201
4-15-77

LA-6735-M

Manual

UC-32

Issued: March 1977

Dr 904

**TRIDENT: A Two-Dimensional, Multigroup,
Triangular Mesh Discrete Ordinates,
Explicit Neutron Transport Code**

by

T. J. Seed
W. F. Miller, Jr.
F. W. Brinkley, Jr.

los alamos
scientific laboratory

of the University of California

LOS ALAMOS, NEW MEXICO 87545

An Affirmative Action/Equal Opportunity Employer

MASTER

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

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

This work was supported by the US Energy Research and Development Administration, Division of Reactor Development and Demonstration.

**Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161
Price: Printed Copy \$5.00 Microfiche \$3.00**

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research and Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

CONTENTS

ABSTRACT - - - - -	1
I. INTRODUCTION- - - - -	2
II. THEORY- - - - -	3
A. The Analytic Transport Equation- - - - -	4
1. Particular Forms of the Divergence Operator - - - - -	4
2. Spherical-Harmonics Expansion of Source Terms - - - - -	4
B. Energy and Angular Discretization- - - - -	6
1. Multigroup Equations- - - - -	6
2. Discrete-Ordinates Equations- - - - -	6
a. Rectangular (x-y) Geometry- - - - -	6
b. Finite-Cylindrical (r-z) Geometry - - - - -	7
C. Ray-Effect Fix-up Option - - - - -	8
1. The x-y Geometry Ray-Effect Fix-up ($S_N + P_{N-1}$) - - - - -	9
2. The r-z Geometry Ray-Effect Fix-up ($S_N + DH_{N-1}$) - - - - -	10
D. Discretization of Spatial Variables- - - - -	11
E. Solution Algorithms- - - - -	14
1. Boundary Conditions - - - - -	14
a. Vacuum Boundary Condition - - - - -	14
b. Reflective Boundary Condition - - - - -	14
c. Special Boundary Condition- - - - -	14
d. White Boundary Condition- - - - -	15
e. Source Boundary Condition - - - - -	15
2. Mesh Sweeps - - - - -	15
3. Adjoint Problems- - - - -	16
4. Iterative Processes - - - - -	16
5. Convergence-Acceleration Methods- - - - -	18
a. Inner-Iteration Rebalance - - - - -	18
b. Outer-Iteration Rebalance - - - - -	19
6. Convergence Tests - - - - -	19
III. A GUIDE TO USER APPLICATION - - - - -	20
A. Overall Program Flow - - - - -	20
B. Details of Program Options - - - - -	20
1. Cross Sections- - - - -	20
a. Input Formats- - - - -	20
b. Cross-Section Mixing- - - - -	22
c. Anisotropic Cross Sections- - - - -	22
d. Adjoint Cross Sections- - - - -	23
e. Cross-Section Checking- - - - -	23
2. Geometry and Boundary Condition Satisfaction- - - - -	23
a. Spatial Mesh- - - - -	23
b. Boundary Conditions - - - - -	24
3. Angular Quadrature Coefficient Specifications - - - - -	24
4. Source Options- - - - -	25

CONTENTS (cont)

5.	Source and Flux Input Options - - - - -	26
a.	Source Input Options- - - - -	26
b.	Flux Input Options- - - - -	26
6.	Flux Dumps and Restart Procedures - - - - -	27
7.	Rebalance Acceleration of Iterations- - - - -	28
8.	Eigenvalue Searches - - - - -	28
9.	Adjoint Computations- - - - -	29
10.	Edit Options- - - - -	30
a.	Zone Edit - - - - -	30
b.	Point Edit- - - - -	31
C.	Data-Input Rules - - - - -	31
D.	Description of Input Data- - - - -	31
1.	Job Title Cards - - - - -	32
2.	Input of Control Numbers- - - - -	32
3.	Problem-Dependent Data- - - - -	32
4.	Edit Input- - - - -	36
E.	Output Description for a Test Problem- - - - -	37
IV.	PROGRAMMING INFORMATION - - - - -	37
A.	Program Structure- - - - -	47
1.	Role and Function of Subprograms- - - - -	47
2.	Program Flow- - - - -	47
3.	Relation of Problem Variables and Program Mnemonics - - - - -	47
4.	Definition of Variables in Common Blocks- - - - -	47
5.	TRIDENT Storage Hierarchy - - - - -	47
6.	Machine-Dependent Subprograms - - - - -	66
a.	LCM/Random Disk and LCM/SCM Installation-Dependent Subroutines - - - - -	66
b.	Other Installation-Dependent Routines - - - - -	66
B.	Decimal Data File- - - - -	66
C.	Binary Data File - - - - -	66
D.	Hardware Requirements- - - - -	66
E.	Software Requirements- - - - -	66
1.	CDC Machines- - - - -	66
2.	TRIDENT for the IBM 360 - - - - -	67
F.	Programming Considerations - - - - -	68
1.	Storage Management- - - - -	68
a.	Variable Dimensioning - - - - -	68
b.	Allocation of LCM - - - - -	68
c.	Computation of Required SCM and LCM Storage - - - - -	68
d.	Computation of Required Random Disk Storage - - - - -	68
e.	Temporary Storage Requirements- - - - -	69
f.	Overstorage of Data in Core - - - - -	69
2.	Restart Tape Composition- - - - -	70
3.	Standard Interface Files- - - - -	70
REFERENCES	- - - - -	70
APPENDIX	SPATIAL FINITE ELEMENT EQUATIONS- - - - -	71

LIST OF FIGURES

1.	A typical regular triangular mesh - - - - -	2
2.	Coordinates in (x-y) geometry - - - - -	5
3.	Coordinates in (r-z) geometry - - - - -	5
4.	Built-in (EQ_6) quadrature directions- - - - -	7
5.	Sample input quadrature directions for left or right reflecting boundary- - - - -	7
6.	Typical up-pointing triangle- - - - -	12
7.	Typical down-pointing triangle- - - - -	12
8.	Use of the special boundary condition to represent a hexagonal cell - - -	15
9.	Typical order of solution on a band of triangles for direction Ω_m - - -	16
10.	Simplified logical flow diagram for TRIDENT - - - - -	21
11.	Example of x-or r-coordinate (small numbers) and triangle (large numbers) indexing - - - - -	24
12.	Example of a pictorial representation of the mesh by subroutine MAPPER- - - - -	24
13.	Variation of λ during a hypothetical eigenvalue search- - - - -	29
14.	Illustration of format read by LOAD - - - - -	32
15.	Amplified form of flow diagram in Fig. 10 showing subroutine names in overall program logic- - - - -	47

LIST OF TABLES

I.	Forms of $\nabla \cdot \Omega \psi$ - - - - -	5
II.	Fictitious-Source Angular Polynomials - - - - -	9
III.	Tabulation of Lagrange Polynomials- - - - -	12
IV.	Cross-Section Input Order with Los Alamos Format- - - - -	21
V.	Sample Mixture-Specification Parameter- - - - -	22
VI.	Ordering of Anisotropic-Distributed Source Components - - - - -	25
VII.	Restart Data in Card 1- - - - -	27
VIII.	Restart Data in Card 2- - - - -	28
IX.	Options for Special Read Formats- - - - -	31
X.	TRIDENT Input Cards - - - - -	32
XI.	Problem-Dependent Data- - - - -	34
XII.	Edit Input- - - - -	36
XIII.	Structure of the TRIDENT Program- - - - -	48
XIV.	Function of TRIDENT Subroutines - - - - -	49
XV.	Relation of Key Problem Variables to Program Mnemonics- - - - -	51
XVI.	Contents of Blank Common Blocks IA and RIA- - - - -	52
XVII.	Contents of Named Common Block FWBGN1 - - - - -	59
XVIII.	Contents of Named Common Block FWBGN2 - - - - -	61
XIX.	Contents of Named Common Block LOCAL- - - - -	61
XX.	Contents of Named Common Block UNITS- - - - -	62
XXI.	Contents of Named Common Block TSWEET - - - - -	62
XXII.	Contents of Named Common Block RSWEET - - - - -	62

TABLES (cont)

XXIII.	Contents of Named Common Block JSWEEP - - - - -	63
XXIV.	Contents of Named Common Block TBAL - - - - -	63
XXV.	Contents of Named Common Block FICSR- - - - -	63
XXVI.	Contents of Named Common Block INST - - - - -	63
XXVII.	Contents of Named Common Block SEEKIT - - - - -	64
XXVIII.	TRIDENT Bulk-Storage Scheme - - - - -	65
XXIX.	Decimal Data Files- - - - -	66
XXX.	SEEK Assigned Files - - - - -	67
XXXI.	Variables for SEEK Assigned Values- - - - -	67
XXXII.	LCM Storage Parameters- - - - -	69
A-I.	Difference Relationships, Cylindrical Geometry, r-Derivative Terms - - - -	72
A-II.	Geometric Expressions used in Tables A-I through A-VI - - - - -	73
A-III.	Difference Relationships, Cylindrical Geometry, z-Derivative Terms - - - -	73
A-IV.	Difference Relationships, Cylindrical Geometry, Angular Derivative Terms - - - - -	74
A-V.	Difference Relationships, Cylindrical Geometry, Collision Terms- - - - -	74
A-VI.	Difference Relationships, Cylindrical Geometry, Source Terms - - - - -	74
A-VII.	Difference Relationships, Rectangular Geometry, x-Derivative Terms - - - -	74
A-VIII.	Difference Relationships, Rectangular Geometry, y-Derivative Terms - - - -	75
A-IX.	Difference Relationships, Rectangular Geometry, Collision Terms- - - - -	75
A-X.	Difference Relationships, Rectangular Geometry, Source Terms - - - - -	75

TRIDENT:
A TWO-DIMENSIONAL, MULTIGROUP, TRIANGULAR MESH DISCRETE ORDINATES,
EXPLICIT NEUTRON TRANSPORT CODE

by

T. J. Seed, W. F. Miller, Jr.,
and F. W. Brinkley, Jr.

ABSTRACT

1. Program Identification: TRIDENT
2. Computer for Which Program is Designed: CDC-7600, IBM 360
3. Description of Function: TRIDENT solves the two-dimensional-multigroup-transport equations in rectangular (x-y) and cylindrical (r-z) geometries using a regular triangular mesh. Regular and adjoint, inhomogeneous and homogeneous (k_{eff} and eigenvalue searches) problems subject to vacuum, reflective, white, or source boundary conditions are solved. General anisotropic scattering is allowed and anisotropic-distributed sources are permitted.
4. Method of Solution: The discrete-ordinates approximation is used for the neutron directional variables. An option is included to append a fictitious source to the discrete-ordinates equations that is defined such that spherical-harmonics solutions (in x-y geometry) or spherical-harmonics-like solutions (in r-z geometry) are obtained. This option is useful for cases in which ray-effect distortions are severe. A spatial-finite-element method is used in which the angular flux is expressed as a linear polynomial in each triangle that is discontinuous at triangle boundaries. Both inner (within-group) and outer iteration cycles are accelerated by either whole-system or fine-mesh rebalance.
5. Restrictions: Variable dimensioning is used so that any combination of problem parameters leading to a container array less than MAXLEN can be accommodated. On CDC machines MAXLEN can be about 40 000 words of Small Core Memory (SCM) and Large Core Memory (LCM) is used for most group-dependent data. On IBM machines, TRIDENT executes in double precision (8 bytes per word) so that MAXLEN can be on the order of 100 000 or more. Although most problems can be fast core contained on the IBM machine and SCM and LCM contained on the CDC machine, an option exists to automatically overflow the storage of selected large arrays to random disk if necessary.
6. Running Time: A six group, S_2 , 1 700 triangle, k_{eff} calculation of an EBR-II core modeled in x-y geometry requires approximately 4.4 min of CDC-7600 time. This is approximately the same as the TRIPLET running time. Running times vary almost linearly with the total number of unknowns. When the ray-effect correction option is used, the running time is increased by a factor of approximately two, depending upon the problem.
7. Unusual Features of the Program: Provision is made for creation of standard interface output files for S_N constants, angle-integrated (scalar) fluxes, and angular fluxes. Standard interface input files for S_N constants, inhomogeneous sources, cross sections, and the scalar flux may be read. Subroutines DRED and DRIT perform information transfers between LCM and random disk. Data transfer between large and small core are performed by CRED and CRIT. Sequential binary operations are performed by SEEK, REED, and RITE. Flexible edit options as well as a dump and restart capability are provided.
8. Machine Requirements: A maximum of three interface units, open at once, and three system input/output units are required. One random-disk unit is required if the automatic-data-overflow option is used. A large bulk memory is assumed by the code such as the CDC-7600 LCM. This function may, however, be replaced by a portion of fast-core memory with IBM 360 machines or by disk, drum, or tape storage.
9. Related Programs: TRIDENT supersedes the TRIPLET program. With minor modifications, TRIPLET input packages may be used in TRIDENT. LASL now maintains two time-independent, two-dimensional, neutron-transport computer codes. TWOTRAN uses a rectangular mesh and is faster than TRIDENT. The latter code should be used only when the triangular-mesh option is needed (nonorthogonal material boundaries), when oscillations attributable to the diamond-difference scheme are present, or when the ray effect must be eliminated.
10. Material Available: Source deck, test problems, results of executed test problems, and this report are available from the Argonne Code Center.

I. INTRODUCTION

TRIDENT is a computer program for solving the two-dimensional neutron-transport equation in rectangular (x-y) and cylindrical (r-z) geometries using a triangular spatial mesh. The advantages of a triangular mesh in such geometries are several. Systems of physical interest are often of complicated, curved design. While the description of such systems with an orthogonal mesh is always possible, these shapes are often described to greater accuracy by a triangular mesh with many fewer mesh cells. A hexagonal lattice in x-y geometry is often of interest, and such a lattice can be described exactly by a triangular grid. Thus, non-rectangular region or system boundaries can be treated with triangles while preserving convexity, often with a large saving in number of mesh cells.

The increased flexibility of a triangular mesh is not without added cost. Description of the mesh itself is difficult, and users of the TRIDENT program must expect that preparation of the input data for this code will be more difficult than for the orthogonal grid codes. Complexities in the TRIDENT solution algorithms caused by the triangular mesh lead to execution times that are two to three times as long as those for the orthogonal mesh code, TWOTRAN,¹ on a per-mesh-cell basis (the codes are comparable in execution time on a mesh-point basis).

The regular triangular mesh used by TRIDENT is almost as flexible as a general triangular mesh but is much easier to describe. A regular triangular mesh is defined by the following properties:

- (1) adjacent triangles must share two common vertices,
- (2) all triangles must lie on horizontal bands extending across the system. Equivalently, all triangle vertices must lie on horizontal lines extending across the system, and
- (3) each interior vertex must be common to exactly six adjacent triangles.

Note that we do not require that triangles be equilateral or outer boundaries be rectangular. An example of a regular triangular mesh is illustrated in Fig. 1.

There are two reasons why TRIDENT uses a regular mesh. First, specification of a regular triangular mesh is much simpler than specification of a general triangular mesh. Only four pieces of data need be input to the code: the number of

triangles on each band, the band widths $(\Delta y)_j$ or $(\Delta z)_j$ (see Fig. 1), the x-or r-coordinates of the vertices along each horizontal line, and the orientation of the first triangle on each band. This orientation is specified by indicating whether the triangle points up or down. A triangle is said to be upward pointing if it has two vertices on the lower line and one on the upper line forming the band, and vice versa. A regular triangular mesh is determined uniquely by the above data.

The second reason a regular mesh is used by TRIDENT is related to our decision to use explicit methods for solving the discrete-ordinates equations. Explicit methods necessarily sweep the mesh in the direction of neutron travel so that there is a definite order in which the triangles must be solved. This order depends upon the direction of neutron flow across triangle boundaries and is not straightforward as for an orthogonal mesh. Thus the direction of flow across boundaries must be determined and decisions made as to how to progress through the mesh. These decisions are much more complicated and time-consuming for a general triangular mesh than for a regular triangular mesh. Since the order in which the mesh is swept differs with each discrete-ordinates direction, these decisions must be made repetitively in the innermost iteration loops of TRIDENT. If complicated, such decisions would be prohibitively expensive. We restrict to a regular mesh to simplify these decisions as much as possible while retaining most of the flexibility of a general mesh.

In each triangle a piecewise-linear representation for the angular flux in each direction is used. The angular flux is allowed to be discontinuous across triangle boundaries and is determined within

Fig. 1. A typical regular triangular mesh.

each triangle by requiring that the residual be orthogonal to the linear polynomials. This procedure has been found to yield an accurate and stable method.

In addition to its triangular mesh geometry for both x-y and r-z geometries, TRIDENT is unique in its option to eliminate or mitigate ray-effect² distortions that plague some discrete ordinates calculations. A fictitious source^{3,4} is appended to the within-group-discrete-ordinates equations of order N (S_N equations) that is defined to yield spherical-harmonics (P_{N-1}) solutions in x-y geometry and spherical-harmonics-like (DR_{N-1}) solutions in r-z geometry. The addition of the fictitious source, which depends on the angular flux, results in a degradation of the convergence properties of the inner-iteration process increasing the number of iterations by a factor of two or more depending upon the problem. An optional set of input parameters may be used to reduce this penalty in running time while mitigating but not completely eliminating the ray effect.

Since the triangular mesh and ray-effect option result in increased running time, it is clear that TRIDENT should be used for problems where one or both of these options are needed. Thus TRIDENT is complementary to, not a replacement for, the TWOTRAN-II code and should be used only when an orthogonal mesh is incapable of giving accurate answers efficiently, when oscillations attributable to the diamond-difference¹ scheme are present, and/or when a ray-effect-mitigation capability is needed. TRIDENT does, however, supersede TRIPLET,⁵ a triangular-mesh finite-element code operable only in x-y geometry. Most other features and options of the TRIDENT program are similar to those of TWOTRAN-II. TRIDENT was written by modifying TRIPLET, and the codes share the same overlay structure and have many common subroutines. The major features include:

- (1) direct or adjoint capability (the fictitious-source option operates only in the direct mode),
- (2) anisotropic scattering of any order,
- (3) built-in S_N constants (the EQ_N set⁶ is used in lieu of the built-in sets of TRIPLET and TWOTRAN-II),
- (4) the option to input quadrature sets,
- (5) vacuum, reflective, or source boundaries on all sides; a special boundary condi-

tion, useful for hexagonal cell calculations, is allowed on the right-hand boundary in x-y geometry; and a white boundary condition is allowed for all but the left-hand boundary for both geometries,

- (6) nonrectangular domains,
- (7) source, k_{eff} , alpha, or concentration search problems,
- (8) whole-system or fine-mesh rebalance convergence acceleration for inner and outer iterations,
- (9) optional print suppress for large input arrays,
- (10) space-dependent buckling height,
- (11) optional input of flux, source, S_N constants, and cross sections from standard interface files,⁷
- (12) optional output of scalar and angular fluxes and S_N constants in standard interface format,⁷
- (13) an option to mitigate or eliminate the ray effect,
- (14) flexible data management algorithms that allow automatic overflow to random disk for large problems, if necessary.

Because TRIDENT is similar to the TRIPLET code from which it was derived and which it supersedes, much of the TRIPLET manual⁵ applies directly to TRIDENT. To obtain a unified and self-contained presentation for the convenience of the reader of this report, we have reproduced many portions of the TRIPLET manual in what follows. The next section of this report contains the theoretical development of all the methods and approximations used in TRIDENT. Section III is a users guide and Sec. IV contains detailed programming information. The contents of this report follow the guidelines for documentation of digital computer programs⁸ 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 and the spherical-harmonics expansion of sources, are presented and discussed in Sec. II.A. The multigroup treatment of the energy variable, and the discrete-ordinates approximation for the angular variables are treated in Sec. II.B. Section II.C is devoted entirely to a discussion of the ray-effect-fix-up option. In Sec. II.D a discussion of the finite-element methods used

to solve the discrete-ordinates equations on a triangular mesh is provided. The solution of the linear algebraic system obtained by these approximations is discussed in Sec. II.E.

A. The Analytic Transport Equation

The time-independent, inhomogeneous transport equation is written

$$\begin{aligned} \nabla \cdot (\underline{\Omega}\psi) + \sigma_t \psi(\underline{r}, E, \underline{\Omega}) = \\ \iint dE' d\Omega' \psi(\underline{r}, E', \underline{\Omega}') \sigma_s(E' \rightarrow E, \underline{\Omega}' \cdot \underline{\Omega}) \\ + \chi(E) \iint dE' d\Omega' \psi(\underline{r}, E', \underline{\Omega}') v \sigma_f(E') / 4\pi \\ + Q(\underline{r}, E, \underline{\Omega}), \end{aligned} \quad (1)$$

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

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_{off} . 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 TRIDENT program will solve both types of problems as well as perform time-absorption (alpha) and concentration-search calculations.

1. Particular Forms of the Divergence Operator

The form of $\nabla \cdot \underline{\Omega}\psi$ for the two geometries created by TRIDENT is given in Table 1 in terms of the coordinate systems sketched in Figs. 2 and 3. Since an x-y geometry triangular mesh may be used to model all planar configurations, an (r- θ) option is not needed. In both of these geometries we choose ϕ to be the angle of rotation about the μ axis such that $d\Omega = d\mu d\phi$, and

$$\xi = (1 - \mu^2)^{1/2} \sin \phi,$$

$$\eta = (1 - \mu^2)^{1/2} \cos \phi.$$

In both geometries ψ is symmetric in ϕ , and only a hemisphere of angular directions need be considered. In rectangular geometry, the north pole of the hemisphere points in the $\hat{e}_z$ direction; in cylindrical geometry, the north pole of the hemisphere points in the $-\hat{e}_\theta$ direction.

2. Spherical-Harmonics Expansion of Source Terms

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

$$\sigma_s(E' \rightarrow E, \mu_0) = \sum_{\ell=0}^{\text{ISCT}} \frac{2\ell+1}{4\pi} P_\ell(\mu_0) \sigma_{s\ell}(E' \rightarrow E),$$

where $\mu_0 = \underline{\Omega} \cdot \underline{\Omega}' = \mu\mu' + (1 - \mu^2)^{1/2} (1 - \mu'^2)^{1/2} \times \cos(\phi - \phi')$. If this expansion is inserted in Eq. (1), and if the addition theorem is used to expand $P_\ell(\mu_0)$, we can write

$$\begin{aligned} \int d\Omega' \psi(\underline{r}, E', \underline{\Omega}') \sigma_s(E' \rightarrow E, \mu_0) \\ = \sum_{\ell=0}^{\text{ISCT}} \frac{2\ell+1}{2\pi} \sigma_{s\ell}(E' \rightarrow E) \\ \times \sum_{k=0}^{\ell} R_\ell^k(\mu, \phi) \int_{-1}^1 d\mu' \int_0^\pi d\phi' R_\ell^k(\mu', \phi') \psi. \end{aligned} \quad (2)$$

In deriving this expression, the ϕ symmetry of ψ is used, reducing the domain of Ω to a hemisphere and

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

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

eliminating expansion terms odd in ϕ . The spherical harmonics polynomials, R_ℓ^k , are defined by

$$R_\ell^k = \left[\frac{(2 - \delta_{k0})(\ell - k)!}{(\ell + k)!} \right]^{\frac{1}{2}} P_\ell^k(\mu) \cos k\phi, \quad (3)$$

where δ_{k0} is the Kronecker delta (equal to 1 when $k = 0$, and vanishing otherwise) and the P_ℓ^k are the associated Legendre polynomials. These functions are orthogonal; i.e.,

$$\int_{-1}^1 d\mu \int_0^\pi d\phi R_\ell^k(\mu, \phi) R_m^n(\mu, \phi) = \frac{2\pi}{2\ell + 1} \delta_{\ell m} \delta_{kn}. \quad (4)$$

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

$$\psi = \sum_{\ell=0}^{\infty} (2\ell + 1) \sum_{k=0}^{\ell} R_\ell^k \psi_\ell^k, \quad (5)$$

the expansion coefficients are given by

$$\psi_\ell^k = \int_{-1}^1 d\mu \int_0^\pi d\phi R_\ell^k \psi / 2\pi. \quad (6)$$

Fig. 2. Coordinates in (x-y) geometry.

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

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

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

$$\int_0^\infty dE' \sum_{\ell=0}^{\text{ISCT}} (2\ell + 1) \sigma_{s\ell}(E' \rightarrow E) \sum_{k=0}^{\ell} R_{\ell}^k(\mu, \phi) \psi_{\ell}^k + \chi(E) \int_0^\infty dE' \nu \sigma_f(E') \psi_0^0 + \sum_{\ell=0}^{\text{IQAN}} (2\ell + 1) \times \sum_{k=0}^{\ell} R_{\ell}^k Q_{\ell}^k. \quad (7)$$

As implied by the sum in this equation, we have assumed that the source Q is represented by IQAN+1 terms of an expansion like Eq. (5).

B. Energy and Angular Discretization

1. Multigroup Equations

The energy domain of interest is assumed to be partitioned into IGM intervals of width ΔE_g , $g = 1, 2, \dots, \text{IGM}$. By convention, increasing g represents decreasing energy. If we integrate Eq. (7) over ΔE_g we can write

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

$$\sum_{h=1}^{\text{IGM}} \sum_{\ell=0}^{\text{ISCT}} (2\ell + 1) \sigma_{s\ell h \rightarrow g} \sum_{k=0}^{\ell} R_{\ell}^k \psi_{\ell h}^k + \chi_g \sum_{h=1}^{\text{IGM}} \nu \sigma_{fh} \psi_{0h}^0 + \sum_{\ell=0}^{\text{IQAN}} (2\ell + 1) \sum_{k=0}^{\ell} \frac{k}{\ell} \psi_{\ell g}^k. \quad (8)$$

Here, the flux for group g ,

$$\psi_g = \int_{\Delta E_g} \psi dE, \quad (9)$$

is no longer a distribution in energy, but is the total number of particles in the energy interval. For this reason, when group structures are changed, the effect on results must be evaluated by comparing $\psi_g / \Delta E_g$. Because of Eq. (9) energy integrals in TRIDENT are evaluated by simple sums.

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

$$\sigma_{tg} = \int_{\Delta E_g} \sigma_t \psi dE / \int_{\Delta E_g} \psi dE, \quad (10)$$

but, of course, ψ is not known and must be approximated by some means. If in Eq. (10) the angular dependence of ψ is nonseparable, then σ_{tg} will depend on angle. No provision for such dependence is made in TRIDENT. Recipes for taking this dependence into account, as well as for improving the averages $\sigma_{s\ell h \rightarrow g}$ when scattering is severely anisotropic, are given by Bell, Hansen, and Sandmeier.⁹

2. Discrete-Ordinates Equations

The three terms on the right-hand side of Eq. (8) represent sources of neutrons due to scattering, fission, and inhomogeneous sources, respectively. All coupling between the IGM multigroup equation 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 latest values of ψ_g . For details of the iterative processes in TRIDENT see Sec. II.E.4. Dropping the group subscript g we have

$$\nabla \cdot \underline{\Omega}\psi + \sigma_t \psi = S. \quad (11)$$

We will now be concerned with the discretization of Eq. (11) with respect to the variables μ and η .

a. Rectangular (x-y) Geometry

We assume a set of MM quadrature points (μ_m, η_m) has been selected on the unit disk ($\mu^2 + \eta^2 \leq 1$), along with a set of quadrature weights (w_m)

normalized so that $\sum_{m=1}^{\text{MM}} w_m = 1$ arranged, for example,

as depicted in Figs. 4 and 5. The user of TRIDENT may input this quadrature to the code or may use the built-in (EQ_N) sets if desired. As discussed in Sec. II.C however, the built-in sets must be used in conjunction with the ray-effect-fix-up option. Although the built-in quadrature sets are symmetric under 90° rotations, for vacuum, source, and white boundaries in x-y geometry there are no symmetry requirements on input quadratures

and MM may be any positive integer greater than or equal to four. Unlike TRIPLET, TRIDENT requires a minimum of one direction for each quadrant. If the built-in set is used, MM will equal one of a selected set of values for which quadrature values are available ($MM = \frac{N(N+2)}{2}$, where N is the discrete-ordinates (S_N) order restricted to be an even integer in the range $2 \leq N \leq 16$). If reflecting or special boundary conditions are used in x-y geometry, then input quadrature sets must meet certain symmetry requirements:

- Top or bottom reflecting boundary condition - the directions must possess symmetry about the μ axis (see for example Fig. 4).
- Left or right reflecting boundary condition - the directions must possess symmetry about the η axis (see for example Figs. 4 and 5).
- Special boundary condition - the directions must possess symmetry about both the μ and η axis. For hexagonal cell calculations using the special boundary condition, the left, top, and bottom sides must be reflecting, while the right boundary is treated using the special boundary condition.

See Sec. II.E.1 for a further discussion of TRIDENT boundary condition options.

Once the quadrature set is selected the (angular) cell-centered flux for direction m is then assumed to be given by

$$\psi_m(x, y) \approx \psi(x, y, \mu_m, \eta_m),$$

and the angular-flux moments of Eq. (6) are approximated by

$$\psi_{\ell}^k(x, y) \approx \sum_{m=1}^{MM} w_m R_{\ell}^k(\mu_m, \phi_m) \psi_m(x, y), \quad (12)$$

where ϕ_m is given by

$$\begin{aligned} \phi_m &= \tan^{-1} (1 - \mu_m^2 - \eta_m^2)^{1/2} / \eta_m, & \eta_m > 0, \\ &= \tan^{-1} (1 - \mu_m^2 - \eta_m^2)^{1/2} / \eta_m + \pi, & \eta_m < 0. \end{aligned} \quad (13)$$

The set of discrete-ordinates equations approximating Eq. (11) are then

$$\mu_m \frac{\partial \psi_m(x, y)}{\partial x} + \eta_m \frac{\partial \psi_m(x, y)}{\partial y} + \sigma_t \psi_m(x, y) = S_m(x, y) \quad m = 1, 2, \dots, MM, \quad (14)$$

where the angular source S_m is computed from

$$\begin{aligned} S_m &= \sum_{h=1}^{IGM} \sum_{\ell=0}^{ISCT} (2\ell + 1) \sigma_{s\ell h} \tau_g \sum_{k=0}^{\ell} R_{\ell}^k(\mu_m, \phi_m) \psi_{\ell h}^k \\ &+ \chi_g \sum_{h=1}^{IGM} w_{fh} \psi_{oh}^0 + \sum_{\ell=0}^{IQAN} (2\ell + 1) \sum_{k=0}^{\ell} R_{\ell}^k(\mu_m, \phi_m) Q_{\ell g}^k. \end{aligned} \quad (15)$$

b. Finite-Cylindrical (r-z) Geometry

The procedures to be followed to derive the r-z geometry discrete-ordinates equations are identical to those outlined above except for complications introduced by the angular derivative term

Fig. 4. Built-in (EQ_6) quadrature directions.

Fig. 5. Sample input quadrature directions for left or right reflecting boundary.

in the streaming operator (see Table I). We again select a set of MM quadrature points and normalized weights. In addition to the restrictions imposed on input quadrature sets for x-y geometry, TRIDENT further requires that for r-z geometry the quadrature set be symmetric with respect to the η axis regardless of the boundary condition. The reason for this restriction is explained below. Note that the special boundary condition does not apply to and cannot be used in r-z geometry.

The angular-cell-centered angular flux is assumed to be given by

$$\psi_m(r, z) \approx \psi(r, z, \mu_m, \phi_m), \quad (16)$$

and the directional group source and angular-flux moments approximated by Eqs. (15) and (12), respectively. On a given η level, we denote the angular-cell-edge fluxes by $\psi_{m-\frac{1}{2}}(r, z)$ and $\psi_{m+\frac{1}{2}}(r, z)$. We then write the discrete-ordinates approximation to Eq. (11) as

$$\begin{aligned} \mu_m \frac{\partial(r\psi_m)}{r\partial r} + \frac{\alpha_{m+\frac{1}{2}} \psi_{m+\frac{1}{2}}(r, z) - \alpha_{m-\frac{1}{2}} \psi_{m-\frac{1}{2}}(r, z)}{rw_m} \\ + \eta_m \frac{\partial\psi_m(r, z)}{\partial z} + \sigma\psi_m(r, z) = S_m(r, z). \end{aligned} \quad (17)$$

$m = 1, 2, \dots, MM.$

In order that Eq. (17) exhibit the correct behavior for the case of divergenceless flow, it can be shown¹⁰ that

$$\alpha_{m+\frac{1}{2}} - \alpha_{m-\frac{1}{2}} = -w_m \mu_m, \quad (18)$$

with the requirement from neutron conservation that the first ($\alpha_{\frac{1}{2}}$) and last ($\alpha_{M+\frac{1}{2}}$) coefficients on each η level vanish. Note that starting with $\alpha_{\frac{1}{2}} = 0$ and using the recursion relationship of Eq. (18), the $\alpha_{M+\frac{1}{2}}$ will only be zero, in general, if the quadrature set is symmetric with respect to the η axis. Thus, TRIDENT requires quadrature symmetry about this axis in r-z geometry.

Associated with Eq. (17) must be a difference relationship for the angular-mesh cell since there are two unknown angular-flux values, ψ_m , and $\psi_{m+\frac{1}{2}}$. The $\psi_{m-\frac{1}{2}}$ edge flux is assumed known from the previous angular-mesh-cell computation and imposing continuity at angular-mesh-cell boundaries.

The standard diamond-difference¹⁰ assumption (in angle only) is made to relate the edge and cell-centered fluxes,

$$\psi_m(r, z) = \frac{1}{2} [\psi_{m-\frac{1}{2}}(r, z) + \psi_{m+\frac{1}{2}}(r, z)]. \quad (19)$$

Thus, in each angular-mesh cell, we need only solve the transport equation for one function, $\psi_m(r, z)$. To initiate the computation in the first angular-mesh cell for each η level, TRIDENT assumes the step scheme,¹⁰

$$\psi_m = \psi_{m-\frac{1}{2}} = \psi_{m+\frac{1}{2}}.$$

The built-in (EQ_N) quadrature set, as used in TRIDENT, does not satisfy a requirement tacitly assumed above. By a simple change of variables, it can be shown that the ω derivative in the streaming operator (see Table I) can be expressed as a derivative with respect to μ alone. That is, an η derivative does not appear. The difference relationship then should involve points of equal η values. For $N \geq 6$, a few points do not exactly fall upon selected η levels but are very close. In Fig. 4, for example, the η coordinate of points 16, 18, 22, and 24 is 0.2561429, while the η coordinate of points 17 and 23 is 0.2663443. It has been shown⁴ that the r-z geometry discrete-ordinates equations can be derived assuming the angular flux is piecewise constant in η . Thus, since points 16, 17, 18, 22, 23, and 24 can be associated with one η band, there is no inconsistency. This is true for all EQ_N orders.

C. Ray-Effect-Fixup Option

The discrete-ordinates approximation results in distorted flux distributions for a certain class of problems. The physical problems in question are those with more-or-less physically-isolated sources and with low scattering ratios. The flux plots have so-called ray effects² which are anomalous ripples that may be confused with spatial-flux variation due to physical effects. The distortion is due to the fact that a finite number of directions is used resulting, for some problems, in subregions which are not penetrated by rays originating at sources. Increasing the number of angular directions does mitigate the effect since more of the problem area is penetrated by rays originating in

scattering, fission, or distributed sources. The ripples, however, are often remarkably persistent, even when a large number of directions is used.¹¹

In order to eliminate the ray effect in problems where it is likely to appear, TRIDENT includes the option to append a fictitious source to the discrete-ordinates equations defined so that spherical-harmonics (x-y geometry) or spherical-harmonics-like (r-z geometry) solutions result. The spherical-harmonics equations are rotationally invariant and do not demonstrate the ray effect.¹¹

1. The x-y Geometry Ray-Effect Fix-up ($S_N \rightarrow P_{N-1}$)

We append to the discrete-ordinates equations [Eq. (14)] the Miller-Reed^{3,4} fictitious source

$$\mu_m \frac{\partial \psi_m}{\partial x} + \eta_m \frac{\partial \psi_m}{\partial y} + \sigma_t \psi_m = S_m + \beta FS_m$$

$$m = 1, 2, \dots, MM, \quad (20)$$

where

$$FS_m = \sum_{k=1}^{N/2} A_{mk} \frac{\partial \psi_N^{2k-1}}{\partial x} + \sum_{k=1}^{N/2} B_{mk} \frac{\partial \psi_N^{2k-1}}{\partial y} + \sum_{k=0,2,\dots}^{N-2} C_{mk} \frac{\partial \psi_{N-1}^k}{\partial x} + \sum_{k=0,2,\dots}^{N-2} D_{mk} \frac{\partial \psi_{N-1}^k}{\partial y}. \quad (21)$$

For simplicity, we have again suppressed the group subscript. The A_{mk} , B_{mk} , C_{mk} , and D_{mk} are defined in Table II and are stored in the array FS in TRIDENT. We use the inner product notation

$$(c, b) = \int_{-1}^1 d\mu \int_0^\pi d\phi \, c(\underline{\Omega}) b(\underline{\Omega}),$$

and use the parameters, a_{mp}^n , defined by the following procedure.³ We select the MM normalized spherical-harmonic polynomials

$$\left\{ Y_\ell^k \right\}, \quad \ell = 0, 1, 2, \dots, N-1$$

$$k = 0, 1, 2, \dots, \ell$$

and

TABLE II
FICTITIOUS SOURCE ANGULAR POLYNOMIALS

$$A_{mk} = \sum_{p=0}^{N-1} \sum_{n=0}^p Y_p^n(\mu_m, \eta_m) \sum_{m'=1}^{MM} a_{m',p}^n \mu_m Y_N^{2k-1}(\mu_m, \eta_m)$$

$$B_{mk} = \sum_{p=0}^{N-1} \sum_{n=0}^p Y_p^n(\mu_m, \eta_m) \sum_{m'=1}^{MM} a_{m',p}^n \eta_m Y_N^{2k-1}(\mu_m, \eta_m)$$

$$C_{mk} = \sum_{p=0}^{N-1} \sum_{n=0}^p Y_p^n(\mu_m, \eta_m) \sum_{m'=1}^{MM} a_{m',p}^n Y_N^k(\mu_m, \eta_m) (Y_N^k, \mu Y_{N-1}^k)$$

$$D_{mk} = \sum_{p=0}^{N-1} \sum_{n=0}^p Y_p^n(\mu_m, \eta_m) \sum_{m'=1}^{MM} a_{m',p}^n$$

$$\times \left[Y_N^{k+2}(\mu_m, \eta_m) (Y_N^{k+2}, \eta Y_{N-1}^{k+1}) + Y_N^k(\mu_m, \eta_m) (Y_N^k, \eta Y_{N-1}^{k+1}) \right]$$

$$\left\{ Y_N^{2k-1} \right\}, \quad k = 1, 2, \dots, N/2.$$

These polynomials are related to the R_ℓ^k of Eq. (3) by

$$Y_\ell^k = \left(\frac{2\ell+1}{2\pi} \right)^{1/2} R_\ell^k.$$

We arrange these polynomials in a single subscripted array, $\vec{Y}$. We denote MM unknown parameters, for each direction m, as

$$\left\{ a_{m\ell}^k \right\}, \quad \ell = 0, 1, 2, \dots, N-1$$

$$k = 0, 1, 2, \dots, \ell$$

and

$$\left\{ a_{mN}^{2k-1} \right\}, \quad k = 1, 2, \dots, N/2.$$

These are arranged in MM subscripted arrays of length MM that we denote, $\vec{a}_m$. The coefficients, $a_{m\ell}^k$, are determined by solving the following linear algebraic system of equations

$$\vec{a}_m^T \vec{Y}(\mu_m, \phi_m) = \delta_{m'm} \quad m = 1, 2, \dots, MM, \quad m' = 1, 2, \dots, MM. \quad (22)$$

It can be shown³ that with this definition

$$\psi_\ell^k = \sum_{m=1}^{MM} a_{m\ell}^k \psi_m. \quad (23)$$

When generating the fictitious source, all flux moments are calculated using Eq. (23). Flux moments for the scattering and fission source and inhomogeneous-source moments are always calculated using Eq. (12). The β of Eq. (20) is a group-dependent input parameter restricted to $0 \leq \beta_g \leq 1$ for $g = 1, 2, \dots, IGM$. For $\beta = 1$, the Miller-Reed fictitious source is defined such that Eq. (20) yields spherical-harmonics solutions of order $N - 1$. We refer to this then as a $S_N \rightarrow P_{N-1}$ method. Note that with $\beta = 0$, solutions to the standard discrete-ordinates equations result. In order to describe the case of $0 < \beta < 1$, we use the following heuristic argument.

Let A_1 be the discrete ordinates operator and A_2 the spherical harmonics operator. Then Eq. (20) may be written as⁴

$$A_1 \vec{\psi} = \vec{S} + \beta(A_1 - A_2) \vec{\psi}, \quad 0 \leq \beta \leq 1 \quad (24)$$

where the elements of $\vec{\psi}$ and $\vec{S}$ are respectively the angular flux and directional-source values for each quadrature point. It is clear by definition of the fictitious source that

$$FS_m = [(A_1 - A_2) \vec{\psi}]_m \quad m = 1, 2, 3, \dots, MM,$$

where by $[]_m$ we mean the m^{th} component of the vector. Rearranging Eq. (24), we obtain

$$(1 - \beta) A_1 \vec{\psi} + \beta A_2 \vec{\psi} = \vec{S} \quad 0 \leq \beta \leq 1.$$

It is clear that, for any β , the group flux is the solution to a linear combination of discrete-ordinates and spherical-harmonics equations.

As will be pointed out several times, the addition of the fictitious source to the discrete-ordinates equations eliminates the ray effect but results in two significant penalties. First, Eq. (24) is solved iteratively and when the ray effect is severe (the fictitious source is large) the iteration is slow to converge. By selecting a $\beta < 1$ (e.g., $\beta = .9$), the ray effect may be reduced and not completely eliminated with less of a run-time penalty.^{3,4} Second, the spherical-harmonics solutions are less accurate than discrete-ordinates solutions in terms of spatial integral parameters (e.g., total absorption). Thus, if one desires ray-effect mitigation with small loss in integral-parameter accuracy, $\beta < 1$ may be used. Note further that since β is group-dependent the user can mitigate the ray effect only in problem energy groups and then convergence penalties are only paid for those groups for which $\beta > 0$. Although for most problems, the fictitious source is small and does not significantly affect convergence, the user should use the fictitious-source option only when necessary due to the large computer storage required.

2. The r-z Geometry Ray-Effect Fix-up ($S_N \rightarrow DH_{N-1}$)

In the case of cylindrical geometry, we append to the discrete-ordinates equations [Eq. (17)] the Modified r-z Miller-Reed fictitious source^{4,12}

$$\begin{aligned} \mu_m \frac{\partial(r\psi_m)}{r\partial r} + \frac{\alpha_{m+\frac{1}{2}} \psi_{m+\frac{1}{2}} - \alpha_{m-\frac{1}{2}} \psi_{m-\frac{1}{2}}}{rw_m} \\ + \eta_m \frac{\partial\psi_m}{\partial z} + \sigma\psi_m = S_m + \beta MFS_m \end{aligned} \quad m = 1, 2, \dots, MM, \quad (25)$$

where

$$MFS_m = \sum_{k=1}^{N/2} A_{mk} \frac{\partial(r\psi_N^{2k-1})}{r\partial r} + \sum_{k=1}^{N/2} B_{mk} \frac{\partial\psi_N^{2k-1}}{\partial z} + \sum_{k=0,2,\dots}^{N-2} C_{mk} \frac{\partial(r\psi_{N-1}^k)}{r\partial r} + \sum_{k=0,2,\dots}^{N-2} D_{mk} \frac{\partial\psi_{N-1}^{k+1}}{\partial z}. \quad (26)$$

The A_{mk} , B_{mk} , C_{mk} , and D_{mk} are defined in Table II. Note that this fictitious source has a form identical to the x-y geometry fictitious source [Eq. (21)]. However, due to the angular derivative term in the streaming operator, appending this source does not result in spherical-harmonics solutions. The solution is spherical-harmonics-like (DH_{N-1}) and is virtually ray-effect free.^{4,12} We refer to this as a

the diamond-difference scheme, are explicit in nature.

Our task is to develop a discrete (in space) approximation to Eqs. (20) and (25) on a triangular mesh. In the ensuing analysis we will consider the r-z geometry case. The x-y geometry procedures are completely analogous. Consider the discrete-ordinates equations, and again suppress the group subscript

$$\mu_m \frac{\partial(r\psi_m)}{r\partial r} + \frac{\alpha_{m+\frac{1}{2}}\psi_{m+\frac{1}{2}} - \alpha_{m-\frac{1}{2}}\psi_{m-\frac{1}{2}}}{rw_m} + \eta_m \frac{\partial\psi_m}{\partial z} + \sigma\psi_m = \bar{S}_m \quad m = 1, 2, 3, \dots, MM \quad (27)$$

$S_N \rightarrow DH_{N-1}$ method. The use of this source also results in storage and computational-time penalties. The β parameters, therefore, are also applied in the r-z geometry case.

D. Discretization of Spatial Variables

The approximations that have been made so far in this report are independent of the form of the spatial mesh used by a transport code. These approximations are in fact identical to those used in the orthogonal grid code TWOTRAN-II with the exception of the ray-effect-fix-up option. In this section, however, we will depart from the standard diamond-difference scheme used in that code and will develop a set of finite-element methods for spatial discretization of the discrete-ordinates equations on a triangular mesh. In x-y geometry, the basic discretization method¹³ is identical to that used in TRIPLET although the implementation of the scheme in TRIDENT is somewhat different.

Difference schemes for the transport equation fall into two broad categories, which we will refer to as 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. An explicit method, on the other hand, sweeps once through the mesh, solving for the unknowns in the direction in which neutrons are streaming. In the latter case, the matrix to be inverted is triangular, or at least block triangular. The finite-element methods developed below, like

where

$$\bar{S}_m = S_m + \beta MFS_m.$$

The finite-element method used in TRIDENT is derived using a weight-and-integrate technique. The trial space consists of functions that are piecewise linear and discontinuous across triangle boundaries. Note that although TRIPLET allows polynomials of any input order, TRIDENT is restricted to the linear case. If $\psi_{mi}(r, z)$ is our approximation for the exact solution $\psi_m(r, z)$ within the i 'th triangle (see Figs. 6 and 7), then we assume that

$$\psi_{mi}(r, z) \approx \sum_{k=1}^3 \psi_{mi}^k L_i^k(r, z). \quad (28)$$

The polynomial $L_i^k(r, z)$ is the unique linear polynomial that is unity at the point (r_k, z_k) and is zero at the other two points on the i 'th triangle. The $L_i^k(r, z)$ are referred to as Lagrange polynomials and are given in Table III. Choosing the points (r_k, z_k) as the triangle vertices assures that the Lagrange polynomials exist, are linearly independent, and form a basis for the space of polynomials of order less than or equal one.¹³ The unknown coefficients ψ_{mi}^k can be interpreted as the values of the approximate solution $\psi_{mi}(r, z)$ at the points (r_k, z_k) in the i 'th triangle. It is this physical interpretation for ψ_{mi}^k which leads us to the Lagrange representation for $\psi_{mi}(r, z)$. To complete the

TABLE III
TABULATION OF LAGRANGE POLYNOMIALS

$$L_1^{(1)}(r, z) = \frac{(r - r_2)(z_3 - z_2) - (r_3 - r_2)(z - z_2)}{(r_3 - r_2)(z_2 - z_1) - (r_2 - r_1)(z_3 - z_2)}$$

$$L_1^{(2)}(r, z) = \frac{(r - r_3)(z_1 - z_3) - (r_1 - r_3)(z - z_3)}{(r_1 - r_3)(z_3 - z_2) - (r_3 - r_2)(z_1 - z_3)}$$

$$L_1^{(3)}(r, z) = \frac{(r - r_1)(z_2 - z_1) - (r_2 - r_1)(z - z_1)}{(r_2 - r_1)(z_1 - z_3) - (r_1 - r_3)(z_2 - z_1)}$$

Note that for the regular triangular mesh of TRIDENT $z_2 = z_1$.

Fig. 6. Typical up-pointing triangle.

specifications of our trial space, we must assign a unique value to the approximate flux on the triangle boundaries. It is essential to the following analysis that we assume that the flux on a triangle boundary is the limit of the flux as one approaches the boundary in the direction in which neutrons are streaming.

With the above definition of the trial space in which we look for an approximate solution, there are $3I$ unknowns to be determined for each direction m and group g , where I is the total number of triangles. We must therefore derive a set of $3I$ equations for these unknowns. Because we are only

interested in explicit methods, we are required to derive a set of three equations for the three unknowns on a given triangle, assuming that the flux on incoming boundaries is known from boundary conditions or from prior calculation in adjoining cells. An incoming boundary is a triangle boundary across which the neutron flow is directed into the cell. Of course, an incoming boundary for one cell is an outgoing boundary for an adjacent cell, and the definition of an incoming boundary depends upon the direction (μ_m, η_m) under consideration. There are two cases that must be considered: one or two faces may be visible when looking along the direction Ω_m determined by μ_m and η_m . Hence there may be one or two incoming boundaries. In Fig. 6, the arrow labeled Ω_m depicts the case of one incoming boundary while Fig. 7 depicts two incoming boundaries.

The form of Lagrange polynomials, $L_1^k(r, z)$, of Eq. (28) depends upon the origin of the coordinate system as well as the shape of the triangle. For this reason, the authors of TRIPLET found it convenient to map every triangle to a standard form when solving for the unknown fluxes ψ_{m1}^k . Although that approach is advantageous for x-y geometry, the geometry to which TRIPLET was restricted, such mapping proves to be exceedingly cumbersome in r-z geometry. Accordingly, in TRIDENT each triangle is solved, for both x-y and r-z geometries, without such mapping procedures.

To derive a set of three equations for our three unknowns, it is first necessary to define a so-called residual. We will derive a general expression for the residual holding for both up- and down-pointing

Fig. 7. Typical down-pointing triangle.

triangles and for the cases of both one and two incoming boundaries. We use Fig. 6 to derive our equation for the residual. We begin by defining the parameters

$$\gamma_i = \begin{cases} 1.0 & \text{flux incoming on side } i \\ 0.0 & \text{flux outgoing on side } i \end{cases} \\ i = 1, 2, 3.$$

We multiply Eq. (27) by r in anticipation of integrating over a volume element in r - z geometry. We then insert the approximate flux, $\psi_{mi}(r, z)$, into Eq. (27) to obtain an expression for the residual for triangle i and direction m

$$\begin{aligned} R_{mi}(r, z) = & \mu_m \frac{\partial [r\psi_{mi}(r, z)]}{\partial r} + \mu_m \gamma_2 \tilde{r}_2 [\psi_{mB1}(\tilde{r}_2, \tilde{z}_2) - \psi_{mi}(\tilde{r}_2, \tilde{z}_2)] \delta(r - \tilde{r}_2) \\ & + \mu_m \gamma_3 \tilde{r}_3 [\psi_{mi}(\tilde{r}_3, \tilde{z}_3) - \psi_{mB1}(\tilde{r}_3, \tilde{z}_3)] \delta(r - \tilde{r}_3) + \frac{\alpha_{m+1/2} \psi_{m+1/2,1}(r, z) - \alpha_{m-1/2} \psi_{m-1/2,1}(r, z)}{w_m} \\ & + r \eta_m \frac{\partial \psi_{mi}(r, z)}{\partial z} + \eta_m \gamma_1 \tilde{r}_1 [\psi_{mi}(\tilde{r}_1, \tilde{z}_1) - \psi_{mB1}(\tilde{r}_1, \tilde{z}_1)] \delta(z - \tilde{z}_1) \\ & + \eta_m \gamma_2 \tilde{r}_2 [\psi_{mB1}(\tilde{r}_2, \tilde{z}_2) - \psi_{mi}(\tilde{r}_2, \tilde{z}_2)] \delta(z - \tilde{z}_2) + \eta_m \gamma_3 \tilde{r}_3 [\psi_{mB1}(\tilde{r}_3, \tilde{z}_3) - \psi_{mi}(\tilde{r}_3, \tilde{z}_3)] \\ & \times \delta(z - \tilde{z}_3) + r \sigma \psi_{mi}(r, z) - r \bar{S}_m(r, z). \end{aligned} \quad (29)$$

In Eq. (29) the Dirac Delta functions, δ , arise from differentiation of the discontinuous function. We have introduced in Eq. (29), the equations (in r and z) for sides 1, 2, and 3. These are

$$\tilde{r}_j: r = r_j + \frac{\Delta r_j (z - z_j)}{\Delta z_j}, \quad j = 2, 3,$$

and

$$\tilde{z}_j: z = z_j + \frac{\Delta z_j (r - r_j)}{\Delta r_j}, \quad j = 1, 2, 3,$$

with $\Delta z_1 = z_2 - z_1$, $\Delta z_2 = z_3 - z_2$, $\Delta z_3 = z_1 - z_3$,
 $\Delta r_1 = r_2 - r_1$, $\Delta r_2 = r_3 - r_2$, and $\Delta r_3 = r_1 - r_3$.
Note that since the mesh is restricted to be regular,

$$\Delta z_1 = 0, \quad \Delta z_2 = \Delta z, \quad \text{and} \quad \Delta z_3 = -\Delta z.$$

To express the flux discontinuities, we have introduced the notation $\psi_{mB1}(\tilde{r}_k, \tilde{z}_k)$ which we define as the flux on the boundary along side k of triangle i . For simplicity in Eq. (29), we have not explicitly included the expansion of the source term. In fact, the group source, $S_m(r, z)$ is expanded in the Lagrange polynomials analogous to the procedure of Eq. (28), while the fictitious source, $MFS_m(r, z)$, defined in Eq. (26), is treated in a manner identical to the flux derivative terms of Eq. (29).

We observe that, in general, the residual R cannot identically equal zero for any choice of ψ_{mi} , since the exact solution is rarely given by a low-order polynomial. We introduce a set of three linearly independent weight functions $W_j(r, z)$, $j = 1, 2, 3$, and insist that

$$\{W_j, R\} = 0, \quad j = 1, 2, 3, \quad (30)$$

where the symbol $\{ \}$ means integration over the triangle volume. Equation (30) represents a three-by-three linear algebraic system of equations for the unknowns ψ_{mi}^k . With a proper choice of weight functions, this system is nonsingular and can be solved by any of a variety of methods appropriate for small linear systems. In TRIDENT, the weight functions

W_j are chosen to be the polynomials 1, $(r - \bar{r})$ and $(z - \bar{z})$ [1, $(x - \bar{x})$ and $(y - \bar{y})$] in rectangular geometry where the bar denotes the triangle centroid [e.g., $\bar{r} = 1/3(r_1 + r_2 + r_3)$]. These are linearly independent and span the space of polynomials of order less than or equal to 1. Equation (30) is solved by analytical inversion and substitution of the coordinate points, sources, and cross sections in the resulting expression. In the Appendix, we carry the algebra through to obtain the coded expressions for both x-y and r-z geometries. The general expressions hold for both up- and down-pointing triangles.

Numerical experiments have shown that the above-described finite-element method is capable of yielding accurate solutions to the discrete-ordinates equations for both x-y and r-z geometries. TRIPLET, however, included a continuous-finite-element option since for that case the flux for I in lieu of 3I mesh points need be stored. Since the accuracy and running time of TRIDENT are comparable with TWOTRAN on a mesh-point basis, storage requirements are not much different to obtain a given degree of error. Accordingly, the continuous-finite-element option is not included in TRIDENT. To take maximum advantage of fast-bulk memory, however, an automatic-disk-overflow-storage package which allows larger problem configurations to be stored is included in TRIDENT. This feature is described in Sec. IV.A.

Negative-flux values can be produced with the above-described finite-element method. They are encountered, however, much less frequently than with the diamond method. For this reason, there is no negative-flux fix-up¹ in TRIDENT, and if the user is unhappy with such fluxes he must use a finer mesh.

E. Solution Algorithms

1. Boundary Conditions

The equations and methods of Sec. II.D determine the angular flux within a triangle provided the incoming flux on triangle boundaries is known. This incoming flux may be known from prior calculation in adjacent cells. If an incoming-triangle boundary is a system boundary, then the incoming flux must be determined from the appropriate boundary condition. In TRIDENT, one of the boundary conditions described below must be applied to the top, bottom, and each of the two sides of a

system. The top and bottom boundaries are defined to be those parts of the system perimeter coinciding with the top and bottom horizontal lines (node lines), respectively. Note that the top and bottom boundaries are always flat, but side boundaries may approximate some curved surface. It is possible for the top or bottom boundary or both to fail to exist. This happens, for example, if there is a single up-pointing triangle on the top band. In these cases the boundary conditions are meaningless and are not used.

a. Vacuum Boundary Condition

The value of the angular flux on the boundary is set equal to zero for all incoming directions. Unlike some two-dimensional transport codes, (e.g., TWOTRAN), TRIDENT allows a vacuum boundary condition on the left side in r-z geometry. This feature is useful for modeling some toroidal configurations.

b. Reflective Boundary Condition

Reflective boundary conditions may be applied to each of the four sides in any combination. A reflective boundary must be parallel with one of the coordinate axes, and the quadrature must possess certain symmetries (see Sec. II.B.2). If either of the side boundaries is reflective, the quadrature must be symmetric about the line $\mu = 0$, whereas if either the top or bottom boundary is reflective, the quadrature must be symmetric about $\eta = 0$. These requirements are necessary to guarantee that all reflections of directions are contained in the quadrature set.

For a reflective boundary condition, the value of the incoming flux on the boundary is set equal to the value of the outgoing flux at the same position in the direction corresponding to specular reflection. At the right-hand boundary, for example, we set

$$\psi_{\text{incoming}}(-\mu, \eta) = \psi_{\text{outgoing}}(\mu, \eta),$$

$$\mu > 0.$$

c. Special Boundary Condition

A special boundary condition,⁵ useful for hexagonal cell calculations, is provided in TRIDENT. This boundary condition can be used on

the right-hand boundary, in x-y geometry only and must be used in conjunction with reflective boundary conditions on the other three sides. When a special boundary condition is used, the band widths must be symmetric about a horizontal line through the middle of the system. There may be an even or odd number of bands; if there is an odd number of bands, the centermost band is unpaired and may be any size.

Figure 8 illustrates the use of the special boundary to represent an hexagonal cell. This boundary condition is implemented in the following way. We assume there are a total of JT bands. The incoming flux on the J'th band in the direction $(-\mu, -\eta)$ is set equal to the outgoing flux on band $JT + 1 - J$ in the direction (μ, η) . Furthermore, the spatial variation of the flux within the band is reversed. Hence the special boundary condition is really a 180° reflection combined with a spatial transposition about the horizontal centerline of the system.

d. White Boundary Condition

The values of the incoming boundary fluxes are all set equal to the same value. This value is the average of the outgoing fluxes such that the net flow through the boundary is zero. For example, for a right boundary this value is given by

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

Fig. 8. Use of the special boundary condition to represent a hexagonal cell.

where the sums range over outgoing directions. White boundary conditions are allowed on all boundaries except the left and require the boundary to be parallel with a coordinate axis. The white boundary condition is meaningful for reactor-cell calculations in cylindrical geometries where it is applied to the outside radial boundary.

e. Source Boundary Condition

The user of TRIDENT may specify the incoming angular flux in each quadrature direction for each of the four sides in any combination. It is not necessary when using this boundary condition that sides be parallel to coordinate axes. Because it is therefore impossible to guarantee which directions are incoming along a certain boundary, and because an incoming direction at one point on the boundary may be an outgoing direction at other points, the user must enter a value for all directions at every point along a source boundary. (An option is provided whereby the user need enter only one value per triangle.) TRIDENT then selects and uses only the values corresponding to incoming directions. Boundary sources represent a source of particles and are treated as sources by the code.

2. Mesh Sweeps

In an orthogonal grid code like TWOTRAN-II, the mesh sweeps are usually organized so that a quadrant of directions is treated simultaneously. This can be done because neutrons streaming in all directions within a single quadrant move through the spatial mesh in the same order. Consider, for example, the positive quadrant ($\mu > 0$ and $\eta > 0$). For all directions in this quadrant, the left and bottom boundaries of each rectangular mesh cell are incoming boundaries and the top and right boundaries are outgoing boundaries. Therefore the spatial mesh can be solved in a straightforward manner beginning with the lower left corner and ending with the upper right corner.

The manner in which a triangular mesh must be swept is less simple. First, for a given direction the order in which the spatial mesh is solved is no longer straightforward but involves testing the direction of flow across triangle boundaries. Second, this order in which the mesh is swept is not the same for all directions within a quadrant but may be different for each direction. These tests must

therefore be repeated for every quadrature direction (and for every group, of course).

The regular triangular mesh used by TRIDENT and derived from TRIPLET simplifies this testing. A regular triangular mesh retains a regular mesh structure in the axial dimension so that the order in which the bands are solved is straightforward. If $\eta > 0$, the neutron flow is upward and the bands are solved in order beginning with the bottom band. If $\eta < 0$, this order is reversed. If $\eta = 0$, there is no vertical flow and thus no coupling between bands; the bands may in this case be solved in any order. For these reasons we need only consider the order in which the triangles are solved on a single band.

Let us assume that μ is positive. The direction of flow across the left boundary of the leftmost triangle is first computed. The direction of flow across the right side of each triangle is then computed, moving from left to right until an outgoing right boundary is encountered. The first triangle encountered whose right boundary is outgoing can then be solved, because the flux on all incoming boundaries is known. After the fluxes in this triangle have been determined, all triangles to the left can be solved in order moving from right to left. This process of skipping several triangles and then solving from right to left is continued until the flux on all triangles on the band has been determined. If $\mu < 0$, just the reverse of the above procedure is followed. This procedure is illustrated in Fig. 9 for a typical band of triangles. The numbers indicate the order in which the triangles are solved, and the arrows indicate the direction of particle flow.

The direction of flow across the side of a triangle is not very difficult to compute. Let us assume that the x (or r) coordinate of the upper

Fig. 9. Typical order of solution on a band of triangles for direction Ω_m .

node is x_u and the x -coordinate of the lower node is x_l (the triangle side is a straight line connecting these two nodes). We assume the band width is Δy . The following quantity is then computed:

$$A = \mu \Delta y + \eta(x_l - x_u).$$

If $A > 0$, the flow across this boundary is from left to right. By left to right we mean that neutrons stream from the triangle to the left of the boundary into the triangle to the right of the boundary. This may occur when $\mu < 0$ and the flow is in a leftward direction. If $A < 0$, the flow across this boundary is from right to left.

In TRIDENT, the space-angle mesh is swept using a procedure somewhat different than that used in TRIPLET. Referring to Fig. 4, the third quadrant is swept first using the directions in the order indicated. The top-most band is selected and successively swept for the six directions in the manner illustrated in Fig. 9. The directions marked with x in Fig. 4 are solved using the step scheme for the case of r - z geometry as described in Sec. II.B.2b. This top band is then solved for quadrant two. The next band is then selected and quadrants three and two again swept. This procedure is continued until the bottom-most band has been solved. Then, this bottom band is solved for the fourth and first quadrants. TRIDENT then proceeds up to the top band solving for the flux in the fourth and first quadrant directions.

3. Adjoint Problems

The TRIDENT program solves the adjoint transport equation by transposing the scattering and fission matrices and inverting the group order of the problem. The solution of the resulting problem in the direction $\underline{\Omega}$ is then identified with the adjoint solution in the direction $-\underline{\Omega}$. Transposition of the scattering matrix converts the normal predominantly downscattering problem to an upscattering problem. Group-order inversion restores this downscattering dominance and eliminates unnecessary upscattering iteration. The fictitious-source option is not available in the adjoint mode.

4. Iterative Processes

We assumed in deriving the method of Sec. II.D for solving Eqs. (20) and (25) on a triangular

mesh that the sources appearing in these equations were known functions. It is clear from Eqs. (15), (21), and (26) that, if scattering, fission and/or the fictitious source are present, these source functions are not known but depend upon moments of the angular flux. In TRIDENT, the source is generated in an iterative context using the latest approximations for the angular flux. For the initial iterate, a flux guess must be input that permits the generation of the source function. The guess for the fictitious source is automatically set to zero. Physical sources in TRIDENT are represented, like the angular flux, by piecewise polynomial functions discontinuous across triangle boundaries. A Lagrange representation of these sources is used, so that a source is computed at each of the node points within a triangle. For the fictitious source, the spatial derivatives are calculated in a manner analogous to the procedure for the streaming operator discussed in Sec. II.D.

In the following analysis we develop the iterative strategies used in TRIDENT 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.

The multigroup-transport equations can be written in operator notation as

$$L\vec{\psi} + \Sigma\vec{\psi} = (S_s + S_d + S_u + S_F)\vec{\psi} + F\vec{\psi} + \vec{Q}, \quad (31)$$

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, and S_F represents the fictitious source. The g^{th} elements of the vectors $\vec{\psi}$ and $\vec{Q}$ contain the unknown fluxes and the known inhomogeneous sources, respectively, in the g^{th} energy group. The form of the operators appearing in Eq. (31) can be inferred by comparing that equation with Eq. (8).

TRIDENT uses a dual iteration strategy for solving the discrete analog of Eq. (31). The two nested iterations are referred to as outer and inner iterations. The outer iteration represents a sweep

through all the energy 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

$$L\vec{\psi}^{k+1} + \Sigma\vec{\psi}^{k+1} = (S_s + S_d + S_F)\vec{\psi}^{k+1} + (S_u + F)\vec{\psi}^k + \vec{Q}. \quad (32)$$

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

We can solve Eq. (31) 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 Q_g^k to the g^{th} group can then be computed as

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

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

$$L_g\psi_g^{k+1} + \Sigma_g\psi_g^{k+1} - (S_{sg} + S_{Fg})\psi_g^{k+1} = Q_g^k. \quad (33)$$

The operators L_g , Σ_g , S_{sg} , and S_{Fg} represent the g^{th} component of the diagonal matrix operators L , Σ , S_s , and S_F . The above equation cannot be solved easily because of the presence of the self-scatter and fictitious-source terms, which couple all directions. The methods developed in Sec. II.B, II.C, and II.D are capable of solving the discrete form of Eq. (33) if scattering, and fictitious sources are assumed known. Thus a second iteration, the inner iteration, is suggested. In TRIDENT, this iteration takes the form

$$(L_g + \Sigma_g)\psi_g^{k+1, \ell+1} = (S_{sg} + S_{Fg})\psi_g^{k+1, \ell} + Q_g^k,$$

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

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. (32) we have

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

In the above equation we have divided the fission source by the parameters κ_k . These parameters are computed as

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

with $\kappa_0 = 1$ and $\langle \rangle$ representing an integration over group, angle, and space variables. The parameters κ_k approach k_{eff} for the system:

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

5. Convergence-Acceleration Methods

In most problems the inner and outer iterations described in the previous section of this report converge rapidly. There exist problems, however, for which these algorithms require excessive iteration for convergence to a satisfactory degree of accuracy. The TRIDENT program uses the acceleration method known as rebalance to accelerate the convergence of both inner and outer iterations. The code user may select one of two versions of rebalance. We call these two versions fine-mesh and whole-system rebalance. The application of these convergence-acceleration methods to the inner and outer iterations is discussed below.

a. Inner-Iteration Rebalance

When Eqs. (20) or (25) are multiplied by w_m , summed over all MM directions, and integrated over the i^{th} triangle, we obtain the following equation:

$$-FI_1^{(1)} - FI_1^{(2)} - FI_1^{(3)} + FO_1 + AB_1 = Q_1. \quad (34)$$

The quantities appearing in the above balance equation are defined as follows:

- $FI_1^{(k)}$ The inward partial flow of neutrons across the k^{th} boundary of the i^{th} triangle.
- FO_1 The total outward partial flow of neutrons across all three boundaries of the i^{th} triangle.
- AB_1 The total absorption rate of neutrons in the i^{th} triangle.
- Q_1 The total source of neutrons in the i^{th} triangle to the particular group at hand from all other groups and from inhomogeneous sources.

TRIDENT requires that the built-in (EQ_N) quadrature set be used with the fictitious source option. With this set, multiplying the fictitious source by w_m and summing over m yields zero.¹⁴ Thus the fictitious source makes no contribution to Eq. (34). Repeating this procedure, we can obtain a balance equation for each triangle. These balance equations are satisfied if the flows and absorptions appearing in them are computed with a fully converged flux. The balance equations will not usually be satisfied by the unconverged fluxes obtained at some stage of the inner iteration.

The object of the fine-mesh-rebalance acceleration method is to find a set of rebalance factors, f_i , one for each triangle, by which all fluxes are multiplied so that the balance equations are satisfied. We assume, in particular, that the absorption and all outflows are multiplied by the rebalance factor in the given triangle. Inflows, which are really outflows from other adjacent triangles, are multiplied by the rebalance factors corresponding to the triangles from which the flow originates. We then obtain a set of equations for the rebalance factors f_i which takes the form

$$-FI_1^{(1)}f_{i_1} - FI_1^{(2)}f_{i_2} - FI_1^{(3)}f_{i_3} + (FO_1 + AB_1)f_i = Q_1. \quad (35)$$

The subscripts i_1 , i_2 , and i_3 designate the three triangles that are adjacent to the i^{th} triangle. If the i^{th} triangle is adjacent to a system boundary, then the above equation must be altered in a manner which depends upon the boundary condition.

With a consistent ordering of the triangles, Eq. (35) represents a block tri-diagonal system of equations for the rebalance factors f_i . This system of equations is solved iteratively in TRIDENT, and the resulting rebalance factors are applied to all the fluxes.

Whole-system rebalance is similar to fine-mesh rebalance, except that a single rebalance factor is determined by which all fluxes are multiplied. This factor is easily seen to be the ratio of the total source to the sum of the leakage plus the absorption.

We observe experimentally that fine-mesh rebalance almost always yields a large reduction in computing time over that required for the unaccelerated iteration. Since the fictitious source, however, makes no contribution to Eq. (35), fine-mesh rebalance does nothing to accelerate convergence of that source.¹⁴ Fine-mesh rebalance is usually more efficient than whole-system rebalance, although it requires more core storage for the flows and absorptions. There are isolated problems in which the use of fine-mesh rebalance can lead to an unstable algorithm. If the user encounters such difficulty, he is advised to use whole-system rather than fine-mesh rebalance.

Two methods developed for TRIPLET have been incorporated in TRIDENT for the stabilization of fine-mesh rebalance. The first method checks the set of Eqs. (35) for diagonal dominance. Diagonal dominance of this set of equations is guaranteed by adding factors to the diagonal entries and to the sources. This procedure changes the rebalance factors so that they are closer to unity. The second method adds another factor to both the diagonal and the source. This factor is chosen to be a small fraction of the largest diagonal element. This second method also forces the rebalance factors to be closer to unity and tends to damp oscillations that sometimes develop in the application of this method.

b. Outer-Iteration Rebalance

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

sorption using an effective σ_{ag} calculated from the input cross sections by

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

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

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

The source for over-all-group rebalance consists of the group sum of the inhomogeneous source (if any) and the group sum of the fission source (if any). Again the fictitious source vanishes when summed over all directions. If there is a fission source, we perform a source iteration to determine the rebalance factors. If there is no inhomogeneous source, this iteration can also be used to estimate an eigenvalue, say k_{eff} . We choose to do this only for the first two outer iterations, thereafter considering the fission source to be a known inhomogeneous source.

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

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

6. Convergence Tests

There are three levels of iterative processes in the TRIDENT program: (1) the inner iteration, in which the within-group scattering source and/or the fictitious source, and/or the boundary flux at an implicit boundary change, (2) the outer iteration in which the fission or upscattering source changes or which is caused by artificial inner iteration limitation (usually in inhomogeneous source problems), and (3) the parametric eigenvalue search iteration in which, after a converged outer iteration, the

value of a material concentration or a time absorption is changed. Two additional iterations are required for the calculation of fine-mesh rebalance factors, one for the factors themselves and one for the fine-mesh rebalance eigenvalue.

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

$$\lambda(r) = \frac{\text{Fission Source}^r + \text{Inhomogeneous Source}}{\text{Fission Source}^{r-1} + \text{Inhomogeneous Source}} \quad (36)$$

are required to differ 1, less than XLAX before a new eigenvalue guess is computed. All other processes are tested against precisions derived from EPSO. These precisions are

EPSO outer iteration convergence
 EPSI = EPSO inner iteration convergence
 EPSR = 10^{-4} fine-mesh rebalance numbers
 EPSX = EPSO collapsed group fine-mesh convergence,
 both factors ($10 \cdot \text{EPSX}$) and eigenvalue
 (EPSX).

In the inner iteration processes, we require

$$\max_i \left| 1 - \frac{\phi_i^{r-1}}{\phi_i^r} \right| < \text{EPSI}$$

for the r'th iteration where ϕ_i is the average scalar flux in the i'th triangle before application of the rebalance factors. If the number of iterations is greater than the input value IITL, the inner iteration is terminated.

For the iterative computation of the fine-mesh rebalance factors in the inner iteration, we require

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

In the outer process, if both

$$|\lambda - 1| < \text{EPSO} \text{ and } \max_i |1 - f_i| < 10 \cdot \text{EPSX},$$

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

$$|\lambda_{\text{Fine Mesh}}^{(r)} - 1| < \text{EPSX},$$

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

$$1 - \lambda_{\text{FM}}^{(r)} / \lambda_{\text{FM}}^{(r-1)} < \text{EPSX},$$

if IEVT $\neq$ 1 (parametric eigenvalue or inhomogeneous source plus fission problems).

III. A GUIDE TO USER APPLICATION

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

A. Overall Program Flow

A schematic flow chart for TRIDENT is given in Fig. 10. A more detailed flow chart is provided below in the section on programming information.

B. Details of Program Options

1. Cross Sections

a. Input Formats

The TRIDENT program accepts cross sections either from the standard file ISOTXS,⁷ in FIDO format,¹⁵ or in the standard Los Alamos format discussed below. In upscattering problems, the program does not need the special σ^{up} cross section which is required in earlier Los Alamos programs.¹⁶ In TRIDENT, it is assumed that σ^{up} is present, and σ^{up} is automatically removed from card input cross-section sets unless the user tags the input number IHT with a minus sign. Cross sections read with the FIDO format may not contain σ^{up} .

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

TABLE IV
CROSS-SECTION INPUT ORDER
WITH LOS ALAMOS FORMAT

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	$\sigma_{s,g+N+g}$
.	
.	
.	
IHS-2	$\sigma_{s,g+2+g}$
IHS-1	$\sigma_{s,g+1+g}$
IHS	$\sigma_{s,g+g}$
IHS+1	$\sigma_{s,g-1+g}$
IHS+2	$\sigma_{s,g-2+g}$
.	
.	
.	
IHS+M	$\sigma_{s,g-M+g}$

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

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

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

Fig. 10. Simplified logical flow diagram for TRIDENT.

b. Cross-Section Mixing

The user is free, in TRIDENT, 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 $MS \neq 0$, the user must provide three sets of MS numbers which are stored in the vectors $MIXNUM$, $MIXCOM$, and $MIXDEN$. These numbers are used in the following algorithm to manipulate cross-section blocks:

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

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

Suppose we have entered 45 cross sections as input. Then any mixtures that are made must be given block numbers higher* than 45. For example suppose we enter the constants of Table V. For this example we have $MS = 17$ instructions. In the first three instructions, block 46 is cleared (set to zero) and then made up of 0.0478 parts of block 1 and 0.0333 parts of block 20. If blocks 1 and 20 are microscopic cross sections in barns, then 0.0333 and 0.0478 times 10^{24} are the atomic densities. In the second set of instructions, block 47 is cleared and then made up of 0.1179 times the result of adding three-fourths of block 2 to one-fourth of block 3. In the next set of instructions, block 48 is cleared and made up of portions of blocks 15 and 14. If IEVT (the input eigenvalue

TABLE V
SAMPLE MIXTURE-SPECIFICATION PARAMETER

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

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 $\neq$ 3, the line of instructions 48, 48, 0.0 would not alter the composition of block 48. In the final sequence, block 49 is made up of 0.187 times one-half of block 33 and block 34, and provision is made to search for the concentration of this portion of 49 to which is always added 0.1 of the previously mixed block 46. It should be clear that there are many possibilities not covered in these examples, but by examining the FORTRAN instructions above, the user should be able to prepare his own sets of mixture instructions.

c. Anisotropic Cross Sections

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

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

$$\sigma_s(E' \rightarrow E, \mu_0) = \sum_{\ell=0}^{\text{ISCT}} \frac{2\ell+1}{4\pi} P_{\ell}(\mu_0) \sigma_{s\ell}(E' \rightarrow E),$$

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

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

d. Adjoint Cross Sections

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

$$(\sigma_a)_{\text{eff}} = \sigma_t - \sum_{\text{all } h} \sigma_{so,g+h} \quad (37)$$

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

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

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

e. Cross-Section Checking

As input cross sections are processed, Eq. (37) 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 EPSO (outer convergence precision) the user is so informed.

2. Geometry and Boundary-Condition Satisfaction

a. Spatial Mesh

The input of information necessary to specify the spatial mesh is greatly simplified by the restriction to a regular triangular mesh. The first three items necessary for complete specification of the mesh are:

- (1) The number of bands of triangles: JT.
- (2) The number of triangles on each band: IT array, JT entries, from the bottom band to the top band.
- (3) The orientation of the first triangle on each band: ITT array, JT entries, from the bottom band to the top band.

With the above data, the subroutine KSET calculates the total number of triangles (NTC), the total number of triangle vertices (NTX), the number of upward-pointing triangles on the bottom band (NUPB), the number of downward-pointing triangles on the top band (NDNT), the two required indexing arrays (KC and KX). The remaining mesh specification items which must be input are:

- (4) The height of each triangle band: HY array, JT entries, from the bottom band to the top band.
- (5) The radial coordinates of the triangle vertices: X array, NTX entries.
- (6) The cross-section material identification number for each triangle: IDC array, NTC entries.

The radial coordinates are numbered sequentially from the lower left corner to the upper right corner, as indicated in Fig. 11, and are input in this order. Each triangle has an identification number associated with it. These triangle i.d.'s run from left to right, starting

with the bottom band and proceeding upward, as illustrated in Fig. 11.

After all of the problem input has been read by TRIDENT, the above information is converted into a pictorial representation of the mesh by subroutine MAPPER as seen for example in Fig. 12, where an 8-triangle example is shown. The material identification number is printed within each triangle. For this example, the system is homogeneous in material 1. The axial coordinate is printed to the left of each node line (line of triangle vertices). The symbol beside the band index denotes the orientation of the first triangle on each band (up- or down-pointing). The triangle vertices on each node line are numbered from left to right. The TRIDENT user may draw lines between the vertices to complete the picture of the spatial mesh, as illustrated in the example. Furthermore, a table of the radial coordinates is printed immediately before the material map and numbered to give the radial coordinate of the vertices along each node line. The material map is drawn to scale in the radial direction, with a uniform band height. For large meshes with many triangles, the material map is automatically continued onto additional pages. Additional characters are used to indicate those boundaries that are reflecting, special, or white.

For input of the distributed source or flux guess the TRIDENT user can specify a pointwise variation over each triangle. The indexing of these points on a triangle depends upon whether the triangle is upward or downward pointing. For upward-pointing triangles, the numbers start with the left-most point on the bottom row and proceed counter clockwise. For downward-pointing triangles, the

Fig. 11. Example of x- or r-coordinate (small numbers) and triangle (large numbers) indexing.

numbers start with the right-most point on the top row and also proceed counter clockwise. This procedure is illustrated in Figs. 6 and 7.

b. Boundary Conditions

The TRIDENT user must select one of the following five boundary conditions for each of the system boundaries.

- (1) Vacuum boundary conditions - the angular flux on the boundary is set to zero for all incoming directions.
- (2) Reflective boundary conditions - the angular flux on the boundary for incoming directions is set equal to the outgoing flux in the direction corresponding to specular reflection.
- (3) Boundary sources - the angular flux on the boundary for incoming directions is set equal to a user-specified value.
- (4) White boundary conditions - the incoming angular flux values are all set to the average of the outgoing flux.

For the right boundary in x-y geometry the user is allowed the

- (5) Special boundary condition - the angular flux on the boundary of the j 'th band for incoming directions $(-\mu, \eta)$ is set equal to the outgoing flux on band $JT-j+1$ for directions $(\mu, -\eta)$.

Use of reflecting or special boundary conditions imposes certain requirements on the S_N quadrature set, as discussed in Sec. II.B. Additionally, side boundaries must be vertical if reflecting, special, or white boundary conditions are applied. For a special boundary condition, the band heights must be symmetric about the centerline ($HY_{JT-j+1} = HY_j$).

3. Angular Quadrature Coefficient Specifications

The TRIDENT user has the option of obtaining the angular quadrature coefficients from interface file SNCONS or from a built-in set in subroutine SNCON, or the user may enter these coefficients as card input. The input parameter ISN is used to specify the source of these coefficients.

Fig. 12. Example of a pictorial representation of the mesh by subroutine MAPPER.

The built-in constants are the EQ_N set.⁶ If the built-in set is used, input entries for MM, MNPQ, MNEL (total number of quadrature directions, maximum number of quadrature directions in any quadrant, and maximum number of quadrature directions on a given eta level in any quadrant) may be left blank. If the quadrature set is obtained from an interface file or cards, these values must be entered. For the input quadrature set of Fig. 5, for example, MM = 16, MNPQ = 6, and MNEL = 3.

Considerable freedom is allowed the TRIDENT user in the input of quadrature sets. However, in addition to the above-mentioned symmetry requirements, two requirements must be met. The weights, w_m , must be correctly normalized ($\sum_{m=1}^{MM} w_m = 1$), and the quadrature set must correctly integrate the spherical-harmonics polynomials through order ISCT. Subroutine SORTQS sorts input quadrature sets as indicated in Figs. 4 and 5 for use by solution algorithms. Consequently, the sets may be input in any order satisfying the card-input format specifications of Sec. III.D or interface-file format provided in Ref. 7.

4. Source Options

The TRIDENT user may specify an anisotropic distributed source or the boundary flux at any boundary of the system. The inhomogeneous distributed source must be represented by the finite-spherical-harmonics expansion

$$Q_g(\underline{r}, \mu, \eta) = \sum_{\ell=0}^{IQAN} (2\ell+1) \sum_{k=0}^{\ell} R_{\ell}^k(\mu, \phi) Q_{\ell g}^k(\underline{r}),$$

in which IQAN is an input number designating the order of anisotropy of the source, and R_{ℓ}^k are the spherical-harmonics polynomials defined in Eq. (3). In these terms,

$$Q_{\ell g}^k = \frac{1}{4\pi} \int_{-1}^1 d\mu \int_0^{2\pi} d\phi Q_g(\underline{r}, \mu, \eta) R_{\ell}^k(\mu, \phi).$$

We have written the integral in this equation over the entire range of ϕ and used a 4π normalization of

the integral, but it should be clear that Q_g must be symmetric in ϕ [$Q_g(\phi) = Q_g(-\phi)$].

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

$$(2\ell + 1)Q_{\ell g}^k \quad k = 0, 1, \dots, \ell \\ \ell = 0, 1, \dots, IQAN$$

in the order shown in Table VI. 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 polynomials R_{ℓ}^k is computed.

The TRIDENT user is also allowed to specify the value of the incoming flux along any or all boundaries. If the source-boundary-condition integer is -3, the boundary flux is assumed to be constant along the face of each boundary triangle (subsequently referred to as the flat-source mode). If the source-boundary-condition integer is +3, the boundary flux is assumed to have a pointwise variation along the face of each boundary triangle (subsequently referred to as the shaped-source mode). Use of the source-boundary-condition integer as -3 reduces (by one-half) the number of entries required to specify the boundary flux. The boundary sources,

TABLE VI
ORDERING OF ANISOTROPIC-
DISTRIBUTED SOURCE COMPONENTS

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

summed over all groups and integrated over all angles and surface area, are added to the volume-energy-integrated distributed sources (if any) for the normalization of all inhomogeneous sources.

5. Source and Flux Input Options

a. Source Input Options

If a distributed source of anisotropy IQAN is designated, the $NMQ = (IQAN+1)(IQAN+2)/2$ components of the source must be entered for each group in the order listed in Table VI. This may be accomplished with the use of four options designated by the input value of IQOPT and described below. The various source arrays described below are supplied for each group, with the distributed source (if any) specified first, followed by the boundary sources (if any). The directional information must be input in the order indicated, for example, in Figs. 4 and 5, regardless of the ordering of the quadrature set input, if any. The ordering of the source input within each group g is then:

- (1) Distributed source (optional) - order of input depends on IQOPT as described below.

- (2) Right boundary source (optional) - from the bottom triangle to the top triangle on the right side.

$BR_{gkjm}; k=1,2; j=1,NT; m=1,MM$ for $IBR = +3$

or

$BR_{gjm}; j=1,NT; m=1,MM$ for $IBR = -3$.

- (3) Bottom boundary source (optional) - from the left triangle on the bottom band to the right triangle.

$BB_{gkjm}; k=1,2; j=1,NUPB; m=1,MM$ for $IBB = +3$

or

$BB_{gjm}; j=1,NUPB; m=1,MM$ for $IBB = -3$.

- (4) Top boundary source (optional) - from the left triangle on the top band to the right triangle.

$BT_{gkjm}; k=1,2; j=1,NDNT; m=1,MM$ for $IBT = +3$

or

$BT_{gjm}; j=1,NDNT; m=1,MM$ for $IBT = -3$.

- (5) Left boundary source (optional) - from the bottom triangle to the top triangle on the left side.

$BL_{gkjm}; k=1,2; j=1,NT; m=1,MM$ for $IBL = +3$

or

$BL_{gjm}; j=1,NT; m=1,MM$ for $IBL = -3$.

There are two k values since there are two points on one side of a triangle. NUPB is the number of upward-pointing triangles on the bottom band, and

NDNT is the number of downward-pointing triangles on the top band. The above notation is that of standard FORTRAN (the innermost set of indices cycled most rapidly). For example, for $IBL = -3$, the boundary flux BL is entered in a continuous stream, the first JT numbers for $m = 1$, the next JT numbers for $m = 2$, etc.

The IQOPT parameter provides the TRIDENT user with five options for entering the distributed source. These options are:

IQOPT

OPTION

- 0 No input entered. The complete distributed source array, $Q_{g\ell k i}$, is automatically set to zero.
- 1 Enter an energy spectrum ($GR_{g\ell}$, $g=1,IGM$) for each anisotropic component ℓ . The source array is then formed as $Q_{g\ell k i} = GR_{g\ell}$; $k = 1,3; i=1,NTC$ for each ℓ and g .
- 2 Enter the complete source array as ($Q_{g\ell k i}$; $k=1,3; i=1,NTC$) for $\ell=1,NMQ$ and $g=1,IGM$. See Secs. II.D and III.B.2a for the ordering of the three points on each triangle and the ordering of the NTC triangles on the mesh. The above entries are input as a block of $3*NTC$ numbers for each ℓ and g .
- 3 Enter first a spectrum ($GR_{g\ell}$, $g=1,IGM$) and then a spatial shape ($F_{g\ell k i}$; $k=1,3; i=1,NTC$) for $\ell=1,NMQ$. The source array is then formed as $Q_{g\ell k i} = GR_{g\ell} F_{g\ell k i}$; $k=1,3; i=1,NTC$ for each ℓ and g . See Sec. II.D and III.B.2a for the ordering of the three points on each triangle and the ordering of the NTC triangles on the mesh. The above entries are input as blocks of IGM numbers and $3*NTC$ numbers for each ℓ .
- 5 The distributed source (but not boundary sources) is read from a standard interface file FIXSRC.

The quantity NTC is the total number of triangles in the spatial mesh with three points per triangle for which the source and flux must be stored.

b. Flux Input Options

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

ISTART

OPTION

- 6 A problem restart dump is read from DUMPIT and edited with a minimum of computation.

- 5 Same as option +5.
- 3 Same as option +3, isotropic component only.
- 2 Same as option +2, isotropic component only.
- 1 Same as option +1, isotropic component only.
- 0 No flux guess entered. A fission guess of unity in every mesh cell is automatically supplied.
- +1 Enter an energy spectrum ($GR_{g\ell}$, $g=1, IGM$) for each anisotropic component ℓ . The flux guess is then formed as $FLUX_{g\ell k i} = GR_{g\ell}$; $k=1,3$; $i=1, NTC$ for each ℓ and g .
- +2 Enter the entire flux array as ($FLUX_{g\ell k i}$; $k=1,3$; $i=1, NTC$) for $\ell=1, NM$ and $g=1, IGM$. See Secs. II.D and III.B.2a for the ordering of the 3 points on each triangle and the ordering of the NTC triangles on the mesh. The above entries are input as a block of $3*NTC$ numbers for each ℓ and g . Imagine that the fluxes are read by:

```
DO 17 G=1,IGM
DO 17 L=1,NM
17 READ ((FLUX(G,L,K,I), K=1,3), I=1,NTC)
```
- +3 Enter first a spectrum ($GR_{g\ell}$, $g=1, IGM$) and then a spatial shape ($F_{\ell k i}$; $k=1,3$; $i=1, NTC$) for $\ell=1, NM$. The flux array is then formed as $FLUX_{g\ell k i} = GR_{g\ell} F_{\ell k i}$; $k=1,3$; $i=1, NTC$ for each ℓ and g . See Secs. II.D and III B.2a for the ordering of the 3 points on each triangle and the ordering of the NTC triangles on the mesh. The above entries are input as blocks of IGM numbers and $3*NTC$ numbers for each ℓ . Imagine the fluxes to be read by:

```
DO 17 L=1,NM
READ (GR(G), G=1,IGM)
READ ((F(K,I), K=1,3), I=1,NTC)
DO 17 G=1,IGM
DO 17 K=1,3
DO 17 I=1,NTC
17 FLUX(G,L,K,I) = GR(G)*F(K,I)
```
- +5 An entire scalar flux guess is read from standard interface file RTFLUX or ATFLUX.
- +6 A problem restart dump is read from DUMPIT and computation proceeds in the normal manner.

The fictitious source guess is always assumed to be zero.

6. 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 M minutes where M is a program variable which can be set to meet particular installation requirements. A final dump is always taken after the successful completion

of a problem, and a time-limit dump is taken after a user-specified period of time. Dumps are written alternately on two versions of the generic-named file DUMPIT depending on which is free; and output message is written to indicate which version contains the latest dump.

When problem execution is continued using a restart dump, certain input parameters can be changed and edit specifications can be added or modified. It is even possible to use the program to obtain additional edits from a final dump. The same request may be made of a nonconverged problem with one or more outer iterations being required to generate the angular fluxes for the edit.

To restart a problem, a special problem input deck is required consisting of three sections. The first section is the same as the normal problem from number of title cards through integer input with the value of ISTART set to ± 6 . During restart all other values entered on those integer cards are ignored. The second section consists of two cards which enable the user to change certain integer and floating-point input parameters. The first card is entered in a 6(I1,I11) format and the second card in a 6(I1,E11.3) format. The I1 entry is zero or one with the zero value indicating no change in the corresponding input parameter. The one value indicates that the user desires the input parameter changed to the eleven-field-length value. The data required for cards one and two are depicted in Tables VII and VIII respectively.

The third section of restart input is the edit section. The composition of the edit input section is determined by the value of IEDOPT after card 1 is processed. For example, if the original problem

TABLE VII
RESTART DATA IN CARD 1

Field Pair	Input Parameter	
1	IITL	maximum number of inner iterations
2	ITLIM	time limit
3	IEDOPT	edit input indicator
4	IFP	final-flux-print indicator
5	I4	final-fission-print indicator
6	IFO	interface-file-output indicator

TABLE VIII
RESTART DATA IN CARD 2

Field Pair	Input Parameter	
1	EPSO	convergence precision (may be used for example with ISTART=6, to tighten the convergence for a problem with a final dump)
2	POD	parameter-oscillation damper used in eigenvalue searches
3	XLAL	search-lambda lower limit
4	XLAH	search-lambda upper limit
5	XLAX	fine-mesh-search precision

had IEDOPT > 0 and IEDOPT was not changed, the original edit input must be re-entered.

The sign of ISTART determines the iteration path following a restart. If ISTART=+6, computation is restarted at the exact point at which the dump was taken. In the case of a final-dump problem restart and with no change in EPSO, the final-print section is entered. If ISTART=-6 and a final-dump problem restart is requested, the edits are immediately processed. Otherwise this option is treated as the positive entry.

7. Rebalance Acceleration of Iterations

The user is given two choices of rebalance schemes for acceleration of the inner and outer iterations. Particle balance may be enforced over the entire system (whole-system rebalance) or each triangle (fine-mesh rebalance). Normally, more rapid convergence will be achieved with fine-mesh rebalance, at the expense of 7*NTC additional words of fast core. Experience has shown that for some problems, the fine-mesh-rebalance algorithm is unstable. For such problems one must use whole-system rebalance to obtain a solution. When the special boundary condition is used, only whole-system rebalance is permitted.

8. Eigenvalue Searches

It is possible in TRIDENT to adjust nuclide concentrations or the value of the time absorption to achieve a desired value of k_{eff} . This value is

taken to be unity (criticality) unless the parametric eigenvalue trigger (IPVT) is set to unity. In this case, the parametric value of k_{eff} is entered as an input number. If IPVT=2, a 1/v absorber of value PV is added to the problem in each space cell.

The modification of cross-section concentrations takes place as indicated in Sec. III.B.1.b. This type of problem is run when the eigenvalue type indicator (IEVT) is 3. If IEVT is 2 (time-absorption computation), the value EV/v_g is added to the absorption and total cross sections in each group. Here v_g is the speed associated with energy group g.

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 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. (36) unity

In the following description of NEWPAR, it is helpful to refer to Fig. 13 in which the deviation of λ from unity is plotted against outer-iteration number. For the initial system, NEWPAR continues outer iteration until two successive values of λ differ by less than EPSO. For subsequent sequences of λ values a different convergence precision, XLAX, is used. After the first converged λ sequence is obtained, the initial value of the eigenvalue (EV) is altered by EVM, an input value. If $\lambda > 1$ (multiplying system), the new eigenvalue is equal to $EV + EVM$; if $\lambda < 1$ (decaying system), the new value is $EV - EVM$. These alterations correspond to the addition or the subtraction of an absorption, e.g., as in a time-absorption search or a poison-concentration search.

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 $|1 - \lambda|$

Fig. 13. Variation of λ during a hypothetical eigenvalue search.

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, a linear fit is used and the new eigenvalue is computed from

$$(EV)_{\text{new}} = (EV)_{\text{old}} + POD * EQ * (1 - \lambda), \quad (38)$$

where POD is an input "parameter oscillation damper" which may be used to restrict the amount of change in the eigenvalue. In Eq. (38), EQ is a measure of the slope of the curve. When $|1 - \lambda| > XLAH$, $(1 - \lambda)$ in Eq. (38) is replaced by XLAH (with the correct sign) to prevent too large a change in EV. After $|1 - \lambda| < XLAL$, the value of EQ is fixed and kept constant until convergence to prevent numerical

difficulty in the approximation of the derivative when λ is close to unity.

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

Convergence in time-absorption calculations is typically one-sided. If EV 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.

9. Adjoint Computations

The TRIDENT 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 Ω is then identified with the solution of the adjoint equation in direction $-\Omega$.¹⁰

The inversion of the group order is made because the transposition of the scattering matrices usually converts a downscattering problem to an up-scattering 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 in a direct problem. On the other hand, an output flux from one adjoint problem is in the proper group order for use in another adjoint problem. The group order of the group speeds and the fission spectrum is inverted by the program. Note that the fictitious-source option may not be used in the adjoint mode.

10. Edit Options

The TRIDENT 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 triangles which have the same zone number. The user defines a zone by entering a set of NTC numbers (NEDZ array) which associate with each triangle on the fine mesh a zone identification number (zone i.d.). The triangles of an edit zone need not be contiguous. For each zone edit, a map of the zone i.d. similar to the material map is printed by EDMAP. For each group and zone, a table containing the zone volume, net leakage, buckling absorption, source, integral flux, average flux, and macroscopic activities (for cross-section positions 1 through IHT) is given. The macroscopic activity $A_k^j(g, IPOS)$ in zone k and group g for cross-section position IPOS is defined by

$$A_k^j(g, IPOS) = \sum_1 C(g, IPOS, m_1) \phi_1 V_1 \text{ for } i \in \text{zone } k,$$

where m_1 is the material i.d. (cross-section block identification number) for triangle i, $C(g, IPOS, m)$ is the cross section for group g in position IPOS

for material m, V_1 is the triangle volume, and ϕ_1 is the average flux in triangle i. Thus A_k^j is the activity computed with the macroscopic cross section actually used in the problem, summed over all triangles in zone k.

For each zone edit, the TRIDENT 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

$$A_k^j(g, IPOS) = \sum_1 C(g, IPOS, m_1) \phi_1 V_1 \delta_{jm_1} \text{ for } i \in \text{zone } k.$$

Here δ_{jm_1} equals unity if material j equals material m_1 , the mixture table density (MIXDEN) if material j is a "constituent" of material m_1 , and is zero otherwise. A "constituent" means that material j appears as an entry in the MIXNUM array with density MIXDEN (see Sec. III.D) that is used to form material m_1 . Thus if material j is used to form a material j', which is used to form material m_1 , then material j is not a "constituent" of material m_1 within this definition.

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

$$A_k^j(g, IPOS) = \sum_1 C(g, IPOS, j) \phi_1 V_1 \text{ for } i \in \text{zone } k.$$

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 TRIDENT 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 $v \times$ fission rate divided by the zone volume), normalized to that of a user-designated zone. The zone relative power density (unnormalized) is defined by

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

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 TRIDENT provides the user with the option of obtaining the pointwise variation across each triangle of the scalar flux and activities. The user must enter the triangle i.d.'s over which the point edits are desired (NEDPT array). For each point edit, a map of the point arrangement indexing on a triangle is printed (see Figs. 6 and 7 for example) by subroutine ARRMAT. This is followed by the pointwise scalar flux for the specified triangles. The user is also provided by the NPMA (number of point microscopic activities) parameter the option of obtaining a pointwise microscopic activity for any desired material. This pointwise microscopic activity for material j in triangle i at point k is

$$A_{ki}^j(g, IPOS) = C(g, IPOS, m_i) \phi_{ki},$$

where ϕ_{ki} is the scalar flux at point k in triangle i. The point edit is based on the angular flux stored on the file named TEMPAF.

C. Data-Input Rules

Except for the control parameters, cross sections, and edit parameters, all floating-point numbers and integers are read into TRIDENT 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 IX. Five illustrations of the special formats are given in Fig. 14. These illustrate:

- 1 - Zero is repeated 47 times.
- 2 - Zero is repeated 470 times.
- 3 - Four interpolants are inserted between 0.0 and 5.0 giving six data numbers: 0.0, 1.0, 2.0, 3.0, 4.0, 5.0.
- 4 - Four interpolants are inserted between 0.0 and 5.0, two between 5.0 and 7.0, and 7.0 is repeated 10 times.
- 5 - After reading 0 and 4 we skip to the next card and read 7.

A special routine, PRINTS, 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. When fluxes and fission rates are printed, they are oriented with the origin at the lower left of the page, corresponding to the picture drawn by MAPPER.

D. Description of Input Data

In the following pages the input data for TRIDENT 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 and 5, (3) problem-dependent data on subsequent cards, and (4) edit input.

TABLE IX
OPTIONS FOR SPECIAL READ FORMATS

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

I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2												I1 I2											
-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--	-------	--	--	--	--	--	--	--	--	--	--	--

TABLE X (cont)

<u>Number of Word on Card</u>	<u>Name of Variable</u>	<u>Comments</u>
11	IEVT	0/1/2/3 (inhomogeneous-source/ k_{eff} -calculation/time-absorption or alpha/concentration-search) eigenvalue type.
12	ISTART	-6/-5/-3/-2/-1/0/1/2/3/5/6 input flux guess and starting options. See Sec. III.B.5.b.
CONTROL INTEGERS (12I6)-----CARD 2		
1	IGEOM	Geometry option (1/2) x,y/r,z.
2	MT	Total number of materials (cross-section blocks including anisotropic cross sections) in the problem.
3	MTPS	Number of input material sets from the interface file ISOTXS. <u>Caution:</u> each material set from this file yields ISCT+1 materials. See IDLIB in Table XI.
4	MCR	Number of input materials from the code-dependent input file. If this number is negative, FIDO format cross sections are read.
5	MS	Number of mixture instructions. See Sec. III.B.1 and items MIXNUM, MIXCOM, and MIXDEN below.
6	IHT	Row of total cross section in the cross-section format. If IHT < 0, code assumes that there is no σ^{up} in the cross-section table.
7	IHS	Row of within-group scattering cross section in the cross-section format.
8	IHM	Total number of rows in the cross-section format.
9	IQOPT	0/1/2/3/5 options for input of inhomogeneous source. See Sec. III.B.5.a.
10	IQAN	Order of anisotropy of inhomogeneous distributed source.
11	IPVT	0/1/2 (none/ k_{eff} /alpha) parametric eigenvalue entered. See entry PV.
12	IITL	Maximum number of inner iterations allowed per group.
CONTROL INTEGERS (3I4,10I6)-----CARD 3		
1	MM	Total number of quadrature directions.
2	MNPQ	Maximum number of quadrature directions in any quadrant.
3	MNEL	Maximum number of quadrature directions on a given eta level in any quadrant. (The three values above are used when the quadrature set is input via cards or a standard interface file. The data are not used when the built-in quadrature set is selected.)
4	IACC	0/1 (whole-system/fine-mesh) rebalance acceleration method.
5	ITLIM	0/N (no/N-second) time limit. If an integer number of seconds is entered, a restart dump is taken after this number of seconds and the problem is terminated.
6	IEDOPT	0/1 (no/yes) edit input to be entered.
7	I1	0/1 (yes/no) full-input-flux-print-suppression indicator.
8	IFP	0/1/2 (all/isotropic/none) final-flux-print indicator.
9	I3	0/1/2 (all/mixed/none) cross-section-print indicator.
10	I4	0/1 (yes/no) final-fission-print indicator.
11	I5	0/1/2/3 (all/unnormlized/normalized/none) source-print indicator.
12	ISDB	0/1 (whole-system/space-dependent) buckling-height indicator.
13	IFO	0/1 (no/yes) interface-file output is created.

TABLE X (cont)

<u>Number of Word on Card</u>	<u>Name of Variable</u>	<u>Comments</u>	
CONTROL FLOATING-POINT DATA(6E12.4)-----			CARD 4
1	EV	Eigenvalue guess. It is satisfactory to enter 0.0 for IEVT = 2 and 1.0 for IEVT = 3.	
2	EVM	Eigenvalue modifier used only if IEVT > 1. See Sec. III.B.8.	
3	PV	Parametric value of k_{eff} for subcritical or supercritical systems or for 1/v absorption. See Sec. III.B.8.	
4	XLAL	Lambda lower limit for eigenvalue searches. See Sec. III.B.8.	
5	XLAH	Search lambda upper limit.	
6	XLAX	Search lambda convergence precision for second and subsequent values of the eigenvalue.	
CONTROL FLOATING-POINT DATA (6E12.4)-----			CARD 5
1	EPSO	Convergence precision.	
2	NORM	Normalization factor. Total number of source or fission particles normalized to this number if it is nonzero. No normalization if NORM is zero.	
3	POD	Parameter oscillation damper used in eigenvalue searches. See Sec. III.B.8.	

TABLE XI
PROBLEM-DEPENDENT DATA

<u>Block Name and Dimension</u>	<u>Format</u>	<u>Number of Entries</u>	<u>Comments</u>
IT(JT)	S(I)	JT	Integers defining the number of triangles on each band.
ITT(JT)	S(I)	JT	1/2 (up/down) integers indicating orientation of first triangle on each band.
C(IHM,IGM,MIN)			Three options are available for reading cross sections. The LASL input format may not be mixed with the FIDO format. 1. <u>LASL INPUT</u> . If MCR.GT.0, MCR blocks of IHM*IGM numbers are read in a 6E12.5 format. Each block is preceded by an identification card read in a 18A4 format. MIN = MCR + MTPS*(ISCT+1). 2. <u>FIDO INPUT</u> . If MCR.LT.0, MCR blocks of data are created from FIDO input. 3. <u>INTERFACE FILE ISOTXS</u> . When MTPS.GT.0, MTPS material sets are read from standard file ISOTXS. On this file each material set consists of ISCT+1 cross-section blocks for the isotropic and ISCT anisotropic cross sections. The first component of the first material is stored in cross-section block MCR+1, the first component of the second material is stored in cross-section block MCR+ISCT+2, etc. Should the ISOTXS file not contain ISCT anisotropic components, zeroes are supplied for the components not

TABLE XI (cont)

<u>Block Name and Dimension</u>	<u>Format</u>	<u>Number of Entries</u>	<u>Comments</u>																								
IDLIB(MTPS)	S(I)	MTPS	present. If the ISOTXS file contains more components than needed, only the first ISCT+1 components are read. Position numbers of material sets to be read from ISOTXS. <u>Do not enter unless MTPS.GT.0.</u> The material sets need not be in order of increasing identification number. Number of entries depends on option. See Sec. III.B.5.b.																								
Input FLUX Guass FLUX(NM,NTC,3)	S(E)		<table><tr><th><u>Option</u></th><th><u>Number of entries</u></th></tr><tr><td>-6</td><td>Problem restart dump from unit DUMPIT for immediate edits</td></tr><tr><td>-5</td><td>Input from RTFLUX or ATFLUX standard interface file</td></tr><tr><td>-3</td><td>IGM, then NTC*3</td></tr><tr><td>-2</td><td>IGM sets of NTC*3</td></tr><tr><td>-1</td><td>IGM</td></tr><tr><td>0</td><td>None</td></tr><tr><td>1</td><td>NM sets of IGM</td></tr><tr><td>2</td><td>NM groups of (IGM, then NTC*3)</td></tr><tr><td>3</td><td>IGM, then NM sets of NTC*3</td></tr><tr><td>5</td><td>Same as -5.</td></tr><tr><td>6</td><td>Problem restart dump from unit DUMPIT for problem continuation</td></tr></table>	<u>Option</u>	<u>Number of entries</u>	-6	Problem restart dump from unit DUMPIT for immediate edits	-5	Input from RTFLUX or ATFLUX standard interface file	-3	IGM, then NTC*3	-2	IGM sets of NTC*3	-1	IGM	0	None	1	NM sets of IGM	2	NM groups of (IGM, then NTC*3)	3	IGM, then NM sets of NTC*3	5	Same as -5.	6	Problem restart dump from unit DUMPIT for problem continuation
<u>Option</u>	<u>Number of entries</u>																										
-6	Problem restart dump from unit DUMPIT for immediate edits																										
-5	Input from RTFLUX or ATFLUX standard interface file																										
-3	IGM, then NTC*3																										
-2	IGM sets of NTC*3																										
-1	IGM																										
0	None																										
1	NM sets of IGM																										
2	NM groups of (IGM, then NTC*3)																										
3	IGM, then NM sets of NTC*3																										
5	Same as -5.																										
6	Problem restart dump from unit DUMPIT for problem continuation																										
Input Source Q(NMQ,NTC,3)	S(E)		Number of entries depends on option. See Sec. III.B.5.a. <table><tr><th><u>Option</u></th><th><u>Number</u></th></tr><tr><td>0</td><td>None</td></tr><tr><td>1</td><td>NMQ sets of IGM</td></tr><tr><td>2</td><td>IGM groups of NMQ sets of NTC*3</td></tr><tr><td>3</td><td>NMQ groups of (IGM, then NTC*3)</td></tr></table>	<u>Option</u>	<u>Number</u>	0	None	1	NMQ sets of IGM	2	IGM groups of NMQ sets of NTC*3	3	NMQ groups of (IGM, then NTC*3)														
<u>Option</u>	<u>Number</u>																										
0	None																										
1	NMQ sets of IGM																										
2	IGM groups of NMQ sets of NTC*3																										
3	NMQ groups of (IGM, then NTC*3)																										
Boundary Sources BR(NPP,JT,MM) BB(NPP,NUPB,MM) BT(NPP,NDNT,MM) BL(NPP,JT,MM)	S(E)		The right, bottom, top, and left boundary sources (flux) are read in that order, one group at a time. For IBR, IBB, IBT, or IBL = +3, NPP = 2. For IBR, IBB, IBT, or IBL = -3, NPP = 1. See Sec. III.B.5.a.																								
WGT(MM)	S(E)	MM	S_N quadrature weights. <u>Enter only if ISN = 0.</u>																								
COSMU(MM)	S(E)	MM	S_N μ cosine directions. <u>Enter only if ISN = 0.</u>																								
COSETA(MM)	S(E)	MM	S_N η cosine directions. <u>Enter only if ISN = 0.</u>																								
HY(JT)	S(E)	JT	Height of triangle bands.																								
X(NTX)	S(E)	NTX	Radial-coordinates of triangle vertices. Entered in a single array from left to right, starting with the bottom line of the bottom band and ending with the top line of the top band.																								
IDCS(NTC)	S(I)	NTC	Cross-section material identification numbers. These numbers assign a cross-section block to each triangle. If these numbers are negative, an anisotropic scattering source is calculated in that triangle, but the numbers need not be negative when ISCT > 0.																								
CHI(IGP)	S(E)	IGM	Fission fractions. Fraction of fission yield emerging in each group.																								
VEL(IGP)	S(E)	IGM	Group speeds. Used only in time-absorption calculations.																								

TABLE XI (cont)

Block Name and Dimension	Format	Number of Entries	Comments
FSG(IGM)	S(E)	IGM	Fictitious-source group constants (called β in Sec. II.C). These are limited to the interval $0 \leq \beta_g \leq 1$. <u>Do not enter if IFS = 0.</u>
MIXNUM(MS)	S(I)	MS	Numbers identifying cross-section block being mixed. See Sec. III.B.1.b. <u>Do not enter if MS = 0.</u>
MIXCOM(MS)	S(E)	MS	Numbers controlling cross-section-mixture process. See Sec. III.B.1.b. <u>Do not enter if MS = 0.</u>
MIXDEN(MS)	S(E)	MS	Mixture densities. See Sec. III.B.1.b. <u>Do not enter if MS = 0.</u>
BHT(NTC)	S(E)	NTC or 1	Buckling heights (in cm if cross sections are in barns) used to simulate z-dimension of an x-y geometry problem by adding an absorption given by

$$\sigma_{a,BHT} = \frac{\sigma_t}{3} [\pi / (BHT \sigma_t + 1.4209)]^2.$$

Here 1.4209 is twice the Milne problem extrapolation distance, and σ_t is the total cross section. For ISDB = 0, a single, whole-system buckling height is entered. For ISDB = 1, a buckling height is entered for each triangle. For BHT = 0, no buckling absorption is added.

used in the calculation. Normally sources are in units of particles/length³/solid angle/sec (the energy dependence is removed by the multigroup approximation, i.e., $fQdE$ is used, see Sec. II.B), microscopic cross sections are in units of barns $\times$ length²/cm², nuclide number densities in units of 10^{24} $\times$ number/length³, and velocities are habitually measured in units of length/ 10^{-8} sec.

With the exception of the cross sections from the code-dependent input file, all the data of Table XI is loaded by the LASL block loader using the special formats described in Sec. III.C. We

denote these formats by S(I) for integer and S(E) for floating-point numbers.

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 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. Detailed input is given in Table XII.

TABLE XII
EDIT INPUT

EDIT CONTROL INTEGERS (216)-----			EDIT 1
1	NZEDS	Number of zone edits.	
2	NPEDS	Number of point edits.	
ZONE-EDIT CONTROL INTEGERS, Enter only if NZEDS > 0 (416)-----			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.	

TABLE XII (cont)

IDCA(NCA)	S(I)	NCA	Cross-section material identification numbers for constituent activities. <u>Enter only if NCA > 0.</u>
IDMA(NMA)	S(I)	NMA	Cross-section material identification numbers for microscopic activities. <u>Enter only if NMA > 0.</u>
NEDZ(NTC)	S(I)	NTC	Zone identification numbers. These numbers assign a zone number to each triangle.
POINT-EDIT CONTROL INTEGERS, <u>ENTER ONLY IF NPEDS > 0 (216)</u> -----EDIT 3			
1	NTPE		Number of triangles to be included in the point edit.
2	NPMA		Number of microscopic activities calculated in the point edit.
IDMA(NPMA)	S(I)	NPMA	Cross-section material identification numbers for microscopic activities. <u>Enter only if NPMA > 0.</u>
NEDP(NTPE)	S(I)	NTPE	Triangle identification numbers to be included in the point edit. <u>No entries are made if NTPE = NTC;</u> the array is automatically set to the triangle i.d. numbers.

E. Output Description for a Test Problem

The TRIDENT program comes with a set of ten test problems. The problem output presented on the following pages is a problem designed to illustrate several of the TRIDENT options. Each page of the output is numbered, and we refer to these numbers in the text below.

As can be seen from the first output page, this problem is a one-group, S_2 , linearly anisotropic-scattering source problem in x-y geometry. The integer and floating-point input control data are printed on output p. 1. The remaining problem input is printed on output pp. 2 through 4.

Following the edit input on p. 4, the table of triangle x-coordinates is printed. Each node number corresponds to an identically numbered node point on the material map. From this table and the material map on p. 5, we see that the system is a 5.0-cm-high, and 0.5-cm-wide rectangle with the left side reflecting and all other boundaries vacuum. From the symbols on the material map, the orientation (up-or-down-pointing) of the first triangle on each band is known and the detailed triangular mesh may be drawn by connecting successive node points on alternate node lines, as shown.

After the material map, the fission spectrum, the velocities, and the unnormalized and normalized sources are printed.

Following the summary of convergence precisions on output p. 6, a monitor of the progress of the calculation is printed. The column headed "rebalance convergence" contains the maximum deviation from one

of any rebalance factor for the rebalance performed after each outer iteration. If the number of inner iterations in a group exceeds IITL, we print the maximum flux error at that time immediately to the right of the number of inner iterations.

Also on p. 6 is the final monitor line, followed by the balance table of pp. 6 and 7. A print of the cell-averaged scalar flux and the anisotropic components is on p. 7. The flux print starts with the top band and proceeds downward, with triangle one corresponding to the leftmost triangle in that band. The fission rate (in this case all zero) is printed on p. 8 and is the last item printed unless edit options are exercised.

The first item of the zone edit, beginning on output p. 8, is the zone map, similar to the material map, which associates each triangle with a zone number. This is followed by a table, for each group, containing for each zone the volume, net leakage, buckling absorption, total source, integral flux, average flux, and macroscopic activities. The table of relative power densities appears last.

Finally, the point edit begins on output p. 9. The point arrangement for both the upward-and downward-pointing triangles is indicated in the point arrangement map. The point flux is then printed for each triangle identification number requested for the point edit.

IV. PROGRAMMING INFORMATION

In this section we give some of the details of the TRIDENT program. The material contained in this

THIS CASE WAS PROCESSED BY THE TRIDENT CODE OF 02/16/77 ON 02/16/77
 TRIDENT TEST PROBLEM 1
 P-1 SCATTERING, 1 GROUP, DISTRIBUTED SOURCE, X-Y GEOMETRY
 LEFT REFLECTIVE AND ALL OTHER BOUNDARIES VACUUM

p. 1

0 ITH 0/1 DIRECT/ADJOINT THEORY
 1 ISCT 0/N ISOTROPIC/N-TM ORDER ANISOTROPIC SCATTER
 2 ISN SN ORDER (0/0/-) BUILT-IN/HEAD CARDS/STANDARD INTERFACE FILE,
 1 IGM NO. OF GROUPS
 0 IFS FICTITIOUS SOURCE OPTION (0/1) STANDARD SN/SPHERICAL HARMONICS
 5 JT NUMBER OF BANDS
 1 IBL 0/1/-3/+3 VACUUM/REFLECTIVE/FLAT SOURCE/SHAPED SOURCE LEFT BOUNDARY CONDITION
 0 IBR 0/1/2/-3/+3/4 VACUUM/REFLECTIVE/SPECIAL/FLAT SOURCE/SHAPED SOURCE/WHITE RIGHT BOUNDARY CONDITION
 0 IBB 0/1/-3/+3/4 VACUUM/REFLECTIVE/FLAT SOURCE/SHAPED SOURCE/WHITE BOTTOM BOUNDARY CONDITION
 0 IBT 0/1/-3/+3/4 VACUUM/REFLECTIVE/FLAT SOURCE/SHAPED SOURCE/WHITE TOP BOUNDARY CONDITION
 0 IEVT 0/1/2/3 SOURCE/K-EFF/ALPHA/CONCENTRATION CALCULATION
 -1 ISTART -6/-5/-3/-2/-1/0/1/2/3/5/6 STARTING OPTIONS (SEE MANUAL)

1 IGEOM GEOMETRY OPTION (1/2) X-Y/R-Z GEOMETRY
 2 MT TOTAL NUMBER OF MATERIALS
 0 MTPS NUMBER OF MATERIALS FROM LIBRARY
 2 MCR NUMBER OF MATERIALS FROM CARD INPUT
 0 MS NUMBER OF MIXTURE INSTRUCTIONS
 3 IMT ROW OF TOTAL CROSS SECTION
 4 IMS ROW OF SELF-SCATTER CROSS SECTION
 4 IMM LENGTH OF CROSS SECTION TABLE
 1 IQOPT 0/1/2/3/5 SOURCE INPUT OPTIONS
 0 IQAN 0/N ISOTROPIC/N-TM ORDER ANISOTROPIC SOURCE
 0 IPVT 0/1/2 NONE/K-EFF/ALPHA PARAMETRIC EIGENVALUE TYPE
 100 IITL MAXIMUM NUMBER OF INNER ITERATIONS

4 MM TOTAL NUMBER OF DIRECTIONS
 1 MNPQ MAXIMUM NUMBER OF DIRECTIONS PER QUADRANT
 1 MNEL MAXIMUM NUMBER OF ETA LEVELS IN A QUADRANT
 1 IACC 0/1 SYSTEM/FINE MESH REBALANCE OPTION
 0 ITLIM 0/N NO/N SECONDS TIME LIMIT
 1 IEDOPT 0/1 NO/YES EDIT INPUT,
 -0 I1 0/1 YES/NO FULL INPUT FLUX PRINT
 -0 IFP 0/1/2 ALL/ISOTROPIC/NONE FINAL FLUX PRINT
 -0 I3 0/1/2 ALL/MIXED/NONE CROSS SECTION PRINT
 -0 I4 0/1 YES/NO FINAL FISSION PRINT
 -0 I5 0/1/2/3 ALL/UNNORMALIZED/NORMALIZED/NONE INPUT SOURCE PRINT
 1 ISDB 0/1 WHOLE SYSTEM/SPACE DEPENDENT BUCKLING HEIGHT
 -0 IFO 0/1 NO/YES STANDARD INTERFACE FILE OUTPUT

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

1.000E-04 EPSO CONVERGENCE PRECISION
 1.000E-00 NCRM NORMALIZATION AMPLITUDE
 1.000E-00 PCD PARAMETER OSCILLATION DAMPER

INPUT TRI / BAND 5
 ALL ENTRIES = 10
 INPUT BAND TYPES 5
 2 1 2 1 2

p. 2

STORAGE REQUIRED ALLOWED
 SMALL CORE 1914 30000
 LARGE CORE 1644 250000

LTYS 60 ITFX 450 LTRF 22
 LTED 450 ITQS 450 LTAR 50
 LTFLO 150 ITQD 50 LTFSA 0

4421 LFNAMP
 630 LFNAP

***** IOISK = 4 *****

FLUX BLOCKING	X-SECTION BLOCKING	SOURCE BLOCKING	FIC. SOURCE BLOCKING
NFAB = 1			
NFXBX = 1			
NFXQYL = 1			
NGFXBX = 1	NGXSAX = 1	NGQBX = 1	NGFSX = 0
NGFXH = 1	NGXSB = 1	NGQR = 1	NGFS = 0

***** INPUT CROSS SECTIONS *****
 1 LOADED FROM CARDS P=10 MATERIAL (MOMENT 0)
 2 LOADED FROM CARDS P=10 MATERIAL (MOMENT 1)

INPUT CROSS SECT 1

GROUP 1
 1 .500000E+00
 2 0.
 3 .150000E+01
 4 .100000E+01

INPUT CROSS SECT 2

GROUP 1
 1 0.
 2 0.
 3 0.
 4 .833333E+00

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

ISOTROPIC COMPONENT

INPUT FLUX ESHAPE 1
ALL ENTRIES = 0.

***** INPUT FOR SOURCE *****

ISOTROPIC COMPONENT

INPUT SOURCE SHAPE 1
ALL ENTRIES = 1.0000E+00

***** QUADRATURE CONSTANTS (S= 2) *****

QUADRANT 1

	MU	ETA	WEIGHT
1	.57735030E+00	.57735030E+00	.25000000E+00

QUADRANT 2

	MU	ETA	WEIGHT
1	.57735030E+00	-.57735030E+00	.25000000E+00

QUADRANT 3

	MU	ETA	WEIGHT
1	-.57735030E+00	-.57735030E+00	.25000000E+00

QUADRANT 4

	MU	ETA	WEIGHT
1	-.57735030E+00	.57735030E+00	.25000000E+00

INPUT RAND WIDTH 5
ALL ENTRIES = 1.0000E+00

INPUT X COORDINATES 36

0.	1.0000E-01	2.0000E-01	3.0000E-01	4.0000E-01	5.0000E-01	0.	1.0000E-01	2.0000E-01	3.0000E-01
4.0000E-01	5.0000E-01	0.	1.0000E-01	2.0000E-01	3.0000E-01	4.0000E-01	5.0000E-01	0.	1.0000E-01
2.0000E-01	1.0000E-01	4.0000E-01	5.0000E-01	0.	1.0000E-01	2.0000E-01	3.0000E-01	4.0000E-01	5.0000E-01
0.	1.0000E-01	2.0000E-01	3.0000E-01	4.0000E-01	5.0000E-01				

INPUT CROSS SEC ID 50
ALL ENTRIES = -1

INPUT FISSION CPFC 1
ALL ENTRIES = 1.0000E+00

INPUT VELOCITIES 1
ALL ENTRIES = 1.0000E+00

INPUT BUCKLING KTS 50

ALL ENTRIES = 0.

p. 4

EDIT INPUT

1 NUMBER OF ZONE EDITS
1 NUMBER OF POINT EDITS

ZONE EDIT NUMBER 1

1 NUMBER OF ZONES
0 NUMBER OF CONSTITUENT ACTIVITIES
0 NUMBER OF MICROSCOPIC ACTIVITIES
0 ZONE NORMALIZATION ID

INPUT ZONE IDS. 50
ALL ENTRIES = 1

POINT EDIT NUMBER 1

50 NUMBER OF TRIANGLES
0 NUMBER OF MICROSCOPIC ACTIVITIES

POINT EDIT OVER ALL 50 TRIANGLES

TABLE OF TRIANGLE X-COORDINATES BOTTOM NODES OF BAND TO LEFT, TOP NODES OF BAND TO RIGHT

NODE	BAND	1	BAND	2	BAND	3	BAND	4	BAND	5
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	
2	1.0000E-01	1.0000E-01	1.0000E-01	1.0000E-01	1.0000E-01	1.0000E-01	1.0000E-01	1.0000E-01	1.0000E-01	
3	2.0000E-01	2.0000E-01	2.0000E-01	2.0000E-01	2.0000E-01	2.0000E-01	2.0000E-01	2.0000E-01	2.0000E-01	
4	3.0000E-01	3.0000E-01	3.0000E-01	3.0000E-01	3.0000E-01	3.0000E-01	3.0000E-01	3.0000E-01	3.0000E-01	
5	4.0000E-01	4.0000E-01	4.0000E-01	4.0000E-01	4.0000E-01	4.0000E-01	4.0000E-01	4.0000E-01	4.0000E-01	
6	5.0000E-01	5.0000E-01	5.0000E-01	5.0000E-01	5.0000E-01	5.0000E-01	5.0000E-01	5.0000E-01	5.0000E-01	

***** FINE MESH MATERIAL MAP *****
 DIRECTION IN WHICH FIRST TRIANGLE POINTS U=UP D=DOWN
 BOUNDARY CONDITION CHARACTERS (-=VACUUM OR SOURCE) (=REFLECTING) (*SPECIAL) (=WHITE)
 LEFT NUMBER INDICATES Y-COORDINATE

\$\$\$\$\$ INPUT COMPLETED \$\$\$\$\$\$

GROUP	FISSION SPECTRUM	VELOCITIES
1	1.000000E+00	1.000000E+00
2	1.000000E+00	0.

----- UNNORMALIZED SOURCE GROUP 1 -----
 DISTRIBUTED SOURCE MOMENT 0 ALL 1.0000000E+00

----- NORMALIZED SOURCE GROUP 1 -----

DISTRIBUTED SOURCE MOMENT 0

ALL 4.00000000E-01

p. 6

EPSC 1.00E-04 XLAX 1.00E-03
 EPSI 1.00E-04 XLAM 5.00E-01
 EPSD 0. XLAL 1.00E-02
 EPSY 1.00E-04 PCO 1.00E+00
 NORM 1.00E+00

ITTI 99

TIME IN MINUTES	OUTER ITERATIONS	INNER ITERATIONS TOTAL/ BY GROUP	NEUTRON BALANCE	EIGENVALUE	EIGENVALUE SLOPE	LAMBDA	REBALANCE CONVERGENCE
8.19E-03	0	0	0.	0.	0.	0.	0.
3.02E-03		1					
1.24E-02	1	6	-8.79262259E-07	0.	0.	1.00000000E+00	1.96285698E-05

 *FINAL DUMP TAKEN ON UNIT

DMPONE

TIME IN MINUTES	OUTER ITERATIONS	INNER ITERATIONS TOTAL/ BY GROUP	NEUTRON BALANCE	EIGENVALUE	EIGENVALUE SLOPE	LAMBDA	REBALANCE CONVERGENCE
2.41E-02	2	2	-2.45200171E-07	0.	0.	1.00000000E+00	1.96285698E-05

FINAL PRINTING

BALANCE TABLE, VOLUME = .2500000E+01

GRP	INH. SOURCE	FISSION SOURCE	IN SCATTER	SELF SCATTER	OUT SCATTER
1	.1000000E+01	0.	0.	.6372856E+00	.2309264E-13

APP	ABSORPTION	LEAKAGE	TOTAL R. RATE	BALANCE
1	.3186428E+00	.6813594E+00	.9559285E+00	-.2218611E+05
2	.3186428E+00	.6813594E+00	.9559285E+00	-.2218611E+05

FINAL PRINTING

FLUX COMPONENTS FOR GROUP 1

ISOTROPIC COMPONENT

BAND	TRIANGLE 1	TRIANGLE 2	TRIANGLE 3	TRIANGLE 4	TRIANGLE 5	TRIANGLE 6	TRIANGLE 7	TRIANGLE 8
5	2.33741E-01	2.67607E-01	2.29547E-01	2.60353E-01	2.19897E-01	2.47200E-01	2.04235E-01	2.20069E-01
4	2.90004E-01	2.49308E-01	2.89910E-01	2.89460E-01	2.80680E-01	2.77837E-01	2.66174E-01	2.60662E-01
3	2.92507E-01	2.61752E-01	2.88338E-01	2.85976E-01	2.79087E-01	2.75089E-01	2.64705E-01	2.58964E-01
2	2.97014E-01	2.93674E-01	2.91915E-01	2.87536E-01	2.81661E-01	2.76220E-01	2.66082E-01	2.59624E-01
1	2.66310E-01	2.34079E-01	2.62092E-01	2.27768E-01	2.51747E-01	2.15881E-01	2.35257E-01	1.97949E-01

COMPONENT 2

BAND	TRIANGLE 1	TRIANGLE 2	TRIANGLE 3	TRIANGLE 4	TRIANGLE 5	TRIANGLE 6	TRIANGLE 7	TRIANGLE 8
5	4.59349E-03	1.35356E-02	2.09616E-02	3.41329E-02	3.82245E-02	5.57081E-02	5.70585E-02	7.88335E-02
4	8.56671E-03	1.73721E-02	3.42961E-02	4.35139E-02	4.03425E-02	6.96742E-02	8.68421E-02	9.60986E-02
3	8.47444E-03	1.48935E-02	3.37931E-02	4.24209E-02	5.95431E-02	6.84136E-02	8.59900E-02	9.51390E-02
2	8.35802E-03	1.70722E-02	3.39531E-02	4.28348E-02	5.98822E-02	6.88855E-02	8.67849E-02	9.54883E-02
1	7.86551E-03	1.16483E-02	2.78284E-02	2.82218E-02	4.92422E-02	4.57747E-02	7.16221E-02	6.50244E-02

COMPONENT 3

BAND	TRIANGLE 1	TRIANGLE 2	TRIANGLE 3	TRIANGLE 4	TRIANGLE 5	TRIANGLE 6	TRIANGLE 7	TRIANGLE 8
5	7.82205E-02	1.03140E-01	2.09616E-02	3.41329E-02	3.82245E-02	5.57081E-02	5.70585E-02	7.88335E-02
4	1.14040E-01	1.23124E-01	3.42961E-02	4.35139E-02	4.03425E-02	6.96742E-02	8.68421E-02	9.60986E-02
3	1.13388E-01	1.22861E-01	3.37931E-02	4.24209E-02	5.95431E-02	6.84136E-02	8.59900E-02	9.51390E-02
2	1.13738E-01	1.22917E-01	3.39531E-02	4.28348E-02	5.98822E-02	6.88855E-02	8.67849E-02	9.54883E-02
1	9.54812E-02	8.66941E-02	2.78284E-02	2.82218E-02	4.92422E-02	4.57747E-02	7.16221E-02	6.50244E-02

COMPONENT 9

BAND	TRIANGLE 1	TRIANGLE 2	TRIANGLE 3	TRIANGLE 4	TRIANGLE 5	TRIANGLE 6	TRIANGLE 7	TRIANGLE 8
5	3.68842E-02	1.44736E-02	3.60736E-02	1.54502E-02	3.50563E-02	1.61644E-02	3.34925E-02	1.70039E-02
4	-7.11044E-04	-2.87094E-03	-7.50537E-04	-2.00974E-03	-4.17100E-04	-1.59636E-03	-8.30231E-04	-1.02015E-03
3	7.33074E-05	-7.90180E-05	6.50771E-05	2.39722E-05	5.45953E-05	1.44242E-04	1.78318E-05	2.40935E-04
2	3.32244E-05	9.63090E-04	1.89318E-03	8.17778E-04	1.33440E-03	6.4684E-04	7.14680E-04	4.73702E-04
1	-1.63051E-02	-3.58888E-02	-1.61281E-02	-3.55397E-02	-1.63266E-02	-3.47767E-02	-1.67991E-02	-3.38023E-02

COMPONENT 10

BAND	TRIANGLE 1	TRIANGLE 2	TRIANGLE 3	TRIANGLE 4	TRIANGLE 5	TRIANGLE 6	TRIANGLE 7	TRIANGLE 8
5	3.29807E-02	1.77984E-02	3.60736E-02	1.54502E-02	3.50563E-02	1.61644E-02	3.34925E-02	1.70039E-02
4	-7.43941E-04	-3.85432E-04	-7.50537E-04	-2.00974E-03	-4.17100E-04	-1.59636E-03	-8.30231E-04	-1.02015E-03
3	-4.00240E-05	2.95404E-04	6.50771E-05	2.39722E-05	5.45953E-05	1.44242E-04	1.78318E-05	2.40935E-04
2	8.95601E-05	2.89154E-04	1.89318E-03	8.17778E-04	1.33440E-03	6.4684E-04	7.14680E-04	4.73702E-04
1	-1.74107E-02	-3.29470E-02	-1.61281E-02	-3.55397E-02	-1.63266E-02	-3.47767E-02	-1.67991E-02	-3.38023E-02

FISSION EDIT

ALL FISSIONS EQUAL TO 0.

..... ZONE EDIT NUMBER 1.....

SEE MATERIAL MAP FOR TABLE OF TRIANGLE X-COORDINATES

p. 8

DIRECTION IN WHICH FIRST TRIANGLE POINTS U=UP D=DOWN

BOUNDARY CONDITION CHARACTERS (-=VACUUM OR SOURCE) (*=REFLECTING) (.=SPECIAL) (.=WHITE)

LEFT NUMBER INDICATES Y-COORDINATE

GROUP 1

ZONE	VOLUME	NET LEAKAGE	BUCKLING ABS	SOURCE	INTEGRAL FLUX	AVERAGE FLUX
1	.2500000E+01	.6813594E+00	0.	.1000000E+01	.6372856E+00	.2549143E+00

MACROSCOPIC ACTIVITIES

X.SFC.

POS	ZONE	
1	3.186428E+01	
2	0.	
3	9.559285E+01	

ALL GROUPS

ZONE	VOLUME	NET LEAKAGE	BUCKLING ABS	SOURCE	INTEGRAL FLUX	AVERAGE FLUX
1	.2500000E+01	.6813594E+00	0.	.1000000E+01	.6372856E+00	.2549143E+00

MACROSCOPIC ACTIVITIES

X.SFC.

POS	ZONE	
1	3.186428E+01	
2	0.	
3	9.559285E+01	

RELATIVE

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32

[illegible]

section is designed to help in the future local modification of the program. Much supplementary information is provided by the program comment cards.

A. Program Structure

1. Role and Function of Subprograms

We describe in Table XIV the function of all the subprograms in TRIDENT. In this table we follow the overlay structure defined in Table XIII. The overlay is of the simple control, input, execute and output kind. Note that on IBM machines with large central memories it may be more efficient to remove the substructure of OVERLAY (2,0).

2. Program Flow

In Fig. 15 we show an amplified version of Fig. 10, depicting the overall flow of the TRIDENT program and showing the names of key subroutines.

3. Relation of Problem Variables and Program Mnemonics

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

4. Definition of Variables in Common Blocks

Tables XVI through XXVII define the variables stored in blank common blocks IA, and RIA, and the named common blocks of TRIDENT. The container array, A, for problem data is also in blank common. Block IA contains input fixed-point parameters, first-word addresses of data stored in the A array, and integer data generated by the program. Block RIA contains input floating-point parameters and floating-point data generated by the program. Block HIA, in blank common, contains Hollerith information concerning the current block of problem data being loaded.

5. TRIDENT Storage Hierarchy

A storage scheme is used in TRIDENT that uses small-core memory (SCM), large-core memory (LCM), and random disk for storage of the largest required arrays. Specifically four arrays are used in this storage scheme that can most easily be visualized as

FLUX(NM,NTF,IGM),
Q(NMQ,NTF,IGM),
C(LTXS,IGM), and
FSM(NFSM,NTF,IGM).

Fig. 15. Amplified form of flow diagram in Fig. 10 showing subroutine names in overall program logic.

TABLE XIII
STRUCTURE OF THE TRIDENT PROGRAM^a

<u>Overlay (0,0)</u>	<u>Overlay (1,0)</u>	<u>Overlay (2,0)</u>	<u>Overlay (3,0)</u>
<u>TRIDENT</u>	<u>INPUT10</u>	<u>GRIND20</u>	<u>OUTPT30</u>
1. OFFUGO	1. LOAD	1. REBAL	1. <u>OUTPT31</u>
2. MONITR	2. FEBYTE	2. <u>GRIND21</u>	a. BALTEL
3. ERROR	3. <u>INPUT11</u>	a. INITIAL	b. FINAL
4. VOLM	a. JBLX	b. INITQ	2. <u>OUTPT32</u>
5. GENPBI	b. NJBLX	c. INITF	a. EDCALL
6. CLEAX	c. DUMPRD	3. <u>GRIND22</u>	b. GENFLO
7. CLEAR	d. DMPDKA	a. OUTER	c. ZEDIT
8. CLEAH	e. KSET	b. INNER	d. EDMAP
9. MPLY	4. <u>INPUT12</u>	c. ABSORB	e. PEDIT
10. PRINTS	a. CSPREP	d. NEWAB	f. AREMAP
11. DUMPER	b. IFINKS	e. SWEEP	g. CONVRT
12. DSKDMP	5. <u>INPUT13</u>	f. FLXSOL	3. <u>IFOUT</u>
13. DMPDSK	a. READQF	g. SETBC	a. IFRITE
14. DREED	b. YFINQF	4. <u>GRIND23</u>	
15. BANDID	6. <u>INPUT14</u>	a. TESTS	
16. LREED	a. GENALP	b. NEWPAR	
17. CONVT	b. SNCON		
18. LSS	c. SORTQS		
19. DOTPRO	d. IFINSN		
20. NAFIX	e. PNGEN		
21. STNAA	f. FSARY		
22. SECONI	g. FACT		
23. SREED	h. YKLGEN		
24. MCRED	i. NUPSP		
25. MDRED	7. <u>INPUT15</u>		
26. FILLU	a. POLXY		
27. FILPU	b. IXCHK		
28. DOPCA	c. BCCHK		
29. LDOPCL	d. HYCHK		
30. DOPCTP	e. MAPPER		
31. SEEK			
32. REED			
33. SUBRD			
34. SUBWR			
35. SHUTOF			
36. OPENRD			
37. DOPC			
38. CRED			
39. DRED			
40. DRIT			
41. LTSSHD			
42. TIMER			

These are implementations of
Committee on Computer Code
Coordination¹⁷ requirements.

^aOn IBM 360 with large bulk core storage, it may be computationally more efficient to remove the substructure of OVERLAY (2,0), locating GRIND21, GRIND22, and GRIND23 in GRIND20. For the IBM version, variable names of more than six characters are shortened to six. The underlined names are overlay segment names.

TABLE XIV
FUNCTION OF TRIDENT SUBROUTINES

<u>(Overlay) Subroutine</u>	<u>Function</u>
(0,0) <u>TRIDENT</u>	Main driver of program. Initializes program parameters; calls input, calculation, and output overlays; and attempts to execute sequential problems.
1. OFFUGO	Finalizes files before exiting.
2. MONITR	Prints resume of convergence parameters, monitor line headings, and outer-iteration monitor data.
3. ERROR	Prints messages, including error notices.
4. VOLM	Generates triangle volumes, and geometric constants.
5. GENPBI	Generates geometric constants to calculate leakages in cylindrical geometry.
6. CLEAX	Sets a fixed-point Small-Core Memory (SCM) block to a constant.
7. CLEAR	Sets a floating-point SCM block to a constant.
8. CLEAH	Sets a Hollerith A6 block to a constant.
9. MPLY	Multiplies an array by a constant.
10. PRINTS	Prints floating-point arrays.
11. DUMPER	Writes a restart dump.
12. DSKDMP	Dumps random disk and/or Large-Core Memory (LCM) to external unit.
13. DMPDSK	Returns dump information from external unit to random disk and/or LCM.
14. DREED	Control routine for all random disk/LCM transfers.
15. BANDID	Generates IBAND array, used in REBAL and MAPPER to obtain identification number of adjacent triangle.
16. LREED	Computes origins and vector positions for all LCM/SCM transfers.
17. CONVT	Converts m-digit integer to character representation, used in MAPPER.
18. LSS	Linear-systems-of-equations solver. Called from FSARY to solve Eq. (22).
19. DOTPRO	Calculates dot products. Used by LSS.
20. NAFIX	Obtains a fixed-point value from a floating-point vector given origin and position.
21. STNAA	Stores fixed-point value in floating-point array.
22. SECONI	Initializes time function and obtains current time and date.
23. SREED	Control routine for binary operations other than random disk/LCM and LCM/SCM data transfers.
24. MCRED	Control routine for all LCM/SCM transfers.
25. MDRED	Control routine for random disk/LCM transfers.
26. FILLU	Provides logical file information for DOPC.
27. FILPU	Provides physical file information for DOPC.
28. DOPCA	Provides output and terminal units for DOPC, and SEEK series routines.
29. LDOPCL	Finds length of an array in units of disk sectors.
30. DOPCTP	Prints random disk tabular information provided by DOPC.
31. SEEK	Assigns unit names and numbers for sequential binary files.
32. REED	Performs all binary information transfers with the exception of random disk/LCM and LCM/SCM transfers.
33. SUBRD	Performs binary read of sequential files.
34. SUBWR	Performs binary write of sequential files.
35. SHUTOF	Closes a binary device.
36. OPENRD	Renames a physical device.
37. DOPC	Assigns random disk space.
38. CRED	Performs LCM/SCM data transfers.
39. DRED	Performs random disk to LCM data transfers.

TABLE XIV (cont)

<u>(Overlay) Subroutine</u>	<u>Function</u>
40. DRIT	Performs LCM to random disk data transfers.
41. LTSSHD	Notifies when random disk transfer is complete.
42. TIMER	Prepares system time and date.
(1,0) <u>INPUT10</u>	Calls five sections of this overlay.
1. LOAD	Los Alamos data loader.
2. FERRYTE	Converts 4-byte floating-point word to 8-byte.
(1,1) <u>3. INPUT11</u>	Reads basic parameter data, checks it, calculates storage locations in the container array and in bulk storage, prints basic parameter data. Calls DUMPRD.
a. JBLX	Calculates the number of bands of the one-group flux block that will fit in fast memory.
b. NJBLX	Calculates the number or words of the one-group flux block that will fit in fast memory.
c. DUMPRD	Reads restart dump.
d. DMPDKA	Defines random disk file.
e. KSET	Calculates integers NTC, NTX, NUPB, NDNT, and indexing arrays KC and KX.
(1,2) <u>3. INPUT12</u>	Clears cross section, flux, and flow bulk storage. Calls CSPREP.
a. CSPREP	Reads cross sections in standard LASL format, FIDO format or from interface files by calling IFINXS. Prints cross sections, performs adjoint transpositions and reversals, checks cross sections, and calculates effective absorptions.
b. IFINXS	Interface input of cross sections from standard file ISOTXS.
(1,3) <u>4. INPUT13</u>	Calls READQF to read source or flux guess.
a. READQF	Reads source or flux guess from cards or calls IFINQF to read source or flux guess from interface files. Prints flux if required.
b. IFINQF	Reads source or flux from standard-interface file.
(1,4) <u>5. INPUT14</u>	Reads S_N constants and generates spherical-harmonic functions.
a. GENALP	Generates angular derivative (α) coefficients.
b. SNCON	Provides built-in library of S_N constants, S_2 through S_{16} .
c. SORTQS	Sorts input quadrature points into required sequence.
d. IFINSN	Interface input of S_N constants from standard file SNCONS.
e. PNGEN	Generates spherical-harmonic polynomials $R_\ell^k(\mu, \phi)$.
f. FSARY	Generates coefficients needed for the ray-effect-fix-up option.
g. FACT	Computes the factorial of an integer.
h. YKLGEM	Generates normalized spherical-harmonic polynomials, $Y_\ell^k(\mu, \phi)$.
i. NUFFS	Stores/fetches variables from a two-dimensional fixed-point matrix treating length of the first dimension as real 8-byte.
(1,5) <u>6. INPUT15</u>	Reads data specified by input parameters, including edit information. Calls POLXY and MAPPER.
a. POLXY	Computes geometric quantities needed in x-y geometry.
b. IXCHK	Checks that radial coordinates form a monotonically increasing sequence on each band.
c. BCCHK	Checks that left or right boundaries are vertical if they are reflecting, or white.
d. HYCHK	Checks that band heights are symmetric about center line if special boundary condition is used.
e. MAPPER	Draws material map of system.
(2,0) <u>GRIND20</u>	Calls overlay (2,1), (2,2), and (2,3). Controls outer iteration process.
1. REBAL	Calculates rebalance factors for both inner and outer iterations, for either fine-mesh or whole-system options. Called by INNER and TESTS.
(2,1) <u>2. GRIND21</u>	Calls INITIAL, INITQ, and INITF.
a. INITIAL	Performs adjoint reversals of fission spectrum and group speeds, mixes cross sections, initializes fission spectrum, initializes source by call to INITQ, generates total cross-section array, calculates fission source by call to INITF.

TABLE XIV (cont)

<u>(Overlay) Subroutine</u>	<u>Function</u>
b. INITQ	Generates source rebalance information, normalizes sources and related quantities if necessary, prints sources if desired.
c. INITF	Calculates and normalizes fission array.
(2,2) 3. GRIND22	Calls OUTER
a. OUTER	Calculates source to group and rebalance source, calls INNER, calculates new fission rate.
b. INNER	Calculates total source to group, sweeps over angles, performs sweep of spatial mesh by call to SWEEP, checks for convergence of inner iterations, applies rebalance factors calculated by REBAL, accumulates outer-iteration-rebalance arrays.
c. ABSORB	Computes total-absorption array for inner rebalance.
d. NEWAB	Accumulates effective-absorption array for outer rebalance.
e. SWEEP	Performs sweep over the spatial mesh a band at a time, solves for angular flux on each triangle, stores outgoing flux for reflecting and special boundary conditions by call to STORBC, stores angular flux on last inner iteration.
f. FLXSOL	Solves for the angular flux on the three triangle vertices given the source, cross section, and triangle dimensions.
g. SETBC	Multiple entry-point routine to treat boundary conditions on the band sweeps.
(2,3) 4. GRIND23	Calls TESTS.
a. TESTS	Calculates rebalance factors for outer iteration acceleration, makes k_{eff} reductions, computes neutron balance, checks for outer convergence, applies rebalance factors and calls NEWPAR for eigenvalue searches.
b. NEWPAR	Computes new parameters for implicit eigenvalue search.
(3,0) OUTPT30	Controls prints, interface-file output, and edits. Takes final dump.
(3,1) 1. OUTPT31	Calls BALTBL and FINAL.
a. BALTBL	Generates and prints neutron-balance table at problem completion.
b. FINAL	Prints scalar flux and moments and fission rate.
(3,2) 2. OUTPT32	Calls EDCALL.
a. EDCALL	Generates net leakage by call to GENFLO, performs zone edit by call to ZEDIT, performs point edit by call to PEDIT.
b. GENFLO	Generates triangle net leakages for use in ZEDIT.
c. ZEDIT	Calculates zone-balance tables, macroscopic activities, and power densities, calculates constituent and microscopic activities if desired.
d. EDMAP	Draws zone edit map of system.
e. PEDIT	Prints point scalar flux and moments from angular flux, computes microscopic activities if desired.
f. ARRMAT	Draws map of point arrangement on a triangle.
g. CONVRT	Converts two-digit integer to character representation. Used in ARRMAT.
(3,3) 3. IFOUT	Creates interface-output files.
a. IFRITE	Writes or prints interface-file data.

TABLE XV

RELATION OF KEY PROBLEM VARIABLES TO PROGRAM MNEMONICS

<u>Program Mnemonic</u>	<u>Problem Variable</u>	<u>Refer to</u>
1. P(N,M)	$R_{\ell}^k(\mu_m, \phi_m)$	Eq. (3)
2. C(Row, Nuclide)	$\sigma_a, \nu\sigma_f, \sigma_t, \sigma_{s,\ell}h+g$	Sect. III.B.1
3. WGT(M)	w_m	Sect. II.B.2
4. COSMU(M)	μ_m	Sect. II.B.2

TABLE XV (cont)

<u>Program Mnemonic</u>	<u>Problem Variable</u>	<u>Refer to</u>
5. COSETA(M)	η_m	Sec. II.B.2
6. FS(N,M)	Fictitious-Source Polynomials	Table II
7. YI(M,M')	$a_{m\ell}^k$	Eq. (22)
8. Y(M,M')	$Y_{\ell}^k(\eta_m, \eta_m)$	Eq. (22)
9. CTOT(I)	$\sigma_{t,i}$	Eq. (10)
10. F(I)	f_i	Eq. (35)
11. AF(K,I)	ψ_i^k	Eq. (28)
12. S(N,K,I)	Source to a group	Sec. II.B.2
13. FLUX(N,K,I)	Flux moments	Sec. II.B.2
14. FISSA(K,I)	Fission source	Sec. II.B.2
15. FSM(N,K,I)	Derivatives of flux moments used in conjunction with FS to form fictitious source	Eqs. (21) and (26)

TABLE XVI

CONTENTS OF BLANK COMMON BLOCKS IA AND RIA

A. COMMON BLOCK IA

<u>Position</u>	<u>Name</u>	<u>Pointer for Array</u>	<u>Remarks</u>
1	ITH		Theory
2	ISCT		Scattering order
3	ISN		Source of S_N constants indicator and S_N order
4	IGM		Number of groups
5			Not used
6	JT		Number of triangle bands
7	IBL		Left-boundary specification
8	IBR		Right-boundary specification
9	IBB		Bottom-boundary specification
10	IBT		Top-boundary specification
11	IEVT		Eigenvalue-type specification
12	ISTART		Flux-input option
13	MT		Total number of materials
14	MIN		Total number of input nuclides from both library and cards
15	MS		Number of mixture instructions
16	IHT		Position in table of total cross section
17	IHS		Position in table of self-scatter cross section
18	IHM		Cross-section table length
19	IQOPT		Source-input options
20	IQAN		Distributed-source-anisotropy order
21			Not used
22	IACC		Rebalance-option indicator
23	IPVT		k_{eff} or alpha-parametric-eigenvalue indicator

TABLE XVI (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for Array</u>	<u>Remarks</u>
24	ISDB		System/fine-mesh-buckling indicator
25	IGEOM		Geometry type
26	IITL		Maximum number of inner iterations
27			Not used
28			Not used
29	MNPQ		Maximum number of directions in a quadrant
30	MNEL		Maximum number of eta levels in a quadrant
31	IEDOPT		Edit-option indicator
32			Not used
33			Not used
34	IFS		Fictitious-source indicator
35	NFSM		Number of flux-moment derivatives in fictitious source
36	NYIM		Number of fictitious-source flux moments
37			Not used
38	NERROR		Input-error indicator
39	NEXTER		Next-case indicator
40	LENDMP		Approximate length of the dump
41	NXDMP		Next-dump-unit indicator
42	LENAF		Approximate length of the scratch angular-flux file
43			Not used
44	IFXERR		Integer print value for subroutine ERROR
45	JLDER		Current subroutine LOAD field number or count when exceeded
46	INLDER		Current subroutine LOAD card number
47	NCLDER		Current subroutine LOAD number of required entries in a block
48			Not used
49			Not used
50			Not used
51	IUP		IHS-IHT-1 (upscatter indicator)
52	IHF		IHT-1 (position of ω_f in cross-section table)
53	IHA		IHT-2 (position of σ_a in cross-section table)
54	IHTR		IHT-3 (position of σ_{tr} transport cross section in cross-section table, if present)
55	IHNN		IHT-4 (position of $\sigma_{n,2n}$ cross section in cross-section table, if present)
56			Not used
57	MM		Number of quadrature directions
58	NM		$((ISCT+1)*(ISCT+2))/2$, number of anisotropic components of flux
59	NMQ		$((IQAN+1)*(IQAN+2))/2$, number of anisotropic-source components
60			Not used
61			Not used
62	IGP		Sum IGM+1
63	NTC		Total number of triangles in system
64	ITMAX		Maximum number of triangles on one band
65	NTX		Total number of x-coordinates
66	NTF		Total number of space points
67			Not used

TABLE XVI (cont)

Position	Name	Pointer for Array	Remarks
68	NUPB		Number of upward-pointing triangles on bottom band
69	NDNT		Number of downward-pointing triangles on top band
70	NMM		Product NM*MM
71			Not used
72	IHMT		Product IHM*MT
73			Not used
74	ISCP		Sum ISCT+1
75	LNBT		Length of BT array
76	LNBB		Length of BB array
77	LNBL		Length of BL array
78	LNBR		Length of BR array
79	ICLIM		Length of the C and AAJ blocks
80	LIT	IT(JT)	Number of triangles on each band
81	LITT	ITT(JT)	Indicator for orientation of first triangle on each band
82	LC	C(IHM,MT)	Cross sections for a group
83	LKC	KC(JT+1)	Pointer for first triangle i.d. on each band
84	LKX	KX(JT+2)	Pointer for first x-coordinate on each node line
85	LIB	IBAND(NTC)	Pointer for adjacent triangle on band above or below
86			Not used
87			Not used
88	LQ	Q(NM,3,NTC)	Distributed source
89	LX	X(NTX)	Radial coordinates of triangle vertices
90	LAE	AE(ITMAX+1)	Area elements for a single band
91	LAS	AS(ITMAX+1)	Sign of AE times 1.0
92	LAF	AF(3,ITMAX+2)	Angular flux on a single band
93	LBT	BT(LMBT,NDNT,MM)	Top-boundary-angular-flux array, actual length is LNBT
94	LBB	BB(LMBB,NUPB,MM)	Bottom-boundary-angular-flux array, actual length is LNBB
95	LBL	BL(LMBL,JT,MM)	Left-boundary-angular-flux array, actual length is LNBL
96	LBR	BR(LMBR,JT,MM)	Right-boundary-angular-flux array, actual length is LNBR
97	LBLR	BLR(2,MNPFQ)	Left-reflecting-boundary-angular-flux array
98	LFSG	FSG(IGM)	Fictitious-source group constants (called β in Eq. (20))
99	LFS	FS(NFSM,MM)	Fictitious-source constants (see Table II)
100	LYI	YI(NYIM,MM)	Constants to calculate fictitious-source angular moments (see p. 23)
101	LFSM	FSM(NFSM,3,ITMAX)	Current flux-moment derivatives to calculate fictitious source
102	LFSMA	FSMA(NFSM,3,ITMAX)	Previous flux-moment derivatives to calculate fictitious source
103			Not used
104			Not used
105	LFL	FLUX(NM,NTF)	Angular moments of angular flux
106	LFLA	FLUXA(NTC)	Scalar flux from previous inner iteration
107			Not used
108	LFIS	FISS(NTF)	Same origin as FISSA
109	LFISA	FISSA(NTF)	Fission source
110			Not used
111	LDC	IDCS(NTC)	Cross-section identification number for each triangle

TABLE XVI (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for Array</u>	<u>Remarks</u>
112			Not used
113			Not used
114			Not used
115			Not used
116			Not used
117	LPI	PI(3)	Geometric constants used for triangle volume averaging
118	LHY	HY(JT)	Band widths
119	LW	WGT(MNPQ,4)	Direction weights
120	LCM	COSMU(MNPQ,4)	Radial-direction cosines
121	LCE	COSETA(MNPQ,4)	Axial-direction cosines
122	LHAF	HAF(ISN/2,3*ITMAX)	Half-angle angular flux for angular derivative
123			Not used
124	LP	P(NM,MM)	Spherical-harmonic polynomials
125	LPBI	PBI(6,ITMAX)	Geometric constants used to calculate neutron flow from a triangle
126			Not used
127			Not used
128	LMN	MIXNUM(MS)	Input mixture numbers (conditional on MS.GT.0)
129	LMC	MIXCOM(MS)	Input mixture instructions (conditional on MS.GT.0)
130	LMD	MIXDEN(MS)	Input mixture densities (conditional on MS.GT.0)
131	LF	F(NTC)	Fine-mesh rebalance factors
132			Not used
133			Not used
134			Not used
135	LFLO	FLO(3,NTC)	Fine-mesh inward flows
136	LAB	AB(NTC)	Fine-mesh absorption removal rate
137	LQQ	QQ(NTC)	Fine-mesh source
138			Not used
139			Not used
140	LBNDI	BNDI(MNPQ,2* ((ITMAX+1)/2))	Incoming (right to left) angular-flux array on band boundary
141	LBNDI	BNDI(MNPQ,2* ((ITMAX+1)/2))	Outgoing (left to right) angular-flux array on band boundary
142	IFP		Final-flux-print indicator (0/1/2 = all/isotropic/none)
143			Not used
144	LHA	HA(ITMAX)	Used in subroutine REBAL for inversion
145	LGA	GA(ITMAX)	Used in subroutine REBAL for inversion
146	LQG	QG(IGP)	Space integral of boundary source
147	LQGB	QGB(IGP)	Total boundary source by group
148	LE1	E1(MNPQ)	Axial directional cosines in quadrant 1
149	LE2	E2(MNPQ)	Axial directional cosines in quadrant 2
150	LE3	E3(MNPQ)	Axial directional cosines in quadrant 3
151	LE4	E4(MNPQ)	Axial directional cosines in quadrant 4
152	LU1	U1(MNPQ)	Radial directional cosines in quadrant 1
153	LU2	U2(MNPQ)	Radial directional cosines in quadrant 2
154	LU3	U3(MNPQ)	Radial directional cosines in quadrant 3
155	LU4	U4(MNPQ)	Radial directional cosines in quadrant 4
156	LBALTB	BALTB(9,IGP)	Balance-table array

TABLE XVI (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for Array</u>	<u>Remarks</u>
157	LBAL	BAL(IGP)	Group neutron balance
158	LCHI	CHI(IGP)	Input fission spectrum
159	LCHIA	CHIA(IGP)	Fission spectrum used in the calculation
160	LVEL	VEL(IGP)	Group velocities
161	LW1	W1(MNPQ)	Quadrature weights in quadrant 1
162	LW2	W2(MNPQ)	Quadrature weights in quadrant 2
163	LW3	W3(MNPQ)	Quadrature weights in quadrant 3
164	LW4	W4(MNPQ)	Quadrature weights in quadrant 4
165	LSOU	SOURCE(NM,NTF)	Total source in a group (same origin as Q)
166	LTMED	(see subroutine EDCALL)	Pointer for temporary array used in edits
167	LAA	AAJ(MT)	Effective absorption cross section
168	LBHT	BHT(NTC)	Space-dependent buckling height
169	NEL1		Number of eta levels in quadrant 1
170	NEL2		Number of eta levels in quadrant 2
171	NEL3		Number of eta levels in quadrant 3
172	NEL4		Number of eta levels in quadrant 4
173	LNUP	NUP(MNEL,4)	Number of directions on an eta level
174	LCTOT	CTOT(NTC)	Effective total cross section
175	JCONV		Final-convergence indicator
176	LWE	WE(MNPQ,4)	Quadrature weight times eta
177	LWM	WM(MNPQ,4)	Quadrature weight times mu
178			Not used
179	ICNT		Iteration trigger used in NEWPAR
180			Not used
181			Not used
182			Not used
183			Not used
184			Not used
185	NGO		Return indicator set in NEWPAR
186			Not used
187			Not used
188	IITNO		Inner iteration number
189			Not used
190	G		Number of current group (integer)
191	ICONV		Secondary-convergence indicator
192	NGOTO		Return indicator set in TESTS
193			Not used
194			Not used
195	IITOT		Total number of inner iterations
196			Not used
197			Not used
198			Not used
199	IFN		Fission calculation indicator set in INITIAL
200	OITNO		Outer iteration number
201	NPQ1		Number of directions in quadrant 1

TABLE XVI (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for Array</u>	<u>Remarks</u>
202	NPQ2		Number of directions in quadrant 2
203	NPQ3		Number of directions in quadrant 3
204	NPQ4		Number of directions in quadrant 4
205	LMBT		First dimension of BT array
206	LMBB		First dimension of BB array
207	LMBL		First dimension of BL array
208	LMBR		First dimension of BR array
209	LNUP1	NUP1(MNEL)	Number of directions on an eta level in quadrant 1
210	LNUP2	NUP2(MNEL)	Number of directions on an eta level in quadrant 2
211	LNUP3	NUP3(MNEL)	Number of directions on an eta level in quadrant 3
212	LNUP4	NUP4(MNEL)	Number of directions on an eta level in quadrant 4
213	LAL1	AL1(4*MNPQ)	r-z geometry angular-derivative constants
214	LAL2	AL2(4*MNPQ)	r-z geometry angular-derivative constants
215	NTAF		Computed by subroutine KSET, not used in code
216			Not used
217	LAL11	AL11(MNPQ)	r-z geometry angular-derivative constants in quadrant 1
218	LAL21	AL21(MNPQ)	r-z geometry angular-derivative constants in quadrant 1
219	LAL12	AL12(MNPQ)	r-z geometry angular-derivative constants in quadrant 2
220	LAL22	AL22(MNPQ)	r-z geometry angular-derivative constants in quadrant 2
221	LAL13	AL13(MNPQ)	r-z geometry angular-derivative constants in quadrant 3
222	LAL23	AL23(MNPQ)	r-z geometry angular-derivative constants in quadrant 3
223	LAL14	AL14(MNPQ)	r-z geometry angular-derivative constants in quadrant 4
224	LAL24	AL24(MNPQ)	r-z geometry angular-derivative constants in quadrant 4
225			Not used
226			Not used
227			Not used
228			Not used
229			Not used
230			Not used
231	IGXSP		Previous cross-section group block
232	IGFXP		Previous flux group block
233	IGQP		Previous source group block
234	IGFSP		Previous fictitious-source group block
235	ISNRD		ISN with sign as input
236			Not used
237			Not used
238			Not used
239			Not used
240			Not used
241			Not used
242			Not used
243			Not used
244			Not used
245			Not used
246			Not used

TABLE XVI (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for Array</u>	<u>Remarks</u>
247			Not used
248	MCRRDS		Signed number of input nuclides requested from cards
249	NOSGUP		Sigma-up included in cross-sections indicator
250	IOLYCS		Overlay indicator (first digit gives primary overlay and second digit gives secondary overlay when read in octal)

B. COMMON BLOCK RIA

<u>Position</u>	<u>Name</u>	<u>Pointer for Array</u>	<u>Remarks</u>
1	EV		Eigenvalue guess
2	EVM		Eigenvalue modifier
3	PV		Parametric value of k_{eff} or alpha
4	XLAL		Search-lambda-lower limit
5	XLAH		Search-lambda-upper limit
6	XLAX		Fine-mesh-search precision
7	EPSO		Outer-convergence precision
8	EPSI		Inner-convergence precision = EPSO
9	EPSR		Within-group-rebalance-convergence precision = 10^{-4}
10	EPSX		Whole-system-rebalance-convergence precision = EPSO
11	NORM		Normalization amplitude
12	POD		New parameter modifier
13	SUMMUO		Weighted sum of mu over outward ($\mu > 0$) directions
14	SUMETU		Weighted sum of eta over upward ($\eta > 0$) directions
15	SUMETD		Weighted sum of eta over downward ($\eta < 0$) directions
16			Not used
17			Not used
18			Not used
19			Not used
20	TN2N		N,2N reaction term used in balance equation
21	XLAPP		Value of lambda from sequence of outer iterations previous to that of XLAP
22	XLAP		Value of lambda from previous sequence of outer iterations
23	E2		Temporary storage
24	E1		Temporary storage
25	EVPP		Eigenvalue from cycle of outer iterations previous to that of EVP
26	EVP		Eigenvalue from previous sequence of outer iterations
27	E4		Temporary storage
28	ALAR		Value of lambda from previous outer iteration
29	TS		Total integrated source to a group
30	E3		Temporary storage
31	EVS		Slope used in eigenvalue search
32	ALA		Parameter lambda
33	TIN		Time at the last dump (also used as temporary storage)
34	FT		Total space-integrated fission source in a group
35	FTP		Previous fission total
36			Not used
37			Not used

TABLE XVI (cont)

<u>Position</u>	<u>Name</u>	<u>Pointer for Array</u>	<u>Remarks</u>
38			Not used
39			Not used
40	ERR		Scalar-flux error from comparison with previous flux
41	FLTERR		Floating-point print value for subroutine ERROR
42	TIMACC		Accumulated problem running time
43	TIMOFF		Floating-point form of the input fixed-point parameter ITLIM
44	TIMSLD		Elapsed time since last dump
45			Not used
46			Not used
47			Not used
48			Not used
49			Not used
50			Not used
51			Not used
52			Not used
53			Not used
54			Not used
55			Not used
56			Not used
57			Not used
58			Not used
59			Not used
60	CZERO		0.0, IBM double precision
61	CONE		1.0, IBM double precision
62	CTWO		2.0, IBM double precision
63	CHALF		0.5, IBM double precision
64	CTWOPI		2π , IBM double precision
65	CPI		π , IBM double precision
66	RTWOPI		$1/(2\pi)$, IBM double precision
67			Not used
68			Not used
69			Not used
70			Not used
71			Not used
72			Not used

TABLE XVII

CONTENTS OF NAMED COMMON BLOCK FWBGN1

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
1	IDUSE	A vector used for the title of the problem in A4 format (length of 18 words)
2	LAST	Length of common block A used by the current problem
3	LASTEC	Length of LCM used by the current problem
4	IGCDMP	Group number of restart dump
5	IPSO	LCM pointer of the source-to-the-group block which is calculated by subroutine

TABLE XVII (cont)

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
		OUTER and used by subroutine INNER
6	LTSO	Length of the source-to-the-group block
7	IPFX	LCM pointer for the first group of the flux block
8	LTFX	Length of the flux block for a group
9	IPXS	LCM pointer for the first group of the cross-section block (each group block contains the cross-section array, the effective-absorption vector and the spatial total cross-section array)
10	LTXS	Length of the cross-section block for a group
11	IPQS	LCM pointer for the first group of the Q-source block
12	LTQS	Length of the Q-source block for a group (zero when not a Q-calculation)
13	IEREC	Not used
14	I4	Final-fission-print indicator (0/1 = yes/no)
15	IFO	Interface-output indicator (0/1 = no/yes)
16	LUFFS	Random disk logical unit number for fictitious source
17	LUQS	Random disk logical unit number for inhomogeneous source
18	LUXS	Random disk logical unit number for cross sections
19	LUFY	Random disk logical unit number for flux moments
20	LUALL	Random disk group unit number for all random sets
21	IPBS	LCM pointer for the first group of the boundary source block
22	IPAB	LCM pointer for the absorption block used in the outer rebalance
23	LTAB	Length of the absorption block used in the outer rebalance
24	IPFLO	LCM pointer for the inward-flows block used in the outer rebalance
25	LTFLO	Length of the inward-flows block used in the outer rebalance
26	IPQQ	LCM pointer for the group-summed Q-source block used in the outer rebalance
27	LTQQ	Length of the group-summed Q-source block used in the outer rebalance
28	ITYPFY	Random-disk-flux-type indicator (Is flux blocked in LCM by group or by bands?)
29	IDISK	Disk-configuration indicator (see Table XXVII)
30	NFQB	For IDISK=0, this is the number of SCM flux-band blocks making up one LCM band block. For IDISK>0, this is the number of SCM flux-band blocks making up one energy group (normally equal to one) (see Table XXVII)
31	LJBK	LCM pointer for array (JBK(NFQB)) of number of bands in each SCM flux-band block
32	LNTB	LCM pointer for array (NTB(NFQB)) of number of triangles in each SCM flux-band block
33	NFXBXL	Number of LCM flux-band blocks making up one energy group. If IDISK>0, NFXBXL=1.
34	NGQBX	Number of inhomogeneous-source LCM-group blocks needed for the entire array of IGM groups (if IDISK>3, NGQBX=1) (see Table XXVII)
35	NGQB	Number of energy groups comprising an inhomogeneous-source LCM-group block (if IDISK>3, NGQB=IGM) (see Table XXVII)
36	NGFXBX	Number of flux LCM-group blocks needed for the entire array of IGM groups (if IDISK>1, NGFXBX=1) (see Table XXVII)
37	NGXSB	Number of energy groups comprising a cross-section LCM-group block. If IDISK>2, NGXSB=IGM (see Table XXVII)
38	NGXSBX	Number of cross-section LCM-group blocks needed for the entire array of IGM groups. If IDISK>2, NGXSBX=1. (see Table XXVII)

TABLE XVII (cont)

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
39	NFXBX	Number of SCM flux-bands blocks making up one energy group
40	NGFXB	Number of energy groups comprising a flux LCM-group block. If IDISK>1, NGFXB=IGM. (see Table XXVII)
41	LNSMFQ	Number of words comprising an SCM flux-band block
42	LTBF	Length of boundary-source block for a group

TABLE XVIII

CONTENTS OF NAMED COMMON BLOCK FWBCN2

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
1	TIMBDP	Minimum time between periodic dumps (0/F = no periodic dumps/seconds)
2	LENRIA	Length of common block RIA which must be saved for restarts
3	MAXLEN	Maximum length of the main common data block A
4	MAXECS	Maximum length of LCM storage available to the problem
5	LENMCB	Length of the named common block FWBCN1 which must be saved for restarts
6	LENCIA	Length of the common parameter block IA which must be saved for restarts
7	LENCSSR	Length of named common block FICSR which must be saved for restarts
8	IRCOVY	Recall overlay indicator for CDC machine usage (0/6HRECALL = no reloading of overlay when in core/reloading overlay when in core)
9	I1	Full-flux-guess-input-print indicator (indicator effective when ISTART is 2 or -2, 0/1 = yes/no)
10	I3	Cross-section-print indicator (0/1/2 = all/mixed/none)
11	I5	Input-source-print indicator (0/1/2/3 = all/unnormalized/normalized/none)

TABLE XIX

CONTENTS OF NAMED COMMON BLOCK LOCAL

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
1	ITLIM	Fixed-point-time-problem removal value (0/N = no/seconds)
2	MCR	Number of nuclides requested from the cards (minus if FIDO format)
3	MTP	Number of library nuclides requested from ISOTXS interface file (MTPS*(ISCT+1))
4	MTPS	Number of nuclide sets requested from ISOTXS interface file (each set yields ISCT+1 blocks)
5	NISOXS	Number of isotopes in the set as read from the ISOTXS file
6	LMTP	Core pointer for the ISOTXS interface-file position-number data block
7	LIMIT	Maximum storage length required by input cross sections and calculation of anisotropic-scattering coefficients
8	LENCLR	Length of the partial block to be cleared during input

TABLE XX
CONTENTS OF NAMED COMMON BLOCK UNITS

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
1	NINP	Problem-code-dependent-decimal-input unit number
2	NOUT	Problem-decimal-output unit number
3	NTTY	Terminal-output unit number
4	IDECOD	Encode/decode unit number
5	IFNOVY	Overlay-file unit number
6	NOTINF	Working-interface unit number
7	NOTEDT	Edit-scratch unit number
8	NOTAF	Angular-flux-scratch unit number
9	NLEAK	Triangle-net-leakages by group unit number

TABLE XXI
CONTENTS OF NAMED COMMON BLOCK TSWEPT

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
1	BTA1	Indicator for incoming/outgoing flux across side 1
2	BTA2	Indicator for incoming/outgoing flux across side 2
3	BTA3	Indicator for incoming/outgoing flux across side 3
4	GAM1	Angular-derivative constant
5	GAM2	Angular-derivative constant
6	IUD	Sweep-up-or down indicator
7	I	Triangle index
8	J	Band index
9	M	Direction index
10	K	Eta-level index
11	IRL	Sweep-right-or left-indicator
12	INP	Number of triangles in previously swept SCM flux band block
13	JFS	IFS+1 (IFS is IA(34))

TABLE XXII
CONTENTS OF NAMED COMMON BLOCK RSWEPT

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
1	X1	Axial coordinate of triangle vertex 1
2	X2	Axial coordinate of triangle vertex 2
3	X3	Axial coordinate of triangle vertex 3
4	BT1	Indicator for incoming/outgoing flux across side 1
5	BT2	Indicator for incoming/outgoing flux across side 2
6	BT3	Indicator for incoming/outgoing flux across side 3
7	FS1	Fictitious source for triangle vertex 1
8	FS2	Fictitious source for triangle vertex 2
9	FS3	Fictitious source for triangle vertex 3
10	GALP	Fictitious-source group constant
11	ID	Triangle-identification number

TABLE XXIII
CONTENTS OF NAMED COMMON BLOCK JSWEEP

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
1	DX	Geometric-constant array
2	FO	Array containing incoming boundary flux
3	DY	Band height
4	S1	Source at triangle vertex 1
5	S2	Source at triangle vertex 2
6	S3	Source at triangle vertex 3
7	ST	Triangle total cross section
8	AHX	2* triangle area
9	U	Radial direction cosine
10	E	Axial direction cosine
11	IR	Rebalance-zone-identification number

TABLE XXIV
CONTENTS OF NAMED COMMON BLOCK TBAL

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
1	WSR	Whole-system/fine-mesh rebalance indicator
2	W	Quadrature weight
3	I5	Flow indicator for rebalance flows
4	I6	Flow indicator for rebalance flows

TABLE XXV
CONTENTS OF NAMED COMMON BLOCK FICSR

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
1	IDFAL	Not used
2	LTFS	Twice the length of one group of the fictitious-source flux-moment-derivative array
3	LTFSA	Length of one group of the fictitious-source flux-moment-derivative array
4	NGFS	Number of groups in a fictitious-source LCM block
5	NGFSX	Number of fictitious-source LCM blocks
6	NDSFS	Number of random-disk sectors comprising fictitious source array
7	IDSFS	Random-disk pointer of fictitious-source array
8	IPFSM	LCM pointer for current fictitious-source flux-moment-derivative array
9	IPFSMA	LCM pointer for previous fictitious-source flux-moment-derivative array
10	LFSBN	Length of one band of the fictitious-source flux-moment-derivative array
11	IGGFS	Fictitious-source LCM group-block indicator

TABLE XXVI
CONTENTS OF NAMED COMMON BLOCK INST

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
1	MULTR	Number of words required for a real variable (2 on IBM)
2	MULTH	Number of words required for a six Hollerith variable (2 on IBM)
3	LENSCT	Length of scratch file for local SEEK assignment
4	LENEDT	Length of edit file for local SEEK assignment
5	LEDDMP	Delta length to be added to computed length of dump file for local SEEK assignment
6	LEDINF	Delta length to be added to computed length of output interface files for local SEEK assignment
7	IVERST	Current version number of scratch file
8	IVERED	Current version number of edit file
9	IVERD1	Current version number of first dump file
10	IVERD2	Current version number of second dump file

TABLE XXVI (cont)

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
11	IVERAF	Current version number of scratch-angular-flux file
12	JRECAF	Current record number of scratch-angular-flux file
13	IVERLK	Current version number of scratch-leakage file

TABLE XXVII

CONTENTS OF NAMED COMMON BLOCK SEEKIT

<u>Position</u>	<u>Name</u>	<u>Contents and Remarks</u>
1	FNREF	Array containing generic names of scratch, dump, interface files
2	FNLOC	Local names of scratch, dump, interface files
3	HOLOUT	Array used for output of binary identification record of SEEK class files
4	HOLIN	Array used for input of binary identification record of SEEK class files

These are respectively the angular-flux moments, inhomogeneous source, if any, cross-section block and fictitious source, if any. The arguments of the arrays are:

- NM - number of angular flux spherical-harmonic moments needed to generate scattering source
- NMQ - number of inhomogeneous-source spherical-harmonic moments
- NTF - number of spatial points in the problem
- IGM - number of energy groups in the problem
- LTXS - length of one group cross-section block
- NFSM - number of fictitious-source terms ($NFSM=2*ISN$).

The storage strategy is depicted in Table XXVIII. The storage configuration corresponding to TWOTRAN-II is the IDISK=5 column (except TWOTRAN-II does not have a fictitious-source option and, hence, does not have the FSM array). In this case, random disk is not needed since all arrays will fit in LCM. We do, however, allow the case where one flux group will not fit in SCM. The NPBL is the number of spatial points for which the array will fit. This parameter is, at most, NTF and at least the maximum number of space points on a triangle band. NPBL is restricted to be in units of bands. That is, the flux for any band is not allowed to be divided between blocks. Although for, simplicity it is not indicated in the Table, the second argument of the FLUX array in LCM may be required to be greater than NTF. This occurs

if $NTF/NPBL$ is not an exact integer. The FLUX array is stored in LCM in blocks of size $LNSMFQ=NM*NPBL$ and the last block will be partially filled with zeroes if NPBL does not divide into NTF evenly.

When IDISK=4, the four arrays will not all fit into LCM. Then, the fictitious-source array is placed on random disk and space is allocated in LCM for as many fictitious-source energy groups (NGFS) as the container space will accommodate. Note that $1 \leq NGFS \leq IGM$. IDISK will only be four if the fictitious-source option is used. When IDISK=3, the three multigroup arrays and the fictitious-source array for one group will not all fit in LCM. The code then determines the maximum number of inhomogeneous-source groups (NGQB) that will fit into LCM. The inhomogeneous-source array is then stored on random disk, along with the fictitious-source array. Although for simplicity it is not indicated in the Table, the third argument of the Q array on random disk may be required to be greater than IGM. This occurs if $IGM/NGQB$ is not an exact integer. The Q array is stored on random disk in blocks of size $NMQ*NTF*NGQB$ and the last block will be partially filled with zeroes if NGQB does not divide into IGM evenly. This is true of all of the multigroup arrays of Table XXVIII.

As the IDISK parameter approaches zero, other arrays are placed on random disk. Finally, when IDISK=0, one flux group array will not fit in LCM. We then include in LCM as many SCM band blocks (NB) as will fit. Note that TRIDENT requires that $NB \geq 1$. If $NB < 1$, a diagnostic will be printed and, if possible, the SCM/LCM container sizes must be adjusted.

TABLE XXVIII
TRIDENT BULK-STORAGE SCHEME

Storage Container	IDISK =0	IDISK=1	IDISK=2	IDISK=3	IDISK=4	IDISK=5
SCM	FLUX(NM,NPBL,1)	FLUX(NM,NPBL,1)	FLUX(NM,NPBL,1)	FLUX(NM,NPBL,1)	FLUX(NM,NPBL,1)	FLUX(NM,NPBL,1)
	C(LTXS,1)	C(LTXS,1)	C(LTXS,1)	C(LTXS,1)	C(LTXS,1)	C(LTXS,1)
	Q(NMQ,NTF,1)	Q(NMQ,NTF,1)	Q(NMQ,NTF,1)	Q(NMQ,NTF,1)	Q(NMQ,NTF,1)	Q(NMQ,NTF,1)
	FSM(NFSM,3*ITMAX,1)	FSM(NFSM,3*ITMAX,1)	FSM(NFSM,3*ITMAX,1)	FSM(NFSM,3*ITMAX,1)	FSM(NFSM,3*ITMAX,1)	FSM(NFSM,3*ITMAX,1)
LCM	FLUX(NM,NB*NPBL,1)	FLUX(NM,NTF,NGFXB)	FLUX(NM,NTF,IGM)	FLUX(NM,NTF,IGM)	FLUX(NM,NTF,IGM)	FLUX(NM,NTF,IGM)
	C(LTXS,1)	C(LTXS,1)	C(LTXS,NGXSB)	C(LTXS,IGM)	C(LTXS,IGM)	C(LTXS,IGM)
	Q ^a (NMQ,NTF,1)	Q ^a (NMQ,NTF,1)	Q ^a (NMQ,NTF,1)	Q ^a (NMQ,NTF,NGQB)	Q ^a (NMQ,NTF,IGM)	Q ^a (NMQ,NTF,IGM)
	FSM(NFSM,NTF,1)	FSM(NFSM,NTF,1)	FSM(NFSM,NTF,1)	FSM(NFSM,NTF,1)	FSM(NFSM,NTF,NGFS)	FSM(NFSM,NTF,IGM)
Random Disk	FLUX(NM,NTF,IGM)	FLUX(NM,NTF,IGM)	C(LTXS,IGM)	Q ^b (NMQ,NTF,IGM)	FSM(NFSM,NTF,IGM)	
	C(LTXS,IGM)	C(LTXS,IGM)	Q ^b (NMQ,NTF,IGM)	FSM(NFSM,NTF,IGM)		
	Q ^b (NMQ,NTF,IGM)	Q ^b (NMQ,NTF,IGM)	FSM(NFSM,NTF,IGM)			
	FSM(NFSM,NTF,IGM)	FSM(NFSM,NTF,IGM)				

^aEven though only NMQ inhomogeneous source moments are needed, we allocate LCM space for NM moments for programming convenience.

^bEven though only NMQ inhomogeneous source moments are needed, we allocate random disk space for NM moments for programming convenience.

The calculations required to determine the array locations between SCM, LCM, and random disk are performed in INPUT11. The encoded procedure for determining these allocations is as follows. When all required arrays other than the flux moments are allocated in SCM, the remaining SCM container size is computed. The maximum number of bands of the one-group flux array that will fit in SCM (NPBL) is determined and the appropriate space allocated. Likewise after all required arrays are allocated in LCM, the remaining container space is allocated to the four arrays in the priority indicated in Table XXVIII. Thus IDISK, NPBL, NB, NGQB, NGXSB, NGFS, and NGFXB are computed based on the given SCM and LCM container sizes, MAXLEN and MAXECS, respectively.

6. Machine-Dependent Subprograms

a. LCM/Random Disk and LCM/SCM Installation-Dependent Subroutines

Bulk-data transfers between LCM and random disk are controlled by the subroutines under consideration by the Committee on Computer Code Coordination (CCCC),¹⁷ DRIT/DRED, and the random-disk file assignment is controlled by the purposed CCCC routine¹⁷ DOPC. The DOPC assigned files contain a maximum of four logical files associated with one grouped file. Bulk-data transfers between LCM and SCM are controlled by the purposed CCCC routines¹⁷ CRIT/CRED.

b. Other Installation-Dependent Routines

TRIDENT requires the use of CCCC subroutines⁷ SEEK/RITE/REED for the assignment and transfer of sequential binary information. Further, date and current time must be provided by the CCCC subroutine⁷ TIMER.

B. Decimal Data File

TRIDENT assigns the decimal files as indicated in Table XXIX. The units are assigned in the main code and may be changed there.

C. Binary Data File

TRIDENT uses three classes of binary files which may be divided into system assigned, DOPC required assigned, and SEEK assigned. The system assigned class contains only the file having the absolute code overlay on the CDC 7600. This unit is not used in the IBM version of the code. The unit number is associated with the variable named IFNOVY. The DOPC-required assigned class contains a maximum of four individual files associated with

TABLE XXIX
DECIMAL DATA FILES

<u>Variable Name</u>	<u>Unit</u>	<u>Remarks</u>
NINP	2	Decimal input file (unit 5 is assigned in IBM version).
NOUT	3	Decimal output file (unit 6 is assigned in IBM version).
NTTY	59	Terminal output file (unit 6 is assigned in IBM version).
IDECOD	63	Encode/decode file (routines CONVI/CONVO, available at the Argonne Code Center, are used in place of unit on IBM version).

one grouped file. The units corresponding to the files are associated with the variables LUFS, LUQS, LUXS, and LUFX. The grouped file is associated with the variable LUALL. The units are assigned values in INPUT11 and DUMPRD. The SEEK-assigned class contains scratch, standard-interface, and interface-like files. A maximum of three files will be open at once. All of the SEEK class files are made to appear similar to standard-interface files in that the first record of each is that of the file identification record.

Since a standard-interface file format for triangular flux, scalar flux, or source has not been established, we have established our own format following the standard-interface format as closely as possible.

The generic names of SEEK-assigned files are given in Table XXX.

Certain variables are used to contain the SEEK-assigned values as is noted in Table XXXI.

D. Hardware Requirements

The TRIDENT 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. CDC 8444 disk units provide peripheral storage.

E. Software Requirements

1. CDC Machines

The code was designed to operate on the CDC 7600 under the CROS operating system¹⁸ which was developed at Los Alamos. The system uses the CDC RUN compiler with a CDC optimizer attached.

Disk units provide storage for input, output, scratch, and resident files.

TABLE XXX
SEEK ASSIGNED FILES

Generic Name	Remarks
DUMPIT	Restart dump file. Only two versions are alternately used to prevent the build up of a multitude of versions. Current version of the file will be used for restart input.
EDITIT	Problem edit information. Scratch file is destroyed at the end of the problem.
RTFLUX	CCCC interface-like file. May be both input and output.
ATFLUX	CCCC interface-like file. May be both input and output.
RAFLUX	CCCC interface-like file. Problem output only.
AAFLUX	CCCC interface-like file. Problem output only.
SNCONS	CCCC interface file. May be input or output.
FIXSRC	CCCC interface-like file. Problem input only.
ISOTXS	CCCC interface file. Problem input only.
TEMPAF	Problem-angular-flux-scratch file. File is destroyed at end of problem.
NLEAK	Problem-leakage-scratch file. File is destroyed at end of problem.

2. TRIDENT for the IBM 360

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

The major change made in the conversion of the code was in the treatment of peripheral storage. The vast amount of fast core available on the IBM-

TABLE XXXI
VARIABLES FOR SEEK ASSIGNED VALUES

Variable Name	Remarks
NOTINF	Working-SEEK unit
NOTEDT	Edit-information unit
NOTAF	Angular-flux unit
NLEAK	Leakage unit

360 is one of the cheaper resources of that machine. It was thus decided to place the data normally kept in LCM (large core memory) in an expanded container array. This was accomplished by using part of the container array (indexed from 1 through MAXLEN) as small core (SCM) and the remainder as LCM (indexed starting at MAXLEN+1 and ending at MAXLEN+MAXECS). The beginning of the LCM array is passed through the initial call to DOPC. It was thus possible to keep the LCM pointer structure of the code with no change in logic. Unlike the CDC machine, the definitions of MAXLEN and MAXECS change such that their sum is then the size of the container array. The size of the container array is changed simply by changing the dimension of common block A and assigning new values to MAXLEN and MAXECS. These changes need be made only in program TRIDENT.

These changes may not be made when restarting a problem.

Because of the amount of core storage available on the IBM machines, it was found to be computationally advantageous to combine the computational overlay GRIND20 with its secondary overlays.

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

- (a) CDC ENCODE/DECODE functions, found in the subroutine LOAD were replaced by subroutine CONVI/CONVO, available at the Argonne Code Center, to obtain core-to-core conversions required for FIDO cross-section loading.
- (b) CDC mathematical subroutines (e.g., COS) were replaced by the corresponding eight-byte-mode IBM routines (e.g., DCOS). CDC in-line functions (e.g., ABS) were replaced by the corresponding eight-byte-mode IBM functions (e.g., DABS).
- (c) A small function called JFIX was included to convert an eight-byte-floating-point-mode value to a truncated integer.
- (d) The IF(EOF) CDC job termination test was replaced with a read with the IBM END parameter.
- (e) CDC overlay cards were removed, overlay programs were changed to subroutines, calls to overlay segments were replaced with calls to the subroutines, and the setting of overlay parameter IFNOVY was eliminated.
- (f) Two parameters used throughout the code (MULTR and MULTH) which express the number of four-byte words contained in floating-point numbers and six-character Hollerith data were set to two to provide the current four-byte functions to the CCCC routine.
- (g) Shortened names (six-character maximum) were required for the following routines:

<u>CDC</u>	<u>IBM-360</u>
TRIDENT	(main program)
INPUT10	INPT10
INPUT11	INPT11
INPUT12	INPT12
INPUT13	INPT13
INPUT14	INPT14
INPUT15	INPT15
GRIND20	GRND20
GRIND21	GRND21
GRIND22	GRND22
GRIND23	GRND23
OUTPT30	OUTT30
OUTPT31	OUTT31
OUTPT32	OUTT32

F. Programming Considerations

1. Storage Management

a. Variable Dimensioning

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

b. Allocation of LCM

The allocation of storage in LCM is handled in the same manner as in SCM. Most of the group-dependent arrays are stored in LCM so the dimensionality is IGM times the core requirements of the array. Four arrays, however, are stored using the automatic-disk-overflow feature discussed in Sec. IV.A.5.

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 cross-section block includes the cross sections C(IHM,MT), the effective absorption AAJ(MT) and the total cross section CTOT(NTC). The first word of this block is LC, and the last word is LQ-1.

A complete list of LCM storage assuming IDISK=5 (see Table XXVIII) is given in Table XXII.

c. Computation of Required SCM and LCM 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 the amount of SCM (LAST) and the amount of LCM (LASTEC). The computation of these two quantities is made and printed out very early in the problem execution. To obtain an estimate for LAST without running a problem, the following approximation may be used. (The approximation assumes the fictitious-source option is not used, and one flux-moment group array is in SCM):

$$\text{LAST} = \text{MT} \cdot \text{IHM} + 7 \cdot \text{JT} + 13 \cdot \text{NTC} \\ + (11 + 2 \cdot \text{MNPQ}) \cdot \text{ITMAX}$$

+ Conditional Arrays + Boundary Arrays.

The variables in the approximate formulae are defined in Table XVI. The sizes of the conditional arrays are:

NTC	for ISDB=1
3*NTC+2*ITMAX	for IACC=1
3*MNEL*ITMAX	for IGEOM=2
12*ISN*ITMAX	for IFS=1.

The sizes of the boundary array are:

NDNT*MM	for IBT=1,
2*NDNT*MM	for IBT=3,
NDNT*MM	for IBT=-3,
NUPB*MM	for IBB=1,
2*NUPB*MM	for IBB=3,
NUPB*MM	for IBB=-3,
JT*MM	for IBR=1, IBR=2, or IBL=1,
2*JT*MM	for IBR=3 or IBL=3,
JT*MM	for IBR=-3 or IBL=-3.

The amount of LCM for IDISK = 4 without the fictitious-source option is given by

$$\text{LASTEC} = \text{IGM} \cdot \text{MT} \cdot (\text{IHM} + 1) + \text{NTC} \cdot (1 + 6 \cdot \text{NM}) \\ + \text{Boundary Array} \\ + 6 \cdot \text{NM} \cdot \text{NTC} + \text{Conditional Block}$$

where the conditional block size is

$$5 \cdot \text{NTC} \text{ or } 5 \quad \text{for IACC=1 or 0.}$$

d. Computation of Required Random-Disk Storage

The simplest way to compute the random-disk storage requirement is to load the problem for a short run and let the code compute IDISK. Then, using the information in Table XXIX, the amount of disk required can be easily calculated. Another method is to compute LASTEC assuming IDISK=5 (4

TABLE XXII
LCM STORAGE PARAMETERS

<u>LCM First Word Address</u>	<u>Length per Block</u>	<u>No. of Blocks</u>	<u>Contents</u>
IPXS	LTXS = $NTC+MT*IHM+MT$	IGM	Cross-section blocks by group
IPFX	LTFX = $NM*NTF$	IGM	Scalar flux and moments
IPBS	LTBF ^a	IGM	Boundary flux or boundary source
IPSO	LTSO = $NM*NTF$	1	Source-to-group including within group scatter
IPQS	LTQS = $NM*NTF$	IGM	Q-source blocks by group, present only if IEVT=0
IPAB	LTAB = (NTC or 1)	1	Absorptions plus net leakages for (fine-mesh or whole-system) rebalance
IPQQ	LTQQ = (NTC or 1)	1	Total inhomogeneous source for (fine-mesh or whole-system) rebalance
IPFSM ^b	LTFSA = $NFSM*NTF$	1	Fictitious source
IPFSMA ^b	LTFSA = $NFSM*NTF$	IGM	Fictitious source from previous iteration

^aLTBF is the length of the boundary arrays and varies depending upon the boundary-condition options.

^bThese are used only with the fictitious-source option.

with IFS=0) according to the above formula. LASTEC is then subtracted from MAXECS, the amount of LCM memory available. If the result is nonnegative, no disk storage is needed. If the result, called LLFL, is negative, its absolute value must be compared with the array sizes of the four arrays of Table XXI. If the fictitious-source option is used and

$$LLFL \leq 6 * ISN * NTC * IGM,$$

then approximately $LTFSA * IGM$ words of disk storage are needed. If*

$$LTFSA * IGM < |LLFL| \leq NM * NTF * IGM$$

and the problem is a source problem (IEVT=0), then approximately $LTFSA * IGM + NM * NTF * IGM$ words of disk storage are needed. This procedure is continued according to the priorities of Table XXVIII for other cases. Random disk storage is arranged in units called sectors. On the CDC 7600, a sector

is 512 words. Due to the truncation processes used to fill arrays into sectors, the amount of random-disk storage is slightly greater than the above calculated values.

e. Temporary Storage Requirements

In various subroutines of TRIDENT, space is needed for temporary storage. A test is made on each occasion that a temporary storage container is needed and the user is informed if more SCM is required than is available. Usually, the problem data requirement is much larger than the temporary storage requirement. Occasionally this will not be the case. For example, the temporary input cross-section storage requirement ($IGM * IHM * MIN$ words) may sometimes exceed available SCM. In this case, more SCM space must be allocated.

f. Overstorage of Data in Core

In TRIDENT, a certain amount of over-storage is used to reduce the total amount of small-core memory (SCM) required; i.e., more than one array may reside in the same SCM locations as the problem progresses. The most important example of this is that the large arrays Q(NM,NTF) and SOURCE (NM,NTF) are stored in the same spaces. The Q array is used in OUTER to generate the total source to a group,

*Note that even though only NMQ moments of the inhomogeneous source are needed, to simplify programming we allocate space of NM moments on random disk.

including the inhomogeneous source, if any. This source is then written in LCM. In INNER this source is read into SCM after each inner iteration to initialize the SOURCE array. Then the rest of the group source is added to SOURCE.

2. Restart-Tape Composition

The restart dump is composed of the following data blocks: restart-parameter information, named common block FWBGN1, named common block FICSR, common block RIA, common block IA, data common block A (only the portion used by the problem), LCM in multiples of SCM, random disk (when present) for fictitious-source, inhomogeneous-source, cross-sections, flux, and angular-flux records. The angular-flux information is only generated in the last pass of the calculation for groups one through the current group. Each block is tailored to give the necessary information to restore the block. The final dump contains the current group (IGCDMP) value of zero.

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

1. IGM Number of groups.
2. ISFAP Final-dump indicator.
3. NORDAF Convergence indicator.
4. NTC Number of triangles.
5. NM Number of flux components.
6. NDISK Disk indicator.

3. Standard-Interface Files

The standard-interface files read and written by TRIDENT are version III files.⁷ As far as possible, the codes which process these files are subroutines. The user should note that in reading, the current version of a file is used as input, and in writing, the next version of the file is used as output of data.

REFERENCES

1. 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 (1975).
2. K. D. Lathrop, "Ray Effects in Discrete Ordinates Equations," Nucl. Sci. Eng. 32, 357 (1968).
3. W. F. Miller, Jr. and W. H. Reed, "Discrete Ordinates-to-Spherical Harmonic Conversions for Ray Effect Mitigation in x-y Geometry," Proc. Conf. Computational Methods in Nuclear Engineering, Charleston, SC (April 15-17, 1975).
4. W. F. Miller, Jr. and W. H. Reed, "Ray-Effect Mitigation Methods for Two-Dimensional Neutron Transport Theory," (to appear in March 1977 issue of Nucl. Sci. Eng.).
5. W. 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 (1973).
6. B. G. Carlson, "Tables of Symmetric Equal Weight Quadrature EQN Over the Unit Sphere," Los Alamos Scientific Laboratory report LA-4734 (1971).
7. B. M. Carmichael, "Standard Interface Files and Procedures for Reactor Physics Codes, Version-III," Los Alamos Scientific Laboratory report LA-5486-MS (February 1974).
8. "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).
9. G. I. Bell, G. E. Hansen, and H. A. Sandmeier, "Multitable Treatments of Anisotropic Scattering in S_N Multigroup Transport Calculations," Nucl. Sci. Eng. 28, 376 (1967).
10. 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.
11. K. D. Lathrop, "Remedies for Ray Effects," Nucl. Sci. Eng. 45, 255 (1971).
12. W. F. Miller, Jr., "Transport and Reactor Theory January 1 - March 31, 1976," Los Alamos Scientific Laboratory report LA-6325-PR (May 1976).
13. W. H. Reed and T. R. Hill, "Triangular Mesh Methods for the Neutron Transport Equation," Proc. Conf. on Mathematical Models and Comp. Techniques for Anal. of Nuc. Systems, Ann Arbor, Michigan (April 9-11, 1973).
14. W. F. Miller, Jr., "Coarse-Mesh Rebalance Methods Compatible with the Spherical Harmonic Fictitious Source in Neutron Transport Calculations," Los Alamos Scientific Laboratory report LA-6145-MS (November 1975).
15. W. W. Engle, Jr., "A Users Manual for ANISN, A One-Dimensional Discrete Ordinates Transport Code with Anisotropic Scattering," Union Carbide Corporation report K-1693 (March 1967).
16. K. D. Lathrop, "DTF-IV, A Fortran Program for Solving the Multigroup Transport Equation with Anisotropic Scattering," Los Alamos Scientific Laboratory report LA-3373 (November 1965).
17. J. W. Lewellen, U. S. Energy Research and Development Administration, and G. E. Bosler, Los Alamos Scientific Laboratory, personal communication, August 1976.
18. Ray Davenport, Ed., "Elementary Guide to the Control Data 7600," Programmer's Information Manual, Vol. 5A, Los Alamos Scientific Laboratory (1972).

APPENDIX
SPATIAL FINITE ELEMENT EQUATIONS

In this Appendix, we provide a detailed derivation of the spatial difference equations encoded in TRIDENT. We will proceed through the derivation in r-z geometry and simply state the results for x-y geometry.

Noting that the residual has been premultiplied by r, we begin by inserting Eq. (29) into Eq. (30) to obtain

$$\begin{aligned}
 2\pi \int_{\Delta} \int_V dv \frac{1}{r} W_j(r, z) \left\{ \mu_m \left[\frac{\partial(r \psi_{m1}(r, z))}{\partial r} + \gamma_2 \tilde{r}_2 (\psi_{mB2}(\tilde{r}_2, \tilde{z}_2) - \psi_{m1}(\tilde{r}_2, \tilde{z}_2)) \right. \right. \\
 \times \partial(r - \tilde{r}_2) + \gamma_3 \tilde{r}_3 (\psi_{m1}(\tilde{r}_3, \tilde{z}_3) - \psi_{mB3}(\tilde{r}_3, \tilde{z}_3)) \partial(r - \tilde{r}_3) \Big] + \frac{\alpha_{m+\frac{1}{2}} \psi_{m+\frac{1}{2}1}(r, z) - \alpha_{m-\frac{1}{2}} \psi_{m-\frac{1}{2}1}(r, z)}{w_m} \\
 + \eta_m \left[r \frac{\partial(\psi_{m1}(r, z))}{\partial z} + \gamma_1 \tilde{r}_1 (\psi_{m1}(\tilde{r}_1, \tilde{z}_1) - \psi_{mB1}(\tilde{r}_1, \tilde{z}_1)) \partial(z - \tilde{z}_1) \right. \\
 + \gamma_2 \tilde{r}_2 (\psi_{mB2}(\tilde{r}_2, \tilde{z}_2) - \psi_{m1}(\tilde{r}_2, \tilde{z}_2)) \partial(z - \tilde{z}_2) + \gamma_3 \tilde{r}_3 (\psi_{mB3}(\tilde{r}_3, \tilde{z}_3) - \psi_{m1}(\tilde{r}_3, \tilde{z}_3)) \partial(z - \tilde{z}_3) \Big] \\
 \left. + r \sigma \psi_{m1}(r, z) - r \bar{S}_m(r, z) \right\} = 0, \quad j = 1, 2, 3. \quad i = 1, 2, \dots, \text{NTC}. \quad (\text{A-1})
 \end{aligned}$$

The notation is defined in Sec. II.D and we again use the triangle orientation of Fig. 6. We next insert Eq. (28) into Eq. (A-1) yielding

$$\begin{aligned}
 \int_{z_1}^{z_3} dz \int_{\tilde{r}_3}^{\tilde{r}_2} dr W_j(r, z) \left\{ \mu_m \left[\sum_{k=1}^3 \psi_{m1}^k \frac{\partial(r L_1^k(r, z))}{\partial r} + \gamma_2 \tilde{r}_2 \sum_{k=2}^3 L_1^k(\tilde{r}_2, \tilde{z}_2) (\psi_{mB12}^k - \psi_{m1}^k) \partial(r - \tilde{r}_2) \right. \right. \\
 + \gamma_3 \tilde{r}_3 \sum_{k=1,3}^3 L_1^k(\tilde{r}_3, \tilde{z}_3) (\psi_{m1}^k - \psi_{mB13}^k) \partial(r - \tilde{r}_3) \Big] + \sum_{k=1}^3 L_1^k(r, z) \left(\frac{\alpha_{m+\frac{1}{2}} \psi_{m+\frac{1}{2}1}^k - \alpha_{m-\frac{1}{2}} \psi_{m-\frac{1}{2}1}^k}{w_m} \right) \\
 + \eta_m \left[r \sum_{k=1}^3 \psi_{m1}^k \frac{\partial L_1^k(r, z)}{\partial z} + \gamma_1 \tilde{r}_1 \sum_{k=1}^2 L_1^k(\tilde{r}_1, \tilde{z}_1) (\psi_{m1}^k - \psi_{mB11}^k) \delta(z - \tilde{z}_1) \right. \\
 + \gamma_2 \tilde{r}_2 \sum_{k=2}^3 L_1^k(\tilde{r}_2, \tilde{z}_2) (\psi_{mB12}^k - \psi_{m1}^k) \partial(z - \tilde{z}_2) + \gamma_3 \tilde{r}_3 \sum_{k=1,3}^3 L_1^k(\tilde{r}_3, \tilde{z}_3) (\psi_{mB13}^k - \psi_{m1}^k) \partial(z - \tilde{z}_3) \Big] \\
 \left. + r \sigma \sum_{k=1}^3 L_1^k(r, z) \psi_{m1}^k - r \bar{S}_m(r, z) \right\} = 0 \quad j = 1, 2, 3. \quad i = 1, 2, \dots, \text{NTC}. \quad (\text{A-2})
 \end{aligned}$$

In Eq. (A-2), ψ_{mB1j}^k refers to the angular flux in direction m on the triangle i , boundary j , and node k . The expression resulting from the operations in Eq. (A-2) are extremely tedious and are tabulated term-by-term. Table A-I provides the r -derivation terms for each $W_j(r, z)$. The shorthand defined in Table A-II is used in this and subsequent tables.

The z -derivative terms for each $W_j(r, z)$ are listed in Table A-III. The angular derivative, collision, and source terms appear in Tables A-IV through A-VI, respectively. Tables A-VII through A-X give expressions for rectangular geometry. The x - y geometry equation, analogous to Eq. (A-2) is

$$\begin{aligned} & \int_{y_1}^{y_3} dy \int_{x_3}^{x_1} dx W_j(x, y) \left\{ \mu_m \left[\sum_{k=1}^3 \psi_{m1}^k \frac{\partial L_1^k(x, y)}{\partial x} + \gamma_2 \sum_{k=2}^3 L_1^k(x_2, y_2) (\psi_{mB12}^k - \psi_{m1}^k) \partial(x - x_2) \right. \right. \\ & + \gamma_3 \sum_{k=1,3}^3 L_1^k(x_3, y_3) (\psi_{mB13}^k - \psi_{m1}^k) \partial(x - x_3) \left. \right] + \eta_m \left[\sum_{k=1}^3 \psi_{m1}^k \frac{\partial L_1^k(x, y)}{\partial y} \right. \\ & + \gamma_1 \sum_{k=1}^2 L_1^k(x_1, y_1) (\psi_{mB11}^k - \psi_{m1}^k) \partial(y - y_1) + \gamma_2 \sum_{k=2}^3 L_1^k(x_2, y_2) (\psi_{mB12}^k - \psi_{m1}^k) \partial(y - y_2) \\ & \left. \left. + \gamma_3 \sum_{k=1,3}^3 L_1^k(x_3, y_3) (\psi_{mB13}^k - \psi_{m1}^k) \partial(y - y_3) \right] + \sigma \sum_{k=1}^3 L_1^k(x, y) \psi_{m1}^k - \bar{S}_m(x, y) \right\} = 0 \end{aligned}$$

$$j = 1, 2, 3$$

$$i = 1, 2, \dots, \text{NTC.} \quad (\text{A-3})$$

TABLE A-I
DIFFERENCE RELATIONSHIPS, CYLINDRICAL GEOMETRY, r -DERIVATIVE TERMS

$W_j(r, z)$	Integrated Expression
1	$\frac{\mu_m \Delta z}{6} [(-\bar{\gamma}_3 D1 \psi_{m1}^1 + \bar{\gamma}_2 D2 \psi_{m1}^2 + (\bar{\gamma}_2 D3 - \bar{\gamma}_3 D5) \psi_{m1}^3 + \gamma_2 (D2 \psi_{mB12}^2 + D3 \psi_{mB12}^3) - \gamma_3 (D1 \psi_{mB13}^1 + D5 \psi_{mB13}^3)]$
$(r - \bar{r})$	$\frac{\mu_m \Delta z}{36} [(-\bar{\gamma}_3 G1 - F1) \psi_{m1}^1 + (\bar{\gamma}_2 G2 - F2) \psi_{m1}^2 + (\bar{\gamma}_2 G3 - \bar{\gamma}_3 G4 - F3) \psi_{m1}^3 + \gamma_2 (G2 \psi_{mB12}^2 + G3 \psi_{mB12}^3) - \gamma_3 (G1 \psi_{mB13}^1 + G4 \psi_{mB13}^3)]$
$(z - \bar{z})$	$\frac{\mu_m \Delta z^2}{36} [\bar{\gamma}_3 \Delta r_3 \psi_{m1}^1 + \bar{\gamma}_2 \Delta r_2 \psi_{m1}^2 + (\bar{\gamma}_2 E3 - \bar{\gamma}_3 E5) \psi_{m1}^3 + \gamma_2 (\Delta r_2 \psi_{mB12}^2 + E3 \psi_{mB12}^3) + \gamma_3 (\Delta r_3 \psi_{mB13}^1 - E5 \psi_{mB13}^3)]$

TABLE A-II
GEOMETRIC EXPRESSIONS USED IN TABLES A-I THROUGH A-VI

D1	$r_3 + 2r_1$	F2	$(r_1 + 2r_2 + r_3) \frac{3\Delta r_1}{2}$
D2	$r_3 + 2r_2$	F3	$(r_1 + r_2 + 2r_3) \frac{3\Delta r_1}{2}$
D3	$r_2 + 2r_3$	G1	$\Delta r_2^2 - \Delta r_1 E4$
D4	$r_2 + 2r_1$	G2	$\Delta r_3^2 + \Delta r_1 E6$
D5	$r_1 + 2r_3$	G3	$\Delta r_1^2 - \Delta r_3 E5$
D6	$r_1 + 2r_2$	G4	$\Delta r_1^2 + \Delta r_2 E3$
E1	$r_3 + 5r_1$	G5	$\Delta r_2^2 + \Delta r_3 E1$
E2	$r_3 + 5r_2$	G6	$\Delta r_3^2 - \Delta r_2 E2$
E3	$r_2 + 5r_3$	H1	$(r_2 - 5r_1)\Delta r_1 + r_3 \Delta r_2 + 3(r_1 + r_2)\Delta r_3$
E4	$r_2 + 5r_1$	H2	$3(r_2 + r_3)\Delta r_1 + (r_3 - 5r_1)\Delta r_2 + r_1\Delta r_3$
E5	$r_1 + 5r_3$	H3	$r_2\Delta r_1 + 3(r_1 + r_3)\Delta r_2 + (r_1 - 5r_3)\Delta r_3$
E6	$r_1 + 5r_2$	I1	$4r_1 - 2r_2 - r_3$
$\bar{\gamma}_1$	$1 - \gamma_1$	I2	$2r_1 + 4r_2 - r_3$
$\bar{\gamma}_2$	$1 - \gamma_2$	I3	$r_1 + r_2 + 8r_3$
$\bar{\gamma}_3$	$1 - \gamma_3$		
F1	$(2r_1 + r_2 + r_3) \frac{3\Delta r_1}{2}$		

TABLE A-III
DIFFERENCE RELATIONSHIPS, CYLINDRICAL GEOMETRY, z-DERIVATIVE TERMS

$W_j(r, z)$	Integrated Expression
1	$\eta_m/6 [- (\bar{\gamma}_1 \Delta r_1 D4 + \bar{\gamma}_3 \Delta r_3 D1) \psi_{m1}^1 - (\bar{\gamma}_1 \Delta r_1 D6 + \bar{\gamma}_2 \Delta r_2 D2) \psi_{m1}^2 - (\bar{\gamma}_2 \Delta r_2 D3 + \bar{\gamma}_3 \Delta r_3 D5) \psi_{m1}^3$ $- \gamma_1 \Delta r_1 (D4 \psi_{mB11}^1 + D6 \psi_{mB11}^2) - \gamma_2 \Delta r_2 (D2 \psi_{mB12}^2 + D3 \psi_{mB12}^3) - \gamma_3 \Delta r_3 (D1 \psi_{mB13}^1 + D5 \psi_{mB13}^3)]$
(r-r)	$\eta_m/36 [- \bar{\gamma}_1 \Delta r_1 G5 - \bar{\gamma}_3 \Delta r_3 G1) \psi_{m1}^1 - (\bar{\gamma}_1 \Delta r_1 G6 + \bar{\gamma}_2 \Delta r_2 G2) \psi_{m1}^2 - (\bar{\gamma}_2 \Delta r_2 G3 + \bar{\gamma}_3 \Delta r_3 G4) \psi_{m1}^3$ $- \gamma_1 \Delta r_1 (G5 \psi_{mB11}^1 + G6 \psi_{mB11}^2) - \gamma_2 \Delta r_2 (G2 \psi_{mB12}^2 + G3 \psi_{mB12}^3) - \gamma_3 \Delta r_3 (G1 \psi_{mB13}^1 + G4 \psi_{mB13}^3)]$
(z-z)	$\eta_m \Delta z / 36 [(2\bar{\gamma}_1 \Delta r_1 D4 + \bar{\gamma}_3 \Delta r_3^2 - F1) \psi_{m1}^1 + (2\bar{\gamma}_1 \Delta r_1 D6 - \bar{\gamma}_2 \Delta r_2^2 - F2) \psi_{m1}^2$ $- (\bar{\gamma}_2 \Delta r_2 E3 + \bar{\gamma}_3 \Delta r_3 E5 + F3) \psi_{m1}^3 + 2\gamma_1 \Delta r_1 (D4 \psi_{mB11}^1 + D6 \psi_{mB11}^2)$ $- \gamma_2 \Delta r_2 (\Delta r_2 \psi_{mB12}^2 + E3 \psi_{mB12}^2) + \gamma_3 \Delta r_3 (\Delta r_3 \psi_{mB13}^1 - E5 \psi_{mB13}^3)]$

TABLE A-IV
DIFFERENCE RELATIONSHIPS, CYLINDRICAL GEOMETRY, ANGULAR DERIVATIVE TERMS

$W_j(r, z)$	Integrated Expression
1	$\frac{\Delta z}{36 w_m} [\alpha_{m+k_2} (F1 \psi_{m+k_2 1}^1 + F2 \psi_{m+k_2 1}^2 + F3 \psi_{m+k_2 1}^3) - \alpha_{m-k_2} (F1 \psi_{m-k_2 1}^1 + F2 \psi_{m-k_2 1}^2 + F3 \psi_{m-k_2 1}^3)]$
$(r-\bar{r})$	$\frac{\Delta r_1 \Delta z}{360 w_m} [\alpha_{m+k_2} (H1 \psi_{m+k_2 1}^1 + H2 \psi_{m+k_2 1}^2 + H3 \psi_{m+k_2 1}^3) - \alpha_{m-k_2} (H1 \psi_{m-k_2 1}^1 + H2 \psi_{m-k_2 1}^2 + H3 \psi_{m-k_2 1}^3)]$
$(z-\bar{z})$	$\frac{\Delta r_1 \Delta z^2}{360 w_m} [\alpha_{m+k_2} (I1 \psi_{m+k_2 1}^1 + I2 \psi_{m+k_2 1}^2 - I3 \psi_{m+k_2 1}^3) - \alpha_{m-k_2} (I1 \psi_{m-k_2 1}^1 + I2 \psi_{m-k_2 1}^2 - I3 \psi_{m-k_2 1}^3)]$

TABLE A-V
DIFFERENCE RELATIONSHIPS, CYLINDRICAL GEOMETRY,
COLLISION TERMS

$W_j(r, z)$	Integrated Expression
1	$\frac{\Delta z}{36} (F1 \psi_{m1}^1 + F2 \psi_{m1}^2 + F3 \psi_{m1}^3)$
$(r-\bar{r})$	$\frac{\Delta r_1 \Delta z}{360} (H1 \psi_{m1}^1 + H2 \psi_{m1}^2 + H3 \psi_{m1}^3)$
$(z-\bar{z})$	$\frac{\Delta r_1 \Delta z^2}{360} (I1 \psi_{m1}^1 + I2 \psi_{m1}^2 - I3 \psi_{m1}^3)$

TABLE A-VI
DIFFERENCE RELATIONSHIPS, CYLINDRICAL GEOMETRY,
SOURCE TERMS^a

$W_j(r, z)$	Integrated Expression
1	$\frac{\Delta z}{36} (F1 S_{m1}^1 + F2 S_{m1}^2 + F3 S_{m1}^3)$
$(r-\bar{r})$	$\frac{\Delta r_1 \Delta z}{360} (H1 S_{m1}^1 + H2 S_{m1}^2 + H3 S_{m1}^3)$
$(z-\bar{z})$	$\frac{\Delta r_1 \Delta z^2}{360} (I1 S_{m1}^1 + I2 S_{m1}^2 - I3 S_{m1}^3)$

^aThis table refers only to inhomogeneous, scattering and fission source terms. The fictitious source involves spatial derivatives and is handled in a manner analogous to Tables A-I and A-III.

TABLE A-VII
DIFFERENCE RELATIONSHIPS, RECTANGULAR GEOMETRY, x-DERIVATIVE TERMS

$W_j(x, y)$	Integrated Expression
1	$\frac{\mu_m \Delta y}{2} [(-\bar{\gamma}_3 \psi_{m1}^1 + \bar{\gamma}_2 \psi_{m1}^2 + (\gamma_3 - \gamma_2) \psi_{m1}^3 + \gamma_2 (\psi_{mB12}^2 + \psi_{mB12}^3) - \gamma_3 (\psi_{mB13}^1 + \psi_{mB13}^3)]$
$(x-\bar{x})$	$\frac{\mu_m \Delta y}{6} [(-\Delta x_1 \gamma_3 \psi_{m1}^1 - \Delta x_1 \gamma_2 \psi_{m1}^2 - (\bar{\gamma}_2 \Delta x_3 + \bar{\gamma}_3 \Delta x_2 + \Delta x_1) \psi_{m1}^3 + \gamma_2 (\Delta x_1 \psi_{mB12}^2 - \Delta x_3 \psi_{mB12}^3)$ $+ \gamma_3 (\Delta x_1 \psi_{mB13}^1 - \Delta x_2 \psi_{mB13}^3)]$
$(y-\bar{y})$	$\frac{\mu_m \Delta y^2}{6} [(\gamma_3 - \gamma_2) \psi_{m1}^3 + \gamma_2 \psi_{mB12}^3 - \gamma_3 \psi_{mB13}^3]$

TABLE A-VIII
DIFFERENCE RELATIONSHIPS, RECTANGULAR GEOMETRY, y-DERIVATIVE TERMS

$W_j(x, y)$	<u>Integrated Expression</u>
1	$\eta_m/2 [- (\bar{\gamma}_1 \Delta x_1 + \bar{\gamma}_3 \Delta x_3) \psi_{m1}^1 - (\bar{\gamma}_1 \Delta x_1 + \bar{\gamma}_2 \Delta x_2) \psi_{m1}^2 - (\bar{\gamma}_2 \Delta x_2 + \bar{\gamma}_3 \Delta x_3) \psi_{m1}^3$ $- \gamma_1 \Delta x_1 (\psi_{mB11}^1 + \psi_{mB11}^2) - \gamma_2 \Delta x_2 (\psi_{mB12}^2 + \psi_{mB12}^3) - \gamma_3 \Delta x_3 (\psi_{mB13}^1 + \psi_{mB13}^3)]$
(x-x)	$\eta_m/6 [(\gamma_1 - \gamma_3) \Delta x_1 \Delta x_3 \psi_{m1}^1 + (\gamma_2 - \gamma_1) \Delta x_1 \Delta x_2 \psi_{m1}^2 + (\gamma_3 - \gamma_2) \Delta x_2 \Delta x_3 \psi_{m1}^3$ $- \gamma_1 \Delta x_1 (\Delta x_3 \psi_{mB11}^1 - \Delta x_2 \psi_{mB11}^2) - \gamma_2 \Delta x_2 (\Delta x_1 \psi_{mB12}^2 - \Delta x_3 \psi_{mB12}^3)$ $- \gamma_3 \Delta x_3 (\Delta x_2 \psi_{mB13}^3 - \Delta x_1 \psi_{mB13}^1)]$
(y-y)	$\eta_m \Delta y/6 [- \gamma_1 \Delta x_1 \psi_{m1}^1 - \gamma_1 \Delta x_1 \psi_{m1}^2 + (\gamma_2 \Delta x_2 + \gamma_3 \Delta x_3) \psi_{m1}^3 + \gamma_1 \Delta x_1 (\psi_{mB11}^1 + \psi_{mB11}^2)$ $- \gamma_2 \Delta x_2 \psi_{mB12}^3 - \gamma_3 \Delta x_3 \psi_{mB13}^1]$

TABLE A-IX
DIFFERENCE RELATIONSHIPS, RECTANGULAR GEOMETRY,
COLLISION TERMS

$W_j(x, y)$	<u>Integrated Expression</u>
1	$\frac{\Delta x_1 \Delta y}{6} (\psi_{m1}^1 + \psi_{m1}^2 + \psi_{m1}^3)$
(x-x)	$\frac{\Delta x_1 \Delta y}{72} [(\Delta x_3 - \Delta x_1) \psi_{m1}^1 + (\Delta x_1 - \Delta x_2) \psi_{m1}^2$ $+ (\Delta x_2 - \Delta x_3) \psi_{m1}^3]$
(y-y)	$\frac{\Delta x_1 \Delta y^2}{72} (2 \psi_{m1}^3 - \psi_{m1}^2 - \psi_{m1}^1)$

TABLE A-X
DIFFERENCE RELATIONSHIPS, RECTANGULAR GEOMETRY,
SOURCE TERMS^a

$W_j(x, y)$	<u>Integrated Expression</u>
1	$\frac{\Delta x_1 \Delta y}{6} (S_{m1}^1 + S_{m1}^2 + S_{m1}^3)$
(x-x)	$\frac{\Delta x_1 \Delta y}{72} [(\Delta x_3 - \Delta x_1) S_{m1}^1 + (\Delta x_1 - \Delta x_2) S_{m1}^2$ $+ (\Delta x_2 - \Delta x_3) S_{m1}^3]$
(y-y)	$\frac{\Delta x_1 \Delta y^2}{72} (2 S_{m1}^3 - S_{m1}^2 - S_{m1}^1)$

^aThis table refers only to inhomogeneous, scattering, and fission source terms. The fictitious source involves spatial derivatives and is handled in a manner analogous to Tables A-VII and A-VIII.