000E+00	1,4238835E-01	5,4424081E-01	1,4236548E-01

GP	ABSORPTION	NEUTRON BALANCE	RIGHT LEAKAGE	NET LEAKAGE
1	1,3955199E-01	-1,1208414E-07	7,4703970E-02	7,4703970E-02
2	2,5290857E-01	2,3809314E-07	1,1550641E-01	1,1550641E-01
3	2,7506563E-01	1,7757775E-07	1,4228617E-01	1,4228617E-01
SUM	6,6752619E-01	1,1700006E-07	3,3249655E-01	3,3249655E-01

```
-0.         PV
 1,0000E-02 XLAL
 5,0000E-01 XLAM
 1,0000E-03 XLAX
 1,0000E-04 EPS OUTER
 1,0000E-04 EPS INNER
 3,9701E-04 EPS OUTER REBALANCE
 1,0000E-02 EPS TN
 1,0000E+00 POO
 1,0000E+00 NORM
```

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

1,48E-02 12 111 3 1,17888856E-07-3,38766748E-02 1,52155815E+00 1,00002483E+00 3,51782624E-06

EXECUTION TIME 1,4942E-02

DUMP WRITTEN ON UNIT 7

FINAL RADII

	RADII
1	0,
2	,10000000E+00
3	,58001276E+01
4	,98001276E+01

********** G R O U P 1 F L U X **********

FLUX COMPONENTS (ISOTROPIC)

I	RAV	CELL CNTR,FLUX	R=LEFT	LEFT FLUX	RIGHT FLUX	R=RIGHT	C,M,ZONE
1	1,0000E-02	1,7548707E-02	0,	1,7580135E-02	1,7517279E-02	2,0000E-02	1
2	3,0000E-02	1,7514399E-02	2,0000E-02	1,7533790E-02	1,7495008E-02	4,0000E-02	1
3	5,0000E-02	1,7482048E-02	4,0000E-02	1,7497380E-02	1,7466716E-02	6,0000E-02	1
4	7,0000E-02	1,7428958E-02	6,0000E-02	1,7465416E-02	1,7392501E-02	8,0000E-02	1
5	9,0000E-02	1,7229895E-02	8,0000E-02	1,7384104E-02	1,7074085E-02	1,0000E-01	1
6	6,7001E-01	1,7297217E-02	1,0000E-01	1,7220496E-02	1,7373937E-02	1,2400E+00	2
7	1,8100E+00	1,6676168E-02	1,2400E+00	1,7169457E-02	1,6182879E-02	2,3801E+00	2
8	2,9501E+00	1,5357635E-02	2,3801E+00	1,6143191E-02	1,4572080E-02	3,5201E+00	2
9	4,0901E+00	1,3503775E-02	3,5201E+00	1,4570136E-02	1,2437415E-02	4,6601E+00	2
10	5,2301E+00	1,1005361E-02	4,6601E+00	1,1452573E-02	9,5581491E-03	5,8001E+00	2
11	6,2001E+00	9,2354918E-03	5,8001E+00	9,4937944E-03	8,8771891E-03	6,6001E+00	3
12	7,0001E+00	7,4838161E-03	6,6001E+00	8,7249032E-03	6,2427291E-03	7,4001E+00	3
13	7,8001E+00	5,3164552E-03	7,4001E+00	6,1923210E-03	4,4405894E-03	8,2001E+00	3
14	8,6001E+00	3,7390183E-03	8,2001E+00	4,4102069E-03	3,0678297E-03	9,0001E+00	3
15	9,4001E+00	2,4615278E-03	9,0001E+00	3,0450365E-03	1,8780191E-03	9,8001E+00	3

FISSION RATE

1	0,	4	0,	6	3,48712E-03	8	3,09610E-03	10	2,21668E-03	12	3,01787E-05	14	1,50757E-05
2	0,	5	0,	7	3,34192E-03	9	2,72236E-03	11	0,	13	2,14359E-05	15	9,92488E-06
3	0,												

********** G R O U P 2 F L U X **********

FLUX COMPONENTS (ISOTROPIC)

I	RAV	CELL CNTR,FLUX	R-LEFT	LEFT FLUX	RIGHT FLUX	R-RIGHT	C,M,ZONE
1	1,0000E-02	1,8192928E-02	0,	1,8139258E-02	1,8146597E-02	2,0000E-02	1
2	3,0000E-02	1,8173905E-02	2,0000E-02	1,8179704E-02	1,8168106E-02	4,0000E-02	1
3	5,0000E-02	1,8183022E-02	4,0000E-02	1,8174218E-02	1,8191827E-02	6,0000E-02	1
4	7,0000E-02	1,8194916E-02	6,0000E-02	1,8193250E-02	1,8196583E-02	8,0000E-02	1
5	9,0000E-02	1,8108586E-02	8,0000E-02	1,8191555E-02	1,8025616E-02	1,0000E-01	1
6	6,7001E-01	1,8486319E-02	1,0000E-01	1,8448990E-02	1,8523648E-02	1,2400E+00	2
7	1,8100E+00	1,7836091E-02	1,2400E+00	1,8362978E-02	1,7309203E-02	2,3801E+00	2
8	2,9501E+00	1,6441574E-02	2,3801E+00	1,7273841E-02	1,5609307E-02	3,5201E+00	2
9	4,0901E+00	1,4483910E-02	3,5201E+00	1,5604725E-02	1,3363096E-02	4,6601E+00	2
10	5,2301E+00	1,1866774E-02	4,6601E+00	1,3370947E-02	1,0362602E-02	5,8001E+00	2
11	6,2001E+00	1,0146257E-02	5,8001E+00	1,0391072E-02	9,9014419E-03	6,6001E+00	3
12	7,0001E+00	8,9250569E-03	6,6001E+00	9,0868004E-03	7,9633134E-03	7,4001E+00	3
13	7,8001E+00	7,8793202E-03	7,4001E+00	7,9763959E-03	6,1822445E-03	8,2001E+00	3
14	8,6001E+00	5,3683388E-03	8,2001E+00	6,1826666E-03	4,5540110E-03	9,0001E+00	3
15	9,4001E+00	3,7723468E-03	9,0001E+00	4,5181051E-03	2,9965886E-03	9,8001E+00	3

FISSION RATE

1	0,	4	0,	6	3,72684E-03	8	3,31462E-03	10	2,39234E-03	12	3,59858E-05	14	2,16451E-05
2	0,	5	0,	7	3,59576E-03	9	2,91996E-03	11	0,	13	2,85438E-05	15	1,52101E-05
3	0,												

********** G R O U P 3 F L U X **********

FLUX COMPONENTS (ISOTROPIC)

I	RAV	CELL CNTR,FLUX	R-LEFT	LEFT FLUX	RIGHT FLUX	R-RIGHT	C,M,ZONE
1	1,0000E-02	1,2196536E-02	0,	1,2266901E-02	1,2126171E-02	2,0000E-02	1
2	3,0000E-02	1,2215028E-02	2,0000E-02	1,2190185E-02	1,2241470E-02	4,0000E-02	1
3	5,0000E-02	1,2320709E-02	4,0000E-02	1,2256219E-02	1,2385198E-02	6,0000E-02	1
4	7,0000E-02	1,2486766E-02	6,0000E-02	1,2394020E-02	1,2579512E-02	8,0000E-02	1
5	9,0000E-02	1,2683183E-02	8,0000E-02	1,2584957E-02	1,2781408E-02	1,0000E-01	1
6	6,7001E-01	1,3995656E-02	1,0000E-01	1,3869113E-02	1,4122199E-02	1,2400E+00	2
7	1,8100E+00	1,3655437E-02	1,2400E+00	1,4020612E-02	1,3290262E-02	2,3801E+00	2
8	2,9501E+00	1,2690016E-02	2,3801E+00	1,3271203E-02	1,2108829E-02	3,5201E+00	2
9	4,0901E+00	1,1345458E-02	3,5201E+00	1,2111746E-02	1,0579171E-02	4,6601E+00	2
10	5,2301E+00	9,6497753E-03	4,6601E+00	1,0599498E-02	8,7000525E-03	5,8001E+00	2
11	6,2001E+00	8,6904466E-03	5,8001E+00	8,7363035E-03	8,6445897E-03	6,6001E+00	3
12	7,0001E+00	8,2544366E-03	6,6001E+00	8,7137620E-03	7,7951112E-03	7,4001E+00	3
13	7,8001E+00	7,2317574E-03	7,4001E+00	7,8664656E-03	6,5970492E-03	8,2001E+00	3
14	8,6001E+00	5,9494709E-03	8,2001E+00	6,6387062E-03	5,2602356E-03	9,0001E+00	3
15	9,4001E+00	4,5735750E-03	9,0001E+00	5,2869655E-03	3,8601845E-03	9,8001E+00	3

FISSION RATE

1	0.	4	0.	6	2.82152E-03	8	2.55831E-03	10	1.94539E-03	12	3.32819E-05	14	2.39883E-05
2	0.	5	0.	7	2.75294E-03	9	2.28724E-03	11	0.	13	2.91584E-05	15	1.84407E-05
3	0.												

********** G R O U P S U M **********

FLUX COMPONENTS (ISOTROPIC)

I	RAV	CELL CNTR.FLUX	R-LEFT	LEFT FLUX	RIGHT FLUX	R-RIGHT	C.M.ZONE
1	1.0000E-02	4.7938170E-02	0.	4.8086293E-02	4.7790047E-02	2.0000E-02	1
2	3.0000E-02	4.7904132E-02	2.0000E-02	4.7903679E-02	4.7904585E-02	4.0000E-02	1
3	5.0000E-02	4.7985779E-02	4.0000E-02	4.7927817E-02	4.8043741E-02	6.0000E-02	1
4	7.0000E-02	4.8110640E-02	6.0000E-02	4.8052685E-02	4.8168595E-02	8.0000E-02	1
5	9.0000E-02	4.8020863E-02	8.0000E-02	4.8160617E-02	4.7881109E-02	1.0000E-01	1
6	6.7001E-01	4.9779192E-02	1.0000E-01	4.9538598E-02	5.0019785E-02	1.2400E+00	2
7	1.8100E+00	4.8167696E-02	1.2400E+00	4.9553048E-02	4.6782344E-02	2.3801E+00	2
8	2.9501E+00	4.4489226E-02	2.3801E+00	4.6688235E-02	4.2290216E-02	3.5201E+00	2
9	4.0901E+00	3.9333144E-02	3.5201E+00	4.2286607E-02	3.6379681E-02	4.6601E+00	2
10	5.2301E+00	3.2521911E-02	4.6601E+00	3.6423018E-02	2.8620804E-02	5.8001E+00	2
11	6.2001E+00	2.8072195E-02	5.8001E+00	2.8721170E-02	2.7423221E-02	6.6001E+00	3
12	7.0001E+00	2.4663310E-02	6.6001E+00	2.7325466E-02	2.2001154E-02	7.4001E+00	3
13	7.8001E+00	1.9627533E-02	7.4001E+00	2.2035182E-02	1.7219883E-02	8.2001E+00	3
14	8.6001E+00	1.5056828E-02	8.2001E+00	1.7231580E-02	1.2882076E-02	9.0001E+00	3
15	9.4001E+00	1.0807450E-02	9.0001E+00	1.2880107E-02	8.7347923E-03	9.8001E+00	3

FISSION RATE

1	0.	4	0.	6	1.00355E-02	8	8.96903E-03	10	6.55642E-03	12	9.94425E-05	14	6.07091E-05
2	0.	5	0.	7	9.71061E-03	9	7.92956E-03	11	0.	13	7.91382E-05	15	4.35756E-05
3	0.												

```
********************************************************* ZONE EDIT  1 *********************************************************

INPUT IDCA ARRAY       2
           2           3

INPUT NEOZ ARRAY      15
           1           1           1           1           1           2           2           2           2           2
           3           3           3           3           3
GROUP  1 ACTIVITY FOR POSITIONS 1 THRU IHT
    IZ
     1  3,2759083E-05    0,                4,0948854E-05
     2  6,8477563E-02    1,9858493E-01     1,6434615E-01
     3  7,1038317E-02    1,0052998E-01     2,6542499E-01
GROUP  2 ACTIVITY FOR POSITIONS 1 THRU IHT
    IZ
     1  1,7112754E-04    0,                1,8824029E-04
     2  1,1022750E-01    2,3515217E-01     2,7189469E-01
     3  1,4250378E-01    1,5353943E-01     2,5746643E-01
GROUP  3 ACTIVITY FOR POSITIONS 1 THRU IHT
    IZ
     1  4,7059623E-04    0,                4,7765517E-04
     2  1,7328668E-01    2,4248936E-01     2,4248936E-01
     3  1,0138174E-01    4,2158742E-03     1,5177147E-01
GROUP  4 ACTIVITY FOR POSITIONS 1 THRU IHT
    IZ
     1  6,7448285E-04    0,                7,8684432E-04
     2  3,5191183E-01    6,7622646E-01     6,7873020E-01
     3  3,1492384E-01    2,5828528E-01     6,7466288E-01



                  CONSTITUENT ACTIVITIES FOR MATERIAL  2
GROUP  1 ACTIVITY FOR POSITIONS 1 THRU IHT
    IZ
     1  0,                0,                0,
     2  6,8477563E-02    1,9858493E-01     1,6434615E-01
     3  7,1755876E-04    2,0809204E-03     1,7221410E-03
GROUP  2 ACTIVITY FOR POSITIONS 1 THRU IHT
    IZ
     1  0,                0,                0,
     2  1,1022758E-01    2,3515217E-01     2,7189469E-01
     3  1,4394321E-03    3,0707886E-03     3,5505993E-03
GROUP  3 ACTIVITY FOR POSITIONS 1 THRU IHT
    IZ
     1  0,                0,                0,
     2  1,7328668E-01    2,4248936E-01     2,4248936E-01
     3  3,0113387E-03    4,2158742E-03     4,2158742E-03
GROUP  3 ACTIVITY FOR POSITIONS 1 THRU IHT
    IZ
     1  0,                0,                0,
     2  3,5191183E-01    6,7622646E-01     6,7873020E-01
     3  5,1683296E-03    9,3675831E-03     9,4886145E-03



                  CONSTITUENT ACTIVITIES FOR MATERIAL  3
GROUP  1 ACTIVITY FOR POSITIONS 1 THRU IHT
    IZ
     1  0,                0,                0,
     2  0,                0,                0,
```

```
     3  7.0320759E-02    9.8449062E-02    2.6370284E-01
GROUP  2 ACTIVITY FOR POSITIONS 1 THRU IHT
    IZ
     1  0.               0.               0.
     2  0.               0.               0.
     3  1.4106435E-01    1.5046864E-01    2.5391583E-01
GROUP  3 ACTIVITY FOR POSITIONS 1 THRU IHT
    IZ
     1  0.               0.               0.
     2  0.               0.               0.
     3  9.8370398E-02    0.               1.4755560E-01
GROUP  3 ACTIVITY FOR POSITIONS 1 THRU IHT
    IZ
     1  0.               0.               0.
     2  0.               0.               0.
     3  3.0975551E-01    2.4891770E-01    6.6517427E-01
```

ZONE	RELATIVE POWER DENSITY
1	0.
2	8.421746E-01
3	1.578254E-01

********** POINT EDIT 1 **********

INPUT IOMA ARRAY 1
ALL ENTRIES = 1

INPUT NEOP ARRAY 5
1 2 3 4 5

POINT MICROSCOPIC ACTIVITY FOR MATERIAL 1

GROUP 1 ACTIVITY FOR POSITIONS 1 THRU IMT
MESH

1	1.0548081E-03	1.0510368E-03	0.	0.	1.3185101E-03	1.3137959E-03
2	1.0520274E-03	1.0497005E-03	0.	0.	1.3150342E-03	1.3121256E-03
3	1.0498428E-03	1.0480030E-03	0.	0.	1.3123035E-03	1.3100037E-03
4	1.0479249E-03	1.0435500E-03	0.	0.	1.3099062E-03	1.3044375E-03
5	1.0430462E-03	1.0244451E-03	0.	0.	1.3038078E-03	1.2805564E-03

GROUP 2 ACTIVITY FOR POSITIONS 1 THRU IMT
MESH

1	5.4717774E-03	5.4439792E-03	0.	0.	6.0189552E-03	5.9883771E-03
2	5.4539112E-03	5.4504319E-03	0.	0.	5.9993023E-03	5.9954751E-03
3	5.4522654E-03	5.4575480E-03	0.	0.	5.9974919E-03	6.0033028E-03
4	5.4579749E-03	5.4589748E-03	0.	0.	6.0037724E-03	6.0048722E-03
5	5.4574666E-03	5.4076849E-03	0.	0.	6.0032132E-03	5.9484534E-03

GROUP 3 ACTIVITY FOR POSITIONS 1 THRU IMT
MESH

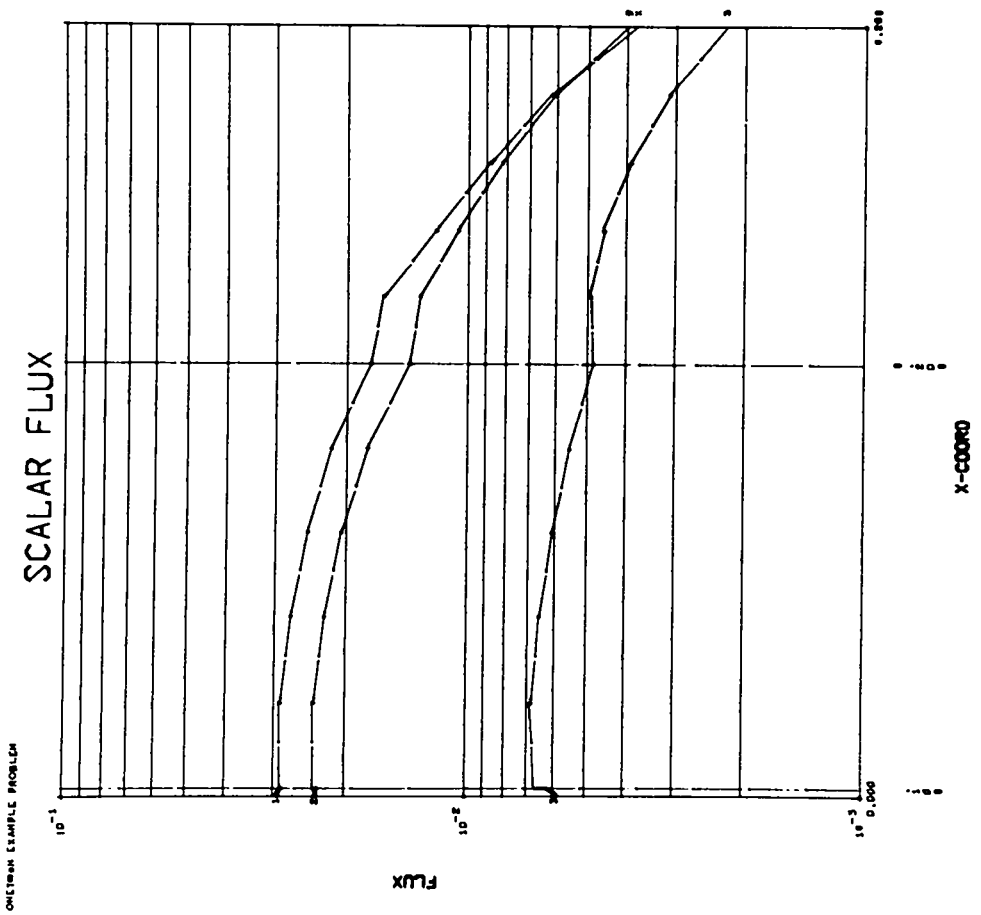
1	1.4720281E-02	1.4551405E-02	0.	0.	1.4941085E-02	1.4769676E-02
2	1.4628222E-02	1.4689764E-02	0.	0.	1.4847646E-02	1.4910111E-02
3	1.4707463E-02	1.4862238E-02	0.	0.	1.4928075E-02	1.5085171E-02
4	1.4872824E-02	1.5095415E-02	0.	0.	1.5095916E-02	1.5321846E-02
5	1.5101949E-02	1.5337690E-02	0.	0.	1.5328478E-02	1.5567755E-02

GROUP 4 ACTIVITY FOR POSITIONS 1 THRU IMT
MESH

1	2.1246866E-02	2.1046421E-02	0.	0.	2.2278550E-02	2.2071849E-02
2	2.1134161E-02	2.1189897E-02	0.	0.	2.2161982E-02	2.2217712E-02
3	2.1209571E-02	2.1367789E-02	0.	0.	2.2237870E-02	2.2398478E-02
4	2.1378724E-02	2.1597940E-02	0.	0.	2.2409595E-02	2.2631156E-02
5	2.1602462E-02	2.1769820E-02	0.	0.	2.2635499E-02	2.2796765E-02

***** PLOTS COMPLETED *****

INTERFACE FILES WRITTEN


SCALAR FLUX
FLUX
X-COORD
0.000
0.500
1.000
10^{-1}
10^{-2}
10^{-3}

IV. PROGRAMMING INFORMATION

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

A. Program Structure

1. Role and Function of Subprograms

We describe in Table VII the function of all the subprograms in ONETRAN.

2. Relation of Problem Variables and Program Mnemonics

In much of the material in this manual we have used variables actually appearing in the FORTRAN of the program. A list of the relations between problem variable symbols and program variable names is given in Table VIII.

3. Definition of Variables in Common Blocks

Tables IX and X define the variables stored in blank common block IA and the named common block of ONETRAN. The container array, A, for problem data is also in blank common. Block IA contains problem input parameters, first word addresses of data stored in the A array, and data generated by the program.

TABLE VII

FUNCTION OF ONETRAN SUBROUTINES

Subroutine	Function
ONETRAN	Main driver of program. Initializes program parameters; calls input, initialization, computation, and output routines.
	Input Functions
INPUT1	Reads header and control integer and floating point variables, performs some checking of input data.
INPUT2	Calculates commonly used integers, large and small core storage pointers; calls various input subroutines; reads problem-dependent input arrays, performs more checking of input data.
SNCON	Reads or generates S_N quadrature constants; calculates some indexing arrays and spherical harmonic polynomials.
IFINSN	Reads S_N constants from interface file ISNCON.
CSPREP	Reads cross sections in standard LASL format, FIDO format, or from interface file by calling IFINXS. Prints cross sections, performs adjoint transpositions and reversals of cross sections, checks cross sections, and stores cross sections in LCM.
IFINXS	Interface input of cross sections from standard interface file ISOTXS.
READF	Reads initial flux guess from cards or standard interface file by calling IFINF.
IFINF	Reads initial flux guess, either scalar or angular flux, from standard interface file ITFLUX or IAFLUX.
READQ	Reads distributed and boundary sources from cards or standard interface file by calling IFINQ.
IFINQ	Reads distributed and boundary sources from standard interface file FIXSRC.

TABLE VII (continued)

Subroutine	Function
	Initialization Functions
INITAL	Performs mixing of cross sections, modifies coarse-mesh boundaries for critical size calculations, calculates geometric functions by call to GEOFUN, initializes inhomogeneous sources by call to INITQ and fission arrays by call to INITF, calculates macroscopic cross section arrays.
GEOFUN	Calculates various geometric functions on the coarse and fine mesh.
INITQ	Generates volume and surface integrals of inhomogeneous sources for rebalance, normalizes sources, stores boundary sources in boundary flux array.
INITF	Computes $\chi\ \nu\Sigma_f$ array for fission source and transposes for adjoint problems, calculates volume integral for fission source and normalizes fluxes.
	Computation Functions
MONITR	Prints résumé of convergence parameters, monitor line headings, and outer iteration monitor data.
OUTER	Performs a single outer iteration, contains the group loop. Calculates source to the group by call to SOURCE, performs the inner iterations by call to INNER, calculates group sums by call to SUMS.
SOURCE	Calculates source to the group from inhomogeneous sources, fission in all groups, and inscattering from other groups. Calculates total source for inner iteration rebalance.
INNER	Performs the inner iterations for a group. Adds within-group scattering to source, performs sweeps over space-angle mesh, solving the 2x2 system for the edge angular fluxes, calculates rebalance flows and absorptions, performs rebalance or Chebyshev accelerations, and checks convergence of inner iterations.
SETBC	Sets the angular flux boundary condition on either the left or right boundary. Called by INNER.
REBAL	Performs inversion of tridiagonal matrix for group coarse-mesh rebalance factors. Called by INNER and GREBAL.
SUMS	Accumulates quantities in system balance table for each group. Renormalizes fission source to group and calculates λ for k_{eff} calculations.
GREBAL	Computes fission source, normalizes fission source and flux moments, computes group rebalance factors by call to REBAL and performs outer iteration acceleration.
NEWPAR	Computes new parameters for implicit eigenvalue search.
	Output Edit Functions
SUMARY	Prints system balance table for each group and final iteration monitor line.
FINAL	Controls final edit output. Prints flux moments, angular flux, and fission rate by call to EDIT. Reads zone and point edit input. Allocates temporary storage for edits and performs zone and point edits by call to ZEDIT and PEDIT. Calls PLOTTR routine if specified. Writes interface file output by call to IFRITE.
EDIT	Prints scalar flux and components, angular flux, and fission rate.
ZEDIT	Calculates zone macroscopic activities, constituent activities, microscopic activities, and power densities.
PEDIT	Calculates pointwise microscopic activities.
IFRITE	Writes interface files SNCONS, FIXSRC, RTFLUX or ATFLUX, and RAFLUX or AAFLUX.
TIMEXF	Writes angular flux file NTIMEX for initial condition to TIMEX code. Called by FINAL.

TABLE VII (continued)

Subroutine	Function
	Service Routines
DUMPER	Reads or writes restart dump.
PRINTP	Prints input control integer and floating point variables.
MAPPER	Draws material map of system.
LOAD	Los Alamos data loader.
WRITE	Generalized output routine for printing 1D, 2D, or 3D arrays, either integer or floating point.
ERROR	Prints error messages and sets fatal error trigger.
REED	Handles all binary reading operations including rewind and bulk memory transfers (LCM).
RITE	Handles all binary writing operations including end of file and rewind and bulk memory transfers.
PLOTTR	Plots scalar flux on film file NFILM. Calls numerous system-dependent plotting routines.

TABLE VIII

RELATION OF PROBLEM VARIABLES TO PROGRAM MNEMONICS

Program Mnemonic	Subroutine	Problem Variable	Refer to
PN(NM,MM)	SNCON,INNER	$R_n(\Omega)$	Eqs. (10),(11)
C(IHM,MT)	SOURCE,INNER	$\sigma_a, \nu\sigma_f, \sigma, \sigma^n_{s,h\to g}$	Sec. II.B.1.
WGT(MM)	SNCON,INNER	w_m	Sec. II.B.2.
U(MM)	SNCON,INNER	μ_m	Sec. II.B.2.
WMU(MM)	SNCON,INNER	$w_m\ \mu_m$	Sec. II.B.2.
CT(IT)	INITAL,INNER	σ	Sec. II.B.1.
CS(IT)	INITAL,INNER	$\sigma^o_{s,g\to g}$	Sec. II.B.1.
BP(MM)	SNCON,INNER	$\alpha_{m+\frac{1}{2}}/w_m$	Sec. II.B.2.
BM(MM)	SNCON,INNER	$\alpha_{m-\frac{1}{2}}/w_m$	Sec. II.B.2.
FL(IM+1)	INNER,REBAL	$FL_{k\pm\frac{1}{2}}$	Eq. (45)
FR(IM+1)	INNER,REBAL	$FR_{k\pm\frac{1}{2}}$	Eq. (46)
AB(IM)	INNER,REBAL	AB_k	Eq. (47)
F(IM)	INNER,REBAL	f_k	Eq. (49)
Z(10,IT)	INNER	z_i	Table V
V(2,IT)	INNER	V-, V+	Table V
AI(IM+1)	INNER	A-,A+	Table V
Q(NM,2,IT)	SOURCE,INNER	Source to group moments	Sec. II.B.2.
FLUX(NM,2,IT)	INNER	Flux moments, inner iteration ℓ+1	Sec. II.B.2.
FLUXA(2,IT)	INNER	Scalar flux, inner iteration ℓ	Eq. (44)
FLUXB(2,IT)	INNER	Scalar flux, inner iteration ℓ-1	Eq. (44)
S1,S2	INNER	$S_{i-\frac{1}{2}}, S_{i+\frac{1}{2}}$	Sec. II.C.

TABLE VIII (continued)

Program Mnemonic	Subroutine	Problem Variable	Refer to
PSIB	INNER	ψ_b	Eq. (35)
AFE(NLEV,IT)	INNER	$\psi_{m+\frac{1}{2}}$	Sec. II.C.
AFC(2,IT)	INNER	$\psi_{i+\frac{1}{2}}$	Eq. (35)
AF1,AF2	INNER	$\psi_{i-\frac{1}{2}}$, $\psi_{i+\frac{1}{2}}$	Eq. (35)
RNORM	INNER	$1/\rho(B)$	Eq. (52)
CN	INNER	$\omega_{\ell+1}$	Eq. (55)
EVR	SOURCE,GREBAL	$1/k_{eff}$	Sec. II.D.5.
ALA	GREBAL	λ	Eq. (58)
XLA	GREBAL	λ_x	Sec. II.D.7.
EV	GREBAL	Eigenvalue	Sec. III.B.9.

TABLE IX

CONTENTS OF BLANK COMMON BLOCK IA*

Position	Name	Pointer for Array	Remarks
1	ITH		Theory
2	ISCT		Scattering order
3	ISN		S_N order
4	IGM		Number of energy groups
5	IM		Number of coarse-mesh intervals
6	IBL		Left boundary condition indicator
7	IBR		Right boundary condition indicator
8	IEVT		Eigenvalue type specification
9	ISTART		Starting option indicator
10	IQOPT		Source input option indicator
11	IGEOM		Geometry indicator
12	IQUAD		Source of S_N constants indicator
13	MT		Total number of materials
14	MTP		Number of cross-section materials from ISOTXS file
15	MCR		Number of cross-section materials from cards
16	MS		Number of mixture instructions
17	IHT		Position of total cross section in table
18	IHS		Position of self-scatter cross section in table
19	IHM		Cross-section table length
20	IDEN		Space-dependent density factor trigger
21	IQAN		Distributed source anisotropy order
22	IQL		Left boundary source trigger
23	IQR		Right boundary source trigger
24	IACC		Inner iteration acceleration option indicator
25	OITM		Outer iteration limit
26	IITL		Inner iteration limit until $\|1 - \lambda\| < 10*EPSO$

*Blank entries are available for future use.

TABLE IX (continued)

Position	Name	Pointer for Array	Remarks
27	IITM		Inner iteration limit after $\|1 - \lambda\| < 10*EPSO$
28	IFISS		Fission fraction-type indicator
29	IEDOPT		Edit option trigger
30	IPLOT		Plotting option trigger
31	ITLIM		Time limit for problem
32	IPVT		Parametric eigenvalue trigger
33	IFO		Interface file output trigger
34	I1		Input flux print suppression trigger
35	I2		Final flux print indicator
36	I3		Cross-section print indicator
37	I4		Final fission print trigger
38	I5		Source print indicator
39	I6		Fine-mesh geometry print suppression trigger
40	IANG		Angular flux storage indicator
41			
42			
43			
44			
45			
46			
47			
48			
49			
50			
51			
52			
53			
54			
55			
56			
57			
58			
59			
60			
61	EV		Eigenvalue
62	EVM		Eigenvalue modifier
63	PV		Parametric value of k_{eff} or alpha
64	XLAL		Search lambda lower limit
65	XLAH		Search lambda upper limit
66	XLAX		λ convergence precision
67	EPSO		Outer convergence precision
68	EPSI		Inner convergence precision
69	EPSX		Outer rebalance convergence precision
70	EPST		Chebyshev norm convergence precision
71	POD		Parameter oscillation damper
72	NORM		Normalization amplitude

TABLE IX (continued)

Position	Name	Pointer for Array	Remarks
73	BHGT		Buckling height
74	BWTH		Buckling width
75			
76			
77			
78			
79			
80			
81	TIME		Problem execution time
82	TOUT		Time in OUTER routine
83	IDUMP		Time limit dump trigger. Set in OUTER
84	EPSR		Convergence precision in REBAL for periodic boundary condition
85			
86			
87			
88			
89			
90			
91			
92			
93			
94			
95	TIMBDP		Time before dump
96	TACC		Time accumulated in OUTER
97	IGCDMP		Group index when dump was taken
98	TIN		Time for INNER
99	TSLDMP		Time since last dump
100	NLEV		Number of quadrature ξ levels
101	MIN		Total number of input cross-section blocks = MCR + MTP*(ISCT+1)
102	IHNN		Position of $\sigma_{n,2n}$ in cross-section table = IHT-4
103	IHF		Position of $\nu\sigma_f$ in cross-section table = IHT-1
104	IHA		Position of σ_a in cross-section table = IHT-2
105	MM		Number of quadrature angles
106	NM		Number of spherical harmonic and flux moments
107	NMQ		Number of distributed source moments
108	M2		MM/2
109	NN		ISN/2
110	IP		Number of coarse-mesh points = IM+1
111	IGP		IGM+1
112	IHMT		IHM*MT
113	ISCP		ISCT+1
114	M2P		M2+1
115	IHTR		Position of σ_{tr} in cross-section table = IHT-3

TABLE IX (continued)

Position	Name	Pointer for Array	Remarks
116	ITP		Number of fine mesh points = IT+1
117			
118	IMGP		IM*IGM
119	IT		Number of fine mesh intervals
120			
121	IPGP		IP*IGM
122	IFISP		Zone-dependent fission fraction trigger
123	LFLM		Last address of flux block
124	EVR		Eigenvalue reciprocal = 1/EV
125	KM		IM for zone-dependent fission fractions, 1 otherwise
126			
127	KEND		Last LCM position used
128	LAST		Last small core position used
129			
130			
131	LIHR	IHR(IM+1)	Number of fine-mesh intervals per coarse mesh
132	LW	W(MM)	Point weights
133	LU	U(MM)	Point cosines
134	LWM	WM(MM)	Point weights*cosines
135	LBP	BP(MM)	$\alpha_{m+\frac{1}{2}}/w_m$
136	LBM	BM(MM)	$\alpha_{m-\frac{1}{2}}/w_m$
137	LDM	MD(MM)	Reflected direction index
138			
139			
140	LUB	UB(ISN)	Level cosines
141	LWB	WB(ISN)	Level weights
142	LUSTRT	USTRT(NLEV)	Starting direction cosines
143			
144	LPN	P(NM,MM)	Spherical harmonic functions
145	LLI	LI(MM)	Level indices
146	LFT	FT(2*ISCT+1)	Factorials
147			
148			
149			
150	LC	C(IHM,MT)	Cross-section blocks for a group
151			
152	LCT	CT(IT)	Total cross-section * density
153	LCS	CS(IT)	Self-scatter cross-section * density
154	LCA	CA(IT)	Absorption cross-section * density
155			
156	LDC	IDC(IM+1)	Cross-section identification number for each coarse-mesh interval
157	LMN	MIXNUM(MS)	Mixture numbers
158	LMC	MIXCOM(MS)	Mixture commands
159	LMD	MIXDEN(MS)	Mixture densities
160			

TABLE IX (continued)

Position	Name	Pointer for Array	Remarks
161	LDEN	DEN(IT)	Fine-mesh densities
162			
163			
164			
165			
166	LQ	Q(NM,2,IT)	Distributed source moments
167	LQR	QR(MM/2)	Right boundary source
168	LQL	QL(MM/2)	Left boundary source
169	LFL	FLUX(NM,2,IT)	Flux moments
170	LFLA	FLUXA(2,IT)	Scalar flux for iterate k-1
171	LFLB	FLUXB(2,IT)	Scalar flux for iterate k-2
172			
173			
174			
175	LBL	BL(MM)	Left boundary flux
176	LBR	BR(MM)	Right boundary flux
177	LAFE	AFE(NLEV,IT)	Edge angular flux for angular extrapolation
178	LAFC	AFC(2,IT)	Mesh cell edge angular flux
179	LB1	B1(MM/2)	Left boundary flux, previous iteration
180	LB2	B2(MM/2)	Right boundary flux, previous iteration
181			
182			
183			
184			
185			
186			
187			
188	LRAD	RAD(IM+2)	Coarse-mesh boundary radii
189	LIDR	IDR(IT+2)	Coarse-mesh zone identifier on the fine mesh
190	LH	H(IM+1)	Fine-mesh Δr_i in each coarse mesh
191	LAI	AI(IT+1)	Fine-mesh area elements
192	LV	V(2,IT+1)	Fine-mesh volume elements
193			
194	LZ	Z(10,IT)	Finite element polynomials on the fine mesh
195			
196			
197	LR	R(IT+1)	Fine-mesh radii r_i
198			
199	LRM	RM(IM+1)	Coarse-mesh modifiers for delta calculations
200	LRDA	RADA(IM+1)	Modified coarse-mesh boundaries
201	LDEL	DEL(IM+1)	Coarse-mesh thickness
202			
203			
204			
205			

TABLE IX (continued)

Position	Name	Pointer for Array	Remarks
206			
207			
208	LQG	QG(IGP)	Inhomogeneous source to a group
209	LFG	FG(IGP)	Fission source to a group
210	LSIN	SIN(IGP)	In-scatter to a group
211	LSS	SS(IGP)	Self-scatter in a group
212	LSOU	SOUT(IGP)	Out-scatter from a group
213	LRL	RL(IGP)	Right boundary net leakage for a group
214	LNL	NL(IGP)	System net leakage for a group
215	LABG	ABG(IGP)	Absorption in a group
216	LBAL	BAL(IGP)	Balance number in a group
217	LCHI	CHI(IM,IGM)	Fission matrix ($\chi\ \nu\sigma_f$) or fission fractions χ
218	LVEL	VEL(IGP)	Group speeds
219	LAF	AF(IGP)	Chebyshev acceleration factors for each group
220	LLB	LB(IGP)	Left boundary group albedo
221	LRB	RB(IGP)	Right boundary group albedo
222			
223			
224			
225			
226			
227	LF	F(IM,IGM)	Coarse-mesh rebalance factors
228	LFR	FR(IM+1,IGM)	Coarse-mesh boundary right flows
229	LFLL	FL(IM+1,IGM)	Coarse-mesh boundary left flows
230	LAB	AB(IM)	Effective rebalance absorption
231	LQQ	QQ(IM)	Total rebalance source (fission + in-scatter + inhomogeneous)
232	LQQG	QQG(IM,IGM)	Total inhomogeneous source on the coarse mesh
233	LCR	CR(IM,IGM)	Effective absorption on the coarse mesh
234	LHA	HA(IM)	Work vector for rebalance inversion
235	LGA	GA(IM)	Work vector for rebalance inversion
236	LFGG	FGG(IM,IGM)	Fission matrix $\chi\ \nu\sigma_f$/EV
237	LSGG	SGG(IM,IGM)	Scattering matrix
238			
239			
240			
241			
242			
243			
244	LENC		Length of LCM cross-section block: LDC-LC
245	LENQ		Length of LCM inhomogeneous source block: LFL-LQ
246	LENF		Length of LCM flux block: LAFE-LFL
247	LENS		Length of LCM source to group block: NM*2*IT
248	LNAF		Length of LCM angular flux block: 2*IT
249	LNFS		Length of LCM fission spectrum: IM*IGM
250			

TABLE IX (continued)

Position	Name	Pointer for Array	Remarks
251	LNFG		Length of LCM fission matrix block: IM*IGM
252	LNSG		Length of LCM scattering matrix block: IM*IGM
253			
254			
255			
256	KC		Origin of LCM cross-section array
257	KQ		Origin of LCM source array
258	KF		Origin of LCM flux array
259	KS		Origin of LCM source to group array
260	KAF		Origin of LCM angular flux array
261	KFS		Origin of LCM fission spectrum array
262	KAFST		Origin of LCM starting direction angular flux array
263	KFG		Origin of LCM fission matrix array
264	KSG		Origin of LCM scattering matrix array
265			
266			
267			
268			
269	ALR		Right boundary albedo for a group
270	ALL		Left boundary albedo for a group
271	SUMMUL		$\sum_m w_m \mu_m$ for leftward directions
272	SUMMUR		$\sum_m w_m \mu_m$ for rightward directions
273	OITNO		Outer iteration number
274	IITOT		Inner iteration total
275	E1		$1 - \lambda$
276	E2		$\|1 - \lambda\|$
277	E3		Rebalance factor error
278			
279	EVP		Eigenvalue for previous outer iteration
280	EVPP		Eigenvalue for previous-previous outer iteration
281	ALA		λ
282	ALAR		λ for previous outer iteration
283	XLAP		λ_x for previous iteration
284	XLAPP		λ_x for previous-previous iteration
285	EVS		Eigenvalue slope, used in NEWPAR
286	ICNT		NEWPAR trigger to indicate $\|1 - \lambda\| <$ XLAL
287			
288			
289	TS		Total source to a group
290	IITNO		Inner iteration number
291	G		Group index
292	TF		Total fission source to a group
293	AFA or AF		Chebyshev acceleration factor for a single group
294	NGO		Convergence trigger set in NEWPAR
295	NGOTO		Problem path trigger set in GREBAL

TABLE IX (continued)

Position	Name	Pointer for Array	Remarks
296	ICONV		Inner iteration convergence trigger
297			
298			
299			
300			

TABLE X

CONTENTS OF NAMED COMMON BLOCK/UNITS/

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

Position	Name	Contents and Remarks
1	NINP	Problem code-dependent decimal input
2	NOUT	Problem decimal output
3	NDMP1	First restart dump unit
4	NDMP2	Second restart dump unit
5	ISNCON	Interface form of S_N constants
6	ISOTXS	Interface form of multigroup cross section file ISOTXS
7	IFIXSR	Interface form of inhomogeneous source (Q-source)
8	IAFLUX	Interface form of angular flux
9	ITFLUX	Interface form of total flux
10	NFILM	Plotting routine output
11	NEXTRA	Scratch unit used in subroutine LOAD
12	NTIMEX	Special angular flux file for TIMEX initial condition

4. Machine-Dependent Subprogram

a. LCM System Routines

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

```
CALL REED(0,KF+(IG-1)*LENF,FLUX,LENF,1)
```

which is equivalent to

```
CALL ECRD(FLUX,KF+(IG-1)*LENF,LENF,IER).
```

In these statements FLUX is the SCM container array, KF+(IG-1)*LENF is the location of the first word of the IG^{th} group flux array in LCM, and LENF words are transferred. IER is an error indicator.

b. General System Routines

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

Use of an end-of-file test is made in INPUT1 to detect the last case of a sequence of cases. The test must be replaced by an equivalent statement to obtain a normal exit.

The subroutine PLOTTR, which plots the scalar flux on a film file NFILM, calls several LASL plotting subroutines. These 15 routines are described with comment cards in the code listing to facilitate the user's conversion of the routine to the plotting software of his installation.

B. External and Internal Data Files

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

C. Hardware Requirements

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

D. Software Requirements

1. CDC Machines

The code was designed to operate on the CDC 7600 under the CROS operating system[17] which was developed at Los Alamos. The system uses the CDC RUN compiler with a CDC optimizer attached. The disk units provide storage for input, output, scratch, and resident files.

2. ONETRAN for the IBM-360

Although ONETRAN was written for the CDC 7600, the coding was performed so that the conversion to the IBM-360 would involve as few changes as possible. Past experience has found that the four-byte (single precision) floating-point mode is adequate for most problems.

The major change made in the conversion of ONETRAN is the treatment of peripheral storage. The vast amount of fast core available on the IBM-360 is one of the cheaper resources of that machine. Thus the data normally stored in LCM (large core memory) is stored directly after the A container array in fast core. The CDC 7600 system routines ECRD and ECWR in subroutines REED and RITE, respectively, are replaced by simple routines which move data to and from sections of the A container array. It is thus possible to keep the LCM pointer structure of the code with no change in logic and with a slight overhead in time for data movement.

In addition to the storage reorganization, the following changes are made to effect the IBM conversion of ONETRAN:

1. The subroutine DATE1, called from INPUT1, must be provided by the user to return the date as an A8 word. A local system routine must be provided for SECOND to return the floating-point value of the current time in order for the periodic and time limit dump options to work.
2. A separate subroutine is provided to process the FIDO cross-section format. The CDC 7600 algorithm to read this format uses a rewind command, resulting in a prohibitively large amount of wait time on IBM systems.
3. Hollerith 6H constants throughout the code are typed as double precision (REAL*8).
4. The IF(EOF,NINP) CDC job termination test in INPUT1 is replaced with a read using the IBM END parameter.
5. Several options are present in subroutine REED to treat the reading of interface file identification records. The information in this identification record is presently bypassed by dummy reads.

E. Programming Considerations

1. Storage Management

a. Variable Dimensioning

A single container array, A, in common is used for the blocks of data required in executing a problem. The storage of all data is consecutive and compact in the A array so that the size of a problem is limited by the total storage required rather than by the size of individual parameters. A pointer word is associated with each data block and is used to index A to locate the block. For example, LFL is the first word address of the flux

TABLE XI

ONETRAN FILE REQUIREMENTS

Name	Logical Unit	Contents	Remarks
NINP	10	Problem code-dependent decimal input.	The user may wish to equate this file to the system input file.
NOUT	9	Problem decimal output.	The user should equate this file to the system decimal printed output file.
NDMP1	7	Restart dump.	This unit is used to receive the first restart dump when the problem is not restarted from a previous dump. The unit must contain the restart dump information when the problem is restarted and will then be used to receive the second restart dump (NDMP2 receives the first dump).
NDMP2	5	Restart dump.	Second restart dump unit.
NEXTRA	18	Scratch file.	The file is used in the decimal mode by subroutine LOAD for Hollerith conversions rather than the core-to-core conversions given by the FORTRAN statements of ENCODE and DECODE on CDC machines.
IAFLUX	31	Interface form of angular flux (either adjoint or regular).	The code requires that this unit be used when a flux guess is requested from the angular flux interface file. The unit is rewound and the records of the first file are used as the input guess. Output of the angular fluxes in interface form is also placed on this file. The file is rewound prior to processing the fluxes and an end of file is placed on the file after the last write. Data for one problem only is kept on this file.
ITFLUX	30	Interface form of total flux (either adjoint or regular).	The code requires that this unit be used when a flux guess is requested from the total flux interface file. The unit is rewound and the records of the first file are used as the input guess. The interface form of the total flux is prepared on this file as problem output by rewinding the file and writing the file in standard format. An end of file is placed on the file after the last write instruction.
ISNCON	32	Interface form of S_N constants.	When the file is used as input, the file is rewound and read. When used as output the file is rewound and written, including an end of file.
IFIXSR	33	Interface form of both distributed and boundary sources.	This file is used as input for the cell-centered inhomogeneous source. Boundary sources (if any) are also obtained from this file.
ISOTXS	34	Interface form of the cross-section multigroup file ISOTXS.	This file is only used as input when cross sections are requested from an interface file library.
NFILM	12	Film file.	This file is used as output of the plotting subroutine PLOTTR. The LASL plotting software generates a magnetic tape that is used to generate film output by an FR-80 or SC-4020 film recorder. The PLOTTR routine could be modified by the user to generate CALCOMP plotter output.
TIMEX	15	TIMEX angular flux file.	The subroutine TIMEXF generates a binary file of angular fluxes to be used as initial conditions by the TIMEX code.

block in A and A (LFL) is the first word of the flux array. When subroutine calls are written, the address of a data block, say A(LFL), is passed through the argument call. In the subroutine the data block is variably dimensioned so that it may be easily indexed by its subscripts, e.g., FLUX(N,I,J).

b. Allocation of Large Core Memory (LCM)

The allocation of storage in large core memory (LCM) is handled in the same manner as core storage. Most of the group-dependent arrays are stored in LCM so the dimensionality is IGM times the core requirement of the array. For example, there are IGM*NM*2*IT LCM locations required for FLUX(NM,2,IT).

Certain blocks of data are stored contiguously in core so that they may be read in and out of LCM in a single stream. For example, the flux block includes FLUX(NM,2,IT), FLUXA(2,IT), FLUXB(2,IT), BL(MM), and BR(MM). The first word of this block is LFL, and the last word is LAFE-1. The cross-section block includes the cross sections C(IHM,MT), the total cross section, CT(IT), the scattering cross section, CS(IT), and the absorption cross section CA(IT). The first word of this block is LC, and the last word is LDC-1. A complete list of LCM storage is given in Table XII.

c. Computation of Required Storage

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

```
LAST=MT*IHM + IT*(24+NLEV+4*NM)+7*IM*IGM
```

The amount of LCM is given by

```
LASTEC=IGM*(MT*IHM+2*NM*2*IT+3*MM+5*IT+3*IM*IGM)
+ NM*2*IT+Conditional Blocks
```

where the conditional block size is

```
2*IT        if IACC = ± 3
```

plus

```
2*IT*IGM*(MM+NLEV)      if IANG ≠ 0.
```

d. Temporary Storage Requirements

The amount of fast core storage actually calculated for LAST is the maximum of two quantities. The total SCM required for problem execution and temporary SCM required for problem input. Usually, the problem data requirement is much larger than the temporary storage requirement during input, but occasionally, the input-cross-section requirement (IGM*IHM) is largest.

TABLE XII

LCM STORAGE PARAMETERS

LCM First Word Address	Length per Block	Number of Blocks	Contents
KC	LENC=IHM*MT+3*IT	IGM	Cross-section blocks by group
KQ	LENQ=NM*2*IT+2*MM/2	IGM	Inhomogeneous distributed and boundary sources
KF	LENF=NM*2*IT+2*(2*IT)+2*MM	IGM	Scalar flux and moments, boundary fluxes, and fluxes from previous iterations
KFS	LNFS=IGM*IM	IGM	χ or $\chi \nu\sigma_g$ array for each group
KFG	LNFG=IGM*IM	IGM	$\chi \nu\sigma_f \phi$ array for each group
KSG	LNSG=IGM*IM	IGM	$\sigma_s \phi$ array for each group
KS	LENS=NM*2*IT	1	Source to group array
KAF	LNAF=2*IT	MM*IGM	Angular flux array by group. Stored only if IANG≠0
KAFST	LNAF=2*IT	NLEV*IGM	Starting direction angular flux array by group. Stored only if IANG≠0

At the end of problem execution, additional temporary storage is required to perform the edits. This temporary storage is reallocated for each zone and point edit. Temporary storage is also required if the interface file output is requested. This temporary storage is also usually less than the problem data storage. The actual allocation is performed in subroutine FINAL.

e. Overstorage of Data in Core

In ONETRAN, a certain amount of overstorage is used to reduce the total amount of small core memory (SCM) required; i.e., more than one array may reside in the same SCM location as the problem progresses. This is done primarily with the CHI(IM,IGM) and FGG(IM,IGM) arrays. A similar overstorage is performed when the temporary storage is allocated for input in the INPUT2 subroutine.

2. Restart Tape Composition

The restart dump is composed of the following records: common block length LENIA, common block IA, data common block A, and LCM data blocks in the order in which they appear in LCM. The final dump contains the current group (IGCDMP) value of zero. Both the reading and the writing of the restart dumps is performed by subroutine DUMPER.

3. Standard Interface Files

The standard interface files read and written by ONETRAN are Version III files.[5] The coding which process these files are all written as separate subroutines. All files are rewound prior to either reading or writing so that the interface files for several problems may not be stacked on the same file. In the reading of the interface files, the first record containing the file identification data HNAME,(HUSE(I),I=1,2), IVERS is skipped by a dummy read statement. For input or output of the scalar or angular flux files, no physical unit distinction is made for regular or adjoint problems. If a standard interface file is used for an input flux guess and a standard interface file output is requested, the input file information is destroyed. Since the discontinuous representation (two values per mesh cell) of the distributed source and fluxes is incompatible with the standard interface file (one value per mesh cell), only the cell-centered values of these quantities are read or written.

ACKNOWLEDGEMENTS

This version of ONETRAN is derived from a diamond-difference version originally written by K. D. Lathrop. The application of the discontinuous finite element scheme to the transport equation was developed by Wm. H. Reed.

REFERENCES

1. T. R. Hill and Wm. H. Reed, "TIMEX: A Time-Dependent Explicit Discrete Ordinates Program for the Solution of the Multigroup Transport Equation," to be published as a Los Alamos Scientific Laboratory report.
2. Wm. H. Reed, "The Effectiveness of Acceleration Techniques for Iterative Methods in Transport Theory," Nucl. Sci. Eng. 45, 245 (1971).
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. Wm. H. Reed, T. R. Hill, F. W. Brinkley, and K. D. Lathrop, "TRIPLET: A Two-Dimensional, Multigroup, Triangular Mesh, Planar Geometry, Explicit Transport Code," Los Alamos Scientific Laboratory report LA-5428-MS (October 1973).
5. B. M. Carmichael, "Standard Interface Files and Procedures for Reactor Physics Codes, Version III," Los Alamos Scientific Laboratory report LA-5486-MS (February 1974).
6. W. W. Engle, Jr., "A User's Manual for ANISN, a One-Dimensional Discrete Ordinates Transport Code with Anisotropic Scattering," Union Carbide Corporation report K-1693 (March 1967).
7. "Guidelines for the Documentation and Recommended Programming Practices to Facilitate Interchange of Digital Computer Programs," American Nuclear Society Standard 10.3, ANSI N 413 (August 1973).
8. G. I. Bell, G. E. Hansen, and H. A. Sandmeier, "Multitable Treatments of Anisotropic Scattering in S_n Multigroup Transport Calculatons," Nucl. Sci. Eng. 28, 376 (1967).
9. B. G. Carlson and K. D. Lathrop, "Transport Theory - Method of Discrete Ordinates," Chapter III of Computing Methods in Reactor Physics, Gordon and Breach, NY (1968), p. 173.
10. Ibid, p. 185.
11. Ibid, p. 211.
12. K. D. Lathrop, "DTF-IV - A FORTRAN IV Program for Solving the Multigroup Transport Equation with Anisotropic Scattering," Los Alamos Scientific Laboratory report LA-3373 (1965).
13. Wm. H. Reed and K. D. Lathrop, "Truncation Error Analysis of Finite Difference Approximations to the Transport Equation," Nucl. Sci. Eng. 41, 237 (1970).
14. G. I. Bell and S. Glasstone, Nuclear Reactor Theory, Van Nostrand Reinhold, NY (1970), p. 260.

15. B. G. Carlson and G. I. Bell, "Solution of the Transport Equation by the S_n Method," Proceedings Second U.N. International Conference on Peaceful Uses of Atomic Energy 16, 535 (1958).

16. G. H. Golub and R. S. Varga, "Chebyshev Semi-iterative Methods, and Second-Order Richardson Iterative Methods, Part I," Numerische Mathematik 3, 147 (1961).

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

☆ US GOVERNMENT PRINTING OFFICE. 1975—677-182/61