ORNL-2027

ORNL CSD 7

Users Guide to
MORSE-SGC

S.K. Fraley

BLANK PAGE

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

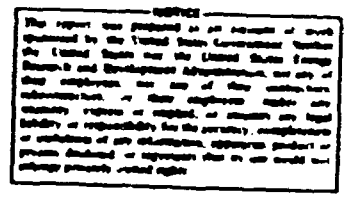
ORNL/CSD-7
UC-32 - Mathematics & Computers

Contract No. W-7405-eng-26

COMPUTER SCIENCES DIVISION

USERS GUIDE TO MORSE-SGC

S. K. Fraley



NOTE:
This work supported by
DEFENSE NUCLEAR AGENCY
Under Subtask PE 074

MARCH 1976

UNION CARBIDE CORPORATION, NUCLEAR DIVISION
operating the
Oak Ridge Gaseous Diffusion Plant • Oak Ridge National Laboratory
Oak Ridge Y-12 Plant • Paducah Gaseous Diffusion Plant
for the
ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION

TABLE OF CONTENTS

ABSTRACT

The MORSE-SGC (Super Grouped Cross Section) code is a version of the MORSE code which reduces the core size requirements to solve problems with large cross-section storage requirements. This version of the MORSE code is available with both an improved combinatorial geometry package and a KENO geometry package. Other changes have also been implemented in the code to make problem formulation more convenient.

INTRODUCTION

The MORSE-SGC (Super Grouped Cross Sections) code is a version of the MORSE[1] code which reduces the core size requirements to solve problems with large cross-section storage requirements. This is accomplished by storing the cross-section data on direct access devices and retrieving supergroups of cross section data (blocks consisting of one or more energy groups) as needed in the Monte Carlo transport calculations. The supergroup sizes are calculated by the program to fit within the core size allocated.

Input data is in the free form or fixed form FIDO[2] structure, except for the geometry input instructions which have not been changed. The cross section data is read in the AMPX[2] working library format. A program (LAVA)[2] is available to convert cross sections from the ANISN[3] format (used in the previous version of MORSE) to the AMPX working library format.

The cross-section mixing process has been changed to eliminate the requirement for all mixtures and elements to all orders of scattering (all Legendre coefficients) to be in core simultaneously. Only one order of one element and one mixture need be in core simultaneously, and the mixing is done before space is allocated to geometry data (or other large data blocks not needed for the mixing process).

MORSE-SGC is a flexibly dimensioned program. Array names are passed in subroutine calls, and blank common is not used for data storage. Use of descriptive array nam in the subroutines allows the user to follow program logic much easier than before. This feature is very desirable since the user may need to write analysis routines which are problem dependent, and the user needs to be able to follow the program logic to do so.

BLANK PAGE

Several standard analysis routines are built into the MORSE-SGC version. These routines can be used to provide useful analyses for many problems and will serve as a model in the development of specialized analysis routines.

The BANKR subroutine has been eliminated. Analysis routines are called directly from the subroutines which previously called BANKR if the analysis routine is flagged as needed in the input data.

Problem size is no longer set by the size of blank common in the main program (eliminating the need to recompile the main program for each problem). Core availability is obtained from subroutine ALOCAT which examines the core allocated to the program by the computer system and returns the amount of core storage available for program data.

The MORSE-SGC version is available with either the combinatorial geometry[4] package or the KENO[5] geometry package.

This users manual is intended to document the difference between the MORSE-SGC version and the MORSE as described in Ref. 1. The reader is assumed to be familiar with the contents of Ref. 1.

Definition of Storage Locations and Variable Names

The following tables describe the main variables, their storage locations, and the pointers to their first location in core storage.

Table 1. Storage Locations

Pointer*	Variable Name	Comments	Length
1**	ENER		NNGA+NGGA+2
LFP1	VELTH		NNGA+NGGA
LFP2	Initial wt. standards		MAXGP*MXREG
LFP3	Path parameters		3*MAXGP*MXREG
LFP4	EPROB		(NNGA+NGGA)*MXREG
LFP5	FWLO		MXREG
LFP6	FSE		(NNGA+NGGA)*MEDIA
LFP7	GWL		NNGA*MXREG
LFP8	FS		NGPFS
LFP9	BFS		NGPFS
LFP10	SIGT		(NNGA+NGGA)*MEDIA
LFP11	SIGS		(NNGA+NGGA)*MEDIA
LFP12	SIGGP		NNGA*MEDIA
LFP13	SIGNU		(NNGA+NGGA)*MEDIA
LFP14	NRD		(3+4*NSCT+2*NPL)*MEDIA*NSGPS
LFP15	MWA		2*(NNGA+NGGA)*MEDIA
LFP16	Current wt. standards		3*MAXGP*MXREG
LFP17	Split and RR counters		8*MAXGP*MXREG
LFP18	Current FWLO		MXREG
LFP19	Scattering counters		8*(NNGA+NGGA)*MXREG
LFP20	NSCA		MEDIA
LFP21	Geometry data		NADD
LFP22	Neutron bank		12*NMOST
LFP23	Fission bank		7*NMOST
LAP1	Analysis data		
LSP1	NSGN		NNGA+NGGA
LSP2	NSGL		2*NSGPS
LSP3	NSGP		NSGPS
LFP24	Supergrouted x-sec data		

*In COMMON INPUT.
**First array location.

Table 2. Definitions of Variables in Main Storage

Variable Name	Definition
ENER(IG)	Upper energy boundary of group IG (in eV).
VEL(IG)	Velocity corresponding to the mean energy for neutron groups and the speed of light for gamma-ray groups (in cm/sec).
FS(IG)	Unbiased source spectrum - unnormalized fraction of source particles in each energy group - transformed to cumulative distribution function (c.d.f.) by SØRIN.
BFS(IG)	Biased source spectrum - relative importance of each energy group - transformed to biased c.d.f. by SØRIN.
WTHI(IG, NREG)	Weight above which splitting is performed (vs group and region).
WTLØ(IG, NREG)	Weight below which Russian roulette is performed (vs group and region).
WTAV(IG, NREG)	Weight to be assigned Russian roulette survivors (vs group and region).
PATH(IG, NREG)	Exponential transform parameters (vs group and region).
NSPL(IG, NREG)	Splitting counter (vs group and region).
WSPL(IG, NREG)	Weight associated with NSPL, i.e., the sum of weights which have split (vs group and region).
NØSP(IG, NREG)	Counter for full bank when splitting was requested (vs group and region).
WNØS(IG, NREG)	Weight associated with NØSP.
RRKL(IG, NREG)	Russian roulette death counter (vs group and region).
WRKL(IG, NREG)	Weight associated with RRKL.

Table 2. Definitions of Variables in Main Storage (Contd.)

Variable Name	Definition
RRSU(IG, NREG)	Russian roulette survival counter (vs group and region).
WRSU(IG, NREG)	Weight associated with RRSU.
INIWHI (IG,NREG)	Initial values of WTHI array.
INIWLØ (IG,NREG)	Initial values of WTLØ array.
INIWAV (IG,NREG)	Initial values of WTAV array.
FWLØ(NREG)	Weights to be assigned to fission daughters (vs region).
INIFLØ (NREG)	Initial values of FWLØ.
GWLØ(IG, NREG)	Weights to be assigned to secondary particles (vs group and region).
EPRB(IG, NREG)	Relative importance of energy groups after scattering (vs group and region). Present only if IEBIAS > 0.
NSCT(IG, NREG)	Number of real scatterings (vs group and region).
WSCT(IG, NREG)	Weight associated with NSCT.
NALB(IG, NREG)	Number of albedo scatterings (vs group and region).
WALB(IG, NREG)	Weight associated with NALB.
NFIZ(IG, NREG)	Number of fissions (vs group and region).

Table 2. Definitions of Variables in Main Storage (Contd.)

Variable Name	Definition
WFIZ(IG, NREG)	Weight associated with NFIZ.
NGAM(IG, NREG)	Number of secondary productions (vs group and region).
WGAM(IG, NREG)	Weight associated with NGAM.
NSCA(IMED)	Scattering counter (vs cross-section medium).
FSE(IG, IMED)	Source spectrum for fission-induced neutrons for each group - input as frequency of group IG.

Table 3. Neutron Bank

Pointer*	Variable Name
LNP1	US
LNP2	VS
LNP3	WS
LNP4	XS
LNP5	YS
LNP6	ZS
LNP7	WATES
LNP8	AGES
LNP9	BLZNTS
LNP10	IGS
LNP11	NAMES
LNP12	NAMEXS
LNP13	NMEDS
LNP14	NREGS

*In COMMON NUTRON.

Table 4. Layout of Analysis Data Storage Area

Location Labels	Variable Name	Length
LAP1	LABELS	20 * NRESP
LØCRSP	RESP	
LØCXD	EXTR	NEX * NMTG
	XD YD ZD RAD TO FACT	6 * ND
LØCIB	EXTR	
	IB EP DELE	3 * NE omitted if NE=0
LØCCØ	IARR	NMTG
LØCT	CØS	NA
LØCUD	T DELT	2 * ND * NT
	UD SUD SUD2	3 * ND * NRESP
LØCSD	SD SSD SSD2	3 * ND * NRESP
LØCQE	SUD & SSD Units	20
	QE SQE SQE2	3 * NE * ND
LØCQT	SQE Units	20
	QT SQT SQT2	3 * NT * ND * NRESP
LØCQTE	SQT Units	20

Table 4. Layout of Analysis Data Storage Area (Contd.)

Location Labels	Variable Name	Length
	QTE SQTE SQTE2	3 * NT * NE * ND
LACQAE →	SQTE Units	20
	QAE SQAE SQAE2	
LMAX →	SQAE Units	20

Table 5. Analysis Arrays

Mnemonic Variable Name	Purpose
LABEL (J,NR)	80-character Hollerith label for each response function.
RESP (IG,NR)	Value of response function.
EXTR (IG,NX)	Extra arrays of length NMTG.
XD (ID)	x-coordinate of detector ID.
YD (ID)	y-coordinate of detector ID.
ZD (ID)	z-coordinate of detector ID.
RAD (ID)	Distance from source (assumed to be at XSTRT, YSTRT, ZSTRT) to detector.
TO (ID)	Minimum flight time to detector.
FACT (ID)	Detector-dependent normalization.
EXTR (ID,NXN)	Extra arrays of length ND.
IB (IE)	Energy bin group number.
EP (IE)	Lower energy of bin.
DELE (IE)	Widths of bin.
IARR (IE)	Key to correspondence between analysis group numbers and random walk group numbers.
CØS (IA)	Cosine of angle bin limits.
T (IT,ID)	Time bin limits.
DELT (IT,ID)	Widths of time bins.
UD (NR,ID)	Uncollided response.
SD (NR,ID)	Total response.

Table 5. Analysis Arrays (Contd.)

Mnemonic Variable Name	Purpose
QE (IE,ID)	Energy-dependent fluence.
QT (NR,IT,ID)	Time-dependent response.
QTE (IT,IE,ID)	Time and energy-dependent fluence.
QAE (IA,IE,ID)	Angle- and energy-dependent fluence.

Index	Maximum Value	Purpose
NR	NRESP	Response function.
IG	NMTG	Energy group.
NX	NEX	Extra array (length NMTG).
ID	ND	Detector.
NXN	NEXND	Extra array (length ND).
IE	NE	Energy bin (one or more groups).
IA	NA	Angle bin.
IT	NT	Time bin.

Table 6. Definition of Variables in Common APOLL

Variable	Definition
DDF	Starting particle weight as determined in SORIN.
DEADWT(5)	The summed weights of the particles at death. The four deaths are: Russian roulette, escape, energy, and age limit. The fifth, DEADWT(5), is unused.
DFF	Normalization for adjoint problems.
ETA	Number of mean-free-path (m.f.p.) between collisions.
ETATH	Distance in cm to the next collision if the particle does not encounter a change in total cross section.
ETAUSD	Flight path in m.f.p. that has been used since the last event.
XTRA(10)	Used for temporarily storing alphanumeric information on 'time used' immediately before printing it.
ITERS	Current batch number.
ITIME	Not used.
ITSTR	Switch indicating that secondary fissions are to be the source for the next batch if > 0.
MAXTIM	The elapsed clock time at which the problem is terminated.
MGPREG	Product of number of weight standard groups (MAXGP) and regions (MXREG).
INALB	An index indicating that an albedo scattering has occurred if > 0
NDEAD(5)	Number of deaths of each type (see DEADWT).
NEWNM	Name of the last particle in the bank.
NGEOM	Not used.
NLAST	The last cell in main storage that was used by the cross-section storage or is set aside for banking.

Table 6. Definition of Variables in Common APOLL (Contd.)

Variable	Definition
NMEM	The location of the next particle in the bank to be processed.
NMGP	Not used.
NMTG	The total number of energy groups (both primary and secondary) for which there are cross sections.
NPSCL(13)	An array of counters of events for each batch: (1) sources generated (2) splittings occurring (3) fissions occurring (4) gamma rays generated (5) real collisions (6) albedo scatterings (7) boundary crossings (8) escapes (9) energy cutoffs (10) time cutoffs (11) Russian roulette kills (12) Russian roulette survivors (13) gamma rays not generated because bank was full.
NSIGL	Not used.
NXTRA(10)	Not used.
NSUP	Pointer to particles in the neutron bank which are in the current supergroup.
NINF	Pointer to particles in the neutron bank which are not in the current supergroup.

Table 7. Definitions of Variables in Common FISBNK

Variable	Definition
NFISBM	Location of cell zero of the fission bank.
NFISH	Number of fissions produced in the previous batch (generation).
FTØTL	Total fission weight from all collisions in this batch (generation).
FWATE	Total weight of fission neutrons stored in bank in this batch (generation).
WATEF	Weight of fission neutron stored in bank.
AWATE	Average weight of particles in the current batch.
NPART	Number of particles in the current batch.

Table 8. Definition of Variables in NUTRON Common

Variable	Definition
NAME	Particle's first name.
NAMEX	Particle's family name. (Note that particles do not marry.)
IG	Current energy group index.
IGØ	Previous energy group index.
NMED	Medium number at current location.
MEDØLD	Medium number at previous location.
NREG	Region number at current location.
U,V,W	Current direction cosine.
UØLD,VØLD,WØLD	Previous direction cosines.
X,Y,Z	Current location.
XØLD,YØLD,ZØLD	Previous location.
WATE	Current weight.
ØLDWT	Previous weight. (Equal to WTBC if no path length stretching.)
WTBC	Weight just before current collision.
IBLZN	Current zone number
IBLZØ	Previous zone number.
AGE	Current age.
ØLDAGE	Previous age.
LNP1-14	Pointers for Neutron bank
LNP15-21	Pointers for Fission bank

Table 9. Definitions of Variables in Common QDET

L∅CRSP	Location of cell zero of response functions.
L∅CXD	Location of cell zero of detector positions.
L∅CIB	Location of cell zero of energy bins.
L∅CC∅	Location of cell zero of angle (cosine) bins.
L∅CT	Location of cell zero of time bins.
L∅CUD	Location of cell zero of array UD.
L∅CSD	Location of cell zero of array SD.
L∅CQE	Location of cell zero of array QE.
L∅CQT	Location of cell zero of array QT.
L∅CQTE	Location of cell zero of array QTE.
L∅CQAE	Location of cell zero of array QAE.
LMAX	Last cell used in storage.
EFIRST	Upper energy limit of first energy bin.
EGT∅P	Upper energy limit of first gamma-ray bin.

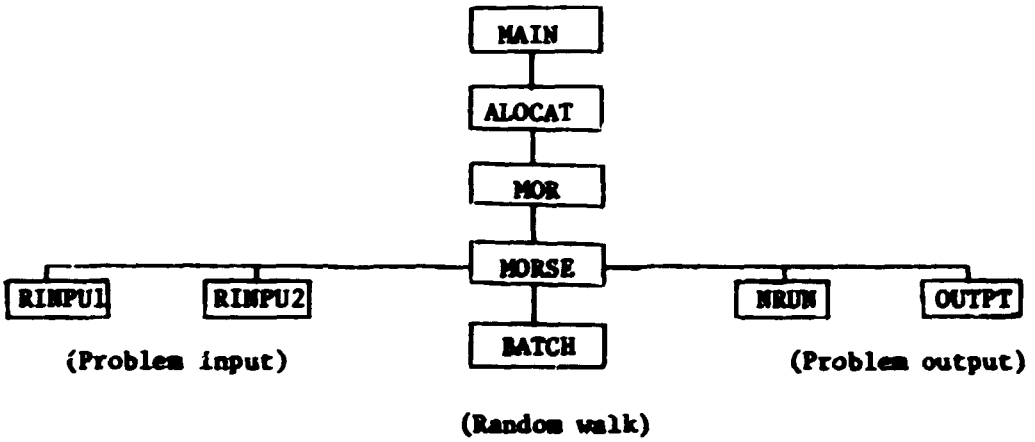
Description of Subroutines

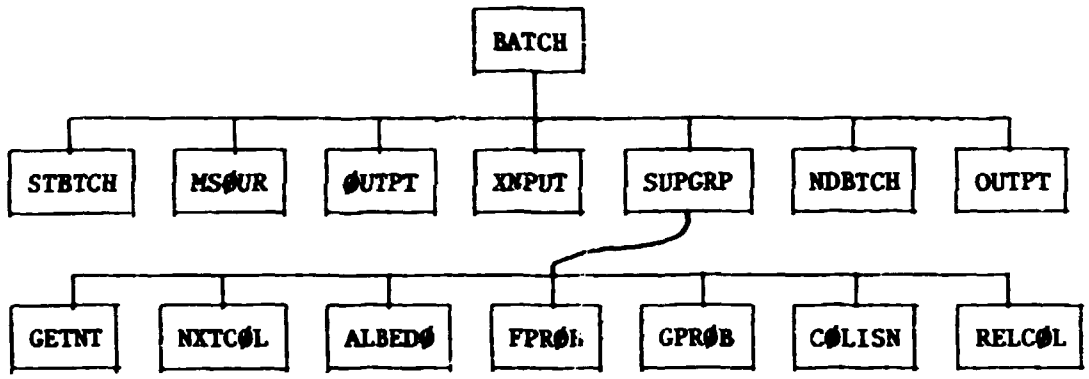
This section gives a brief description of the subroutines which are unique to the MORSE-SGC version and also indicates changes made to subroutines as described in Ref. 1.

MAIN Program

The main program reads the first data block and opens all necessary I/O units. It then passes program control to subroutine MORSE (via subroutines ALOCAT and MOR).

Main Elements of Problem Processing





Subroutine ACCNRM

This program utility subroutine will accumulate and normalize a two-dimensional array over the first dimension.

Subroutine ADJNT

This subroutine will invert arrays FS, BFS, VELTH, ENER. WTHI, WTLO, WTAV, PATH, EPROB, and GWLO (as needed) for an adjoint problem.

Subroutine ALOCAT

This subroutine will determine the amount of core assigned to the program which can be used for data storage.

Subroutine ANGLES

This routine performs the functions that were previously in ANGLES and LEGEND.

Subroutine BADMOM

No change.

Subroutine BATCH

This subroutine is called by subroutine MORSE to perform the processing of batches of particles during the random-walk process.

Subroutine CLEAR

This program utility subroutine will set a two-dimensional array to zero.

Subroutine COLISN

Changed calling sequence to pass array names.

Subroutine DATE

No change.

Subroutine DIREC

No change.

Subroutine EUCLID

Changed calling sequence to pass array names.

Subroutine FBANK

Changed calling sequence to pass array names.

Subroutine FIND

No change.

Subroutine FISGEN

Changed calling sequence to pass array names.

Subroutine FLUXST

Changed calling sequence to pass array names.

Subroutine FPROP

Changed calling sequence to pass array names.

Subroutine FSOUR

Changed calling sequence to pass array names.

Subroutine GAMGEN

Changed calling sequence to pass array names.

Subroutine GETETA

Changed calling sequence to pass array names.

Subroutine GETMUS

No change.

Subroutine GETNT

Changed calling sequence to pass array names. Also now has two entry points: GETNT which retrieves values from the neutron bank and stores them in common NUTRON and STORNT which stores values from common NUTRON into the neutron bank.

Subroutine GOMST

Changed calling sequence to pass array names.

Subroutine GPROB

Changed calling sequence to pass array names.

Subroutine GTIOUT

Changed calling sequence to pass array names.

Subroutine GTMED

No change.

Subroutine INSCOR

No change.

Subroutine INVERT

This program utility subroutine will invert a one-dimensional array.

Subroutine INVRT1

This program utility subroutine will invert a two-dimensional array.

Subroutine INVRT2

This subroutine will invert the cross-section arrays which are stored as one dimensional-array with pointers to the beginning of each group.

Function IWEEK

No change.

Subroutine MAMENT

No change.

Subroutine MOR

This subroutine is called by ALOCAT, and it calls subroutine MORSE with the proper calling sequence.

Subroutine MORSE

This subroutine calls the input routines, the random walk routine (subroutine BATCH), and several output routines. It also monitors the CPU time required for each of these steps.

Subroutine MSOUR

Changed calling sequence to pass array names.

Subroutine NDBTCH

Performs functions previously in NBATCH with different calling sequence.

Subroutine NRUN

Changed calling sequence to pass array names.

Subroutine NSIGTA

This subroutine looks up the total cross section and the nonabsorption probability for a given group.

Subroutine NXTCOL

Changed calling sequence to pass array names.

Subroutine OUTPT

Changed calling sequence to pass array names.

Subroutine OUTPT2

No change.

Subroutine PTHETA

Changed calling sequence to pass array names.

Function Q

No change.

Subroutine RINPU1

This subroutine performs data input.

Subroutine RINPU2

This subroutine controls the processing of input data to establish the data configurations needed during execution of the random walk routines.

Subroutine SCORIN

Changed calling sequence to pass array names and incorporates functions of subroutine ENRGYS.

Subroutine SHIFT

This program utility subroutine will transfer data from one array to another.

Subroutine SHUFLE

This subroutine will change cross-section pointer arrays to the proper position after the cross sections have been supergrouped.

Subroutine SIZEX

This subroutine determines the size of the cross-section data to be stored.

Subroutine SORIN

Changed calling sequence to pass array names.

Subroutine SORT

This subroutine will sort the particles in the neutron bank by supergroups. All particles in the current supergroup will be put into the top of the bank, all others into the bottom.

Subroutine SOURCE

Changed calling sequence to pass array names.

Subroutine STBTCH

Changed calling sequence to pass array names.

Subroutine STRUN

No change.

Subroutine SUPER1

This subroutine calculates the lengths of the various cross sections to be supergrouped.

Subroutine SUPER2

This subroutine will calculate the supergroup number for each cross section group.

Subroutine SUPGRP

This subroutine performs the random walk for all particles within the same supergroup. It includes the coding formerly found in subroutine TESTW.

Subroutine TESTW

Incorporated into subroutine SUPGRP.

Subroutine TIMER

No change.

Subroutine TRNPSE

This program utility subroutine will transpose a two-dimensional array.

Subroutine VAR2

No change.

Subroutine VAR3

No change.

Subroutine XMPUT

This subroutine will retrieve the cross sections for a given supergroup from disk storage and put them in core.

Subroutine XSEC1

This subroutine reads the ID record and energy group boundaries off the cross-section input tape.

Subroutine XSEC2

This subroutine reads cross-section data off the input tape for elements in the mixing table and stores them on disk. It also calculates the storage space that will be needed to store the mixed cross sections.

Subroutine XSEC3

This subroutine mixes the cross-sections and stores the mixed cross-sections on disk.

Subroutine XSEC4

This subroutine will invert the cross-sections if an adjoint problem is being solved. It also sums the two-dimensional neutron gamma cross-sections into the one-dimensional gamma production cross-sections and accumulates and normalizes the two-dimensional cross-sections as needed.

Subroutine XSEC5

This subroutine calculates the angles and probabilities and stores them on disk by supergroup.

Subroutine XSEC6

This subroutine reads cross sections off disk, reorders them into supergroups, and stores them back on disk. Note: This subroutine is called before subroutine XSEC5.

Analysis Routines

Subroutines to analyze the random walk process can be called after the occurrence of a number of events. The user determines which subroutines are to be called by a set of flags in the 5$ array of the input data. Most analysis subroutines are problem-dependent and must be written by the user; however, several common analysis routines have been included. The descriptions of these routines follow.

Subroutine BDRYX

This subroutine is a surface-crossing fluence estimator and will be called whenever a geometry zone boundary is crossed.

The fluence estimate is given by

$$\frac{WATE}{\mu} \qquad \text{for } \mu \geq 0.01$$

or

$$\frac{WATE}{.005} \qquad \text{for } \mu < 0.01$$

where μ = absolute value of the cosine of the angle between the partile track and the surface normal.

A more detailed explanation of this subroutine is given in Ref. 1.

Subroutine RELCØL

The fluence estimate, from the normal collisions in RELCØL, is given by

$$\frac{WATE \cdot e^{ARG} \cdot p(C\emptyset SSCT, IG\emptyset)}{R^2}$$

where WATE is the statistical weight of the particle leaving the collision, ARG is the negative of the number of mean free paths from collision to detector, R is the distance from collision to detector, p(CØSSCT,IGØ) is the probability, per steradian, for a particle with energy in the incoming group IGØ scattering to a lower energy group through angle $\cos^{-1}$(CØSSCT) and CØSSCT is the cosine of the angle between the incoming direction and the line from the collision site to the detector. Note that an array is generated by PTHETA which gives the probability, per steradian, of scattering from group IGØ to group IGQUIT through an angle $\cos^{-1}$ (THETA).

Subroutine SDATA

Estimates of fluence at several point detectors are made for each source event. The source is assumed to emit isotropically and is located at the point X,Y,Z. The contribution is

$$\frac{(WATE)e^{ARG}}{4\pi R^2}$$

where WATE is the statistical weight of the source particle,

ARG is the negative of the number of mean free paths from source to detector, and

R is the distance from source to detector.

Subroutine SGAM

Estimates of fluence at several point detectors are made for each secondary particle generated. The source is assumed to emit isotropically and is located at the point X,Y,Z. The contribution is

$$\frac{(WATE)e^{ARG}}{4\pi R^2}$$

where WATE is the statistical weight of the source particle,

ARG is the negative of the number of mean free paths from source to detector, and

R is the distance from source to detector.

MORSE-SGC Input Instructions

Card A (20A4)

Title card

Card B (Z12)

RANDOM - starting random number

FIDØ DATA BLØCK 1

1$ [14]

1.	IADJM	set greater than zero for an adjoint problem.
2.	NSTRT	number of particles per batch.
3.	NMØST	maximum number of particles allowed for in the bank(s); may equal NSTRT + 1 if no splitting, fission, and secondary generation.
4.	NITS	number of batches.
5.	NQUIT	number of sets of NITS batches to be run without new input data.
6.	NØT USED	set = 0.
7.	ISTAT	set greater than zero to store Legendre coefficients.
8.	NSPLT	set greater than zero if splitting is allowed.
9.	NKILL	set greater than zero if Russian roulette is allowed.
10.	NPAST	set greater than zero if exponential transform is invoked (subroutine DIREC required).
11.	NØLEAK	set greater than zero if nonleakage is invoked.
12.	IEBIAS	set greater than zero if energy biasing is allowed.
13.	NKCALC	the number of the first batch to be included in the estimate of k; if ≤ 0 no estimate of k is made.

14.	NØRMF	the weight standards and fission weights are unchanged if ≤ 0; otherwise fission weights will be multiplied, at the end of each batch, by the latest estimate of k and the weight standards are multiplied by the ratio of fission weights produced in previous batch to the average starting weight for the previous batch. For time-dependent decaying systems, NØRMF should be > 0.

2$ [5]

1.	MEDIA	number of cross-section media.
2.	MMIX	number of mixing operations (elements times density operations) to be performed.
3.	MEDALB	albedo scattering medium is absolute value of MEDALB; if MEDALB = 0, no albedo information to be read in, MEDALB < 0, albedo only problem - no cross sections are to be read, MEDALB > 0, coupled albedo and transport problem.
4.	MXREG	number of regions described by geometry input.
5.	MFISTP	set greater than zero if fissions are allowed.

3$ [20]

1.	NNGA	number of neutron groups to be analyzed.
2.	NGGA	number of gamma groups to be analyzed.
3.	NNGTP	set greater than zero if a completely coupled n, γ problem is desired (both neutrons and gammas are treated as primary particles). NNGA and NGGA must be equal to the number of neutron groups and gamma groups on the input cross-section tape if this option is used.
4.	NØT USED	set = 0.
5.	NØT USED	set = 0.
6.	NØT USED	set = 0.
7.	NØT USED	set = 0.
8.	NCØEF	number of coefficients for each mixture, including P_0.

9.	NSCT	number of discrete angles, set = 0 for isotropic scattering (usually NCØEF/2 integer).
10.	MAXGP	group number of last group for which Russian roulette or splitting or exponential transform is to be performed.
11.	IRDSG	switch to print the cross sections as they are read if > 0.
12.	ISTR	switch to print cross sections as they are stored if > 0.
13.	IFMU	switch to print intermediate results of μ's calculation if > 0.
14.	IMØM	switch to print moments of angular distribution if > 0.
15.	IPRIN	switch to print angles and probabilities if > 0.
16.	IPUN	switch to print results of bad Legendre coefficients of > 0.
17.	IXTAPE	the value of IXTAPE is the logical tape unit for the cross section input data.
18.	NØT USED	set = 0.
19.	NØT USED	set = 0.
20.	NØT USED	set = 0.

4$ [4]

1.	ISØUR	source energy group if greater than zero; if less than or equal to zero, source input cards will be required.
2.	NGPFS	number of groups for which the source spectrum is to be defined. Must equal NNGA + NGGA for an adjoint problem.
3.	ISBIAS	set not equal to zero if the source energy is to be biased.
4.	NSØUR	set less than or equal to zero for a fixed source problem; otherwise the source is from fission generated in a previous batch.

5$ [28]

1.	ND	number of detectors (set = 1 if ≤ 0).
2.	NNE	number of primary particle (neutron) energy bins to be used (must be ≤ NE).
3.	NE	total number of energy bins (set = 0 if ≤ 1).
4.	NT	number of time bins for each detector (set = 0 if NT ≤ 1).
5.	NA	number of angle bins (set = 0 if ≤ 1).
6.	NRESP	number of energy-dependent response functions to be used (set = 1 if ≤ 0).
7.	NEX	number of extra arrays of size NNGA + NGGA to be set aside (useful, for example, as a place to store an array of group-to-group transfer probabilities for estimator routines).
8.	NEXND	number of extra arrays of size ND to be set aside (useful, for example, as a place to store detector-dependent counters).
9-28.	IFLAG(20)	(see Table 10)

T (Block 1 must be terminated with a T)

Table 10. Subroutine Called if IFLAG(I) ≥ 1

I	Subroutine	Called After	Called By	Comments
1	NRUN	end of NITS batches	MØRSE	always called
2	NBATCH	end of each batch	BATCH	always called
3	STBTCH	beg. of batch	BATCH	always called
4	STRUN	beg. of run	MØRSE	no
5	SDATA	source	MSØUR	yes
6	SPLIT	split	SUPGRP	no
7	FISSIN	fission	FPRØB	no
8	SGAM	sec. particle generation	GSTØRE	yes
9	RELCØL	Real collision	SUPGRP	yes
10	ALBEDØ	albedo	SUPGRP	no
11	BDRYX	boundary crossing	NXTCØL	yes
12	ESCAPE	escape	NXTCØL	no
13	ECUT	energy cut	SUPGRP	no
14	TKILL	time kill	SUPGRP	no
15	RRKILL	rr kill	SUPGRP	no
16	RRSURV	rr survival	SUPGRP	no
17	GAMLST	sec. particle generation, no room in banks.	GSTØRE	no
18	NØT USED			
19	NØT USED			
20	NØT USED			

GEOMETRY INPUT DATA - Non-FIDØ Input
Geometry Input Data is Inserted Here

FIDØ DATA BLØCK 2

6* [10]

1. TMAX — maximum 360/91 c.p.u. time in minutes allowed for the problem on the computer.
2. TCUT — age in seconds at which particles are retired; if TCUT = 0, no time kill is performed.
3. WTSTRT — weight assigned to each source particle.
4. AGSTRT — starting age for source particles.
5. XSTRT — x-coordinate for source particle.
6. YSTRT — y-coordinate for source particle.
7. ZSTRT — z-coordinate for source particle.
8. UINP
9. VINP — starting source particle direction cosines. If all are zero, isotropic directions are chosen.
10. WINP

7* [NNGA]

1. (VELTH(I),I=1,NNGA) — the neutron velocities for all energy groups being analyzed. If not input, calculated values will be used based on the group energy limits.

8* [4*MAXGP*MXREG] required only if NSPLT+NKILL+NPAST > 0

1. ((WTHI(I,J),I=1,MAXGP)J=1,MXREG) — weight above which splitting will occur.
2. ((WTLØ(I,J),I=1,MAXGP)J=1,MXREG) — weight below which Russian roulette is played.
3. ((WTAV(I,J),I=1,MAXGP)J=1,MXREG) — weight given those particles surviving Russian roulette.

4. ((PATH(I,J),I=1,MAXGP)J=1,MXREG) path length stretching parameters for use in exponential transform (usually 0 ≤ PATH < 1).

9* [(NNGA+NGGA)*MXREG] required only if IEBIAS > 0.

1. ((EPROB(I,J)I=1,NNGA+NGGA,J=1,MXREG) values of the relative energy importance of particles leaving a collision in region J going to group I.

10* [MXREG+((NNGA+NGGA)*MEDIA] required only if MFISTP > 0. Initialized values will be used if not input.

1. (FWLØ(I,J),I=1,MXREG) values of the weight to be assigned to fission neutrons. (Initialized to 1.0 if not input.)

2. ((FSE(I,J),I=1,NNGA+NGGA,J=1,MEDIA) fraction of fission-induced source particles in group I for medium J. (Uses Chi of element with largest $\nu\Sigma_f$ if not input.)

11* [NNGA*MXREG if IADJM ≤ 0 / or / NGGA*MXREG if IADJM > 0] required only if coupled neutron-gamma ray problem, but not completely coupled.

1. ((GWLØ(I,J),I=1,NNGA or NGGA),J=1,MXREG) probability of generating a gamma at each collision. Alternately (depending on GPROB) the values of the weight to be assigned to the secondary particles being generated. NNGA groups are read for each region in a forward problem and NGGA for an adjoint.

12$ [NMIX]

1. (KM(I),I=1,NMIX) medium number.

13$ [NMIX]

1. (KE(I),I=1,NMIX) element identifiers.

14* [NMIX]

1. (RHØ(I),I=1,NMIX) density.

15$ NØT USED

16$ NØT USED

17* [NGPFS] required only if ISØUR ≤ 0

1. (FS(I),I=1,NGPFS) the unnormalized fraction of source particles in each group. Uses FSE of media number one if not input and MFISTP > 0.

18* [NGPFS] required only if ISØUR $\leq$ 0 and ISBIAS $\neq$ 0.

1. (BFS(I),I=1,NGPFS) the relative importance of a source in group I.

T (Block 2 must be terminated with a T).

Analysis Data

Card AA (20A4)

title or units for total response for all detectors.

Cards AB (20A4*NRESP)

NRESP* title or units for each total response for all detectors.

Card AC (20A4) required only if NE > 1

units for energy-dependent fluence for all detectors.

Card AD (20A4) required only if NT > 1

units for time-dependent total response for all detectors.

Card AE (20A4) required only if NT > 1 and NE > 1

units for time- and energy-dependent fluence for all detectors.

Card AF (20A4) required only if NA > 1

units for angle- and energy-dependent fluence for all detectors.

FIDØ DATA BLØCK 3

19* [ND*3]

((X(I),Y(I),Z(I)),I=1,ND) (X,Y,Z) coordinates of ith detector.

20* [(NNGA+NGGA)*NRESP]

((RFV(I,J),I=1,NNGA+NGGA),J=1,NRESP) response function values.

21$ [NE] required only if NE > 1

Energy group numbers defining the lower limit of energy bins.

22* [NT*ND] required only if NT > 1

((T(I,J),I=1,NT),J=1,ND)

NT values of upper limits of time bins for each detector (in order of increasing time and detector number).

23* [NA] required only if NA > 1

Values of upper limits of angle (cosine) bins.

T (Block 3 must be terminated with a T).

Combinatorial Geometry Input Instructions

CARD CGA (2I5,10X,10A6)

IVØPT - option which defines the method by which region volumes are determined; if

IVØPT = 0, volumes set equal to 1,

IVØPT = 1, concentric sphere volumes are calculated,

IVØPT = 2, slab volumes (1-dim.) are calculated,*

IVØPT = 3, volumes are input by card.

IDBG - if IDBG > 0, subroutine PR is called to print results of combinatorial geometry calculations during execution. Use only for debugging.

JTY - alphanumeric title for geometry input (columns 21-80).

CARDS CGB (2X,A3,I4,6D10.3)

One set of CGB cards is required for each body and for the END card (see Table 11). Leave columns 1-6 blank on all continuation cards.

ITYPE - specifies body type or END to terminate reading of body data (for example BØX, RPP, ARB, etc.). Leave blank for continuation cards.

IALP - body number assigned by user (all input body numbers must form a sequence set beginning at 1). If left blank, numbers are assigned sequentially. Either assign all or none of the numbers. Leave blank for continuation cards.

FPD(I) - real data required for the given body as shown in Table 11. This data must be in cm.

---

*Not operational.

CARDS CGC (2X,A3,I5,9(A2,I5))

Input zone specification cards. One set of cards required for each input zone, with input zone numbers being assigned sequentially.

IALP - IALP must be a nonblank for the first card of each set of cards defining an input zone. If IALP is blank, this card is treated as a continuation of the previous zone card.
IALP - END denotes the end of zone description.

NAZ - total number of zones that can be entered upon leaving any of the bodies defined for this input region (some zones may be counted more than once). Leave blank for continuation cards for a given zone. (If NAZ ≤ 0 on the first card of the zone card set, then it is set to 5.) This is used to allocate blank common. Alternate IIBIAS(I) and JTY(I) for all bodies defining this input zone.

IIBIAS(I) - specify the "ØR" operator if required for the JTY(I) body.

JTY(I) - body number with the (+) or (-) sign as required for the zone description.

CARDS CGD (14I5)

MRIZ(I) - MRIZ(I) is the region number in which the "Ith" input zone is contained (I = 1, to the number of input zones). Region numbers must be sequentially defined from 1.

CARDS CGE (14I5)

MMIZ(I) - MMIZ(I) is the medium number in which the "Ith" input zone is contained (I = 1, to the number of input zones). Medium numbers must be sequentially defined from 1. (Media 0 is used for external void and media 1000 is used for internal void).

CARDS CGF (7D10.5) (Omit if IVØPT ≠ 3)

VNØR(I) - volume of the "Ith" region (I = 1 to MXREG, the number of regions).

Table 11. Input Required on CGB Cards for Each Body Type

Card Columns Body Type	ITYPE 3-5	LALP 7-10	Real Data Defining Particular Body 11-20	 21-30	 31-40	 41-50	 51-60	 61-70	Number of Cards Needed
Box	BØX	IALP is assigned by the user or by the code if left blank.	Vx	Vy	Vz	H1x	H1y	H1z	1 of 2
			H2x	H2y	H2z	H3x	H3y	H3z	2 of 2
Right Parallelepiped	RPP		Xmin	Xmax	Ymin	Ymax	Zmin	Zmax	1
Sphere	SPH		Vx	Vy	Vz	R	-	-	1
Right Circular Cylinder	RCC		Vx	Vy	Vz	Hx	Hy	Hz	1 of 2
			R	-	-	-	-	-	2 of 2
Right Elliptic Cylinder	REC		Vx	Vy	Vz	Hx	Hy	Hz	1 of 2
			R1x	R1y	R1z	R2x	R2y	R2z	2 of 2
Ellipsoid	ELL		V1x	V1y	V1z	V2x	V2y	V2z	1 of 2
			L	-	-	-	-	-	2 of 2
Truncated Right Cone	TRC		Vx	Vy	Vz	Hx	Hy	Hz	1 of 2
			L1	L2	-	-	-	-	2 of 2
Right Angle Wedge	WED		Vx	Vy	Vz	H1x	H1y	H1z	1 of 2
			H2x	H2y	H2z	H3x	H3y	H3z	2 of 2

Table 11. Input Required on CGB Cards for Each Body Type (Contd.)

Card Columns Body Type	ITYPE 3-5	IALP 7-10	Real Data Defining Particular Body 11-20	 21-30	 31-40	 41-50	 51-60	 61-70	Number of Cards Needed
Arbitrary Polyhedron	ARB		V1x	V1y	V1z	V2x	V2y	V2z	1 of 5
			V3x	V3y	V3z	V4x	V4y	V4z	2 of 5
			V5x	V5y	V5z	V6x	V6y	V6z	3 of 5
			V7x	V7y	V7z	V8x	V8y	V8z	4 of 5
			Face Descriptions (see note below)						
Termination of Body Input Data	END								

NOTE: Card 5 of the arbitrary polyhedron input contains a four-digit number for each of the six faces of an ARB body. The format is 6D10.3, beginning in column 11. See the ARB write-up in Section 4.7 for an example.

Data Guide for KENO Geometry to MORSE-SGC

Card 1 Title Card (80 characters)

Card 2 (Enter data in free format - Hollerith data is terminated by two consecutive blanks - all other entries are terminated by a single blank. Any number of blanks may be interspersed between entries. A new card may be started after any entry. nRx, n*x, or n$x will repeat the entry x n times. nZ produces n zeroes).

NBXMAX - number of array units in the x direction.

NBYMAX - number of array units in the y direction.

NBZMAX - number of array units in the z direction

If NBXMAX*NBYMAX*NBZMAX = 0, the problem is treated as a nonarray problem.

Enter geometry data next

FGEOM FGEOM may be one of the following Hollerith keywords:

CUBE, CUBOID, SPHERE, CYLINDER, XCYLINDER, YCYLINDER, HEMISPHERE, HEMISPHE+Z, HEMISPHE-Z, HEMISPHE+X, HEMISPHE-X, XHEMICYL+Y, XHEMICYL-Y, HEMISPHE+Y, HEMISPHE-Y, XHEMICYL+Z, XHEMICYL-Z, YHEMICYL-X, YHEMICYL+Z, YHEMICYL-Z, ZHEMICYL-X, ZHEMICYL+Y, ZHEMICYL-Y, CORE BDY, YHEMICYL+X, ZHEMICYL+X

NOTE: FGEOM may be no more than 12 characters long.

A CUBE has +X, = +Y = +Z and -X = -Y = -Z. Note that the +X dimension need not equal the -X dimension of the cube; i.e., the origin need not be at the center of the cube.

a CUBOID is a rectangular parallelepiped and may be described anywhere relative to the origin.

A SPHERE must be centered about the origin.

A CYLINDER has its length described along the Z axis and its center line must lie on the Z axis.

A XCYLINDER has its length described along the X axis and its center line must lie on the X axis.

A YCYLINDER has its length described along the Y axis and its center line must lie on the Y axis.

A HEMISPHERE must have its flat portion centered about the origin at Z = 0.0 and exists only in the positive Z direction.

A HEMISPHE(B) (C) must have its flat portion centered about the origin at (C) = 0.0 and exists only in the BC direction (B = + or -, C = X, Y, or Z). For example, HEMISPHE+Z is the same as the previously described HEMISPHERE and HEMISPHE-Z is the mirror image of HEMISPHE+Z, therefore existing only in the negative Z direction.

A (B)HEMICYL(C) (D) is a half cylinder whose axis is the B axis (B = X, Y, or Z) and exists only in the CD direction (C = + or -, D = X, Y, or Z). (Examples: ZHEMICYL+Z, YHEMICYL-Z, XHEMICYL+Y).

A CORE BDY defines a CUBOID which fits tightly around the array. It is used to define the coordinates of the array in some overall coordinate system. It is only necessary if the array is to be surrounded by other regions.

Starting 2 or more spaces after the geometry word, the following data is entered, separated by one or more blanks. A new card may be started after any entry.

MAT	Media number.
IMP	Weighting region.
XX(1)	Radius for sphere, cylinders, hemispheres, hemicylinders, +x dimension for cube or cuboid, or -x dimension for a core card.
XX(2)	-x dimension for cube or cuboid region, +z for cylinder, +x for x cylinder, +y for y cylinder, + length for hemicylinder, -y dimension for a core card, omit XX(2) for a sphere or hemisphere.

XX(3)	+y dimension for cuboid region, -z for cylinder, -x for x cylinder, -y for y cylinder, - length for hemicylinder, -z for a core card, omit XX(3) for a sphere, hemisphere, or cube.	
XX(4)	-y dimension for cuboid region	omit for all other geometry types
XX(5)	+z dimension for cuboid region	
XX(6)	-z dimension for cuboid region	

If the geometry describes an array with more than one type of unit, then the units are defined by FGEOM being input as BOX TYPE I, where I is an integer which will be used to describe where the unit goes in the array. At least two blanks must occur between TYPE and I. The geometry regions describing I then follow. The last region of any box type must be a cube or cuboid. The definition of a box type is terminated by inputting another box type, by a core card, or by the end of the geometry data. The end of the geometry data is signaled by an END GEOM.

Within a given box type, or the external regions surrounding the array, each region must be completely enclosed by the next region.

After the END GEOM card, if there was more than one box type in the array, the following mixed box orientation data is read to describe where the units are placed in the array.

The first field contains the box type, followed by three sets of three fields that are treated like FORTRAN DO loops. The arrangement of boxes may be considered as consisting of a three-dimensional matrix of box type numbers, with the box position increasing in the position X, Y, and Z direction, respectively. Each set of mixed box orientation data consists of the following parameters, separated by one or more blanks.

LTYPE The box type. LTYPE must be greater than zero and less than or equal to the maximum number of box types.

IX1 The starting point in the X direction. IX1 must be at least 1 and less than or equal to NBXMAX.

IX2 The ending point in the X direction. IX2 must be at least 1 and less than or equal to NBXMAX.

INCX The number of boxes by which increments are made in the positive X direction. INCX must be greater than zero and less than NBXMAX.

IY1 The starting point in the Y direction. IY1 must be at least 1 and less than or equal to NBYMAX.

IY2 The ending point in the Y direction. IY2 must be at least 1 and less than or equal to NBYMAX.

INCY The number of boxes by which increments are made in the positive Y direction. INCY must be greater than zero and less than NBYMAX.

IZ1 The starting point in the Z direction. IZ1 must be at least 1 and less than or equal to NBZMAX.

IZ2 The ending point in the Z direction. IZ2 must be at least 1 and less than or equal to NBZMAX.

INCZ The number of boxes by which increments are made in the positive Z direction. INCZ must be greater than zero and less than NBZMAX.

An important feature of this type of data description is that if any portion of an array is defined in a conflicting manner, the last card to define that portion will be the one that determines the array's box type configuration. To utilize this feature, one can fill an entire array with the most prevalent box type and then superimpose the other box types in their proper places to accurately describe the array.

Mixed-box orientation data is terminated with an END BOX.

The product of NBXMAX*NBYMAX*NBZMAX* the number of geometry regions must be less than 2 to the 31st power.

APPENDIX A

Explanation of Terms Used in the Combinatorial Geometry Description

The combinatorial geometry package allows the description of the shape of physical structures by the combination of certain basic geometric shapes (bodies) such as rectangular parallelepiped, right circular cylinders, etc. (A complete list of these basic shapes is given in the input description.)

These basic shapes are combined using three logical operators: 1) OR, 2) AND, and 3) NOT. The OR operator is used explicitly and the AND and NOT operators are used implicitly.

An input zone is a physical shape which is described by a string of basic shapes (bodies) and logical operators.

The geometry package will subdivide each input zone into code zones which can be described without the use of the OR operator, so that each input zone consists of one or more code zones.

In addition to describing the shape of an input zone, it is also necessary to indicate which cross section information is to be used. Each material (mixture) for which cross section data is stored during problem execution is assigned a media number. An input zone can contain only one medium. Each input zone, therefore, has associated with it a media number (not necessarily unique). This associated media number may be set to 0 or 1000 as flags to indicate external voids or internal voids, respectively. External voids are used to indicate when a particle has leaked from the system under consideration. Any particle which enters an external void is assumed to leak. The system under consideration must

be completely surrounded by external void as a flag to stop tracking a particle when it leaks. Internal void is a vacuum through which particles travel without any interaction.

In many problems it is desirable to use different Russian roulette kill weights, survival weights, etc. in different parts of the geometry. To do this, one or more importance regions or regions are used. An input zone must be within a single region; and, therefore, each input zone has associated with it a region number.

REFERENCES

1. M. B. Emmett, "The MORSE Monte Carlo Radiation Transport Code System," ORNL-4972 (Feb., 1975).

2. N. M. Greene, J. L. Lucius, W. E. Ford, III, R. Q. Wright, and L. M. Petrie, "AMPX - A Modular Code System for Generating Coupled Multigroup Neutron-Gamma Libraries from ENDF/B," ORNL-TM-3706 (to be published).

3. W. E. Engle, Jr., "A Users Manual for ANISN," K-1693 (March, 1967).

4. E. A. Straker, W. H. Scott, Jr., and N. R. Bryn, "The MORSE Code with Combinatorial Geometry," DNA-2860T (May, 1972).

5. L. M. Petrie and N. F. Cross, "KENO IV - An Improved Monte Carlo Criticality Program," ORNL-4938 (1975).