ornl

ORNL/TM-9355

OAK RIDGE NATIONAL LABORATORY

MARTIN MARIETTA

Applications Guide to the MORSE Monte Carlo Code

S. N. Cramer

OPERATED BY
MARTIN MARIETTA ENERGY SYSTEMS, INC.
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

DISCLAIMER

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
NTIS price codes—Printed Copy: A05; Microfiche A01

ORNL/TM--9355

DE85 018145

Engineering Physics and Mathematics Division

APPLICATIONS GUIDE TO THE MORSE MONTE CARLO CODE

S. N. Cramer

Manuscript Completed: May 1985
Date of Issue: August 1985

This Work Sponsored by
Defense Nuclear Agency
Under
Interagency Agreement No. 40-65-65

Prepared by the
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831
operated by
Martin Marietta Energy Systems, Inc.
for the
U.S. DEPARTMENT OF ENERGY
under Contract No. DE-AC05-840R21400

PREFACE

This report has been compiled from notes on the MORSE code presented by the author at a course entitled "Monte Carlo Methods in Nuclear Reactor Analysis" at the European Community Research Establishment at Ispra, Italy, in October 1984. Other ORNL reports by the author which also consist of lecture notes for this course are:

> "Applications Guide to the RSIC Distributed Version of the MCNP Code," ORNL/TM-9641.
>
> "Applications Guide to the RSIC Distributed Version of the KENO-V Code," ORNL/TM-9642.
>
> "Variance Reduction Methods Applied to Deep-Penetration Monte Carlo Problems," ORNL/TM-9643.

The lecture notes from the entire course (a total of 17 lectures by seven different authors) will eventually be published by Harwood Academic Publishers, Paris and New York. For further information, contact the course coordinator: Dr. H. Rief, Ispra Establishment, 21020 Ispra (Varese), Italy.

Appreciation is extended to Ms. M. B. Emmett, the author of the MORSE manual and keeper of the in-house and RSIC versions of the code, for her review of these notes. Appreciation is also extended to Dr. J. S. Tang and other MORSE users at ORNL for their specific comments, and to Ms. Nancy Hatmaker and other RSIC staff members for their help. Finally, a special acknowledgment is due to Ms. Katie Ingersoll who typed the draft of this report and also typeset the final report.

TABLE OF CONTENTS

ABSTRACT

A practical guide for the implementation of the MORSE-CG Monte Carlo radiation transport computer code system is presented. The various versions of the MORSE code are compared and contrasted, and the many references dealing explicitly with the MORSE-CG code are reviewed. The treatment of angular scattering is discussed, and procedures for obtaining increased differentiality of results in terms of reaction types and nuclides from a multigroup Monte Carlo code are explained in terms of cross-section and geometry data manipulation. Examples of standard cross-section data input and output are shown. Many other features of the code system are also reviewed, including (1) the concept of primary and secondary particles, (2) fission neutron generation, (3) albedo data capability, (4) DOMINO coupling, (5) history file use for post-processing of results, (6) adjoint mode operation, (7) variance reduction, and (8) input/output. In addition, examples of the combinatorial geometry are given, and the new array of arrays geometry feature (MARS) and its three-dimensional plotting code (JUNEBUG) are presented. Realistic examples of user routines for source, estimation, path-length stretching, and cross-section data manipulation are given. A detailed explanation of the coupling between the random walk and estimation procedure is given in terms of both code parameters and physical analogies. The operation of the code in the adjoint mode is covered extensively. The basic concepts of adjoint theory and dimensionality are discussed and examples of adjoint source and estimator user routines are given for all common situations. Adjoint source normalization is explained, a few sample problems are given, and the concept of obtaining forward differential results from adjoint calculations is covered. Finally, the documentation of the standard MORSE-CG sample problem package is reviewed and on-going and future work is discussed.

I. INTRODUCTION

The MORSE code is a large general-purpose multigroup Monte Carlo radiation-transport computer code system[1] that has been undergoing development at Oak Ridge National Laboratory since the late 1960s. Although no claims can be made regarding its superiority in either theoretical details or Monte Carlo techniques, since its inception MORSE has been used more widely than any other Monte Carlo radiation transport code available. The principal reason for this popularity is that the code is relatively easy to use, it is independent of any installation or distribution center, and it can be easily customized to fit almost any specific need.

The MORSE system, which calculates the transport of uncharged particles (neutrons and/or gamma rays) can be used on IBM, CRAY, CDC, UNIVAC, and VAX computers with code packages available from the Radiation Shielding Information Center (RSIC) at ORNL. Some of the features associated with the system and its implementation are:

- Applicable to multiplying or non-multiplying media.
- Can operate in a fixed source or a k_{eff} eigenvalue mode.
- Can be used for static or time-dependent calculations.
- Operable in both forward and adjoint modes.
- Uses cross section and/or albedo data.
- Treats regular and/or delta scattering.
- Can be coupled with discrete ordinates codes (source and/or response mode coupling).
- Has available a general combinatorial geometry, including a torus.
- Contains a multiple array nesting feature for finite or infinite geometry lattices.
- Has a two- and three-dimensional geometry picture drawing capability.
- Can provide collision density plots.
- Includes many standard variance reduction techniques.
- Uses ANISN[2] library multigroup cross section and anisotropic scattering formats, including upscatter for any number of thermal energy groups.
- Can treat separate Klein-Nishina, pair production, and photoelectric reactions for gamma-ray next-event estimation.
- Uses a history file for post-processing of results.
- Has a "user friendly" framework of routines and other aids which alleviate much of the effort necessary for user-written routines.
- Includes a large library of "standard" user routines which may be used directly or serve as models for further modification.

- Is supported by a plethora of related literature and documentation describing the code, multigroup and general theory, programming details, debugging aids, error messages, flow charts, input/output, sample problems, applications, interpretation of results, and other capabilities and limitations.

It is the use of the ANISN format cross sections that has given MORSE much of its installation independence. The ORNL discrete ordinates codes ANISN[2] and DOT[3] are themselves so widely applied that the capability to produce ANISN library multigroup cross sections is almost always present wherever MORSE is to be implemented. Otherwise, the user of MORSE is like the user of a continuous energy Monte Carlo code. He is at the mercy of the code distribution center or the code's home installation for cross sections, or he must process his own cross sections; and gaining proficiency in the theory and application of cross-section processing code systems (AMPX[4] or SCALE,[5] for example, for MORSE) can be as difficult as learning to use the Monte Carlo codes themselves. Cross-section processing for multigroup Monte Carlo codes is even more difficult than that for continuous energy codes in the sense that considerably more physics must be addressed in the form of spectral weighting, resonance self-shielding, etc. (Most continuous energy codes have capabilities for "pseudo-point" cross sections whose creation contains some of these effects but to a lesser degree.)

MORSE does not use the laws of collision mechanics to determine scattering distributions for specific nuclides and reaction types, but it uses a discrete ordinates-type group-to-group transfer (scattering) matrix for each material mixture. The details of the collision mechanics have been absorbed into the multigroup cross-section processing. In a sense, the use of MORSE can be thought of as an extension of ANISN or DOT (or similar codes) into three dimensions. The traditional use of Monte Carlo is the "method of last resort" due to geometrical complexities where the approximations in one- or two-dimensional analysis are judged to be inadequate. If there is some improper use or creation of the cross sections for the discrete ordinates code, then use of the same cross sections in MORSE may give no improvement in the results.

In summary, MORSE uses the same type of multigroup cross sections as several widely used discrete ordinates codes. If expertise in the theory, creation, and application of the cross sections already exists at an installation or institution, the implementation of MORSE there is greatly simplified. The user must always remember that cross-section processing can never be neglected simply because a Monte Carlo code is being used, especially multigroup Monte Carlo. The treatment necessary for a criticality calculation may differ greatly from that for a deep-penetration calculation. As with discrete ordinates codes, a knowledge of the proper generation and use of the cross sections is often more important than a knowledge of the intermechanisms of the multigroup Monte Carlo code.

If the decision to use a Monte Carlo code is based on the desire to obtain more differentiality in results in terms of energy, angle, nuclides, and reaction types, rather than the need for a three-dimensional geometry, then perhaps a continuous energy code is more appropriate than MORSE. However, much of the stigma attached to MORSE with respect to this differentiality is unjustified, as will be explained in the next section.

The multigroup structure of MORSE provides many advantages over continuous energy codes. For example, because MORSE and discrete ordinates codes use the same cross sections, the results obtained with all the codes may be compared for determination of purely geometric effects. MORSE-discrete ordinates code coupling is also facilitated because of the cross sections, as well as use of discrete ordinates-generated albedo data. The adjoint calculation option, either singularly or discrete ordinates coupled, is a relatively simple task to implement with MORSE.

In addition to its use for general reactor and shielding applications, MORSE has had much utilization as a pure research tool for academic purposes. This is in part due to its multigroup formulation and in part due to its amenability to modification, as explained in the next paragraph. Some specific theoretical or Monte Carlo technique can be easily isolated and studied in detail independent of other effects (e.g., a one-group problem to study a phenomenon associated only with the transport process). As a result, MORSE has been used as the basic code in many theses and dissertations.

One of the initial disadvantages a user first encounters in using MORSE can, in time, become a great advantage. This is the necessity for user-written subroutines to describe the source, estimate the results, and do other optional tasks. Due to the longtime and widespread use of MORSE, many "standard" user routines exist which may be used directly or easily modified. All MORSE documentation has been directed toward this user orientation (see Sect. VII for updates to the manual). The original manual,[6] though now out of date in many aspects, contains diagrammed flow charts for all routines, and it is still applicable in this respect. The current manual contains a wealth of information for users in terms of subroutine and common descriptions, variable definitions and locations, computer memory requirements, error messages, etc. The RSIC code package contains the input/output, documentation, and special routines for many representative sample problems.

Before beginning with a detailed description of various parts of the code, a specification must be made as to what code is being described. Due to some of the items discussed above, there are *many* codes called MORSE. The only code directly made reference to in this presentation is that distributed in the RSIC code package CCC 203X, where X designates a particular computer system. The core of this package is the ORNL version of MORSE-CG[1] (*C*ombinatorial *G*eometry, originally developed for the SAM[7] code). MORSE-CG is the most direct descendant of the original MORSE code,[6] and it is maintained in RSIC with timely updates from the ORNL in-house code. The RSIC code library contains packages for seven other MORSE code systems which differ in some ways from CCC 203X. Most of these have been contributed to RSIC from other installations where MORSE has been modified according to local needs. To add to this sometimes resulting confusion, there exists at ORNL a more recent MORSE called MORSE-SGC[8] (*S*uper *G*roup *C*ombinatorial geometry). Although there are some similarities, MORSE-SGC is essentially a completely different code from the other versions, and it lacks many of the features of the other codes. The super group capability is the same as that in KENO-V[9] and provides a capability for use of the code on small computer systems. It was for the SCALE code system version of MORSE-SGC that the MARS (*M*ultiple *A*rray *S*ystem) capability was created. This finite-lattice array-nesting feature has been recently incorporated into MORSE-CG, and it is the only item interchangeable between the two ORNL codes. The next version of MORSE-CG will be called MORSE-CGA.

II. CROSS SECTIONS

Due to the relative ease of generating ANISN format multigroup cross sections, only a nominal number of cross-section libraries are distributed with the RSIC code packages for MORSE. Many code systems, such as AMPX and SCALE (both available from RSIC), produce these cross sections and/or convert other library formats to ANISN format. In addition, RSIC has available an independent Data Library Collection (DLC) which contains many sets of processed cross sections. A review of multigroup cross-section processing is given in Reference 10.

It is, of course, this cross-section treatment which presents the most serious potential problem with multigroup Monte Carlo codes as compared to continuous energy codes. Data sets designed and weighted for specific applications must not be used indiscriminately for many other applications. Cross-section sets for use in reactor core calculations must be given many of the same detailed treatments as those used in any multigroup method of analysis. A particular problem arises in deep-penetration calculations if only one weighting scheme is used in the shield. Just as for discrete ordinates applications, the shield should be divided into several regions, even if it is entirely one material, with appropriate cross-section weighting in each region as indicated by the shape of flux spectrum determined from the fine-group collapsing procedure.

II.A. Angular Scattering

The angular scattering treatment in MORSE is created by discretizing a $P_N(\mu)$ Legendre polynomial expansion such that the $N+1$ mathematical moments of the discrete distribution are the same as those of the continuous truncated polynomial. The excruciating details of the entire development of the generalized Gaussian quadrature scheme to generate the discrete distribution are given in the MORSE manual (Ref. 1, Sections 4.5 and 4.11). The result is a set of $N+1$ equations giving $(N+1)/2$ discrete angles (actually angle cosines) and $(N+1)/2$ probabilities. These discrete angles are located near the peaks of the angular distribution but with enough dispersion to account for the entire cosine range ($-1 \leq \mu \leq 1$). If N is not odd, the method reverts to the next lowest odd integer. The P_0 term corresponds to the group-to-group energy transfer probability, $\Sigma_{g\rightarrow g'}/\Sigma_g$, where the group scattering cross section Σ_g is the sum over all g' including the within-group scattering term ($g'=g$). For isotropic scattering, only the P_0 term is used.

In making cursory observations of these scattering techniques, it must be noted that these discrete angles are laboratory system scattering angles, i.e., angles between the incoming and outgoing particle directions. They are not fixed in space and should never be confused with the fixed angular quadrature used in discrete ordinates analysis. These discrete scattering angles will in general be different for each group-to-group transfer for each mixture of nuclides, and the azimuthal angle is chosen isotropically. Thus with only a small amount of multiple scattering, the continuous angular phase space is as adequately described as for other averaging or weighted Monte Carlo processes. With source distributions which create an initial angular dispersion, low-order P_N truncations are adequate. A P_3 expansion is used for many problem types. (It was the success of the original KENO

criticality multigroup Monte Carlo code, which has a P_1 expansion and 16 groups, and of the ANISN and DOT discrete ordinates codes for deep penetration that led to the creation of MORSE. Its subsequent success, coupled with a decline in funding for pure code development, has contributed to the decline and eventual demise of the ORNL continuous energy Monte Carlo program — 05R[11] and its successors O6R,[12] ESP,[13] and others.)

There are situations where certain combinations of problem characteristics coupled with this angular treatment may produce discrete ordinates-type "ray effects," i.e., non-physical perturbations in the angular distributions and subsequent results. Among these characteristics are:

- low-order P_N expansion,
- little or no multiple scattering,
- a highly directional-dependent angular source distribution, and
- a significant fraction of low-Z (atomic weight) nuclides in the medium mixture.

One of these items appearing singly rarely causes trouble. An increase in the P_N expansion order will usually overcome any trouble associated with the above items.[14] The computer memory requirements and, to a lesser extent, the computation time will increase with larger N. For large N (>7) requests, there may be insufficient data or the data may be faulty. In this case, the code will revert to the highest acceptable N and print appropriate warning messages. There exist other multigroup angular treatments (see, for example, Reference 15); however, the comparison of these methods becomes somewhat difficult for mixtures of nuclides.[16]

The angular scattering method in MORSE always gives non-negative distributions and precludes negative weights in the random walk. The next-event estimation process, however, uses directly the continuous distribution, which will in general have some narrow intervals of negativity. Although this is rarely a problem for neutrons, gamma-ray results from the standard method can be erroneous due to the inability of the continuous distribution to adequately describe the Compton scattering energy-angle delta function relationship with a low-order polynomial expansion. For this reason a special gamma-ray estimation procedure has been recently developed which uses the real (non-polynomial) scattering distribution (see Section V.E.3).

II.B. Cross-Section Flexibility

Differentiality in MORSE results can be increased by increasing the number of energy groups or the number of angular intervals in the estimation procedure input data. Differentiality in results with respect to individual nuclides and reaction types can be obtained by certain manipulations in the MORSE cross-section input procedure for later use in the estimation routines. This cannot be as general as for a continuous energy code and will require some extra effort and computer memory (see Section V.F).

There must exist a cross-section mixture (called a "medium" in MORSE) for each non-void and non-albedo geometry medium, and these must have corresponding integer identification numbers (for a convenient exception to this, see Section V.K). There may also exist extra cross-section media that are not needed in the geometry description but may be used for estimating certain quantities. (In all but the simplest MORSE calculations, the cross-section files should be created independently of the rest of the calculation, using the XCHEKR module — see pages 4.3-11 to 4.3-14 in the manual[1] — and the definition of IXTAPE and JXTAPE. These files can be scrutinized for verification and saved for later calculations where the extra cross sections may be needed in the geometry.)

As an example, consider a situation where it is desired to obtain specific information concerning (1) total collisions, (2) absorption, (3) scattering, (4) fission production, or (5) gamma production from ^{235}U in a mixture of other fissile and/or moderator materials. The cross-section medium with the correct mixture of ^{235}U and other nuclides is used in the random walk geometry. Another cross-section medium is created containing only ^{235}U. This extra cross-section medium is never referenced in the random walk geometry but is available for use in estimator routines for determining the specific desired quantities mentioned above (see Section V.F). It is not required that the extra nuclide actually appear in the system, as is the case sometimes for the determination of central reactivity worths. If further reaction type detail is wanted, e.g., the scattering contribution from a specific inelastic level, the original processing code (such as AMPX) must be rerun. An extra nuclide file would be created here whose entire scattering cross section would consist only of that for the desired level. This extra nuclide is then processed by MORSE into an extra medium and utilized in certain estimator routines as described previously.

II.C. Cross-Section Processing

The cross-section storage in MORSE is now given according to IHT, the position of the total cross section in the ANISN library cross-section table (input card XB). The general positions are given in Table 1 for primary particles. For a coupled neutron-gamma problem, the gamma (secondary particle) data follow in identical form. For no up-scatter data, ISGG=IHT+1. Often IHT=3 and the absorption cross section σ_a is in position 1. (The σ_a is not normally used in MORSE and, in fact, does not appear in the transport equation.) In some libraries IHT=5 and the first two positions contain χ, the fission spectrum, and σ_f, the fission cross section alone. MORSE does not currently process any data above table position IHT-1. The data for the higher order coefficients follow that for P_0 with data in positions equal to and above IHT having zero values. It must be remembered that the group numbers increase as the energy decreases.

In the cross-section input (Section 4.3.3 in the manual), the nuclide identifiers (cards XD) are those given in the cross-section library, NCOEF values for each nuclide including P_0 (NCOEF = 4 for P_3) forming one set. The KE values are numbered consecutively, representing an integer count of each set of nuclide data in the order they are read from the library. If the NMIX value is wrong or if the negative terminator on a KE value is missing, the code may continue processing in some cases but will produce erroneous data. It is convenient to number the KM media consecutively as they are created in the cross-section input and to give the geometry media input corresponding values. The NCOEF

Table 1. Cross-Section Table Positions

Position	Cross-Section Type
.	Not normally used by MORSE
.	
.	
IHT-2	Absorption, σ_a
IHT-1	$\nu \times$ fission, $\nu\sigma_f$
IHT	Total, σ_t
IHT+1	Number of upscatter groups, $\sigma_{g\to g'}$, $g'<g$
.	
.	
IHT+NGP-NDS*	
ISGG	Within-group scatter, $\sigma_{g\to g}$
ISGG+1	Number of downscatter groups, $\sigma_{g\to g'}$, $g'>g$
.	
.	
ISGG+NDS	

*NGP is the number of primary groups.

NDS is the number of groups of downscatter.

should be an even number (odd P_N expansion since NCOEF = $N+1$) and the number of discrete angles NSCT = NCOEF/2. See Section II.A for a discussion of the P_N approximation. If scattering distributions for next-event estimation are needed, ISTAT must be non-zero. The various print options can generate much data if activated. The point cross-section option requires an O6R cross-section capability and has been little used.

Examples of standard cross-section output listings are given in Tables 2-4. This medium (here for Fe, but the number of nuclides is immaterial) is for a coupled 11-neutron-group 5-gamma-ray-group P_3 calculation. In Table 2 the column headings are:

GRP	—	group number; the highest neutron energy group is 1 and the highest gamma-ray energy group is 12.
SIGT	—	Σ_T.
SIGST	—	Σ_S.
PNUP	—	probability of upscatter.
PNABS	—	scattering (non-absorption) probability Σ_S/Σ_T, sometimes > 1 because of (n,xn) neutron reactions and pair production of gamma rays.
GAMGEN	—	gamma-ray production, $\gamma\Sigma_\gamma/\Sigma_T$.
NU*FIS	—	fission production, $\nu\Sigma_f/\Sigma_T$.

Table 2. Cross Sections for Fe

GRP	SIGT	SIGST	PNUP	PNABS	GAMGEN	NU* FIS	Downscatter Probability
1	1.445E−01	1.425E−01	0.0	0.9863	1.0099	0.0	0.5858 0.2492 0.0743 0.0458 0.0369 0.0060 0.0015 0.0005 0.0000 0.0000 0.0000
2	1.312E−01	1.309E−01	0.0	0.9976	0.4039	0.0	0.7808 0.1220 0.0774 0.0182 0.0015 0.0001 0.0000 0.0000 0.0000 0.0000
3	1.078E−01	1.077E−01	0.0	0.9991	0.1683	0.0	0.8063 0.0987 0.0854 0.0096 0.0000 0.0000 0.0 0.0 0.0
4	1.279E−01	1.277E−01	0.0	0.9984	0.0088	0.0	0.9521 0.0434 0.0009 0.0033 0.0003 0.0 0.0 0.0
5	1.459E−01	1.456E−01	0.0	0.9985	0.0041	0.0	0.9911 0.0089 0.0 0.0 0.0 0.0 0.0
6	4.218E−01	4.213E−01	0.0	0.9988	0.0000	0.0	0.9937 0.0063 0.0 0.0 0.0 0.0
7	2.373E−01	2.367E−01	0.0	0.9976	0.0052	0.0	0.9537 0.0463 0.0 0.0 0.0
8	3.746E−01	3.717E−01	0.0	0.9924	0.0164	0.0	0.9893 0.0107 0.0 0.0
9	4.590E−01	4.560E−01	0.0	0.9934	0.0149	0.0	0.9845 0.0155 0.0
10	4.630E−01	4.560E−01	0.0	0.9848	0.0334	0.0	0.9699 0.0301
11	5.238E−01	4.560E−01	0.0	0.8706	0.2930	0.0	1.0000
12	1.111E−01	1.755E−01	0.0	1.5798	0.0	0.0	0.0192 0.0599 0.0597 0.0419 0.8193
13	1.106E−01	1.545E−01	0.0	1.3970	0.0	0.0	0.0675 0.1282 0.0782 0.7262
14	1.300E−01	1.478E−01	0.0	1.1362	0.0	0.0	0.1918 0.2089 0.5993
15	1.829E−01	1.838E−01	0.0	1.0049	0.0	0.0	0.2682 0.7318
16	6.355E−00	3.301E−01	0.0	0.0519	0.0	0.0	1.0000

Table 3. Neutron to Gamma Transfers for Fe

Neutron Group	GAMGEN	Transfer Probabilities
1	1.0099E−00	0.0000 0.0171 0.2787 0.2784 0.4258
2	4.0395E−01	0.0 0.0006 0.1026 0.1632 0.7336
3	1.6828E−01	0.0 0.0040 0.0043 0.0025 0.9892
4	8.7923E−03	0.0 0.1262 0.1361 0.0397 0.6980
5	4.1052E−03	0.0 0.2590 0.2745 0.0794 0.3871
6	2.3119E−05	0.0 0.2642 0.2720 0.0779 0.3859
7	5.2124E−03	0.0 0.3695 0.2707 0.0781 0.2818
8	1.6371E−02	0.0 0.3760 0.2039 0.0906 0.3295
9	1.4887E−02	0.0 0.3623 0.1981 0.0985 0.3411
10	3.3447E−02	0.0 0.3623 0.1981 0.0985 0.3411
11	2.9303E−01	0.0 0.3605 0.1979 0.0991 0.3426

The downscattering table indicates that for the first neutron group 58.58% of scatters will be within group 1, 24.92% will be from group 1 to group 2, and there will be no scattering from group 1 to the last three neutron groups. Table 3 breaks the gamma-ray production from each neutron group into probabilities for each of the five gamma-ray groups. The cumulative probabilities and scattering angle cosines are given for each group-to-group transfer in Table 4. With a probability of 0.5003, a neutron scattering from group 1 to group 2 has a scattering angle cosine of −0.5767, and with a probability of 0.4997, this cosine is 0.5892. The −1.0 probability signals the code that the distribution is isotropic.

Fission neutrons are created isotropically in MORSE. Secondary gamma rays are also produced isotropically unless the method described in the next section is implemented, in which case any anisotropic gamma-ray production data are used.

Table 4. Scattering Probabilities and Angles for Fe

GRP	to GRP	PROB	ANGLE	PROB	ANGLE
1	1	0.8951	0.8905	1.0000	−0.4449
1	2	0.5003	−0.5767	1.0000	0.5892
1	3	−1.0000	0.0	0.0	0.0
1	4	−1.0000	0.0	0.0	0.0
1	5	−1.0000	0.0	0.0	0.0
1	6	−1.0000	0.0	0.0	0.0
1	7	−1.0000	0.0	0.0	0.0
1	8	−1.0000	0.0	0.0	0.0
1	9	−1.0000	0.0	0.0	0.0
1	10	−1.0000	0.0	0.0	0.0
1	11	−1.0000	0.0	0.0	0.0
2	2	0.6948	0.7976	1.0000	−0.5723
2	3	0.5342	−0.6299	1.0000	0.5556
2	4	−1.0000	0.0	0.0	0.0
2	5	−1.0000	0.0	0.0	0.0
2	6	−1.0000	0.0	0.0	0.0
2	7	−1.0000	0.0	0.0	0.0
2	8	−1.0000	0.0	0.0	0.0
2	9	−1.0000	0.0	0.0	0.0
2	10	−1.0000	0.0	0.0	0.0
2	11	−1.0000	0.0	0.0	0.0
3	3	0.6547	0.7456	1.0000	−0.6280
3	4	0.5025	−0.6085	1.0000	0.5602
3	5	−1.0000	0.0	0.0	0.0
3	6	−1.0000	0.0	0.0	0.0
-	-	-	-	-	-
-	-	-	-	-	-
14	14	0.8601	0.9697	1.0000	0.8682
14	15	0.5455	0.8913	1.0000	0.7158
14	16	0.5514	0.4744	1.0000	−0.5744
15	15	0.7700	0.9572	1.0000	0.8189
15	15	0.6310	0.6157	1.0000	−0.5332
16	16	0.6251	0.7139	1.0000	−0.5545

III. MODES OF OPERATION

All MORSE input is divided into three parts, each with its own energy group nomenclature:

1) Cross-section preparation (MORSEC or XCHEKR) — group structure is equal to or a subset of that found on the cross-section library.

2) Random walk (including geometry input) — group structure is equal to or a subset of that created in the cross-section preparation.

3) Estimation (SAMBO analysis) — group structure is equal to or a subset of that used in the random walk.

Table 4.1 and Section 4.4.4 (page 4.4-60) in the manual[1] explain these concepts in terms of input variables. Efficient manipulation of these variables can greatly reduce computer memory requirements.

III.A. Primary and Secondary Particles

Although it is convenient to think of primary and secondary particles in terms of neutrons and gamma rays, a gamma-ray-only problem can be run in the primary particle mode. In fact an entire coupled neutron-gamma-ray calculation can be made in the primary particle mode if the energy group data for all the above three processes so indicate. This is because MORSE, with its multigroup structure, treats neutrons and gamma rays exactly the same for all processes (as in a discrete ordinates code). The only distinction that can be made is in the group number, and gamma-ray production can be treated simply as another group-to-group transfer (neutron group-to-gamma group). In effect, a neutron stops and a gamma ray begins, but because of the particle weighting and probability selection, the relative neutron and gamma-ray populations remain correct. In the two-particle mode, the generated gamma rays are stored in the particle "bank" and processed later. This bank is a portion of computer memory which contains all pertinent parameters (physical and otherwise) for each particle to be processed. The bank is initially filled from the source parameters (input or special routine). The source input energy spectrum differential for the two particle types is also determined solely by the group number, although primary source gamma-ray calculations can be made separately from the neutrons for a mixed-particle source problem. The secondary gamma rays from neutron production can also be run separately through use of the history file feature (see Section III.F). This would be necessary if one wanted to make some changes, such as in importance region specification for gamma-ray transport different from that for neutron transport and gamma-ray production.

Although it may not always be desirable, there are several advantages to running both particle types as primary particles. The gamma-ray production probabilities (input cards 0) are not needed. Gamma-ray groups are selected as for any other neutron group transfer and any anisotropic production data are utilized as for an anisotropic scattering. In the two-particle mode these input data are the probabilities of gamma-ray production and can

be used to control (bias) the number of secondary gamma-ray particle histories in each group and importance region. In the one-particle mode the energy and region importances (cards K) can be used for the same effect. Reference 17 gives a study of these and other possible methods of gamma-ray production in MORSE.

A special uncollided routine (SGAM, see Section V.E.5) is not needed in the one-particle mode for next-event estimation. The uncollided contributions from the secondary gamma rays are made automatically (group-to-group transfer) in the same routine (RELCOL) for the scattered neutrons and gamma rays. Differentiation between the neutron and gamma responses in any mode of operation is accomplished by setting the NNE and NE values (cards BB) appropriately. The neutron response function is set to zero in the gamma-ray groups, and the gamma-ray response function is set to zero in the neutron groups (cards FF). If both response functions are combined into one set, then the results will be combined.

If the primary particle mode is used for time-dependent neutron-gamma-ray calculations, a small amount of special programming is required. In the two-particle mode the velocities for primary (neutron) energy groups are determined from the group energy limits (cards F) and those for secondary particles are set equal to the speed of light. In the primary-particle mode, the gamma-ray velocities must be reset to the correct value in blank common (these values are the second set of group dependent data, see Figure 4.2 in the manual[1]). The programming could be done in a special SOURCE routine or in SCORIN or STRUN (see Sections 4.6.22 and 4.6.26 in the manual).

For energy-dependent neutron-gamma-ray flux calculations, the last neutron group value for the one-particle mode will be in error due to division by the incorrect ΔE. For time-independent calculations and ΔE independent fluxes (i.e., group flux not divided by ΔE), the group energy limits (cards F) can be input as unity. Only group numbers, not energy values, are used in the normal operation of the code.

III.B. Fission Neutron Generation

Fission neutron production is treated in MORSE in much the same manner as gamma-ray production is treated. Fission neutrons and secondary gamma rays may be produced and analyzed simultaneously in the same calculation. The FWLOW value (card M) controls the number of fission neutrons produced by importance region. This number is inversely proportional to the value of FWLOW. A fission spectrum χ (cards N) must be input for each medium with a fissile nuclide. For subcritical systems with a fixed source, this source is used to start each batch, and any fission neutrons produced are stored in the bank and processed later in that batch. This process leads to very long batch calculation times for systems close to critical, since the fission neutrons produce other fissions, etc. The calculation must now be done in the eigenvalue (k_{eff} calculation) mode. All values on card L are set ≥ 1. The neutrons produced in one batch (now a generation) become the source for the next batch. The k_{eff} is the ratio of the fission weight produced in a batch to the batch source weight. A MORSE k_{eff} calculation will generally take more computation than that for a criticality code such as KENO V, since MORSE must

make many code decisions for each neutron regarding the various general options available. For both subcritical and supercritical systems, MORSE keeps the number of neutrons per batch approximately constant by appropriate weight adjustments. The initial batch source spatial distribution must be created from a user-written SOURCE routine (Section V.D). Several batches should be skipped for source convergence before compilation of any results begins. As with any Monte Carlo eigenvalue calculation, the average answers and statistical uncertainty will contain an unremovable theoretical bias, which is not important for most practical applications.

It is also possible to calculate true time-dependent values of k_{eff} and other quantities — in contrast to the usual static mode treatment of the previous paragraph. Here a special BANKR routine is needed (one of the sample problems has this feature, see Section VII). The normal "history-batch" concept takes on a "batch-run" concept (NQUIT on card B is 1 in the static mode). The batches become "histories" and the multiple "runs" (NQUIT values) become "batches" for the accumulation of average results and statistical uncertainty. Some time cutoff must be set (card C) and a large number of batches (NITS) set so that all neutrons reach the time cutoff before the number of batches is exhausted for each "run." Here the time (AGE) of each neutron and its descendants continues to increase from batch to batch. A run is terminated following the batch for which the first neutron time cutoff occurs, and the next run is initiated. This procedure is necessary since there is no correlation between a neutron AGE and the batch (generation) it is in, and there may be a great dispersion in the AGE of neutrons in the same batch. The inclusion of any batches following that with the first time cutoff would bias the result. The batch skipping feature at the beginning of a run must still be utilized to allow correct convergence from the initial source distribution. A detailed description of this and other MORSE eigenvalue capabilities are given in Reference 18.

III.C. Albedo Capability

The default albedo option in MORSE is a specular (mirror image) reflection for a geometry medium number as indicated on card B. This option is applicable only for some symmetry or infinite dimension feature of the system. With reflection, less geometry input data may be needed, but the computation will generally remain unchanged. Source normalization and estimation procedures must be scrutinized to make sure the results will be unchanged from those if no reflection is used. Any next-event estimation is usually incompatible with this option except for a point detector located on (or very near) an axis of symmetry.

The user is free to write his own albedo scattering routine ALBDO, called from subroutine MORSE in the code. A collection of such routines for general use has been assembled and is described in the BREESE report.[19] The CARP[20] code provides DOT-generated data for use in the BREESE routines. These data are expensive to generate and the following albedo scattering data sets have been compiled:

1. 30.48 cm of water,
2. 30.48 cm of ordinary concrete,

3. 22.86 cm of carbon steel,
4. 30.48 cm of concrete covered by 1.27 cm of steel.

Application of the BREESE and CARP capabilities have been favorably compared with experimental results for multilegged ducts and pipes.[21]

MORSE may also be run in an albedo-only mode; i.e., no cross sections are input. This is usually a void-albedo medium geometry. The user must specify all particle distributions at the albedo surface interface in his special routines. This technique takes advantage of the MORSE geometry capabilities and has been used for transport studies of several uncharged, non-nuclear "particles" either for one-group or multigroup problems. Among these phenomena are gases, heat quanta, and material sputtering from first-wall fusion devices.[22,23] Monte Carlo heat transfer has also been investigated in more general detail.[24]

III.D. Adjoint Capability

Due to its multigroup energy structure, MORSE has had much application in the adjoint mode. All that is needed is an indicator set on card B (also set in XCHEKR if the cross sections are pre-processed). All input is done in the same manner as for a forward calculation. Group (energy) dependent inversions and matrix transposition are done automatically. For fission problems, the χ and $\nu\Sigma_f$ are also interchanged by the code. The user must input the adjoint source spectrum (forward response) on cards E in the forward structure. Likewise, the adjoint response (forward source spectrum) must be input on cards FF in the forward structure. For a two-particle mode, coupled neutron gamma-ray adjoint calculation, the GWLOW values (cards O) are input for the gamma groups in the forward structure.

These simple procedures will yield adjoint calculated results that are the same (within statistical uncertainty) as those for a forward calculation only for a physical source and detector which are points in space and isotropic in angular dependence. Otherwise, special attention must be given to the adjoint source selection, normalization, and estimation processes, which are usually more complicated than those for a forward calculation (see Section VI). The calculated adjoint energy spectra are not divided by ΔE and correspond to those from an adjoint discrete ordinates calculation. It is a simple procedure to obtain forward flux spectra from an adjoint multigroup Monte Carlo calculation (see Section VI.G). For coupled neutron-gamma-ray adjoint calculations, the primary-secondary particle connotation reverses. Primary adjoint gamma rays now produce secondary adjoint neutrons. Also, the designation of the various energy group limits mentioned at the beginning of Section III can become confusing, especially when one is trying to skip groups for computer memory conservation in either of the two neutron-gamma-ray modes. It is a good idea to run a forward cross-section processing and compare the Σ_T and Σ_T^* for verification of the group order. The total cross sections should be identical, and the extra calculation time is insignificant.

In observing the output of a MORSE adjoint cross-section processing calculation, it is seen that the reversed total cross-section column SIGT is identical to that for a forward case as in Figs. 2—4. All other values will in general be much different (non-physical in the forward sense). The scattering cross section is the sum of the adjoint downscatter; i.e., $\Sigma_g^* = \sum_{g'} \Sigma_{g' \to g}$, whereas the forward value is $\Sigma_g = \sum_{g'} \Sigma_{g \to g'}$. For some nuclides, or mixture of nuclides, the Σ_g^* can be much greater than Σ_T. This is especially true in the lowest numbered adjoint neutron groups (lowest neutron energies). Multiple scatter in these groups can lead to prohibitively high adjoint particle weights since at each collision the weight is multiplied by PNABS $= \Sigma_g^*/\Sigma_T^*$. A method to alleviate this problem has been proposed and will soon be incorporated into MORSE.[25]

III.E. DOMINO Coupling Code

DOMINO[26] is a much used code which couples DOT discrete ordinates calculations with MORSE. All special routines for normal operation (source, estimation, etc.) have been created. A standard application is to utilize the forward DOT leakage from a cylindrical reactor as a forward source for MORSE at some core-shield interface in order to investigate some streaming path or other geometric anomaly in the shield. In choosing the coupling interface, care must be taken to have some overlap in the geometry of the two calculations so that there is not a coupling at two vacuum boundaries; i.e., any backscatter across the interface must be accounted for.[27] The MORSE geometry may be in any position relative to that of DOT. In addition to cylindrical (R,z) geometry, DOT (R,θ) and (X,Y) geometries may also be used. But in each case the MORSE geometry must be constructed relative to the DOT two-dimensional symmetry; i.e., for a cylinder the three-dimensional MORSE geometry must surround the DOT cylindrical axis. A separate calculation must be made for each DOT surface, if necessary — for cylinder side, top, and bottom calculations. A separate DOT boundary flux file must be written for each surface. For a forward-forward coupling, the MORSE calculation is broken into three parts with a separate source for each surface (see Section V.I for combining MORSE results).

An adjoint DOT flux file may also be used as a forward MORSE scoring function. In this case MORSE input response functions are not used. Use of the history file procedure (special BANKR routines) precludes the need for multiple MORSE calculations for side, top, and bottom cases. One MORSE calculation and then three coupling runs are made.

Other calculational modes are possible — a DOT adjoint source for a MORSE adjoint calculation, and a DOT forward response for a MORSE adjoint. It is also possible to combine calculational sequences:

forward DOT (source) → forward MORSE ← adjoint DOT (response)

or

adjoint DOT (source) → adjoint MORSE ← forward DOT (response).

In the source mode DOMINO creates a current-like source for MORSE by multiplying the flux by the appropriate angle interval cosine and then integrating. To take advantage of the same programming detail, the response mode coupling initially creates the same source-like term, later differentiates it, and at coupling divides by the appropriate cosine as determined by the particle crossing the coupling surface, not the discrete ordinates cosine. This procedure can lead to a numerical error ($\sim$15% for small DOT quadrature) which can be eliminated by removing the cosine multiplication in subroutine ADDEM and the cosine division at coupling in subroutine BDRYX. The 4π factors that appear throughout the DOMINO coding are due to the fact that DOT results are on a per unit solid angle basis, not 4π steradians.

III.F. History File

MORSE has the capability to write a history file (collision tape) for post-processing of any random walk-generated data independent of the random walk. These data can be used for various user-written routines, DOMINO coupling, secondary gamma-ray transport, flux spectra plots, or collision density plots.[28] Although collision density information by medium and importance region is available as standard output in tabular form, visual plots are helpful in complicated geometries. This history file can be small, as for information at one boundary for coupling, or it may be voluminous, as for next-event estimation where information from all source and collision sites must be included in addition to geometry and cross-section data. The option is activated in MORSE from input cards B and G. The available event types and variables are given in Tables 4.2 and 4.22 (page 4.6-30) in the manual.[1] Special BANKR routines are available for direct use or modification for each application of writing and reading a history file.

III.G. Variance Reduction

MORSE contains many of the standard variance reduction techniques. Among these are:

- survival biasing,
- source energy biasing,
- collision energy biasing,
- splitting and Russian roulette,
- exponential transformation,
- non-leakage biasing,
- delta scattering,
- next-event estimation,
- fission neutron production control,
- secondary gamma-ray production control, and
- energy group and time cutoffs.

These procedures are in general implemented by energy group and importance regions. Theoretical details have been given elsewhere.[29] Although the activation of these options by standard input is relatively simple, the determination of the necessary input data is usually one of the most difficult of any of the tasks associated with a Monte Carlo calculation.

Survival biasing (non-absorption weighting) is fixed in the code, but analog capture can be implemented by replacing the Σ_S/Σ_T weight (WATE) multiplication in subroutine COLISN with a random number test. Source and collision biasing, if activated, bias the outgoing energy from each event. Splitting and Russian roulette are implemented in MORSE following each source and collision event. Boundary splitting and Russian roulette may be activated by calling subroutine TESTW from BANKR(7). In making changes such as these (see Section V.L), it is instructive to examine the executive subroutine MORSE (see pages 22-25 in Ref. 6). The use of the exponential transform requires the user-written function DIREC (see Section V.J). The non-leakage option is useful in small systems but can lead to long running times if histories are not sufficiently terminated by escape or other means. Delta (fictitious) scattering for use in increasing the collision density in thin regions, or eliminating complicated geometry ray tracing, is implemented through the use of a special NXTCOL routine. The largest total cross section Σ_M for the current group is chosen, and a collision is real with probability Σ_T/Σ_M and fictitious with probability $(1-\Sigma_T/\Sigma_M)$. Several routines exist for use of next-event estimation (see Section V.E). Fission neutron and gamma-ray production have been discussed previously in this section. The energy group cutoffs are NGPQTN and NGPQTG on card B, and the time cutoff is on card C.

III.H. Input/Output

In addition to the output of the desired responses and fluxes and their fractional standard deviations, much information about the calculation is available which aids in determining the effectiveness of any variance reduction options. The weighted average of all source parameters starting a batch are given: (1) weight, (2) group, (3) direction cosines, (4) position, and (5) time. A summary of event types (integral count, not weighted) is given at the end of the batch as follows: (1) source (will differ from WTAVE for source biasing or if WTSTRT $\neq$ 1), (2) splits, (3) collisions in fissionable media, (4) secondary particle generation (includes fission neutrons), (5) real (scattering) collisions, (6) encounters with albedo media, (7) interior boundary crossings, (8) escapes (outer boundary), (9) group cutoffs, (10) time cutoffs, (11) Russian roulette kills, (12) Russian roulette survivals, and (13) secondary particles lost (secondary gamma rays or fission neutrons) due to insufficient computer memory available (increasing NMOST relative to NSTRT on card B will eliminate this problem). Splitting will be omitted in this case if NMOST is too small.

Following the output of the results, most of the information is summarized for the entire problem. The most important output for adjusting variance reduction options is that given for each group and importance region in terms of (1) scattering, (2) fission, (3) secondary gamma production, (4) splitting, and (5) Russian roulette. The user can add any desired results to this output through the creation of user routines.

The computer memory parameter is set in the main (first) routine in COMMON NC. Various places in the output print the amount of memory used or available for the rest of the calculation. Pre-processing of the cross sections reduces the memory needed for the rest of the calculation. The MORSE manuals[1] give much information regarding various computers and code memory requirements. An explanation of all error and warning messages is also given.

Various input instructions have been described in other sections in connection with specific options. The MORSE input instructions are relatively simple as compared to other general Monte Carlo codes because much of its versatility is included in the user-written routines. The most difficult input in MORSE other than the importance parameters is that for the geometry data described in the next section.

IV. GEOMETRY

MORSE has two geometry specifications: MEDIUM number for chemical compositions and REGION number for importance regions. A medium 1000 corresponds to an internal void, and a medium 0 is an external void, which terminates a history (causes an escape). Source points and estimation detector points cannot be in medium 0. It is convenient to surround the geometry system first by medium 1000 and then by medium 0. This precludes calculation termination due to some small undefined volume. A medium 0 can be used internally in a system as a pure absorber. A next-event estimation trajectory for this or any concave system will terminate upon encountering medium 0 and give an incorrect value of the exponential attenuation. An error will result if this estimation encounters an albedo medium unless the appropriate BREESE routine is used. If the cross-section media numbers do not correspond to those of the geometry for any reason, a user-written routine GTMED(GM,XM) can eliminate input changes. Here GM is the geometry medium the particle has encountered, and XM is the cross-section medium to be used there. Void media, 0 and 1000, and albedo media cannot be inserted for XM in GTMED. This routine can be very simple or very elaborate to allow any desired geometry-cross-section effect, e.g., to utilize extra processed cross sections (see Sect. II.B) in a following calculation without changing the geometry input data.

IV.A. Combinatorial Geometry

The combinatorial geometry (CG) is the basis of MORSE geometry. The system must be described by combinations of the following geometric bodies:

- right parallelepiped,
- sphere,
- right circular cylinder,
- right elliptic cylinder,
- ellipsoid,
- right cone (truncated or not)
- right angle wedge,
- arbitrary polyhedron of 4, 5, or 6 sides, and
- circular torus.

Each of these bodies may be arbitrarily oriented in space. In the input data the parallelepiped is called BOX, and there is a coordinate-fixed body called Right Parallelepiped (RPP). In the CG associated with MARS (next section) the right parallelepiped BOX and the right angle wedge WED or RAW may be rotated by coordinate axis designation or also by angle (BPP and WPP).

Bodies are combined into volumes called "zones" by three operators:

- combination of bodies using an "OR" operator,
- exclusion of bodies using a minus ("−") operator, and
- overlap of bodies using a plus ("+") operator (or blank default).

Figure 1 illustrates these operations for the six possible volumes from two spheres:

a. All volume that is in *either* sphere 1 or sphere 2.

b. All volume that is in *both* sphere 1 *and* sphere 2.

c,d. All volume that is in one sphere *but not also* in the other sphere.

e,f. All volume that is in either sphere *without regard* to the other sphere.

Consider a more complicated case, as shown in Fig. 2. Here a two-legged cylindrical duct has a rounded outer surface at the intersection of the two legs (one-quarter of a sphere). The zones, indicated as circled numbers, are:

1. outside the duct,
2. duct wall (three shaded volumes), and
3. inside the duct.

The bodies, indicated as uncircled numbers, are:

1. small sphere,
2. large sphere,

3,4. wedges,

5,7. small cylinders,

6,8. large cylinders, and

9. right parallelepiped (outer-most body).

The large and small spheres and cylinders have corresponding equal radii. Each cylinder extends only to a plane defined by the center point of the spheres. The wedges are needed only to define the intersections of the cylinders and they must completely enclose the cylinders. Here wedge 3 is larger than wedge 4 since the horizontal cylinders are larger than the vertical ones. Any of the geometry bodies with a plane surface and correct orientation could be used for the cylindrical intersection definition. If it were desired to have a rounded surface at the inner side of the cylindrical intersections, then use of the torus body would be necessary.

The zones are described in the figure so that the spheres and cylinders do not overlap but define the desired intersections.

- Zone 1 is all volume that is inside body 9 but outside the large sphere and cylinders (bodies 2, 6, and 8).

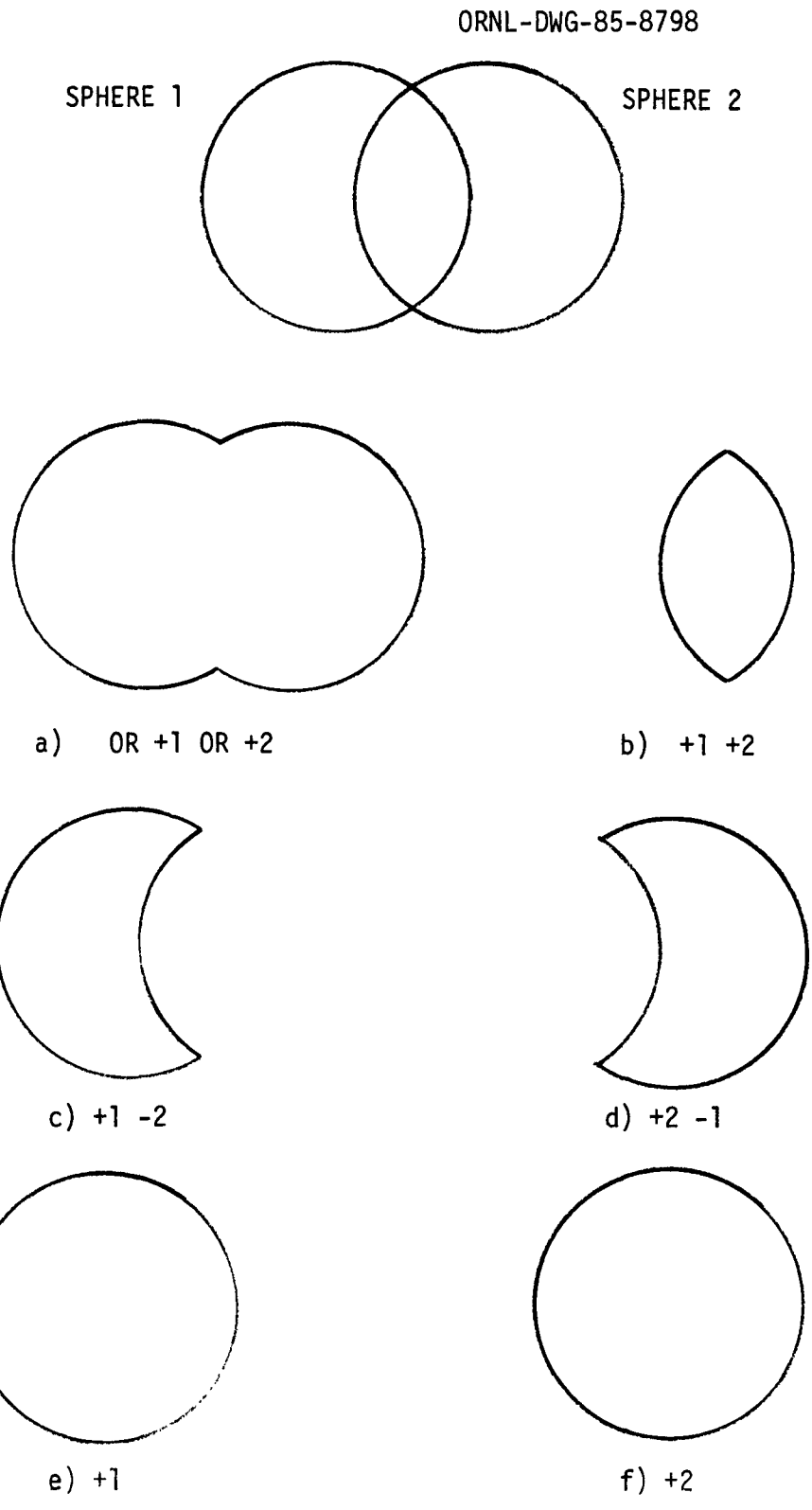
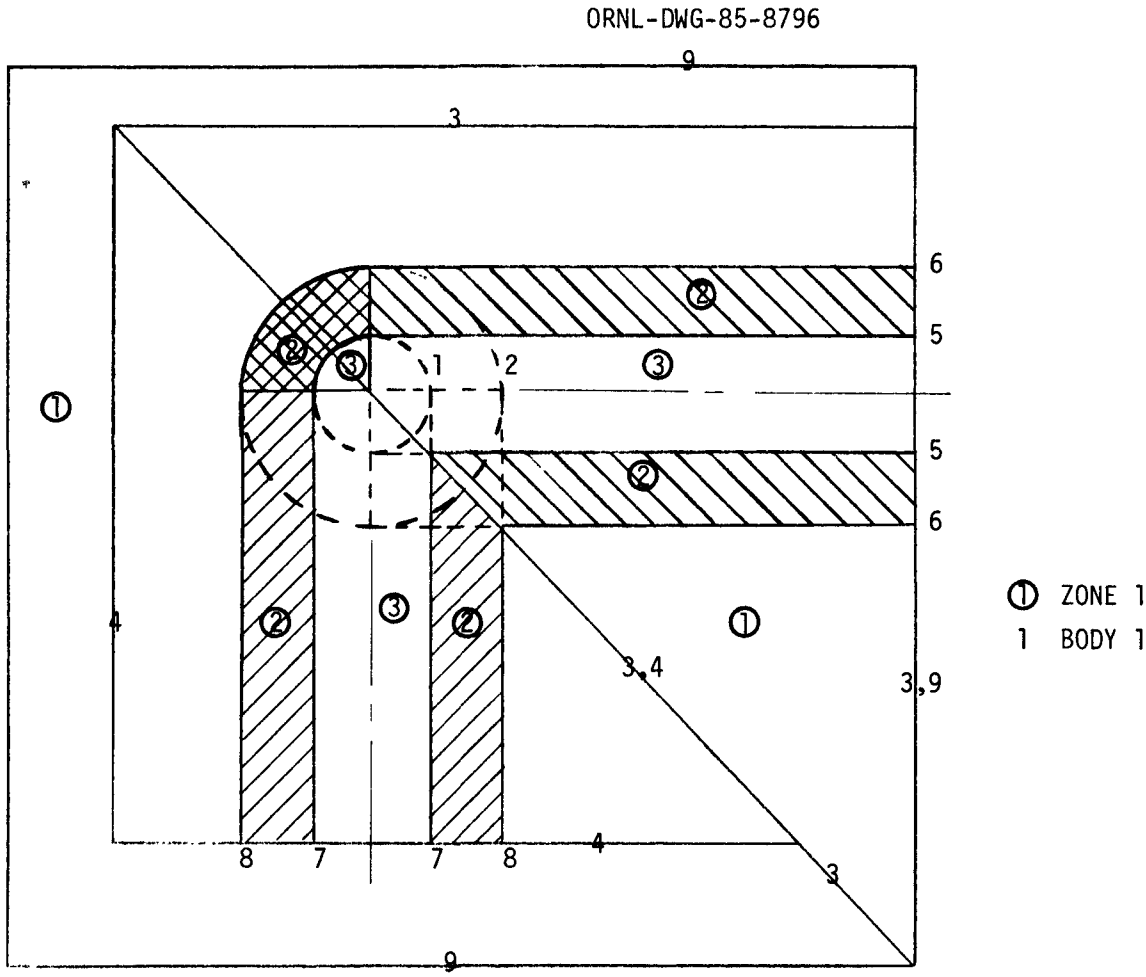


Fig. 1. Combinatorial Geometry Operations.



① +9 -2 -6 -8

② OR +6 -5 +3 OR +8 -7 +4 OR +2 -1 -6 -8

③ OR +5 +3 OR +7 +4 OR +1 -5 -7

Fig. 2. Combinatorial Geometry Zone Descriptions.

- Zone 2 is all volume that is either (OR) inside the large horizontal cylinder (body 6), outside its inner cylinder (body 5), and inside the wedge (body 3) (use of the wedge excludes the cylinder overlap volume, i.e., the volume inside the cylinder but outside the wedge); *OR* inside the large vertical cylinder (body 8), outside its inner cylinder (body 7), and inside the wedge (body 4); *OR* inside the large sphere (body 2), outside the small sphere (body 1), and outside the two large cylinders (bodies 6 and 8) (since the small cylinders are contained within the large cylinders, their exclusion from the spheres is implicit with the exclusion of the large cylinders).
- Zone 3 is all volume that is either (OR) inside the small horizontal cylinder (body 5) and inside the wedge (body 3); *OR* inside the small vertical cylinder (body 7) and inside the wedge (body 4); *OR* inside the small sphere (body 1) and outside the small cylinders (bodies 5 and 7).

In the zone definitions in Fig. 2, the "+" and "−" operations are performed first, and then the "OR" operations. All operations between OR operators (and after the last OR) define a "code zone," in contrast to the larger zone (sometimes called "input zone"). Media and regions (and universes, see next section) are defined by (input) zones; i.e., each of the three zones will be given a medium number corresponding to the appropriate cross-section medium and a region number corresponding to that for importance (Russian roulette, splitting, etc.).

Code zones can be removed from an input zone and defined separately as other input zones. For example, if the different shaded areas of the duct wall were different cross-section media, then Zone 2 would have to be broken into three separate zones without the OR operators. This would also be the case if the duct wall were all one medium but three different importance regions. If it were necessary to have more importance regions, then more zones would have to be created with more body designations just for that purpose. If Zones 1 and 3 were the same medium and region, then they could be combined into one zone with four OR operators. The last two entries in Zone 3 (-5 and -7) are not necessary. However, if Zone 3 were broken into three separate zones of different media or regions, these items would be required. It may sometimes be beneficial for efficiency of computation time or problem setup clarity to create extra zones not required for media or region specification. If one large zone is adjacent to many smaller zones, the code may run faster if the large zone is divided into a few smaller ones. This is due to tracking decisions internal to the geometry coding. Other examples of CG input are given in the MORSE manual[1] (and also in Ref. 30), along with explanations of error messages and other comments about the geometry module subroutines.

IV.B. Array Geometry (MARS)

The MARS multiple array geometry module of the SCALE system has been recently incorporated into MORSE-CG (now designated MORSE-CGA). This feature is similar (but with some differences) to that of KENO V and KENO V.a.[9] It is possible with MARS to model very complicated geometries with very little effort when much of the geometry is made up of repeating segments, such as in a large commercial reactor.[31] Figure 3 shows a computer-drawn MARS geometry for a (very) simple reactor core and a domed containment structure. However, if the core had contained 500 identical elements instead of 5 as shown in the figure, the input description would have been only very slightly more complicated. In the input description, Fig. 4, the number of CG body types (10) would not increase; only the array data (cards 25 and 26) would change to reflect the increase to 500 elements.

MARS geometry can be constructed in modular form with as much detail given to each module as necessary. Any repeating modules or segments need to be modeled only one time in the regular CG method (see preceding section) in any convenient orientation or translation of coordinates for ease of input data. These segments are then placed in an array in any number, order, or orientation. Each of these different segments is called a "universe" and is numbered starting from 1. The "zero universe" is the physical coordinate system of the problem. Each zone in CG must have a universe number just as for medium and region number. If there are no repeating segments, everything is in the zero universe.

Array 1 contents are: ORNL Dwg. 81-5763

FOR Z = 1

Y \ X	1	2	3
3	0	1	0
2	1	1	1
1	0	1	0

Array 1 is a 3 by 3 by 1 array.
"0" is a vacant cell position.
"1" refers to Universe 1.

AN IRREGULAR ARRAY

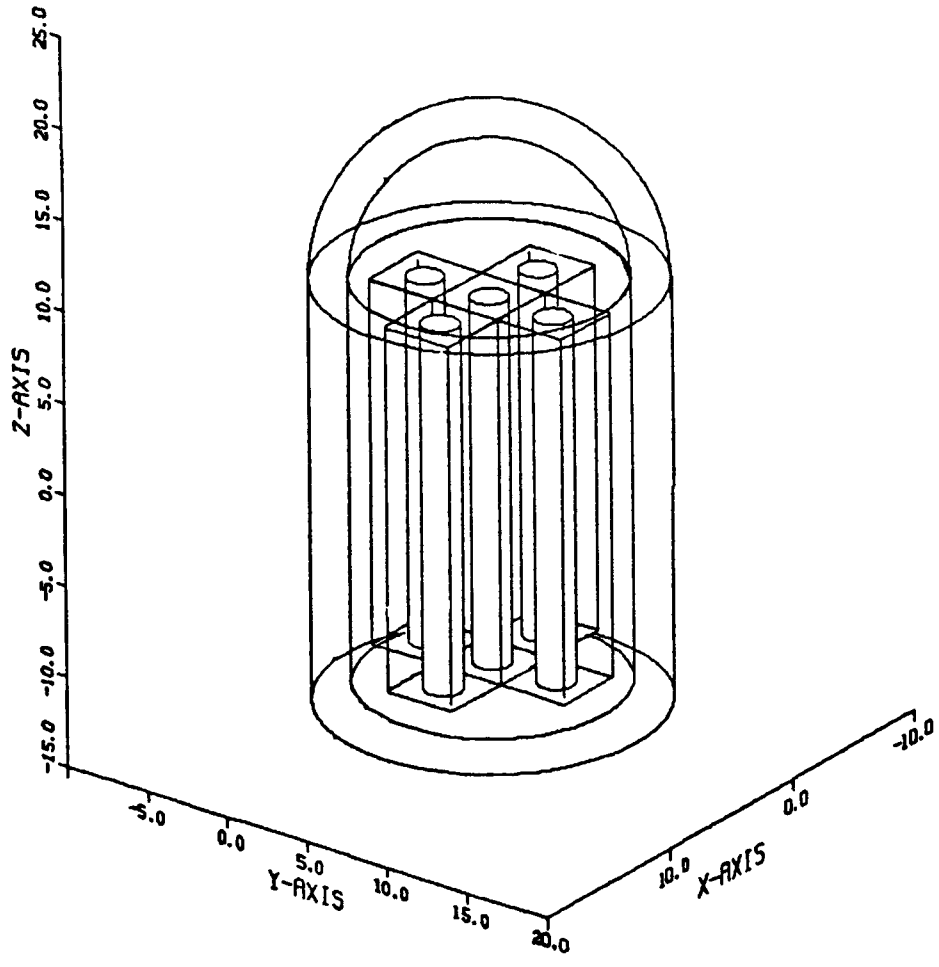


Fig. 3. MARS Array Geometry Example (JUNEBUG Plot).

```
                                THE FOLLOWING IS A LIST OF CARD IMAGE INPUT

CARD NO.                                  C O L U M N  N O .

                  1111111112222222222333333333344444444445555555555666666666677777777778
         12345678901234567890123456789012345678901234567890123456789012345678901234567890

    1      10 100 10 100 2 1 0 0 0 1.0 0.40
    2     AN UNCOMMON ARRAY SHAPE
    3      3*0 100
    4     RPP   0.0 12.0 0.0 12.0 -10.0 10.0
    5     RCC   6.0 6.0 -11.0 0.0 0.0 22.0 7.0
    6     RCC   6.0 6.0 -12.0 0.0 0.0 23.0 9.0
    7     RPP   4.0  8.0  0.0 12.0  -10.0 10.0
    8     RPP   0.0 12.0 4.0 8.0 -10.0 10.0
    9     RPP   -2.0 2.0 -2.0 2.0 0.0 20.0
   10     RCC   0.0 0.0 1.0 0.0 0.0 18.0 1.0
   11     RPP   -1.0E+08 1.0E+08 -1.0E+08 1.0E+08 -1.0E+08 1.0E+08
   12     SPH   6.0 6.0 11.0  7.0
   13     SPH   6.0 6.0 11.0  9.0
   14       END
   15     SPC   +8 -3 -10 OR +9 -2   OR +2 -4 -5 OR +4 -1 OR +5 -1
   16     CYL   +3 -2 OR +10 -9 -3
   17     AR1   +1  +4  OR +1 +5
   18     BDY   +8 -6
   19     ROD   +7
   20     FIL   +6 -7
   21       END
   22      6*1
   23     3*0 3*1
   24      1000 1 -1 -1000 2 3
   25      3 3 1  0  2
   26     101$$ 0 1 0 3R1 0 1 0   T
   27       1
   28       -1
   29       1
   30     AN IRREGULAR ARRAY$
   31      10.0 7.5
   32      -10.0  20.0 -10.0 20.0 -15.0 25.0
   33      8.5E+03 6.5E+03 5.0E+03
   34         0 1 0
   35      0 1
```

Fig. 4. MARS and JUNEBUG Geometry Input Description.

To define a universe (other than universe zero), a medium −1000 must surround its outer boundary, which must be a rectangular parallelepiped (BOX or RPP). The −1000 medium is never entered by a particle but signals to the code that a universe boundary has been encountered so that the necessary coordinate transformations can be performed for entrance into an adjacent universe.

The number of arrays is not limited, except by computer memory. At least one array must be in the zero universe (if any arrays are used). Universes and/or arrays may be combined and placed in other arrays. The process of placing arrays inside other arrays is called "nesting," and levels of nesting begin at 1 in the zero universe. Arrays must be defined as rectangular parallelepipeds and their lattice positions are to be filled by universes and other arrays. An array to be placed in the lattice position of another array is distinguished from a universe number by a negative sign attached to the array number. There are three different input options for designating the lattice positions of an array. In addition to universe and array positions, there is also a vacant (void) position. The

positions are filled by an X, Y, Z index. For the array in Fig. 3, there is only one Z level (Z = 1) and the X and Y positions are

	X	1	2	3
Y				
1		0	1	0
2		1	1	1
3		0	1	0

Here the 1 refers to the universe with the cylindrical element. If there had been two Z levels, i.e., if the core had been divided into two vertical levels with a second universe designation for the lower level, then in the arrangement above a 2 would replace the 1 to describe the other level.

A different and more complicated example at some intermediate Z level (e.g., Z = 4) would be

	X	1	2	3	4
Y					
1		0	1	1	0
2		3	2	-1	4
3		3	0	0	4
4		3	1	-2	4
5		3	3	4	1
6		0	1	2	0

In this example there are four universes and two other arrays, each described with its own lattice, in addition to the one being described. All Z levels will have the same 4 × 6 XY structure but the lattice positions may be filled differently. Otherwise, different Z levels are not necessary. In all CG and MARS geometry, any bodies, universes, and arrays may have identical outer dimensions while being defined as "inside" or "outside" another volume with the same dimensions. An array might contain only one XYZ position. This is a convenient method of rotating a universe or another lattice of the same size (see page M9.B.26 in reference 5). The MARS module of the SCALE manual gives many examples of geometry input data as well as much other information regarding the code operation.

IV.C. Geometry Plotting

Any but the simplest geometry input descriptions in MORSE should be checked visually. Available for this are two-dimensional plots from PICTURE (part of the RSIC package) and three-dimensional JUNEBUG plots from the SCALE code system. The PICTURE plots are very quickly made, inexpensive, and available in the form of computer listing printouts as well as other methods. Any plane may be cut through the geometry in any direction or orientation for examination of the geometry in terms of either media, regions, or (input) zones. These plots may be very large but cover only a tiny portion of the geometry such that even with printer characters for definition, the smallest details may be seen. Figure 5 shows such a plot for a military vehicle.

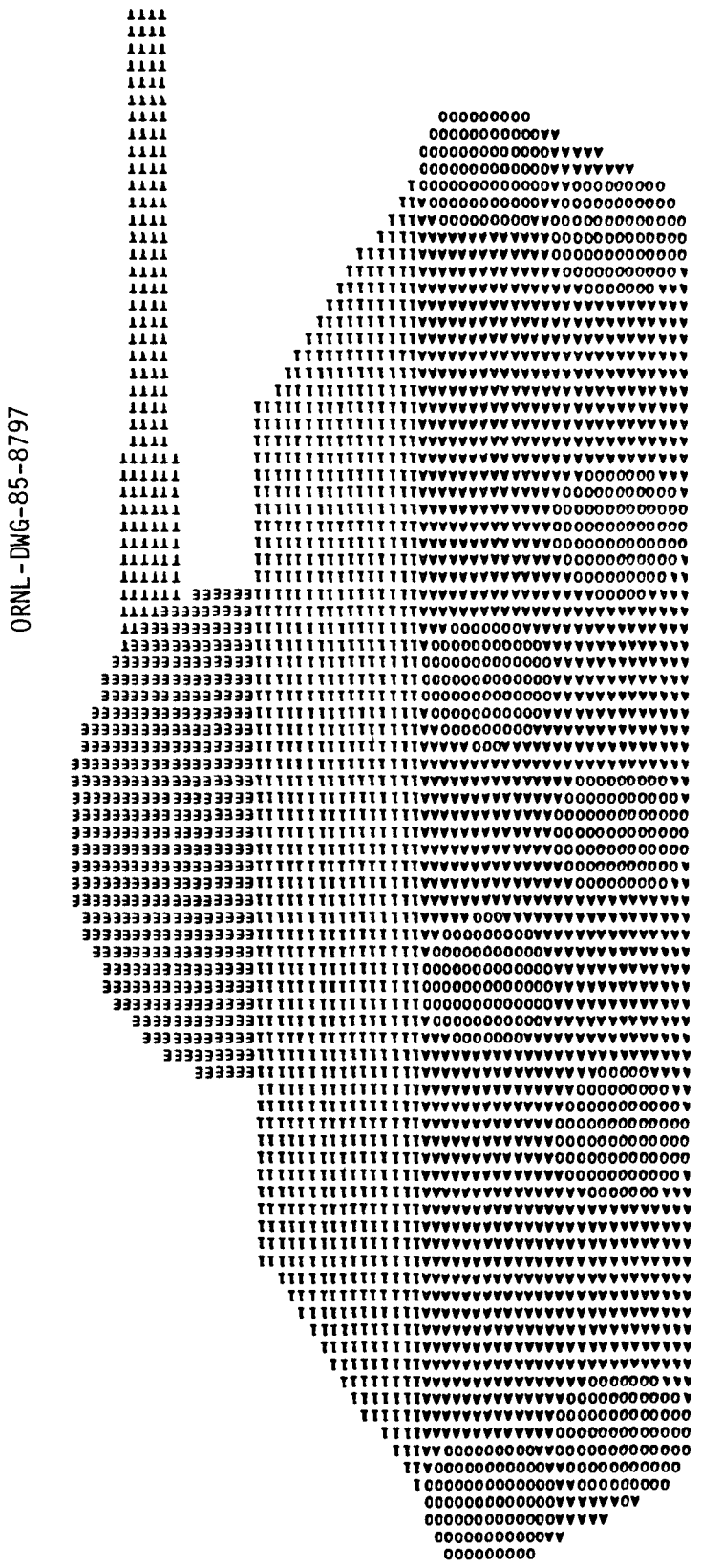


Fig. 5. PICTURE Plot Example.

The JUNEBUG plots are photograph-like drawings suitable for reports and presentations. (Figure 3 is such an example.) The plots are expensive (comparable to a MORSE calculation itself for complicated cases), depending on the line resolution requested and other variables. JUNEBUG plots are needed for geometry testing only for oblique body intersections or curved surfaces where two-dimensional PICTURE plots can be confusing, i.e., where three dimensions are required. (It is sometimes difficult to get a true two-dimensional plot from JUNEBUG.) There are many options available for JUNEBUG, and plots may be made by medium, code zone, or input zone. Any lines may be included or omitted by making these volumes invisible, transparent, or opaque. Any volume or sub-volume of the geometry may be drawn relative to any external point.

It is seen that Fig. 3 is not drawn with high resolution (some of the lines are wavy and the intersections are not sharp). Some lines have been omitted by input option for clarity. If the outer cylindrical volume had been made opaque, then only the upper portion of the array under the dome would have been visible. The JUNEBUG module of the SCALE manual gives examples and information on the workings of the code. All plots in the examples of the MARS module were drawn by JUNEBUG and thus give JUNEBUG input data as well as MARS data. Work is currently in progress or planned to up-grade both MARS and JUNEBUG in terms of computation time, memory requirements, and documentation, especially error messages and diagnostics.

It must be noted that any geometry errors acceptable by the code as valid input data will not be detected by either plotting method unless the exact erroneous volume is encountered. Many PICTURE cuts can easily miss a small error which may cause trouble in the MORSE tracking calculation if a particle encounters the incorrect volume. Further methods of geometry checking, as well as volume determination (in cm^3) for complicated shapes, are given in Section V.H.

V. USER-WRITTEN ROUTINES

It is through the use of user-written routines that the MORSE code system has its greatest flexibility. There are many examples in the RSIC code package and sample problem package. Some of these routines have been used to such an extent that they are almost standard parts of the code. Most of the required effort is directed toward the source description and estimation processes. As a result, the total MORSE input data description is much smaller than a code with little or no user-routine requirements. User routines can be very simple or very elaborate systems with calls to other routines. Almost any information concerning any part of the code can be obtained and printed out. Special data needed in a routine can be programmed directly in the FORTRAN, put in DATA statements, read the first time through the routine, or read in one of several dummy data-reading routines and passed to the user routine through a user-created common.

V.A. Physical Parameters

In writing special routines for MORSE, an understanding of the variables in several standard commons is essential. The primary common is NUTRON, and its variables for each particle are defined in Table 5. Each particle has a unique NAME (an integer). These particle parameters are stored in an area of blank common called the "neutron bank" and are processed in order of the largest NAME (reverse order from creation in SOURCE). The NAMEX is the integer associated with the original source particle and will differ from NAME for new particles created from splitting, fission neutrons, or neutron-produced gamma rays. Most of the parameters in NUTRON common given to the new particles are the same as those for the parent particle. Primary-secondary particles (neutrons or gamma rays) are distinguished only by the energy group indices, IG and IGO (see Section III.A). The physical locations in the geometry where the variables in Table 5 may change are those of (1) source events, (2) pre-collision (scattering) events, (3) post-scattering events, (4) internal (geometry media, region, or input zone) boundary crossings, and (5) outer boundary escapes. On a boundary the current (present) parameters are for the volume about to be entered and the previous parameters for the volume just left. At a collision site these are the pre- and post-collision parameters. It is the statistical weight that is ultimately used to score the desired results, and an understanding of the code and physical meanings of WATE, OLDWT, and WTBC is necessary for creating user routines.

WATE is the current particle weight and its value is initially set in subroutine SOURCE (default input value is WTSTRT, usually equal to 1.0). The OLDWT is set to WATE before each new tracking, collision, and analysis sequence begins and remains unchanged until reset before the next sequence. This resetting is done in subroutine MORSE at the same time all "old" parameters are set to "current" parameters (see page 20, Ref. 6). The other "old" parameters are in general reset at various locations in the sequence. OLDWT is little used in preference to WTBC. The WTBC is set equal to WATE after a distance to collision has been selected in subroutine NXTCOL and will remain unchanged from OLDWT unless the exponential transform and/or non-leakage biasing is invoked.

Table 5. NUTRON Common Variables

Variable	Definition
NAME	Particle's first name
NAMEX	Particle's family name
IG	Current energy group index
IGO	Previous energy group index
NMED	Medium number at current location
MEDOLD	Medium number at previous location
NREG	Region number at current location
U, V, W	Current direction cosines
UOLD, VOLD, WOLD	Previous direction cosines
X, Y, Z	Current location
XOLD, YOLD, ZOLD	Previous location
WATE	Current weight
OLDWT	Previous weight (equal to WTBC if no path length stretching or non-leakage)
WTBC	Weight just before current collision
IBLZN	Current zone number
IBLZO	Previous zone
AGE	Current age
OLDAGE	Previous age

WATE is the only variable altered by biasing and it is set to zero (and the sequence terminated for that NAME) for (1) Russian roulette kill, (2) escape from the outer boundary, (3) energy group limit cutoff, or (4) time (AGE) cutoff. At a splitting the WATE of each particle is successively halved until WATE is below WTHIR; and for Russian roulette survival, WATE is set to WTAVG. At each collision WATE is multiplied by Σ_S/Σ_T to account for absorption.

The physical analogies for the various stages of the random walk are given in Table 6. It is necessary to use WTBC for pre-collision analysis since MORSE calls the collision analysis routines after the collision. Other parameters which change after a collision are U, V, W, and IG. It is necessary to use WTBC for escapes since WATE has been set to zero.

Table 6. Physical Analogies Corresponding to MORSE Weights

o Source	WATE	=	current, J, particles/cm^3 in a volume* particles/cm^2 on a surface
o In flight	WATE	=	current, J, particles flux, ϕ, tracklength or length of flight [There is no in-flight analysis of particle histories (only at source, collisions, and boundaries).]
o Before collision	WTBC	=	collision density, $\Sigma_T\phi$, particles/cm^3
	WTBC/Σ_T	=	flux, ϕ, particles/cm^2
	$\nu \frac{\Sigma_f}{\Sigma_T}$*WTBC	=	fission neutron production, $\nu\Sigma_f\phi$, fission neutrons/cm^3
	$\gamma \frac{\Sigma_\gamma}{\Sigma_T}$*WTBC	=	secondary gamma-ray production, $\gamma\Sigma_\gamma\phi$, gamma rays/cm^3
o After collision	WATE	=	emergence scattering density, $\Sigma_S\phi = \Sigma_S/\Sigma_T$*WTBC particles/cm^3
	WATE/Σ_S	=	flux, ϕ (Σ_S at pre-collision energy)
	WTBC-WATE	=	absorption density, $\Sigma_a\phi$
o On a boundary	WATE	=	partial current, $J_{\pm}$, particles/cm^2
	WATE/$\lvert\cos\theta\rvert$	=	flux, ϕ, particles/cm^2 [θ is the angle between the particle direction and the normal to the surface.]
o Outer boundary	WTBC	=	leakage current, particles/cm^2
	WTBC/$\lvert\cos\theta\rvert$	=	leakage flux, particles/cm^2

*All quantities must be divided by appropriate volumes or surface areas to get the correct units.

V.B. Helpful Routines and Commons

In addition to NUTRON common, there are several other commons a user-written routine may need. Appropriate table (T) or figure (F) numbers in the manual[1] are indicated in Table 7 for some of these commons. There are several standard subroutines which may be called from user routines which return needed variables from these and other commons. Other routines spare the user an enormous amount of work in terms of scoring results and statistical analysis. These routines are listed in Table 8. The first three provide cross-section data as needed. PTHETA and EUCLID are used in the next-event estimation routines. LOOKZ and NORML are available for source, albedo, and boundary crossing routines. FLUXST is called from each user routine from which results are compiled. This and subsequent routines provide all bookkeeping analysis for averaging results, performing statistical analysis, and printing answers. The last three dummy routines allow the user to read in additional data, alter data read in before executions begins, and add additional output capability. A few special versions of these last three routines are distributed with the RSIC code package.

Table 7. MORSE Commons Useful for User Routines

Common		Contents*
NUTRON	—	particle parameters - T4.5
FISBNK	—	fission neutron parameters (k_{eff} calculation only) - T4.12
USER	—	additional parameters - T4.18
PDET	—	analysis data positions in blank common - T4.20
APOLLO	—	additional parameters and blank common positions - T4.4
LOCSIG	—	cross-section data positions in blank common - T4.15
ORGI	—	geometry zone number for LOOKZ call - T4.25
NORMAL	—	normal (perpendicular) vector at a boundary (4.7-19)
GOMLOC	—	geometry data positions in blank common - T4.23
blank	—	general layout - F4.2
		random walk data - T4.8, T4.9, T4.10, T4.11
		cross sections - T4.16
		analysis data - F4.6, T4.21
		geometry data - F4.18

*T and F numbers identify tables and figures in MORSE manual.[1]

Table 8. Standard MORSE Routines for Use in User Routines

Subroutine	Calling Quantities	Returned Quantities
NSIGTA	o group number o medium number	o Σ_T o Σ_S/Σ_T
FISGEN	o group number o medium number	o $\nu\Sigma_f/\Sigma_T$
GAMGEN	o group number o medium number	o γ Σ_γ/Σ_T o generated gamma-ray group number
PTHETA	o group number o medium number o scattering angle	o all group-to-group scattering probabilities for a specific scattering angle
EUCLID	o geometry location 1 o geometry location 2	o number of mean free paths between locations 1 and 2 o intermediate boundary crossing locations
LOOKZ	o geometry location	o input zone number placed in ORGI common o GOMLOC common locates medium, region, etc., in blank common
NORML	o boundary location o crossing direction Ω	o unit normal vector $\bar{n}$ placed in NORMAL common ($\Omega \cdot \bar{n} \leq 0$)
FLUXST	o contribution to result (WATE, WTBC, etc. — see Table 5) o detector number o group number o time variable o angle variable	o ultimately compiles all results and statistical uncertainty printed in output
INSCOR	o dummy routine used to read in data for user routines	
STRUN	o dummy routine for use by user after all data is read in but before a calculation (NQUIT run) begins	
ENDRUN	o dummy routine for use by user after all standard analysis (estimation) operations are complete	

In addition to the routines in Table 8, several random number generators are available for the user. These are given in Table 9 and are most often used in a source routine. The first five generators are FORTRAN functions, and the last three are subroutines. These routines are not derived from a uniform random variable, i.e., EXPRNF is not equal to $-\ell n(\mathrm{RN})$ where RN=FLTRNF, but from another more efficient, not easily recognizable technique. This is the case for many more efficient methods such as selection of a uniform random radius r in a circle of outer radius R. Selection of $r = R\sqrt{RN}$ is slower than $r = R[\max(RN_1, RN_2)]$, i.e., the maximum value of two uniform random numbers. References 32 and 33 give examples of almost any conceivable random number selection

Table 9. Random Number Generators

Function or Routine	Distribution and Interval
R=FLTRNF(DUM)	Uniform $0 \leqslant R \leqslant 1$
R=SFLRAF(DUM)	Uniform $-1 \leqslant R \leqslant 1$
R=EXPRNF(DUM)	Exponential, e^{-R} $0 \leqslant R < \infty$
R=RNMAXF(T)	Maxwellian energy in speed squared V^2, cm^2/sec^2 for temperature T ($V^2=2E/m$), $0 \leqslant V^2 < \infty$
R=FISRNF(DUM)	Watt energy spectrum in cm^2/sec^2 for ^{235}U, $0 \leqslant V^2 < \infty$
AZIRN(SIN,COS)	Sine and cosine of a random azimuthal angle θ (θ uniform on $0 \leqslant \theta \leqslant 2\pi$)
POLRN(SIN,COS)	Sine and cosine of a random polar angle θ ($\cos\theta$ uniform on $-1 \leqslant \cos\theta \leqslant 1$)
GTISO(U,V,W)	Isotropic unit vector (in any order) $U=\cos\theta$, $V=\sin\theta\cos\phi$, $W=\sin\theta\sin\phi$, where θ is a random polar angle and ϕ is a random azimuthal angle

needed in a MORSE user routine. A random variable sometimes needed in a user routine is that from a Gaussian (normal) distribution. A good approximation is R = (RN_1 - RN_2)*FWHM for zero mean and standard deviation σ = FWHM/$2\sqrt{2\ell n 2}$ where FWHM is full width at half maximum. A more exact method[34] is given in Fig. 6 for zero mean and unit standard deviation ($0 \leqslant x < \infty$).

V.C. Subroutine BANKR(I)

The interface routine between the random walk procedure in MORSE and the analysis (user) routines is called BANKR(I). Familiarity with this routine is essential for normal operation of MORSE. Table 10 gives the possible entries to BANKR for various values of I. In the standard routine the negative indices are set for normal use and the positive indices have RETURN statements to be replaced with call statements to user routines. Comment statements in BANKR indicate from what routine in the random walk module the positive index call is made. For some of the lesser used options the call is not currently made, as indicated in the comments, and must be activated if necessary. The most used options are for the physically analogous events: (1) source, (3,4) fission and gamma-ray production, (5) collision, (7) boundary crossing, and (8) escape. Many routines have been written for these options. The code already produces standard output concerning many of the other options. The user is free, of course, to add to this with other routines or initiate

$$f(x) = \sqrt{\frac{2}{\pi}}\, e^{-x^2/2} \qquad 0 \leq x < \infty \qquad \text{the (half) Gaussian}$$

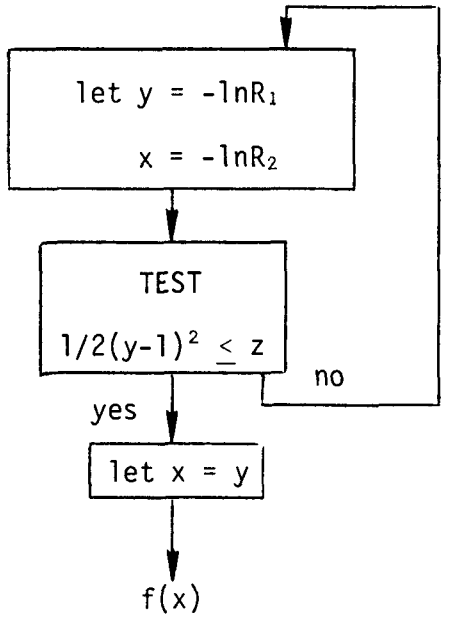


Fig. 6. Gaussian Selection Procedure (from Ref. 34).

other calls from the random walk for information not included in Table 10. It must be remembered that all calls to BANKR are currently made *after* the particular event, so pre-event analysis will need to use pre-event variables. If two successive indices are used in the same calculation, a RETURN statement must be placed between the calls to the user routines. It must also be remembered that the call to BANKR for I=1 is to analyze source events and is in no way connected with the user-written source routine that creates source particles.

Special versions of BANKR are necessary for use with a history file (collision tape). These routines contain the basic elements of the standard BANKR with extra programming for reading and/or writing the information from the BANKR calling indices. A MORSE calculation using a collision tape is only an analysis calculation using user routines. The call to the file-reading BANKR provides exactly the same random walk information (which has been written on the file as specified on input in the random walk calculation) as if the entire calculation were done in one step. These BANKR routines will be different for various combinations of (1) writing data, (2) reading data, (3) analog-type data, (4) next-event data simulated as analog, i.e., make a boundary estimation appear on the tape as an analog crossing, and (5) use with a DOMINO calculation.

Table 10. BANKR Arguments

BANKR Argument	Location of call in walk
−1	After call to INPUT — to set parameters for new problem
−2	At the beginning of each batch of NSTRT particles
−3	At the end of each batch of NSTRT particles
−4	At the end of each set of NITS batches (A new problem is about to begin.)
1	At a source event
2	After a splitting has occurred
3	After a fission has occurred
4	After a secondary particle has been generated
5	After a real collision has occurred (Post-collision parameters are available.)
6	After an albedo collision has occurred (Post-collision parameters are available.)
7	After a boundary crossing occurs (The track has encountered a new geometry region or medium other than albedo or medium zero.)
8	After an escape occurs (The geometry has encountered medium zero.)
9	After the post-collision energy group exceeds the maximum desired
10	After the maximum chronological age has been exceeded
11	After a Russian roulette kill occurs
12	After a Russian roulette survival occurs
13	After a secondary particle has been generated but no room in the bank is available

V.D. Subroutine SOURCE

The standard SOURCE routine and associated input require all particles to be created which

- are at the same point in space,
- are isotropic (UINP = VINP = WINP = 0) or monodirectional ($UINP^2 + VINP^2 + WINP^2 = 1$) in direction,

- have one group-dependent energy spectrum with optional biasing (one group for all source particles may also be set by input),
- have a delta function distribution in time (usually AGSTRT = 0.0),
- have the same weight (WTSTRT usually 1.0).

A system with fissionable media requires a fission spectrum for each fissionable medium to be specified by input. If any other source distribution and/or biasing is required, special programming is required. The user may (1) write a new SOURCE routine (with the standard calling parameters), (2) modify the standard routine, or (3) call another routine from the standard routine. Any standard input variable not reset in the user source remains unchanged and is used for all source particles. The source routine is called one time for each starting particle and all source particles are processed before the random walk begins; i.e., the source routine is not actually in the random walk sequence. NAME and NAMEX are assigned consecutively in MSOUR, the routine which calls SOURCE. In the following examples it is assumed that a subroutine SOUR1(IG,U,V,W,X,Y,Z,WATE,AG) is called at the end of the standard SOURCE routine.

- **Source Example #1 (see Fig. 7)**

Consider a plane source in the YZ plane with particles starting in the +X direction. The plane extends to ±10 cm in both Y and Z directions. The spatial distribution is uniform over the entire plane, but it is desired to start 2/3 of the particles from the −Y half and 1/3 from the +Y half. This bias requires a weight correction. The unnormalized angular distribution for the −Y half is cosine distributed, peaked in the +X direction; i.e., $f(\mu) = \mu$, the polar angle cosine. The angular distribution on the other half is isotropic in the forward direction. The routine for this case is in Fig. 8. If the natural angular distribution on the −Y side had been the same as on the left but given a cosine bias, then an additional correction, WTCOR = 0.5/P1/(2.*U), would be necessary; i.e., natural $f(\mu)$/biased $f(\mu) = 1/2\mu$ due to normalization. If the source had been a volume, rather than a plane, an X selection similar to Y or Z would have given a uniform volumetric source.

- **Source Example #2 (see Figs. 9 and 10)**

In this example it is desired to select spatial points uniformly on the surface of a hemisphere of radius R in the negative X half-space. The angular distribution is isotropic in the outward direction. This is a 5-group problem. Half of the source particles are picked from the input energy spectrum (nothing done here) and half from the spectrum in the user routine. The time distribution is increased from zero by an approximate Gaussian with mean TM seconds and standard deviation TSD. In selection of the outward direction, the cosine between the radius vector and the random direction must be positive; i.e., $\bar{n}\cdot\bar{\Omega} > 0$ where $\bar{n} = i(X-XO) + j(Y-YO) + k(Z-ZO)$ and $\bar{\Omega} = iU + jV + kW$. The normalization of $\bar{n}$ is not needed here. It is possible to select an outward direction on the hemisphere with no rejection; however, this would require a three-dimensional coordinate

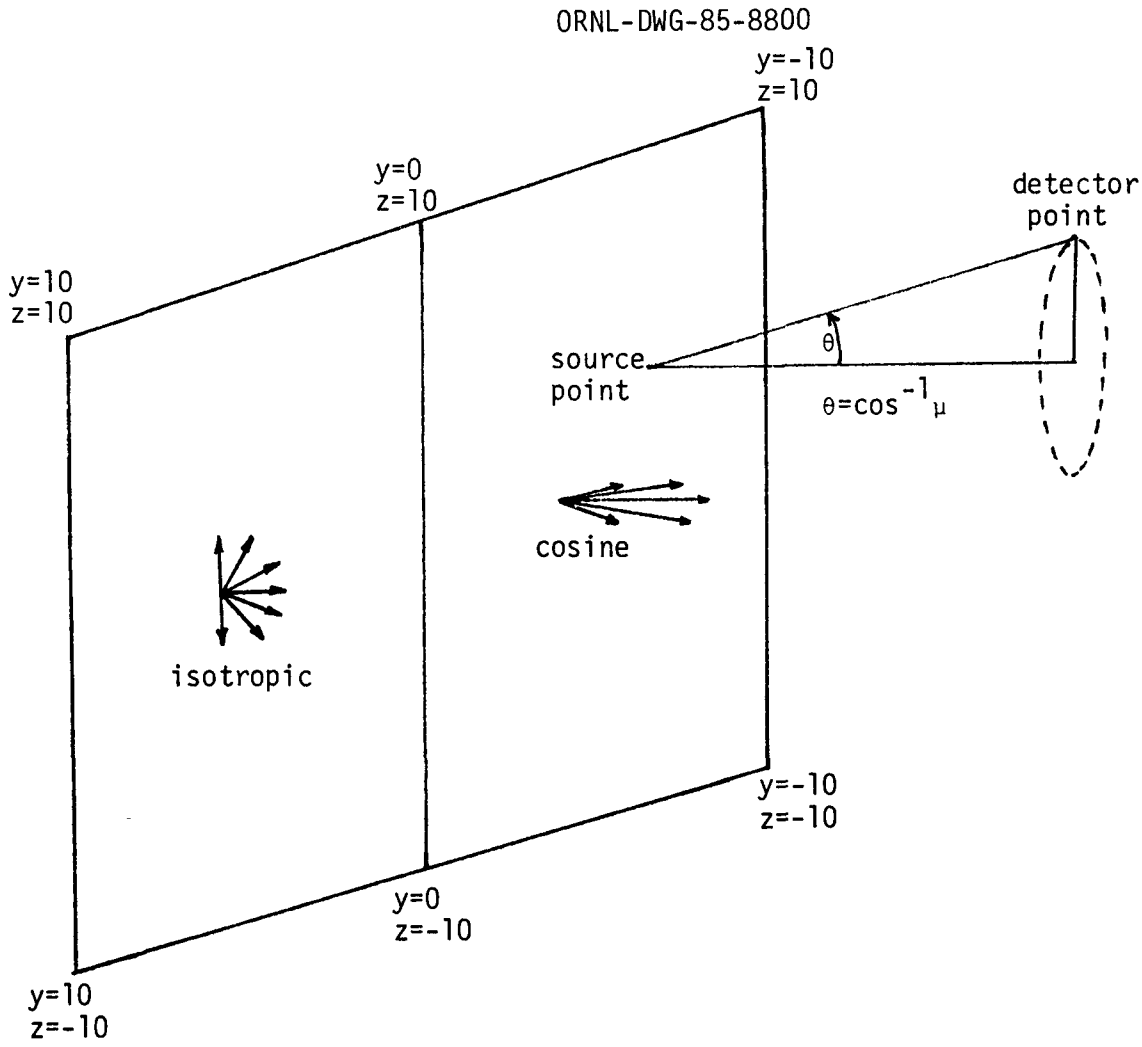


Fig. 7. Source Example 1.

rotation for the direction cosines and would be more difficult to verify. If the spatial distribution in the hemisphere had been uniform in volume, rather than on the surface, the RO for X, Y, and Z would have been multiplied by the cube root of FLTRNF(0) — a random spherical radius. There is no biasing so the weight is unchanged from WTSTRT. If −ABS(XX) were replaced by XX, the source would be uniform on the entire sphere. The rejection of inward directions could be replaced by the following procedure.

A general direction cosine transformation is

$$U = \sqrt{\frac{1-\mu^2}{1-W_0^2}}\,(W_0U_0\,\cos\theta - V_0\,\sin\theta) + \mu U_0\ ,$$

```
      SOUR1(IG,U,V,W,X,Y,Z,WATE,AG)
      DATA P1,P2/0.667,0.333/,X1/0.0/
C     P1 AND P2 ARE THE BIASED PROBABILITIES FOR
C     EACH HALF
C     THE NATURAL VALUES ARE 0.5 FOR EACH
C     WTCOR=NATURAL/BIASED
      X=X1
      RN=FLTRNF(0)
      IF(RN.LE.P1) GO TO 1                    pick correct side
      Y=10.*FLTRNF(0)                         { pick uniformly
      Z=10.*SFLRAF(0)                         { in space
      CALL GTISO(U,V,W)                       { pick isotropically
      U=ABS(U)                                { in +X direction
      WTCOR=0.5/P2                            weight increase
      GO TO 2
    1 CONTINUE                                picked -Y side
      Y=-10.*FLTRNF(0)                        { pick uniformly
      Z=10.*SFLRAF(0)                         { in space
C
      RN=FLTRNF(0)                            { RN = ∫_0^μ f(μ')dμ' / ∫_0^1 f(μ')dμ'   f(μ)=μ, μ=cosθ
C
      U=SQRT(RN)
      CALL AZIRN(SA,CA)                       random azimuthal angle φ
      WU=SQRT(1.-U*U)                           sinθ
      V=CA*WU                                 V=sin θ cosφ
      W=SA*WU                                 W=sin θ sinφ
      WTCOR=0.5/P1                            weight decrease
    2 WATE=WATE*WTCOR                         WATE=WTSTRT in input
      RETURN
      END
```

Fig. 8. Routine for Source Example 1.

$$V = \sqrt{\frac{1-\mu^2}{1-W_0^2}}\ (W_0V_0\ \cos\theta + U_0\ \sin\theta) + \mu V_0\ ,$$

$$W = -\sqrt{(1-\mu^2)\ (1-W_0^2)}\ \cos\theta + \mu W_0\ ,$$

where

U,V,W	=	new (transformed) cosines,
U_0, V_0, W_0	=	old (incident) cosines,
μ	=	cosine of the angle between the old and new directions,
θ	=	random azimuthal angle $0 \leqslant \theta \leqslant 2\pi$.

In this source example, U_0, V_0, W_0 would be the normal vector components at the surface of the sphere:

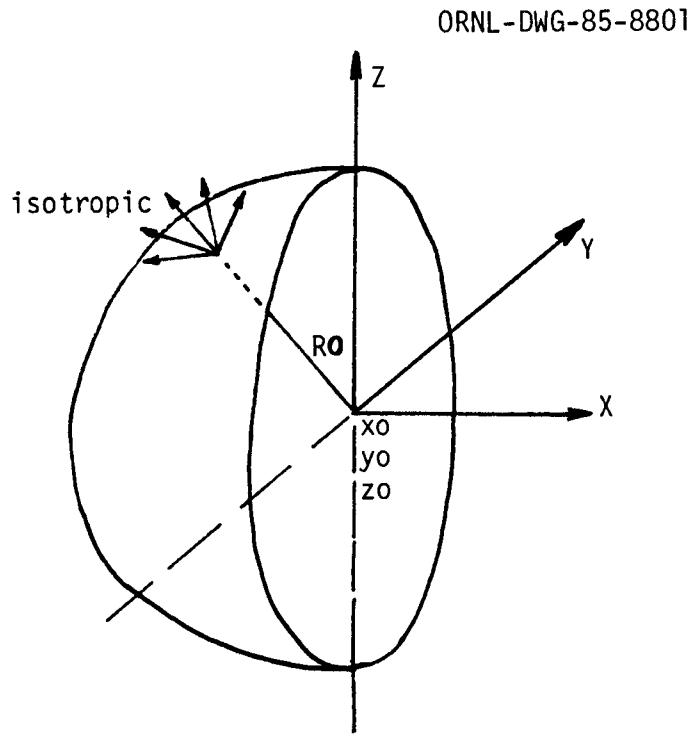


Fig. 9. Source Example 2.

$$
\begin{aligned}
U_0 &= (X - X_0)/\mathrm{R}, \\
V_0 &= (Y - Y_0)/\mathrm{R}, \\
W_0 &= (W - W_0)/\mathrm{R}, \\
R &= \sqrt{(X-X_0)^2 + (Y-Y_0)^2 + (Z-Z_0)^2}.
\end{aligned}
$$

For a more complicated surface, the normal vector components could be obtained from subroutine NORML. Also, $U_0^2 + V_0^2 + W_0^2 = 1$ and $U^2 + V^2 + W^2 = 1$. For isotropic outward emission ($0 \leqslant \mu \leqslant 1$), μ = FLTRNF(O) and $\sin\theta$ and $\cos\theta$ are from AZIRN(SX,CX). If $|W_0| \sim 1$ in the above equations, $U = \sqrt{1-\mu^2}\cos\theta$, $V = \sqrt{1-\mu^2}\sin\theta$, and $W = \mu W_0$.

These general transformation equations are the same as those used in the random walk process for determining the outgoing direction following a collision.[35] The μ is determined from $f(\mu)$, the scattering function, and U_0, V_0, W_0 are UOLD, VOLD, WOLD.

- **Source Example #3 (see Figs. 11 and 12)**

The last example is for a right circular cylinder of radius *RM* and height *H*. The volumetric neutron source is uniform in radius from inner radius *RO* to *RM* and has a cosine distribution along the entire cylindrical axis, $f(z) = \cos(\pi z/H)$. However, no source points are accepted in the shell between radii *R1* and *R2*, nor are they accepted if they are in medium 3. The energy spectrum is from the input data for medium 1, from group 3 only for medium 2, and from a ^{235}U Watt spectrum for medium 4; it is a function of Z for medium 5 and is determined in some function GETENG(IG,Z).

```
      SOUR1(IG,U,V,W,X,Y,Z,WATE,AG)
      DIMENSION SP(5),CSP(5)
      DATA RO,TM,TSD/10.,1.0,0.5/,XO,YO,ZO/0.0,0.0,0.0/
      DATA IST/1/,NIG/5/
      DATA SP/0.1,0.2,0.4,0.3,0.1/
      IF(IST.EQ.0) GO TO 1
      IST=0
      FWHM=2.*SQRT(2.*ALOG(2.))*TSD                    set FWHM
      SUM=0.0
      DO 2 I=1,NIG                                     {sum second
    2 SUM=SUM+SP(I)                                    {source spectrum
      NIG1=NIG-1
      SUM1=0.0
      DO 3 I=1,NIG1                                    {create cumulative
      SUM1=SUM1+SP(I)/SUM                              {distribution, CSP(5)=1.0
    3 CSP(I)=SUM1
    1 CONTINUE
      IF(FLTRNF(0).LT.0.5) GO TO 4                     pick source spectrum
      RN=FLTRNF(0)
      DO 5 I=1,NIG1                                    {pick energy group
      IF(RN.LE.CSP(I)) GO TO 6                         {from second spectrum
    5 CONTINUE
      I=NIG
    6 IG=I
    4 CONTINUE
      CALL GT1SO(XX,YY,ZZ)                             isotropic vector
      X=-RO*ABS(XX)+XO
      Y=RO*YY+YO                                       {uniform positions on
      Z=RO*ZZ+ZO                                       {hemisphere
    7 CALL GTISO(U,V,W)                                isotropic direction cosines
      DOTP=(X-XO)*U+(Y-YO)*V+(Z-ZO)*W                  {dot product—
      IF(DOTP.LT.0.0) GO TO 7                          {if inward direction,
      R1=FLTRNF(0)                                     {try again
      R2=FLTRNF(0)
      AG=TM+FWHM*(R1-R2)                               set time
      RETURN
      END
```

Fig. 10. Routine for Source Example 2.

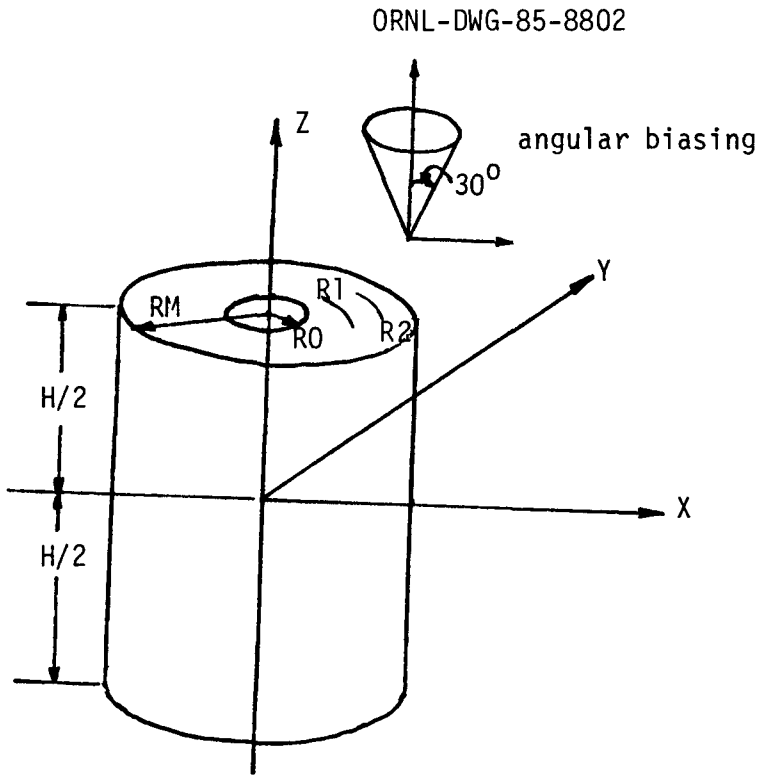


Fig. 11. Source Example 3.

In order to enhance leakage out of the top and bottom of the cylinder, particles are given a biased angular distribution if they are created within HMIN centimeters of the top or bottom. Half these particles are started uniformly within a 30° polar angle cone toward the top or bottom, one-third uniformly in the remaining 60° outward direction, and one-sixth isotropically toward the center. If this source were for the initial batch of an eigenvalue problem, then the radial distribution might also be a cosine, $f(R) = \cos(\pi R/2RM)$, and selected from similarly as for z.

None of these source examples in total would necessarily represent any real system. However, the individual selection of variables are representative of actual MORSE calculations. Also, no attempt has been made to optimize these routines in terms of programming or computing time efficiency. This is rarely necessary for source routines since even complicated source selection time is often insignificant as compared to the rest of the calculation. Ease of verification is of primary concern. However, any doubtful rejection techniques should be tested prior to a long calculation by placing counters and appropriate print statements in the programming to determine the ratio of rejections to acceptances. In fact, the entire source program should be tested to determine if the random selections and weight adjustments actually reproduce (within statistical uncertainty) the original distributions. This could be done by calling FLUXST in the source routine or running a source only (no collisions) calculation with appropriate estimation routines.

```
      SOUR1(IG,U,V,W,X,Y,Z,WATE,AG,NMTG)
      COMMON NBC(1)
      REAL*8 DI
      EQUIVALENCE(NBC(1),BC(1))
      COMMON/ORGI/DI,MA,NZ
      COMMON/GOMLOC/L1,L2,L3,L4,L5,D1,D2,LM,D3,D4,D5,L6
      DATA P1,P2,P3/0.5,0.3333,0.1667/
      DATA RM,RO,R1,R2,/100.,10.,30.,34./
      DATA CONV/1.91322E+12/
      DATA H2,HMIN/50.,25./,PI/3.14159/                    H2=H/2
    1 RN=FLTRNF
C
      ROS=RO*RO                                            RN=(r^2-RO^2)/(RM^2-RO^2)
C
      ROOT=RN*(RM*RM-ROS) + ROS
      RAD=SQRT (ROOT)                                      random radius
      IF(RAD.LT.R2.AND.RAD.GT.R1) GO TO 1                  rejection
      CALL AZIRN(SA,CA)
      X=RAD*CA
      Y=RAD*SA
      RN=FLTRNF(0)
C
C
C     Z=2*H2/P1*ARSIN(RN)                                  RN=(∫_0^z f(z)dz)/(∫_0^{H/2} f(z)dz)
C
C
C
      IF(FLTRNF(0).LT.0.5)Z=-Z
      CALL LOOKZ(X,Y,Z,NBC(L1),NBC(L2),
     *NBC(L3),NBC(L4),NBC(L5),NBC(L6))                     get medium
      NLOC=NZ+LM-1                                         from zone number
      NMED=NBC(NLOC)
      GO TO (2,3,1,5,6),NMED                               reject medium 3
    3 IG=3
      GO TO 2
    5 ENG=FISRNF(0)/CONV                                   energy in eV
      IF(ENG.GT.BC(1)) GO TO 5                             energy too high, reject
      DO 7 I=2,NMTG                                        NMTG=number of groups,
      IF(ENG.GT.BC(I)) GO TO 8                             energy limits are first in BC
    7 CONTINUE
      I=NMTG+1                                             low energies put in last group
    8 IG=I-1
      GO TO 2
    6 IG=GETENG(Z)
    2 CONTINUE
      IF(ABS(Z).LT.HMIN) GO TO 10                          select angle
      RN=FLTRNF(0)
      IF(RN.GT.P1) GO TO 9
      W=1.-FLTRNF(0)*0.134                                 uniform on 0 ≤ θ ≤ π/6
      WTCOR=P3/P1                                          1/6 ÷ P1
      GO TO 11
    9 PT=P1+P2
      IF(RN.GT.PT) GO TO 12
      W=FLTRNF(0)*0.866                                    uniform on π/6 ≤ θ ≤ π/2
      WTCOR=1.0                                            1/3 ÷ P2
   11 WR=SQRT(1.-W*W)
      CALL AZIRN(SA,CA)
      U=WR*CA
      V=WR*SA
      W=SIGN(W,Z)                                          W points out of cylinder
      GO TO 13
   12 WTCOR=0.5/P3                                         U,V,W isotropic from input data
      W=-SIGN(W,Z)
   13 WATE=WATE*WTCOR                                      correction due to angle selection
   10 RETURN
      END
      FUNCTION GETENG(Z)
      coding to give IG as a function of Z for medium 5
      RETURN
      END
```

Fig. 12. Routine for Source Example 3.

V.E. Next-Event Estimation Routines

V.E.1. Estimation to a Point. These routines are the most "standard" of any user routines issued in the MORSE code package. The basic routine is RELCOL called from BANKR(5) after each collision for flux estimation to point detectors. Pre-collision parameters are used (except for WATE) since post-collision parameters have already been determined. The detector locations for the detector loop are obtained from blank common. The dot product determines THETA $= \mu$, the angle of "scatter." PTHETA returns the scattering angle probabilities $P_g = f(\mu)$. EUCLID determines the number of mean free paths (negative value) between the collision site and detector. WATE is used instead of OLDWT since the "scatter" to the detector must also include the pre-collision Σ_S/Σ_T. The FORTRAN SIGN function in the contribution CON to the flux is necessary due to the negative values of the Legendre polynomial expansion of $f(\mu)$. The random energy group selection must use positive values; i.e., the selection is made from $|P_g|/\sum_g |P_g|$ instead of $P_g/\sum_g P_g$ and the estimator includes the correction $P_g/|P_g| * \sum_g |P_g| = \pm \sum_g |P_g|$, depending on the sign of P_g for the selected g. The ISTAT=1 parameter in the cross-section input (card XB) makes the P_g available for RELCOL. Note the comment cards concerning NEX and NEXND in the analysis input data (card BB), as well as those concerning when energy biasing is used in the random walk. Provision is made, although not included, to make more than one estimate to each particular detector. In the FLUXST call, I is the detector (input cards BB and CC), IL is the estimation group unrelated to any random walk values, AGED is the estimation trajectory time increase over the random walk value, and COS is the trajectory angle cosine with respect to some direction corresponding to the angle bin input date (cards LL). The last item, SWITCH, is zero for most cases.

V.E.2. Estimation to Volume or Surface. If the estimation were being made to a body of solid angle $\Delta\Omega$, this value must be determined and replace the $1/R^2$ term in the estimator.[29] When $\Delta\Omega$ is sufficiently large, next-event estimation to a boundary is possible. Here the post-collision trajectory is extended to the boundary crossing, if possible. The estimator WATE*EXP(ARG) is current, J, and must be treated for grazing angle crossings to determine the flux (see Section V.G). Much of the RELCOL programming can be eliminated for next-event estimation to a boundary since there is no random energy selection, but the boundary crossing location of the extended trajectory must be determined.

V.E.3. Gamma-Ray Estimation. The "standard" RELCOL has recently been upgraded to use the Klein-Nishina formulation for gamma-ray estimation.[36] The neutron treatment is unchanged. In the old method, neutrons and gammas were distinguished only by group number. If the new method is used, the cross section input must include the Compton and pair production data cross sections in the first two positions in the ANISN library format. On card XB replace NCOEF by −NCOEF, and the Compton and pair production partial cross sections must be in positions IHT−3 and IHT−4, respectively, in Table 1 [see page 4.6-41 of the MORSE manual revision ORNL-4972/R1 (Ref. 1)]. The energy group limits in eV must be given exactly (cards E) for the gamma-ray groups.

It has recently been pointed out that this extra cross-section treatment is not needed and that all required data are available in the standard libraries.[37] If it is inconvenient to alter the ANISN library table positions in the data or for older versions of the code which do not contain the new features, the following development outlines an interface "fix" to create the ANISN format positions in the cross-section library.

For each nuclide on the cross-section library, the gamma-ray P_0 data given, for each energy group, is σ_g^t, the total cross section, and $\sigma_{g\to g'}^s$, each group-to-group transfer.

The total cross section is

$$\sigma_g^t = \sigma_g^{photo} + \sigma_g^{Compt} + \sigma_g^{pp},$$

where σ_g^{photo} is the photoelectric (absorption) cross section, σ_g^{Compt} is the total Compton (scattering) cross section, and σ_g^{pp} is the pair production (scattering) cross section. For use in a transport code with only one exit particle from a collision,

$$\sigma_g^s = \sigma_g^{Compt} + 2\sigma_g^{pp}$$

and also

$$\sigma_g^s = \sum_{g'} \sigma_{g\to g'}^s .$$

Thus, by equating the two expressions for σ_g^s and using σ_g^t, there are two equations and two unknowns, σ_g^{Compt} and σ_g^{pp}. The σ_g^{photo} is in position IHT-2 in the table (Table I) and σ_g^s can be determined by summing the transfer cross sections. Thus the user can put the two extra cross sections in the first two library positions and move all other table positions down by two locations.

A more permanent update will soon be made in MORSE where the macroscopic cross sections (the Σ for each medium) are adjusted after the cross-section mixing is done rather than for each nuclide in the library. The user must verify that σ_g^{photo} is actually in the absorption position, IHT-2. The absorption cross section is not generally used directly in a transport code, and some evaluators sometimes put other data, such as kerma, in that position.

If it is known with certainty that there are little or no gamma-ray sources or production above the pair production threshold (1.02 MeV), then the Klein-Nishina equation can be programmed directly into RELCOL and the procedures for the pair production neglected.[36]

V.E.4. Other Applications of RELCOL. There is one other version of RELCOL available — one that makes an estimation to all energy groups rather than to a randomly selected group. This creates more contributions and smaller variance but requires more computation time. For a given angle of scatter, some of the probabilities will be zero. It is also inefficient to have the EUCLID call inside the loop over groups. A better procedure would be to create a group-dependent array for mean free paths. A loop with the EUCLID option for intermediate boundary crossing between collision and detector point could then compile the needed data in this array for later use in the loop for the flux estimate for each group. Use of distances and total cross sections (from NSIGTA) would be necessary for determining ARG.

An option which should be considered in each application of RELCOL is the inclusion of probabilities of estimation $P(\bar{r},det)$ for each detector as a function of spatial position.[29] With probability P an estimation is made and the contribution is divided by P. With probability $(1-P)$ the estimation to that detector, and all computation, is omitted. This technique can save much computation time, but the user must create the P function which can be difficult in systems with voids and streaming paths. The inclusion into RELCOL of any of the techniques for eliminating the infinite variance characteristic due to the $1/R^2$ term (once-more-collided or ring detector[29]) would constitute a major programming and debugging effort.

V.E.5. Uncollided Next-Event Routine. Each calculation using the RELCOL routine requires use of an uncollided estimation from source particles. The standard routine is SDATA called from BANKR(1). Since a source event can be thought of as another "scatter" event, SDATA can be thought of as a simple form of RELCOL. A similar routine is needed for secondary generated particles — SGAM (identical to the standard SDATA) called from BANKR(4) for gamma generation — and for non-eigenvalue fission calculations another call to standard (isotropic) SDATA, also from BANKR(4). For eigenvalue calculations the call is from BANKR(1) at the start of the next batch. If a neutron-gamma problem is run in the primary-only mode, no separate uncollided estimate for gamma generation is necessary. This is done implicitly in RELCOL and any anisotropy in the generated gamma is automatically included.

Uncollided estimation may be done analytically, independent of the code, or in a separate calculation and combined with the collided results later. The uncollided calculation can be omitted (no call to SDATA) if it is known with certainty that the contribution is insignificant compared to that of the collision processes. The usual procedure is to use the source parameters generated for the random walk (in NUTRON common) for the uncollided estimates; i.e., estimate from X,Y,Z in group IG, etc. If there were some energy-angle functional form, then the group would be determined in SDATA since the angle is fixed by the source and detector locations. The only difficulty (sometimes a major one) occurs when the source is not isotropic. Then the $1/4\pi$ term in the estimator must be replaced by the angular source distribution used to create the cumulative distribution for random selection in the random walk source routine. Fission neutrons and secondary gamma rays are usually considered to be produced isotropically — if not, any available data may be used in the program.

In the first source routine example (Section V.D) the +Y side of the plane produces particles isotropically in the forward (+X) direction. The standard SDATA would be used here for detectors in +X with $1/4\pi$ replaced by $1/2\pi$ since only half of the 4π solid angle is included. If the geometry actually extended in the −X direction, there would be no uncollided estimates in that direction. For the −Y side of the source plane the distribution is a forward peaked cosine current $f(\mu)=\mu$ (isotropic flux condition). In this case the $1/4\pi$ would be replaced by μ/π, i.e., $1/2\pi*2\mu$ (see Fig. 8). In SDATA $\mu = \cos\theta =$ A/DS for this case. The general angular distribution term in SDATA is

$$\frac{f(\Omega)}{\int_{\Omega} f(\Omega)d\Omega} = \frac{f(\mu)}{2\pi\int_{\mu} f(\mu)d\mu} ,$$

where $f(\mu)$ and its normalization are those used for the random selection in the source routine (see notes in Fig. 8) and μ is determined by the source and detector points. That is, $f(\mu)$ is *selected from* for a random μ in SOURCE and *evaluated for* a specific μ in SDATA. For the case in Fig. 8, $0 \leqslant \mu \leqslant 1$. If the normalization vanishes over any interval (the denominator is zero), the angular space must be broken into segments. If the azimuthal angle were not isotropic, then both the source routine and SDATA would be more complicated. If $f(\mu)$ is a discrete function, the procedure is still the same; i.e., use the discrete value of f_i for the $\Delta\mu_i$ interval which contains μ.

For biased random walk source selection, there are two possibilities for an uncollided estimation: (1) all the biased source parameters are used in SDATA, including the biased $f(\mu)$ and the corrected WATE, and (2) the natural distribution and the uncorrected weight (WTSTRT or 1.0) are used. The latter method would be preferable, for example, for a natural isotropic source with a complicated biasing procedure in the source routine. In the SDATA estimator CON, WATE would be replaced by 1.0 and the $1/4\pi$ term retained.

In general, any special features in RELCOL, such as estimation probabilities, estimation to a boundary, etc., should be applied to the uncollided routines with any appropriate simplifications. Attention should be given to the last value (SWITCH) in the FLUXST call from SDATA. The standard MORSE output format for detector results includes a possible separate uncollided term. If SWITCH < 1, the collided and uncollided results are separate. For SWITCH > 1, the uncollided is still separate, but the total response includes both contributions (assuming that SWITCH=0 in the FLUXST from RELCOL). Comments concerning the other items in FLUXST are as for RELCOL. An uncollided estimate can be made in conjunction with any estimator, such as boundary crossing, for example, when it is desired to know the uncollided contribution. However, the two results cannot be combined (set SWITCH < 1 in SDATA) since analog estimator results contain any uncollided flights from the detector. However, the SDATA uncollided results are usually easier to obtain and will have a smaller variance than those from an analog routine.

V.F. Collision Density and Track Length Estimators

A collision density estimator would normally be called from BANKR(5) after a random walk collision, just as for RELCOL. In fact, this estimator could be included in a RELCOL routine; however, the association of the next-event estimator in MORSE with RELCOL is well set, so another name should be used. The collision density routine in Fig. 13 also includes parts of a track length estimator, which will be completed in the two boundary crossing estimators to follow. Like the source routines in Section V.D, these routines are for demonstration only and should not be considered representative, in total, of a realistic calculation. Individually, the different detectors (first parameter in the call to FLUXST) are representative of many normal and a few unusual uses. These detectors are:

```
      SUBROUTINE COLDEN
      COMMON/NUTRON/NAME,NAMEX,IG,IGO,NMED,MEDOLD,
     *NREG,U,V,W,UOLD,VOLD,WOLD,X,Y,Z,XOLD,YOLD,ZOLD,WATE,
     *OLDWT,WTBC,IBLZN,IBLZO,AGE,OLDAGE
      COMMON/APOLLO/DUM(8),ETATH
      DATA NAM1/0/
      DIMENSION DUM(8)
      DATA VOL1,VOL2,VOL3,VOL4/100.,50.,30.,10./
      DATA XO,YO,ZO,RO/-15.,10.,25.,10./
      CALL NSIGTA(IGO,NMED,TSIG,PNAB)                    get Σ_T, Σ_S/Σ_T
      CALL FISGEN(IGO,NMED,PNUF)                         get νΣ_f/Σ_T
      CON=PNUF*WTBC/VOL1                                 fission production
      CALL FLUXST(1,IGO,CON,0,0,0)
      CON=(WTBC-WATE)/VOL1                               absorption density
      CALL FLUXST(2,IGO,CON,AGE,0,0)
      IF(NREG.NE.3) GO TO 1
      CON=WTBC/TSIG/VOL3                                 collision flux
      IF(NAM1.EQ.NAME) GO TO 3
      NAM1=NAME
      CALL FLUXST(4,IGO,CON,0,0,0)
3     CALL FLUXST(3,IGO,CON,0,0,0)
1     RSTS=(X-XO)**2+(Y-YO)**2+(Z-ZO)**2
      IF(RO.GE.SQRT(RSTS)) GO TO 2
      CALL FLUXST(5,IG,WATE,0,0,0)                       scattering density
      CALL NSIGTA(IGO,8,TSIG8,PNAB8)
      CALL GAMGEN(IGO,8,PGEN,IGG)                        get γΣ_γ/Σ_T
      CON=WTBC*PGEN*TSIG8/TSIG
      CALL FLUXST(6,IGO,CON,0,0,0)
      CALL FLUXST(7,IGG,CON,0,0,0)
2     NDET=NMED+7
      CON=WTBC*ETATH                                     path length flux
      CALL FLUXST(NDET,IG,CON,0,0,0)
      RETURN
      END
```

Fig. 13. Collision Density and Track Length Estimator.

1. Fission production per cm^3, $\nu\Sigma_f\phi$, in the entire system of volume VOL1. If this were an eigenvalue calculation, this and most other estimators would be excluded for the first several batches until the batch number (NBAT + 1 in BANKR) is equal to or greater than NKCALC in APOLLO common (see input card L).

2. Absorptions per cm^3 in the entire system. The age is included here for time-dependent analysis. In addition to the standard k_{eff} calculation, another value could be

$$k_{\text{eff}} = \frac{\nu\Sigma_f\phi}{\Sigma_a\phi+L} ,$$

where the total leakage L is from the ESCAPE routine (see next section). For k_{eff} evaluation, one wants total production divided by total losses. Other k_{eff} estimators are possible,[9] and any significant difference of results should be investigated. In the above method, the three terms must be combined by hand from final results unless a more substantial programming effort is made for a batch-by-batch analysis. The final fractional standard deviation *FSD* would be as follows (see Section V.I):

$$FSD_{k_{\text{eff}}} = \left[FSD_f^2 + \frac{(\Sigma_a\phi)^2 FSD_a^2 + L^2 FSD_L^2}{(\Sigma_a\phi + L)^2}\right]^{1/2} .$$

In any analog calculation the sum of $\Sigma_a\phi+L$ should be exactly unity since all results are normalized to one source particle. Russian roulette kills and survivals will create a statistical departure from unity. These and other particle terminations, such as energy and time cutoffs, can be investigated in more detail from the normal output by appropriate subroutine creation and calls from BANKR (see Table 10).

3. The flux, particles/cm^2, in region 3 only.

4. The first-collision contribution of detector 3. If NAME changes, this is the first collision for that particle. If NAM1 equals NAME, this is a multiple collision of the same particle. Further differentiality can be obtained by creating counters for subsequent (second, third, etc.) collisions.

5. Total scattered particles (no volume normalization) in a sphere of radius RO and center at XO, YO, ZO. No regard is given to medium or region number. Note the use of post-collision parameters IG and WATE.

6. Gamma-ray production, $\gamma\Sigma_\gamma\phi$ (total gammas produced), in the sphere of detector 5 for medium 8. Assume there are seven media in the geometry input (1 through 7) and medium 8 in the cross-section input includes one nuclide with the same density (atoms/barn-cm) as the partial density of the same nuclide in the media inside the sphere (see Section II.B).

7. Gamma production spectrum from the estimator in detector 6; i.e., the gamma production group is used instead of the neutron producing collision group.

8–14. These detectors give the collision site contribution to the track length flux estimator in volumes corresponding to each of the seven media. These volumes could be defined in various other ways. In any volume there are four types of flights: (1) flights that end at a collision that began as a collision, (2) flights that end at a collision that began at a boundary crossing, (3)

flights that end at a boundary crossing that began at a collision, and (4) flights that end at a boundary crossing that began at a boundary crossing. It is necessary that a path length estimation volume be defined by geometry surfaces that are recognized by the code so that BANK(7) is called from NXTCOL when particles encounter them. The surfaces must define changes in medium number, region number, or geometry input zone. For time-dependent track length estimators, some decision must be made as to what time (AGE) to score — possibly the mid-point of the flight. For time collection intervals much larger than flight times, either AGE or OLDAGE is probably acceptable.

V.G. Boundary Crossing and Track Length Estimators

Continuing with the seven media problem, BANKR(7) calls BDRYX each time a medium, region, or geometry input zone change is encountered by a particle. The programming for a boundary crossing estimator involves the definition of a particular boundary of interest (see Fig. 14). In this case, for detector 15, it is assumed that there is only one boundary anywhere in the system between ZBDX ± 0.1 and that it is in the XY plane. Division by the area of this surface gives flux in terms of particles/cm^2. The surface could have been subdivided into Z segments or ± W directions with further tests and detector numbers. A discussion of the grazing angle problem is included in the MORSE manual[1] (Section 4.6.4). The call to FLUXST includes the angle cosine CSW for angle-dependent analysis (input cards MM).

```
      SUBROUTINE BDRYX
      COMMON/NUTRON/NAME,NAMEX,IG,IGO,NMED,MEDOLD,NREG,
     *U,V,W,UOLD,VOLD,WOLD,X,Y,Z,XOLD,YOLD,ZOLD,WATE,OLDWT,
     *WTBC,IBLZN,IBLZO,AGE,OLDAGE
      COMMON/APOLLO/DUM(8),ETATH
      DATA ZBDX/5./,ABDX/25./,GRA/0.01/
      Z1=ZBDX+0.1
      Z2=ZBDX-0.1
      IF(Z.L.Z2.OR.Z.GT.Z1) GO TO 1                    boundary in XY plane
      CSW=ABS(W)
      IF(CSW.LT.GRA)CSW=0.005                          grazing angle
      CON=WATE/CSW/ABDX                                flux
      CALL FLUXST(15,IG,CON,0,CSW,0)
    1 CONTINUE
      CON=WATE*ETATH                                   path length flux
      NDET=MEDOLD+7                                    WATE = WTBC here
      CALL FLUXST(NDET,IG,CON,0,0,0)
      RETURN
      END
```

Fig. 14. Boundary Crossing and Track Length Estimator.

Any number of boundaries may be searched for desired estimation with various methods for acceptance or rejection. The remaining estimator here gives the internal boundary contribution to the path length estimator, i.e., detectors 8–14. At a boundary crossing, NMED is the medium across the boundary in the direction the particle is traveling and MEDOLD is the previous medium where the path length is measured.

The remaining possible contribution to the path length estimator, escape from the outer system boundary, is in the ESCAPE routine (Fig. 15) called from BANKR(8). If the path length volume did not have an outer surface boundary, there would be no contribution to that estimator here. WTBC is used since WATE is set to zero for a particle which encounters medium number zero. If the system outer boundary was surrounded by medium 1000 and then by medium 0, then this routine would be called by BANKR(7) and, in general, some search procedure would be necessary, as in BDRYX for finding the desired boundary if the crossing angle is necessary. Detector 16 is used to illustrate the use of a counter (using unity instead of weight) for determining the number of escapes for both X and Y < 0. This number (multiplied by the total number of starting particles) can be compared with the total number of escapes in the standard output.

The last part of this routine represents a very special example applicable to any estimator. For two outer boundary segments ($-$X, $-$Y, $-$Z and $-$X, $-$Y, $+$Z) it is desired to have energy, angle, and time-dependent results. The standard MORSE output format does not allow both time and angular analysis, although there is a special four-dimensional analysis package available. However, all this can be done by using a separate detector for each time bin in the standard input data. The x position for a point detector location is used to input the limits of the time bins. If there are to be five time bins, then ND, the total number of detectors (card BB), will be 28 — detectors 17–22 for one segment of the boundary and detectors 23–28 for the other. The times in these two sets will be the same — the smallest times in detectors 17 and 23 and the largest time in 22 and 28. It is assumed that all values of AGE will fall between these two values, although other provisions could be made. The leakage current is determined here and divided by the appropriate time interval. Subroutine NORML is used to determine the values in NORMAL common.

V.H. Comments on User Estimator Routines

It is seen that the versatility of estimator routines is almost limitless. Experience and need will govern their creation and use. If two adjacent calls in BANKR are activated in the same calculation, a RETURN statement must be placed between them, e.g., calls to BDRYX and ESCAPE. All results are normalized to one source particle. Any units of the source, $\sec^{-1}$, for example, will be included in all results. All results are normalized by the interval of the collection bins ($2\pi\Delta\mu$ for angles). Any output values not available through ordinary input, such as energy-dependent response, can be programmed directly using Tables 4.18–4.21 in the manual.[1] Often it is desirable to compare calculated flux spectra with experimental results for which the detector resolution has not been removed.[14] This can be simulated in an estimation routine using the approximate Gaussian technique (Fig. 6). A random energy is picked in the estimation group, and this Gaussian smear is applied. If the new energy is now in another group, this new group is used in the call to FLUXST.

```
      SUBROUTINE ESCAPE
      COMMON/NUTRON/NAME/NAMEX,IG,IGO,NMD,MDOLD,NREG,
     *U,V,W,UOLD,VOLD,WOLD,X,Y,Z,XOLD,YOLD,ZOLD,WATE,OLDWT,
     *WTBC,IBLZN,IBLZO,AGE,OLDAGE
      COMMON/APOLLO/DUM(8),ETATH
      COMMON/NORMAL/UN,VN,WN
      COMMON BC(1)
      COMMON/PDET/ND,DUMM(16),LOCXD
      COMMON/GOMLOC/L1,L2,L3,L4,L5,D1,D2,LM,D3,L6,L7
      EQUIVALENCE(BC(1),NBC(1))
      CON=WTBC*ETATH                                    path length flux
      NDET=MEDOLD+7
      CALL FLUXST(NDET,IG,CON,0,0,0)
      IF(X.GT.0.0.AND.Y.GT.0.0) RETURN
      CALL NORML(NBC(L1),NBC(L2),NBC(L3),
     *NBC(L4),NBC(L6),NBC(L7))
      CALL FLUXST(16,IG,1.0,0,0,0)                      escape counter
      CSW=-(U*UN+V*VN+W*WN)                             normal points inward
      IPT=0
      IF(Z.GT.0.0)IPT=1
      DO 3 I=17,22
      LOC=LOCXD+I
      IF(AGE.LT.BC(LOC)) GO TO 4
    3 CONTINUE
      I=22
    4 NDET=I+6*IPT
      CON=WTBC/(BC(I)-BC(I-1))
      CALL FLUXST (NDET,IG,CON,0,CSW,0)                 leakage
      RETURN
      END
```

Fig. 15. Escape and Track Length Estimator.

As for the source routine, any estimation routine must be thoroughly checked before use in a long calculation. The ETATH in the path length estimator should always be equal to SQRT((X-XOLD)**2+(Y-YOLD)**2+(Z-ZOLD)**2). A path length estimator can be used to calculate the volume of irregular, non-concave volumes.[30] Fill the volume with internal void (medium 1000) and impose a unit isotropic flux (cosine current) uniformly on the surface. A track length estimator used in the volume will give the volume in cm^3. This general procedure, independent of volume determination, is a convenient way to extend the geometry checking procedure over that from plotting pictures; i.e., fill the void volumes with particle tracks to check for any geometry input errors. Special versions of path length and also collision density estimation routines are available in some versions of the ENDRUN user routine. There are certain restrictions, such as one medium per region, and no statistical analysis is available. These routines should be used with caution.

If there is any question as to whether to use collision density or track length estimation, it is always helpful to use both in the same volume and compare results. In applying delta (fictitious) scattering[38] and the collision density estimator in optically thin media, BANKR(5) should be called from the special version of NXTCOL for a delta scatter. The cross section used for both real and delta collision estimation is the maximum Σ_T in the system, i.e., CON=WTBC/Σ_M in COLDEN.

V.I. Combining Monte Carlo Results

Often it is desired to combine the results of two *different* quantities X_A and X_B in some way, where these results are either from the same or different calculations. The combined result X_C has the general form $X_C = f(X_A,X_B)$, where

1) for addition

$$f(X_A,X_B) = X_A + X_B ,$$

2) for subtraction

$$f(X_A,X_B) = X_A - X_B ,$$

3) for multiplication

$$f(X_A,X_B) = X_A * X_B ,$$

4) for division

$$f(X_A,X_B) = X_A/X_B .$$

The general form for the combination of variance, σ_A^2 and σ_B^2, is

$$\sigma_C^2 = \sigma_A^2\left(\frac{\partial f}{\partial X_A}\right)^2 + \sigma_B^2\left(\frac{\partial f}{\partial X_B}\right)^2 + \rho_{A,B}\sigma_A\sigma_B\left(\frac{\partial f}{\partial X_A}\right)\left(\frac{\partial f}{\partial X_B}\right) .$$

The correlation coefficient $\rho_{A,B}$ is zero if the two results are completely independent. Even if there is some correlation, such as for eigenvalue calculations, $\rho_{A,B}$ is usually not known and is assumed to be zero. Then, for the four cases

1 and 2) $\sigma_C^2 = \sigma_A^2 + \sigma_B^2 ,$

3 and 4) $\sigma_C^2 = \left(\frac{\sigma_A^2}{X_A^2} + \frac{\sigma_B^2}{X_B^2}\right)X_C^2 ,$

where $X_C = X_A * X_B$ or $X_C = X_A/X_B$.

In MORSE, the statistical uncertainty is given as fractional standard deviation *FSD* of the result X, where

$$FSD = \sigma/X \ .$$

Then for the above cases

$$1 \text{ and } 2)\ FSD_C = \frac{\sqrt{X_A^2 FSD_A^2 + X_B^2 FSD_B^2}}{(X_A \pm X_B)} \ ,$$

$$3 \text{ and } 4)\ FSD_C = \sqrt{FSD_A^2 + FSD_B^2} \ .$$

In the example of the second detector for the collision density estimator (Section V.F), the uncertainty on the k_{eff} was found by applying the above procedures first for the addition in the denominator and then for the ratio.

If the two results X_A and X_B are in fact the *same* quantity from two different calculations, they should be combined as a linear average based on the number of source particles from each calculation (or the number of batches if the number of particles per batch was the same in both cases), N_A and N_B. Then, for case (1) only,

$$X_C = f(X_A, X_B) = \frac{N_A}{N_A + N_B} X_A + \frac{N_B}{N_A + N_B} X_B \ ,$$

$$\sigma_C^2 = \left[\frac{N_A}{N_A + N_B}\right]^2 \sigma_A^2 + \left[\frac{N_B}{N_A + N_B}\right]^2 \sigma_B^2 \ ,$$

$$FSD_C = \sigma_C/X_C \ .$$

V.J. FUNCTION DIREC

If the exponential transform (path stretching, card I) is used, a FUNCTION DIREC *must* be supplied by the user. This routine supplies the preferred direction to the next path selection procedure from the last collision site X,Y,Z (see page 4.4-35 in the manual[1]). The most common use is for the preferred direction to be toward a point detector used in next-event estimation (see Fig. 16). The detector coordinates can be obtained from blank common just as in RELCOL or from a data statement. NUTRON common is also needed. The dot product of the particle direction $\bar{\Omega}$ and the direction toward the detector is $\bar{\Omega} \cdot (\bar{r}_d - \bar{r})/|\bar{r}_d - \bar{r}|$. If DIREC > 0, the track is stretched toward the detector; otherwise, the path is shrunk. A DIREC = 0 gives no change. All weight adjustment is done automatically in GETETA, called from NXTCOL.

```
FUNCTION DIREC
COMMON/NUTRON/NAME,NAMEX,IG,IGO,NMED,MEDOLD,NREG,
*U,V,W,UOLD,VOLD,WOLD,X,Y,Z,XOLD,YOLD,ZOLD,WATE,
*OLDWT,WTBC,IBLZN,IBLZO,AGE,OLDAGE
DATA XD,YD,ZD/10.,20.,30./
DIST=SQRT((XD-X)**2+(YD-Y)**2+(ZD-Z)**2)
DIREC=(U*(XD-X)+V*(YD-Y)+W*(ZD-Z))/DIST
RETURN
END
```

Fig. 16. Example of a Simple FUNCTION DIREC.

In Fig. 16 if XD, YD, ZD defined the center of a sphere, a final statement DIREC = −DIREC would stretch all paths going away from the center and shrink all paths going toward it. If it is desired to stretch paths in some general direction (e.g., if there are several detectors) rather than to a specific point, other forms are needed. A simple DIREC=W would stretch all paths in the +z direction, and DIREC=−W would stretch all paths in the −z direction. For a more complicated case, consider the bent, circular cross-section duct in Fig. 17, with three regions as shown. Assume the entire centerline is in the same XZ plane (y=YC). The material in the duct is medium 1, and the wall material is medium 2 (there is no duct wall or liner). The arrows indicate the preferential directions, here taken as 45° from the walls toward the duct centerline. The routine is given in Fig. 18. Russian roulette and splitting should be used in the wall medium so that particles are not followed too far into it. The interior protruding corners and exit region could be given an even more detailed treatment.

V.K. Subroutine GTMED

The user may interchange geometry (CX) and cross-section (SX) media numbers in this routine without altering any input data. Suppose there is a SX=7 not referenced in the geometry data which is to be used in CX=3. Also, it is desired to interchange cross-section media numbers 1 and 2 in the geometry. The routine is

```
SUBROUTINE GTMED(CX,SX)
SX=CX
IF(CX.EQ.3)SX=7
IF(CX.EQ.1)SX=2
IF(CX.EQ.2)SX=1
RETURN
END
```

The default for this routine is SX=CX. This procedure can be used to change mixtures, densities of the same mixture, etc. Media 0, 1000, and albedo media are not allowed here. A void can be simulated by a mixture with a very small density.

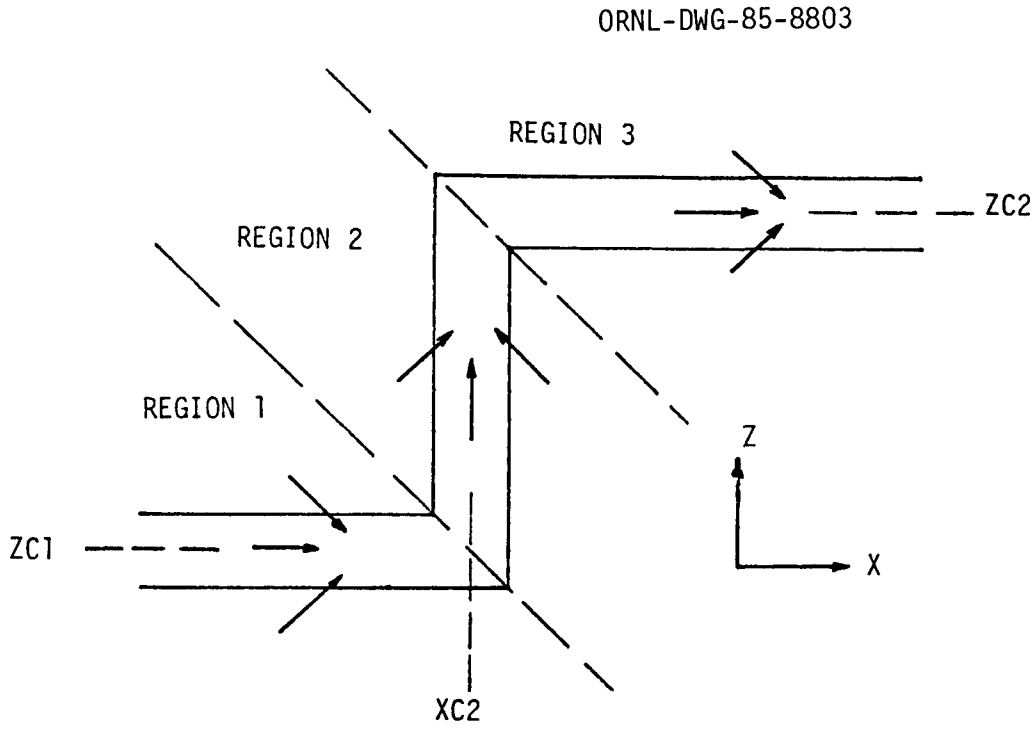


Fig. 17. Geometry for FUNCTION DIREC Example.

```
      FUNCTION DIREC(0)
      COMMON/NUTRON/NAME,NAMEX,IG,IGO,NMED,MDOLD,NREG,
     *U,V,W,UOLD,VOLD,WOLD,X,Y,Z,XOLD,YOLD,ZOLD,WATE,
     *OLDWT,WTBC,IBLZN,IBLZO,AGE,OLDAGE
      DATA RO,YC,ZC1,XC2,ZC2/1.0,0.0,5.,10.,20./
      IF(NMED.EQ.2) GO TO 2
      DIREC=U                                          inside duct
      IF(NREG.EQ.2)DIREC=W
      RETURN
    2 IF(NREG.EQ.2) GO TO 3
      B=YC-Y
      C=ZC1-Z
      IF(NREG.EQ.3)C=ZC2-Z                             horizontal legs
      A=SQRT(C*C+B*B)
      GO TO 4
    3 A=XC2-X
      B=YC-Y                                           vertical leg
      C=SQRT(A*A+B*B)
    4 DIST=SQRT(A*A+B*B+C*C)
      DIREC=(U*A+V*B+W*C)/DIST
      RETURN
      END
```

Fig. 18. Example of a Complicated FUNCTION DIREC.

V.L. Advanced User Programming

The user routines described, or their defaults, are necessary for normal use of MORSE. It is possible to simulate many effects not in the standard MORSE code, or to program in many features ordinarily supplied by input data. Among these are:

- Play Russian roulette and splitting at boundaries rather than at collisions. Call TESTW from BANKR(7) — boundary crossings.
- Set region number by programming a geometric functional form as in the TRIPOLI code[39] from BANKR(5) or BANKR(7). This could be done in subroutine NXTCOL after a collision site has been selected or at a boundary crossing location; done before or after COLISON, or in TESTW. Make sure the standard input data sets aside the correct locations in blank common for the number of regions to be created.
- Set Russian roulette and splitting parameters by functional forms rather than by input data. At the beginning of TESTW replace the WTHIR, WTLOR, and WTAVG from blank common with any new values. Make sure the standard input data sets aside enough room in blank common even though this data is not used.
- Create a special NXTCOL routine to create collision sites and boundary crossings instead of using the standard geometry routines and input data; i.e., write a user geometry "package."

There are, of course, many other possibilities. The user must be aware that any changes such as these can accidentally introduce errors into the code. Familiarity with subroutine MORSE and all standard code features is essential. A substantial series of checks and debugging runs should be made when making changes such as these. Items in this section should be considered only by expert users of the code.

V.M. User Routines with Array Geometry

In the array geometry (MARS) version of MORSE, the position and direction parameters in NUTRON common are in the coordinate system of the current universe (local coordinates). Often the user routine needs these variables in the coordinate system of the zero universe (global coordinates). The user routine must (1) first save these variables with dummy values (X1=X, etc.), (2) CALL NETLEV(2,BC), where BC is blank common, to convert the needed variables in NUTRON common to global coordinates, (3) perform all necessary user routine operations, and finally (4) reset the NUTRON common variables from the dummy values (X=X1, etc.).

V.N. Summary of User Routines

The user routines possibly necessary for a MORSE calculation are:

- before the random walk begins

  SOURCE (creates all particle source parameters)

  SDATA (uncollided next-event estimator)

- during the random walk

  DIREC (for exponential transform)

  GTMED (change cross-section and geometry media)

- estimation analysis, independent of the random walk

  RELCOL (next-event estimation)

  SGAM (uncollided estimation for generated particles)

  COLDEN (collision density and track length)

  BDRYX (boundary crossing and track length)

  ESCAPE (escape and track length)

None of these routines would be necessary for every MORSE calculation. The estimation routines should not change the random walk procedure in any way (except possibly alter the random number sequence when random numbers are needed in the estimators). Except for the comments regarding the source routine (Section V.D), all these routines should be constructed as efficiently as possible in terms of computation time.

The user must be aware that MORSE has had widespread use for many years. It is possible that errors still exist for some little-used combination of options or for any relatively new features, such as the array geometry. However, it should be assumed that trouble anywhere in the code might be caused by a user routine error and/or incorrect input data. As for any large code system, the trouble may appear (error message, termination, unreasonable results, etc.) far from where an error was introduced.

VI. ADJOINT USER ROUTINES

The ease of using MORSE in the adjoint mode has been outlined previously (see Section III.D), as have the general reasons for using adjoint methods.[29,40] Between the source and estimation processes the adjoint calculation is handled completely automatically by MORSE. However, the creation of source and estimation routines and the associated normalization are not always as apparent and straightforward as they are for a forward problem. The general relationships for phase space P are:

- adjoint source, $S^*(P)$ = forward response, $R(P)$
- adjoint response, $R^*(P)$ = forward source, $S(P)$
- total response $\lambda_{FOR} = \int \phi(P)R(P)dP = \lambda_{ADJ} = \int \phi^*(P)S(P)dP$

  (The λ's are equivalent, the subscripts only denote the mode of calculation.)
- factor to apply to MORSE forward (FOR) or adjoint (ADJ) results due to code normalization to one source particle

$$N_{FOR} = \int S(P)dP \qquad \text{(often unity)}$$

$$N_{ADJ} = \int R(P)dP \qquad \text{(often not unity)}$$

It is usually desired to obtain adjoint results in a form as if the calculations were in the forward mode; i.e., multiply the adjoint results by N_{ADJ}/N_{FOR}. In the equality $\lambda_{FOR} = \lambda_{ADJ}$ it is assumed that the normalizations have been applied. In reality, the equality is $N_{FOR}\lambda_{FOR} = N_{ADJ}\lambda_{ADJ}$, where the λ's are code results normalized to one (forward or adjoint) source particle. In applying the adjoint normalization, more care must be taken with volumes, surface areas, angular distribution, etc., than is usually necessary for forward calculations. Appendix B in Ref. 41 gives a discussion of the units and dimensionality associated with forward and adjoint comparison. A summary of this analysis in terms of units of each quantity is as follows:

$$\text{forward flux } \phi(\bar{r},E,\bar{\Omega}) = \frac{\text{forward particle tracklength}}{\text{volume}-\text{energy}-\text{solid angle}} = \frac{\text{cm}}{\text{cm}^3-\text{eV}-\text{steradian}} = \frac{1}{\text{cm}^2-\text{eV}-\text{steradian}}$$

$$\text{forward source } S(\bar{r},E,\bar{\Omega}) = \frac{\text{forward particle}}{\text{volume}-\text{energy}-\text{solid angle}} = \frac{1}{\text{cm}^3-\text{eV}-\text{steradian}}$$

$$\text{forward response } R(\bar{r},E,\bar{\Omega}) = \frac{\text{response}}{\text{forward particle tracklength}} = \frac{1}{\text{cm}}$$

$$\text{adjoint flux } \phi^*(\bar{r},E,\bar{\Omega}) = \frac{\text{adjoint particle tracklength}}{\text{volume}-\text{energy}-\text{solid angle}} = \frac{\text{cm}}{\text{cm}^3-\text{eV}-\text{steradian}} = \frac{1}{\text{cm}^2-\text{eV}-\text{steradian}}$$

$$\text{adjoint source } S^*(\bar{r},E,\bar{\Omega}) = \frac{\text{adjoint particle}}{\text{volume}-\text{energy}-\text{solid angle}} = \frac{1}{\text{cm}^3-\text{eV}-\text{steradian}}$$

$$\text{adjoint response } R^*(\bar{r},E,\bar{\Omega}) = \frac{\text{response}}{\text{adjoint particle tracklength}} = \frac{1}{\text{cm}}$$

Equating $S(\bar{r},E,\bar{\Omega}) = R^*(\bar{r},E,\bar{\Omega})$ and $R(\bar{r},E,\bar{\Omega}) = S^*(\bar{r},E,\bar{\Omega})$ gives for both equalities,

$$\text{adjoint particle} = \frac{\text{response volume}-\text{energy}-\text{solid angle}}{\text{forward particle tracklength}} = \frac{\text{cm}^3-\text{eV}-\text{steradian}}{\text{cm}} = \text{cm}^2-\text{eV}-\text{steradian}$$

Substitution of the above quantity into the adjoint flux $\phi^*(\bar{r},E,\bar{\Omega})$ gives

$$\phi^*(\bar{r},E,\bar{\Omega}) = \frac{\text{response}}{\text{forward particle}} .$$

This expression can be interpreted as the importance of a forward particle, the classical definition of the adjoint flux.[42]

When the adjoint mode has been selected for some reason, it is still convenient to think in terms of the forward mode estimation of the results. That is, if a forward source and estimator had been used, what is the corresponding adjoint procedure? The adjoint source mechanics must simulate the opposite effect of that for forward estimation, and the adjoint estimator mechanics must simulate the opposite effect of that for the forward source. Use of the above relationship for λ is helpful. It should be emphasized that *S(P)* and *R(P)* can take on other than energy dependence. In doing forward Monte Carlo and the associated mechanics of source and estimation routines, it is helpful to think in terms of particles (current). Flux is the quantity from the estimation routines used for multiplication with a response function. In an adjoint calculation, particularly for a multigroup code like MORSE, the code mechanics are the same as for the forward case (see Section 4.10 in the MORSE manual[1]). However, adjoint source and response specification and normalization are sometimes difficult, or confusing, due to the flux (not current) terms in the integrals for $\lambda_{FOR} = \lambda_{ADJ}$. The next six sections discuss these items for point, surface, and volumetric adjoint sources and estimators.

VI.A. Point Adjoint Source

This case corresponds to a forward next-event estimation to a point. Since this is a forward flux estimator (currents are not considered), the adjoint source is selected directly from *R(P)*; that is, $\lambda_{FOR} = \int \phi(P)R(P)dP$:

$S^*(\bar{r}) = R(\bar{r}) = \delta(\bar{r}-r_o)$, point source (card D).

$S^*(E) = R(E) = R_g$, standard source spectrum input (cards E1) using the forward response. The code reverses the group structure and creates the cumulative distribution.

$S^*(\bar{\Omega}) = R_g(\bar{\Omega})$; if the response is isotropic (as is usually the case), $S^*(\bar{\Omega}) = d\bar{\Omega}$ because estimates are scored identically in the forward case regardless of the direction of the trajectory toward the point. Use GTISO(U,V,W) for initial adjoint particle directions.

The point adjoint source normalization is given in general form and then followed by the usual cases

$$
\begin{aligned}
N_{\mathrm{ADJ}} &= \int S^*(P)dP \\
&= \int_{\bar{r}} R(\bar{r})d\bar{r} \int_E R(E)dE \int_{4\pi} R_g(\bar{\Omega})d\bar{\Omega} \\
&= \int_{\bar{r}} \delta(\bar{r}-\bar{r}_o)d\bar{r} \sum_g R_g \int_{4\pi} d\bar{\Omega} \\
&= (1)(\sum_g R_g)(4\pi) \\
&= 4\pi \sum_g R_g
\end{aligned}
$$

The adjoint source energy normalization, $\sum_g R_g$, is done automatically by the MORSE code for all cases.

VI.B. Surface Adjoint Source

The two usual forward surface estimators are flux, $\phi = J/|\mu|$, and current, J. Although the estimation of ϕ in $\lambda_{\mathrm{FOR}} = \int \phi(P)R(P)dP$ has angular dependence, it is assumed that $R(P)=S^*(P)$ does not. However, if λ is a current (leakage across a boundary), then $\lambda_{\mathrm{FOR}} = \int \phi(P)\mu dP = \int J(P)dP$. But the response associated with $\phi(P)$ is $R(P) = \mu$. Therefore, $S^*(\bar{\Omega}) = \mu$ to simulate a forward current estimator.

$S^*(\bar{r}) = \mathrm{dA}$ Pick adjoint source points uniformly on the surface area for which forward estimates would be scored independent of position.

$S^*(E) = R_g$ Same as for point adjoint source.

$S^*_\phi(\bar{\Omega}) = \mathrm{d}\bar{\Omega}$ Same as for point source flux. For forward estimation in only one direction such as leakage flux at the outer boundary, adjoint particles are selected uniformly in 2π steradians.

$S^*_J(\bar{\Omega}) = \mu$ For a leakage estimator, select adjoint directions from a cosine distribution relative to the normal of the surface (see Fig. 8). For an interior boundary total current, half the particles start in each direction.

The normalization for the flux and current sources are:

$$N^{\phi}_{\mathrm{ADJ}} = \int_A dA \sum_g R_g \int_{4\pi} d\bar{\Omega}$$

$$= 4\pi A \sum_g R_g \ .$$

$$N^{\mathrm{J}}_{\mathrm{ADJ}} = A \sum_g R_g \int_0^{2\pi} d\phi \left[-\int_{-1}^{0} \mu d\mu + \int_0^1 \mu d\mu \right] \qquad -1 \leqslant \mu \leqslant 1$$

$$= 2\pi A \sum_g R_g \ .$$

The normalization of particle emission from only one side of a boundary ($0 \leqslant \mu \leqslant 1$) would be 1/2 of each of the above terms.

VI.C. Volume Adjoint Source

The two common forward volume flux estimators, collision density and track length, are simulated by the same adjoint source. Forward current estimators are not considered here.

$S^*(\bar{r}) = \mathrm{dV}$	Select positions uniformly in the volume.
$S^*(E) = R_g$	Same as for point.
$S^*(\bar{\Omega}) = \mathrm{d}\bar{\Omega}$	Same as for point if forward estimation is isotropic.

A volume adjoint source normalization is similar to that for the boundary flux:

$$N_{\mathrm{ADJ}} = 4\pi V \sum_g R_g \ .$$

If a calculation involves some unusual forward response function, such as a collimated detector, so that $S^*(\bar{\Omega})$ is uniform in some interval other than 4π or 2π, this must be included both in the source emission and normalization. The functional form of any variable, continuous or discrete, must be accounted for in both processes. Adjoint particles should not be started where the forward response is zero, i.e., for $R_g = 0$; for groups below that corresponding to ECUT in a simulated forward calculation; or in volumes, on surfaces, at points, or in directions for which no forward scores would have been made for a particular evaluation of λ. Only one (forward) detector can be evaluated in one adjoint calculation, in contrast to a forward calculation. However, the adjoint mode can evaluate this detector response for multiple (forward) sources, which the forward mode cannot. All of the forward response non-zero phase space must be sampled for the adjoint source. If natural sampling is inadequate for good statistics, then the same general biasing techniques for forward source selection should be applied.

The user estimation routines for MORSE adjoint calculations are identical to those for the forward mode in terms of code variables and techniques used. The adjoint flux ϕ^* is the importance function regardless of what estimator is used. In the following sections, the adjoint response $R^*(P)$ = forward source $S(P)$ is given for each type of geometric detector. There can be multiple adjoint detectors of different types (forward sources) for one calculation.

VI.D. Point Adjoint Detector

This case corresponds to adjoint flux from next-event estimation (RELCOL) including an uncollided estimate (SDATA):

$R^*(\bar{r}) = S(\bar{r}) = \delta(\bar{r}-\bar{r}_o)$	Point detector input (cards CC).
$R^*(E) = S_g$	Standard response input (cards FF) using the forward source spectrum (unnormalized fraction of forward source particles in each group, i.e., what would go into card E1 for a forward case); see comments regarding $S^*(E)$.
$R^*(\bar{\Omega}) = S(\bar{\Omega})$	For an isotropic forward source $R^*(\bar{\Omega}) = 1/4\pi$; otherwise, the angular dependence of the estimation trajectory from RELCOL or SDATA must be considered.

The forward source normalization (adjoint detector) is

$$N_{\text{FOR}} = \int_{\bar{r}} S(\bar{r})d\bar{r} \int_{E} S(E)dE \int_{4\pi} S(\bar{\Omega})d\bar{\Omega}$$

$$= \int_{\bar{r}} \delta(\bar{r}-\bar{r}_o)d\bar{r} \sum_{g} S_g \int_{4\pi} \frac{1}{4\pi} d\bar{\Omega}$$

$$= \sum_{g} S_g \ .$$

The result of an adjoint calculation λ_{ADJ} is, when normalized to give the results in terms of a unit forward source,

$$\lambda_{\text{FOR}} = \lambda_{\text{ADJ}} \frac{N_{\text{ADJ}}}{N_{\text{FOR}}} \ ,$$

where N_{ADJ} is the adjoint source normalization from any of the above cases (Section VI.A, B, or C). It is seen that the ratio of λ_{ADJ}/N_{FOR} is the same whether or not $\sum_g S_g = 1$ or $R^*(\bar{\Omega}) = 1$ or $1/4\pi$. In scoring with $R^*(E)=S_g$ in the λ_{ADJ} result, any normalization in S_g will cancel with that in $N_{FOR} = \sum_g S_g$. Likewise, the normalization of $R^*(\bar{\Omega})$ will also cancel in the λ_{ADJ} and N_{FOR}; i.e., if $S(\bar{\Omega})$ in the N_{FOR} above has the $1/4\pi$, then the estimator in λ_{ADJ} must also contain $1/4\pi$. Any variable or constant term appearing in the λ_{ADJ} estimator must appear exactly the same in N_{FOR}.

VI.E. Surface Adjoint Detector

Just as for a forward case, a surface adjoint estimator can evaluate J^* or $\phi^* = J^*/|\mu|$ (WTBC or WTBC/$|\mu|$). However, only ϕ^* can be used to score with the forward source $S(P)$ from $\lambda_{ADJ} = \int \phi^*(P)S(P)dP$. Using J^* would be equivalent to scoring with another forward source $\mu S(P)$, but $S(P)$ is fixed and cannot be changed. To score total forward leakage $J=\lambda_{ADJ}$ would require $S^* = R = \mu$ to be the proper adjoint leakage source, regardless of the adjoint scoring function $S = R^*$ used here. Standard grazing angle techniques must be employed for ϕ^* evaluation at a boundary.

It is the adjoint current J^* that is used in coupling techniques in the response mode, i.e., adjoint MORSE–forward DOT in DOMINO, just as J is used for forward MORSE–adjoint DOT. This is because DOT provides a flux rather than a current, as for a physical source:

$$\lambda_{FOR} = \int \phi_{MOR}\phi^*_{DOT}(\bar{n}\cdot\bar{\Omega})d\bar{\Omega} = \lambda_{ADJ} = \int \phi^*_{MOR}\phi_{DOT}(\bar{n}\cdot\bar{\Omega})d\bar{\Omega} .$$

The $\phi_{MOR}(\bar{n}\cdot\bar{\Omega})$ or $\phi^*_{MOR}(n\cdot\bar{\Omega})$ is just J or J^* and

$$\lambda_{FOR} = \int J\phi^* d\bar{\Omega} = \lambda_{ADJ} = \int J^*\phi d\bar{\Omega} .$$

For the surface adjoint detector:

$R^*(\bar{r}) = S(\bar{r}) = 1/A$	Divide by the surface area for each score.
$R^*(E) = S_g$	Same as for point adjoint detector.
$R^*(\bar{\Omega}) = S(\bar{\Omega})$	Forward source angular distribution.

For azimuthally symmetric adjoint estimation across (forward emission from) only one side of a surface, $S(\Omega) = f(\mu)/2\pi$, and $f(\mu) = 1$ for isotropic emission, 2μ for cosine emission, and $\delta(\mu-\mu_o)$ for monodirectional emission, which could be scored only with next-event adjoint estimation to a surface; i.e., no adjoint particle could cross exactly in the right direction. The functional form of any other continuous or discrete $S(\bar{\Omega})$ would be scored depending on the adjoint direction, -Ω, crossing the surface; i.e., the $\Omega_{ADJ} = -\Omega$ is

in the opposite sense of the forward source emission. The forward source (adjoint response) normalization is, for only one side of a surface ($0 \leqslant \mu \leqslant 1$),

$$N_{\mathrm{FOR}} = \int_A dA/A \sum_g S_g \int_0^{2\pi} \frac{d\phi}{2\pi} \int_0^1 f(\mu) d\mu$$

$$= \sum_g S_g \int_0^1 f(\mu) d\mu \ .$$

In determining the result for a unit forward source λ_{FOR}, as for the point adjoint detector above, the source spectrum normalization cancels in the $\lambda_{\mathrm{ADJ}}/N_{\mathrm{ADJ}}$ ratio. Likewise if the inverse of the area A were not included in the estimate for λ_{ADJ}, it would be included in the N_{FOR}; i.e, $S(\bar{r}\,)$ would be unity and the integral would give A.

VI.F. Volume Adjoint Detector

The procedure here would be the same for both adjoint collision density or track length estimators in a volume where forward source selection would normally be uniform in volume and isotropic in direction:

$R^*(\bar{r}) = S(\bar{r}) = 1/\mathrm{V}$ Divide by the source volume.

$R^*(E) = S_g$ Same as for point adjoint detector.

$R^*(\bar{\Omega}) = S(\bar{\Omega}) = 1/4\pi$ Forward source angular distribution.

$$N_{\mathrm{FOR}} = \int_V dV/V \sum_g S_g \int_{4\pi} d\bar{\Omega}/4\pi = \sum_g S_g \ .$$

As for the surface adjoint response, if the $1/V$ and $1/4\pi$ were replaced by unity in the evaluation of λ_{ADJ}, these terms would appear in N_{FOR} giving the same λ_{FOR}. The same comments apply also to the source spectrum normalization.

Time-dependent adjoint calculations are performed the same as forward calculations with regard to the time variable. The elapsed time Δt is the same whether time goes "forward" or "backward." Thus, after the other variables are selected from the adjoint source $S^* = R$, the time is selected from the forward $S(t)$, often $\delta(t - t_o)$. The adjoint particle is scored with the forward response $R(t)$, usually unity; i.e., the particles are scored the same regardless of the value of the time variable. The time cutoff and time collection intervals are also the same as for the forward case.

Several general comments can be made concerning the adjoint results. If it is desired to calculate total forward flux ϕ or current J, i.e., $\lambda_{\mathrm{ADJ}} = \phi$ or J, then the adjoint source $S^* = R$ would be unity in all groups. That is, adjoint particles are selected uniformly in

group number since $R(E) = 1$ for ϕ or J calculations in the forward mode. Energy group dependent adjoint results have no correspondence to the forward calculated group results, but the integrated quantities are the same, $\lambda_{FOR} = \lambda_{ADJ}$, when the appropriate normalizations are applied.

VI.G. Forward Energy-Dependent Results from Adjoint Calculations

It is possible to obtain forward energy-dependent results from adjoint calculations by scoring in the adjoint response $R^* = S$ with the energy group of the adjoint source rather than the energy group of the estimation.[29] To score forward flux, the adjoint source $S^* = R$ would be unity for all groups. In the call to FLUXST in the adjoint estimation routine, use IG (or IGO) in the normal place, and in an additional calling parameter use the group number selected for that particle in the source routine, IGS. This could be done by filling a user array in common in the source routine, i.e., COMMON/SOUGP/ISG(NSTRT) where NSTRT is the number of source particles, and setting ISG(NAMEX)=IG for each source particle. Any secondary generated particle will have the same NAMEX value as the generating particle. The call to FLUXST in the estimation routine would use IGS=ISG(NAMEX) as the additional calling parameter. This procedure could also be done by use of an unused NUTRON common variable, e.g., AGE for a time-independent calculation. Set AGE=IG in the source routine and IGS=AGE for the call to FLUXST (make allowance for any fixed-floating point number roundoff). The incrementation of AGE in subroutine NXTCOL must be deactivated. The first card in SUBROUTINE FLUXST must be changed to accept the additional calling parameter. Before the final loop, in the variable IB, change IGE (the standard energy parameter) to IGS. If this procedure is used with adjoint MORSE in DOMINO, the standard FLUXST is called with IGS as the first parameter since the coupling code automatically selects the correct DOT forward flux.

The energy-dependent results from the above procedure will be the same as those from a forward calculation (within statistics) when given the appropriate normalizations. This is the same as making a separate adjoint calculation for each group, as is necessary for determining discrete ordinates forward flux from adjoint calculations. This procedure of obtaining forward differential results from an adjoint calculation can be extended, in principle, to any variable by scoring (calling FLUXST) with the variable of the adjoint source rather than the value at the time of estimation. That is, divide the source into angular or spatial intervals (like energy intervals) and score with these intervals when a particle reaches the detector.

VI.H. Example of MORSE Adjoint Calculations

Given here are two examples of adjoint MORSE calculations which are extensions of those given earlier.[43] The forward procedures are also given. Consider a sphere of radius R of one medium centered at X=Y=Z=0 with a uniform volumetric and isotropic angular source. The source energy dependence S_g is different for each of the NMTG groups. It is desired to calculate the total leakage at the outer surface. Start particles with unit weight (WTSTRT=1 on card C) in both cases. The forward calculation is straightforward in MORSE and is as follows:

- The source spectrum is input into cards E1.
- Set UINP=VINP=WINP=0 on card D (isotropic source directions).
- In a user source routine call GTISO(XX,YY,ZZ) for an isotropic vector, set the radius RAD=R*(FLTRNF(0))**0.33333, then X=RAD*XX, Y=RAD*YY, and Z=RAD*ZZ (uniform volumetric source).
- Input a unit response (1.0 in each group) on cards FF.
- Score WTBC in FLUXST in a user routine called from BANKR(8). This gives total leakage at the outer surface.

In the adjoint calculation:

- Input a unit spectrum (1.0 in each group) in cards E1. This is because in the forward calculation the WTBC was multiplied by 1.0 before final scoring.
- Set XINP=YINP=0.0 and ZINP=−R on card D; i.e., all particles start at 0,0,−R. Since the entire problem is spherically symmetric, this procedure correctly simulates a uniform adjoint source selection over the complete surface. Any non-symmetry anywhere in the system would disallow this method. The general procedure would be the same as for X,Y,Z in the forward case for RAD=R.
- In a user source routine, set W=SQRT(FLTRNF(0)) — see $S_J^*(\bar{r})$ in Section VI.B. Call AZIRN(SX,CX) and set U=SQRT(1.-W*W)*CX and V=SQRT(1.-W*W)*SX. It is because of this angular selection that the source position was set at 0,0,−R. Any positions off the coordinate axes would require coordinate transformations for these initial direction cosines, as for the second source example (Section V.D).
- Input the forward source S_g into the response (cards FF) in the forward group order.
- Call a user written COLDEN from BANKR(5) and score CON=WTBC/Σ_T in FLUXST. The Σ_T is from a call to NSIGTA with group number IGO.

The resulting leakage of each calculation normalized to one source particle appears under the TOTAL RESPONSE column for DETECTOR 1 in the MORSE output. To equate the adjoint result to the forward result, the following normalizations are needed:

$$N_{\text{ADJ}} = (4\pi R^2)\ (\pi)\ .$$

This normalization is from Section VI.B; the first term is the surface integration and the second term the angular integration. The energy term, in this case numerically equal to the number of groups NMTG due to a sum over the unit response, is included in the result automatically by the code. The message in older versions of MORSE to do this multiplication by hand should be ignored if the standard NRUN routine is included. It is only the adjoint source energy normalization (referred to as DFF) that is included in the code.

The other normalization term is (see Section VI.F):

$$N_{\mathrm{FOR}} = (4/3\pi R^3)\ (4\pi) \sum_g S_g \ .$$

Each of these terms could have appeared in the adjoint score in COLDEN for λ_{ADJ}, i.e., CON=WTBC/Σ_T/N_{FOR}, and the $\lambda_{\mathrm{FOR}} = \lambda_{\mathrm{ADJ}}\ N_{\mathrm{ADJ}}/N_{\mathrm{FOR}}$ would have been unchanged. The first term in N_{FOR} is $[R^*(r)]^{-1} = V$, the second term is $[R^*(\Omega)]^{-1} = 4\pi$, and the last term could have been easily normalized to unity in the input data. When CON is multiplied by the $R^*(E)=S_g$ in MORSE following the call to FLUXST, any non-normalization in each S_g is included in λ_{ADJ}, so the effects cancel and λ_{FOR} is unchanged. If S_g is zero in any group, then no source particles are started and no adjoint estimation is made in those groups. But since $R_g=1$ in all groups, all groups must be included in both calculations. The final normalization for converting the adjoint-generated result λ_{ADJ} to that given by the forward calculation is

$$\lambda_{\mathrm{FOR}} = N_{\mathrm{ADJ}}/N_{\mathrm{FOR}}{*}\lambda_{\mathrm{ADJ}} = 0.75/\mathrm{R}\ *\ \lambda_{\mathrm{ADJ}}\ ,$$

where λ_{FOR} should agree with the forward result within some statistical uncertainty.

As a second example of a MORSE adjoint calculation, consider the system in Fig. 19. Particles impinge monodirectionally on the volume at X=0.0 in a circular cross-section beam of radius R centered at Y=0.0, Z=0.0. The particle beam is also monoenergetic and this energy is in the highest energy group, IG=1, in the forward structure. It is desired to determine the total and energy-dependent (forward) flux at a detector point (XD,YD,ZD) embedded in the volume as shown.

The steps for the forward calculation are:

- ISOUR=1 (card C) source energy group.
- UINP=1.0, VINP=WINP=0, monodirectional beam.
- In a user source routine set RAD = R*SQRT(FLTRNF(O)) and call AZIRN(SX,CX); set X=0, Y=RAD*SX, Z=RAD*CX — uniform starting positions in the circular beam at the edge of the volume.
- Input unit response — all 1.0s on cards FF.
- Input detector location XD,YD,ZD on card CC.
- Score flux using next-event estimation in the standard RELCOL called from BANKR(5)
- If the detector location is outside the extension of the source beam, R < $\sqrt{ZD^2+YD^2}$, there is no uncollided flux; otherwise, call SDATA from BANKR(1). The standard contribution here must be changed due to the monodirectional source to

$$\mathrm{CON=WATE*EXP(ARG)/AREA}\ ,$$

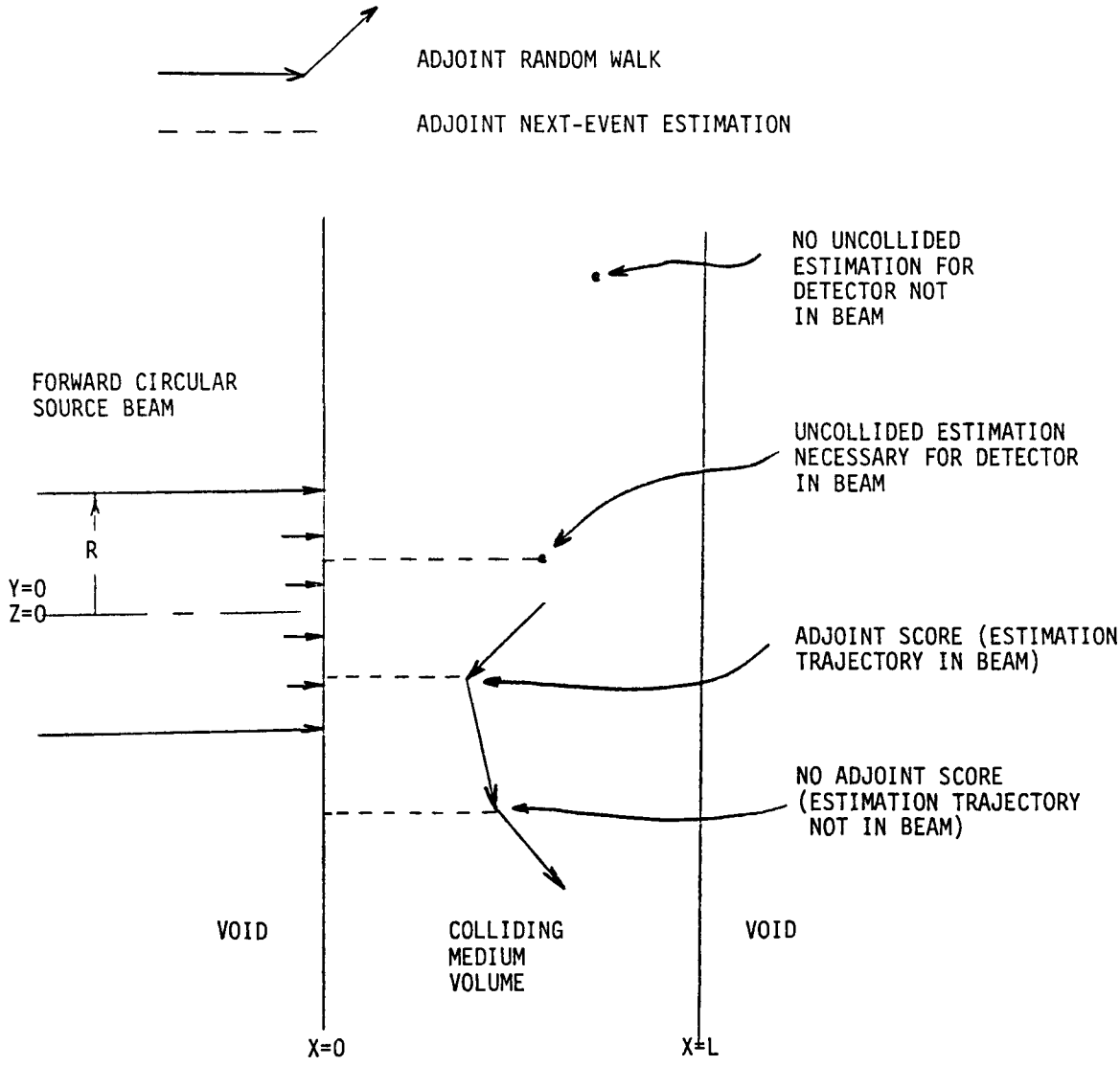


Fig. 19. Second Adjoint Calculation Example.

where AREA is the cross sectional area of the beam, πR^2. This AREA term arises since only one line in the beam can actually pass through the detector position, but a score is made for all source particles.

- The total and energy-dependent flux will be output by inputting data on cards HH. The infinite variance characteristic of the point detector estimator in RELCOL is neglected in this example; however, unrealistic results may occur.

The adjoint calculation of this problem is as follows:

- Set XINP=XD, YINP=YD, ZINP=ZD on card D (point adjoint source).
- Set UINP=VINP=WINP=0 on card D (isotropic source).

- In a source routine, save the source group IGS for each particle (see Section VI.G).
- Input $S_g^* = 1.0$ for all groups on cards E1 ($S_g^* = R$).
- Input data for two detectors on cards BB and CC, both positions at XD=YD=ZD=0.0.
- Input $R_1^* = 1.0$ and $R_g^* = 0.0$ for all other groups on cards FF. ($R_g^* = S$, and $S_1 = 1$, $S_g = 0$, $g \neq 1$. The code reverses the group structure.)
- Call a modified version of RELCOL from BANKR(5). If the collision point is outside the beam, $\sqrt{Y^2+Z^2} > R$, RETURN (no score is made). Otherwise, set XD=0.0, YD=Y, ZD=Z (do not use the usual values from blank common) as the detector point in the beam on the edge of the volume. If the selected energy group IL is not NMTG, the largest group number (highest energy), RETURN. The code reverses the group structure for adjoint calculations so that $R_g^* = 1$ is now in IG=NMTG. This test is included only for efficiency since $R_g^*=0$ for the other groups and the ultimate contribution to the flux will be zero in the other groups. The estimator is unchanged from that in standard RELCOL. No AREA is needed in CON but will be included in N_{FOR}. As for the forward case, the infinite variance characteristic of the estimator is also present here (setting XD just inside the void, e.g., XD = −0.001, may be helpful).
- If an uncollided estimate is necessary [call SDATA from BANKR(1)] in the forward case, it is calculated exactly the same as for the modified RELCOL in the previous paragraph. Here only particles in IG=NMTG contribute and the AREA division is not needed as for the forward uncollided term (all adjoint source particles "see" the detector).

In RELCOL a second call to FLUXST for detector #2 will give forward flux when the energy group used in FLUXST is not IL=NMTG, as in the first call to FLUXST, but rather is the source group IGS for the particle NAMEX value as set in the special source routine. However, the call to FLUXST is still made only when IL=NMTG. Because R_g is non-zero in only one group, no modification is needed in FLUXST (Sect. VI.G). The second call to FLUXST in SDATA is the same as the first call since the group number is the same in both calls.

The adjoint results λ_{ADJ} must be adjusted to compare with the forward results. The adjoint point source normalization is (see Section VI.A) $N_{ADJ}=4\pi$ for the isotropic source. The $\sum_g R_g$ is included in the λ_{ADJ} by the code. The N_{FOR}=AREA$=\pi R^2$ since the area was not divided into the estimation in RELCOL and SDATA as $R^*(\bar{r})$; i.e., $R^*(\bar{r})=1$ not 1/AREA. $\sum_g S_g = 1$ and $f(\mu)=\delta(\mu-\mu_o)$, $\mu_o = 1$ (see Section VI.E). The adjoint results are adjusted to give λ_{FOR} to compare with the forward calculated results as

$$\lambda_{FOR} = 4/R^2 \; \lambda_{ADJ} \; .$$

Both results could be poor due to the point detector estimator. In the adjoint results, detector 1 gives the response and adjoint energy-dependent flux, and detector 2 gives the same response but the forward energy flux from an adjoint calculation. The forward-calculated energy-dependent flux is divided by ΔE in eV, but will agree with the adjoint-calculated forward flux if, in the forward problem, input for the energies (cards E1) is 1.0, for which the code sets $\Delta E = 1.0$.

VI.I. Comments on MORSE Adjoint Calculations

It has been shown that with some extra thought and effort, adjoint MORSE calculations are no more difficult than forward calculations. For both modes, much of the physics and theory is contained within the multigroup cross-section structure. The user is concerned only with source, estimation, and normalization processes. Since many radiation transport problems are more amenable to adjoint solution than to forward solution, a multigroup code such as MORSE greatly increases a user's capability. It has also been shown that not only are total integrated responses available, but also forward energy-dependent quantities. In principle, it is possible to extend this forward differential capability to spatial and anglar dependence by starting and scoring these variables in intervals in the same manner as for the energy intervals of the multigroup structure. It is possible to calculate the forward flux in a detector from each of multiple forward sources in one adjoint calculation, allowing the use of multiple forward response functions. However, statistical uncertainty considerations will usually preclude the utilization of all possible results from any one calculation. By reversing the preceding techniques, obtaining adjoint quantities from forward calculations, multiple sources as well as responses may be evaluated in one forward calculation (see Sect. VI.G and Ref. 29).

In applying source, estimation, and normalization techniques to adjoint calculations, it is sometimes helpful to first look only at uncollided results; i.e., intermediate collisions should have no effect on these general methods. Also, if the source and detector can be described in one or two dimensions, then the techniques can be tested against discrete ordinates methods, which are sometimes more straightforward.

VII. DOCUMENTATION AND SAMPLE PROBLEMS

The basic manual for MORSE-CG is ORNL-4972 (Ref. 1). There are also two revisions to this manual, ORNL-4972/R1 and ORNL-4972/R2, the first containing corrections and updates and the second an addendum of sample problems. The basic manual also includes the PICTURE code for two-dimensional geometry plots. The original MORSE manual, ORNL-4585 (Ref. 6), although now out of date in many respects, contains numerous subroutine flow charts still applicable to much of the current code. Use of these charts in conjunction with FORTRAN listings and the routine descriptions in the current manual is helpful for writing user routines and making changes in the code. The next manual will be for MORSE-CGA, the array geometry version. The manual for this array geometry, MARS, and its three-dimensional geometry plotting package, JUNEBUG, are part of the SCALE system document.[5] There is also a good general MORSE code reference for the Sandia National Laboratory version of MORSE-SGC.[30] Many useful examples for user routines are found in this report. However, this code is completely different from the ORNL code, and caution must be used when applying specific details of one code or manual to the other. Documents related to other features of the MORSE-CG code system or closely related codes are:

- PICTURE[1] (two-dimensional geometry plot)
- JUNEBUG[5] (three-dimensional geometry plot)
- Collision plotting[28] (collision density and response plots)
- DOMINO[26] (MORSE-DOT coupling)
- BREESE[19,21] (albedo routines)
- CARP[20] (albedo data)
- VCS[44] (special adjoint MORSE-forward DOT coupling)
- Criticality calculations in MORSE[18] (general capabilities)

Each new application of MORSE usually results in some new user-routine capability. Reported work often not only explains the theory or techniques used in MORSE, but often gives subroutine listings in the appendices. The input data and special routines for nine sample problems featuring various options in MORSE are given in the MORSE code package from RSIC. The sample problem report ORNL-4972/R2 gives a summary and input data for each problem. The complete output listings are in this report in a microfiche packet. These sample problem features are:

1. neutrons, point source, spherical geometry, boundary crossing,
2. neutron-secondary gamma-ray production for problem #1,
3. time-dependent adjoint for problem #2,
4. k_{eff} for small sphere, SOURCE for first batch,
5. time-dependent k_{eff}, special BANKR,

6. primary gamma-ray source, similar to problem #1,
7. cross-section (XCHEKR) preparation,
8. collision density estimator for problem #1, special ENDRUN,
9. example of PICTURE, two-dimensional geometry plot.

VIII. ON-GOING AND FUTURE DEVELOPMENTS

Among items currently in development or planned for inclusion in the MORSE code system are:

- Documentation and release to RSIC of the MARS geometry and associated JUNEBUG plotting capability. With this addition, MORSE-CG will become MORSE-CGA.
- Improvement in the efficiency of MARS and JUNEBUG.
- Documentation and release of the torus body capability of combinatorial geometry.
- Availability of the gamma-ray Klein-Nishina and pair production next-event estimator by default and independent of the cross-section library.
- Inclusion of an adjoint scattering treatment to alleviate the problem of very large particle weights that occur in some situations for multiple scattering in media with large Σ_S^*/Σ_T^* (adjoint non-absorption probability).
- Continuation of a new DOT-MORSE coupling which, unlike DOMINO, allows the MORSE geometry to be independent of the DOT axis of symmetry. That is, in DOT RZ geometry, the MORSE geometry does not have to be symmetric about the DOT cylindrical axis. At present this coupling is available only in the response mode (forward-adjoint coupling).

REFERENCES

1. M. B. Emmett, "The MORSE Monte Carlo Radiation Transport Code System," ORNL-4972 (1975); also ORNL-4972-R1 (1983) and ORNL-4972-R2 (1984).
2. W. W. Engle, "A User's Manual for ANISN," K-1693, Oak Ridge Computing Technology Center (1967).
3. W. A. Rhoades et al., "The DOT-IV Two-Dimensional Discrete Ordinates Transport Code," ORNL/TM-6529 (1979).
4. N. M. Greene et al., "AMPX, A Modular Code System for Generating Coupled Multigroup Neutron-Gamma Libraries From ENDF/B," ORNL/TM-3706 (1976).
5. "SCALE - A Modular Code System for Performing Standardized Computer Analysis for Licensing Evaluation," ORNL/NUREG/CSD-2/R1.
6. E. A. Straker et al., "The MORSE Code - A Multigroup Neutron and Gamma-Ray Monte Carlo Transport Code," ORNL-4585 (1970).
7. H. Lichtenstein et al., "The SAM-CE Monte Carlo System for Radiation Transport and Criticality Calculations in Complex Configurations (Revision 7.0)," MAGI Report MR-705207; also EPRI CCM-8 (1980).
8. MORSE-SCG module of SCALE (reference 5); also ORNL/CSD-7.
9. KENO V module of SCALE (reference 5).
10. "A Review of Multigroup Nuclear Cross Section Processing," ORNL/RSIC-41 (1978).
11. D. C. Irving et al., "05R, A General Purpose Monte Carlo Neutron Transport Code," ORNL-3622 (1965).
12. C. L. Thompson and E. A. Straker, "06R-ACTIFK Monte Carlo Neutron Transport Code," ORNL-TM-3050 (1969).
13. S. N. Cramer et al., "ESP - A General Monte Carlo Reactor Analysis Code," ORNL-TM-3164 (1972).
14. S. N. Cramer and E. M. Oblow, *Nucl. Sci. Eng. 62*, 532-549 (1977).
15. L. L. Carter and C. A. Forest, *Nucl. Sci. Eng. 59*, 27-45 (1976).
16. J. A. Bucholz, ORNL, private communication.
17. M. Solomito et al., "Methods of Biasing Secondary Gamma Ray Production in Coupled Neutron Gamma Ray Monte Carlo Calculations," ORNL-TM-3530 (1971).
18. C. E. Burgart, "Capabilities of the MORSE Multigroup Monte Carlo Code in Solving Reactor Eigenvalue Problems," ORNL-TM-3662 (1971); also, IAEA Seminar on Numerical Reactor Calculations, Vienna (1972).
19. V. R. Cain and M. B. Emmett, "BREESE-II: Auxiliary Routines for Implementing the Albedo Option in the MORSE Monte Carlo Code," ORNL/TM-6807 (1979).

20. M. B. Emmett and W. A. Rhoades, "CARP, A Computer Code and Albedo Data Library for Use by BREESE, The MORSE Albedo Package," ORNL/TM-6503 (1978).

21. "Neutron Streaming Calculations with MORSE/BREESE for the ORNL-TSF Two-Legged Duct and CRBR Prototypic Coolant Pipe Chaseway Experiments," Science Applications Report SAI 1-138-02-034-00 (1978).

22. S. N. Cramer and E. M. Oblow, "Feasibility Study of a Honeycomb Vacuum Wall for Fusion Reactors," *Nuclear Fusion* **15** (1975); "Reduction of Refluxing Particles into a CTR Plasma by Use of a Honeycomb Wall," *Nuclear Fusion* **16** (1976); also ORNL/TM-4708 and ORNL/TM-4981.

23. R. A. Lillie et al., "Monte Carlo Simulation of Molecular Flow in a Neutral Beam Injector and Comparison with Experiment," *J. Nucl. Energy* **2** (1982); also, "Microwave Transport in EBT Distribution Manifolds Using Monte Carlo Ray Tracing Techniques, *Nucl. Tech./Fusion* **4** (1983).

24. S. K. Fraley et al., "A Monte Carlo Method of Solving Heat Conduction Problems," ORNL/NUREG/CSD-3 (1977).

25. W. H. Scott and V. E. Staggs, "Adjoint Energy Biasing and Thermal Neutron Diffusion in the MORSE and VCS Codes," Science Applications Report SAI-133-81-384-LJ (1981).

26. M. B. Emmett, "DOMINO-II, A General Purpose Code for Coupling DOT-IV Discrete Ordinates and Monte Carlo Radiation Transport Calculations," ORNL/TM-7771 (1981).

27. T. J. Hoffman et al., *Nucl. Sci. Eng.* **48**, 179-188 (1972).

28. E. A. Straker and M. B. Emmett, "Collision Site Plotting Routines and Collision Density Fluence Estimates for the MORSE Monte Carlo Code," ORNL-TM-3585 (1971).

29. S. N. Cramer and J. S. Tang, "Variance Reduction Methods Applied to Deep Penetration Monte Carlo Problems," ORNL/TM-9643 (1985).

30. S. A. Dupree and R. E. Lighthill, "Sandia National Laboratories CDC7600 Version of MORSE-SGC," SAND80-1337 (1982).

31. R. M. Westfall et al., "Criticality Analyses of Disrupted Core Models of Three Mile Island Unit 2," ORNL/CSD/TM-106 (1979).

32. C. J. Everett and E. D. Cashwell, "A Third Monte Carlo Sampler," LA-9721-MS, Los Alamos (1983).

33. E. J. McGrath et al., "Techniques for Efficient Monte Carlo Simulation," ORNL-RSIC-38 (1974).

34. H. Kahn, "Applications of Monte Carlo," AECU-3259, Rand Corp. (1954).

35. E. D. Cashwell and C. J. Everett, "Monte Carlo Method for Random Walk Problems," Pergamon Press, New York (1959).

36. S. N. Cramer, *Nucl. Sci. Eng.* **79**, 417-425 (1981).

37. R. A. Lillie and T. A. Gabriel, ORNL, private communication.

38. S. N. Cramer, *Nucl. Sci. Eng.* **65**, 237-253 (1978).

39. J. C. Nimal et al., "TRIPOLI-2, Three-Dimensional Polyenergetic Monte Carlo Radiation Transport Program," available from RSIC as CCC-372.

40. S. N. Cramer, et al., "Monte Carlo Techniques for Analyzing Deep Penetration Problems," Proceedings of an International Meeting on Advances in Nuclear Engineering Computational Methods, Knoxville, TN (April 1985)

41. T. J. Hoffman, "The Adjoint Difference Method and Its Application to Deep Penetration Radiation Transport," NEUT 0150-2, University of Tennessee, Ph.D. thesis (1971).

42. D. C. Irving, "The Adjoint Boltzmann Equation and Its Simulation by Monte Carlo," ORNL-TM-2879 (1970).

43. T. J. Hoffman, unpublished notes.

44. W. A. Rhoades et al., "Vehicle Code System (VCS) User's Manual," ORNL-TM-4648 (1974); also, ORNL-TM-4664 (1974).

INTERNAL DISTRIBUTION

1–5. L. S. Abbott
6. R. G. Alsmiller, Jr.
7. D. G. Cacuci
8–12. S. N. Cramer
13. H. L. Dodds
14. M. B. Emmett
15. T. A. Gabriel
16. D. T. Ingersoll
17. J. O. Johnson
18. R. A. Lillie
19–23. F. C. Maienschein
24. J. V. Pace
25. C. V. Parks
26. L. M. Petrie
27. W. A. Rhoades
28. R. W. Roussin
29. J. C. Ryman
30. R. T. Santoro
31. C. O. Slater
32. J. S. Tang
33. D. K. Trubey
34. M. W. Waddell
35. C. R. Weisbin
36. R. M. Westfall
37. A. Zucker
38. P. W. Dickson, Jr. (Consultant)
39. G. H. Golub (Consultant)
40. D. Steiner (Consultant)
41. Central Research Library
42. Y-12 Document Ref. Section
43. Laboratory Records ORNL, RC
44. ORNL Patent Office
45–49. EPMD Reports Office

EXTERNAL DISTRIBUTION

50. Office of the Assistant Manager for Energy Research and Development, DOE-ORO, Oak Ridge, TN 37830, Attention: S. W. Ahrends/M. Rohr
51–77. Technical Information Center
78–139. DNA Radiation Transport Distribution